



**Universidad de las Ciencias Informáticas**

*Sistema de Gestión Deportivo  
para la  
Universidad de las Ciencias Informáticas*

**Trabajo de Diploma para optar por el título de  
Ingeniero en Ciencias Informáticas**

**AUTORES:**

**Yamila Vigil Regalado**

**Erich Fouces Cabana**

**TUTOR:**

**Ing. Yaimí Trujillo Casañola**

**Ciudad de La Habana, junio del 2007**

**“Año del 49 Aniversario de la Revolución”**

# DECLARACIÓN DE AUTORÍA

Declaramos ser autores de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmo la presente a los \_\_\_\_ días del mes de \_\_\_\_\_ del año \_\_\_\_\_.

\_\_\_\_\_

Yamila Vigil Regalado

\_\_\_\_\_

Erich Fouces Cabana

\_\_\_\_\_

Yaimí Trujillo Casañola

*"...es el Deporte una maravillosa actividad, que no sólo ayuda a formar el carácter, no sólo ayuda a forjar hombres de espíritu y de cuerpo fuertes, sino que también alienta al pueblo, entretiene al pueblo, entusiasma al pueblo, y hace feliz al pueblo"*

*Fidel Castro Ruz*

## **AGRADECIMIENTOS**

## **DEDICATORIA**

**De Yamila:**

**De Erich:**

## **RESUMEN**

La actividad física y deportiva ayuda a crear en el ser humano un conjunto de valores sociales y educativos esenciales en su desarrollo. Constituye factor de inserción, de participación en la vida social, de tolerancia, de aceptación de las diferencias y de respeto de las normas. La Universidad de la Ciencias Informáticas (UCI), tiene la función de formar jóvenes con un alto nivel docente y productivo. La práctica de deporte constituye un ente principal en la vida universitaria, ya que asegura la autodisciplina y el optimismo en los estudiantes, permite el fortalecimiento del carácter y la voluntad de la entrega responsable en el trabajo y contribuye a la solidaridad en la comunidad universitaria, el orgullo y el tesón por el ejercicio. El presente trabajo tiene como objetivo fundamental realizar el análisis y diseño del Sistema de Gestión de la Deportiva para la Universidad de las Ciencias Informáticas (SGD-UCI) y la implementación de los módulos de Gestión de gimnasio, Nivelación de pruebas de eficiencia física y Administración. Una vez culminado este trabajo, se obtendrá una versión del sistema capaz de mejorar la gestión de la información deportiva que hoy día se lleva a cabo en la universidad de forma manual, lo que provoca lentitud en el desarrollo de los procesos, pérdida de la información, no almacenamiento, ni seguimiento de la información para posteriores consultas e investigaciones.

<b>INTRODUCCIÓN</b> .....	<b>15</b>
Fig. I.1 Organización del deporte universitario en el mundo.....	15
Fig. I.2 Sistema competitivo en Cuba.....	16
Fig. I.3 Organización del deporte universitario en Cuba.....	17
<b>Capítulo 1: Fundamentación Teórica</b> .....	<b>22</b>
<b>1.1 Introducción.</b> .....	<b>22</b>
<b>1.2 Definiciones y términos relacionados con el deporte.</b> .....	<b>22</b>
<b>1.3 Antecedentes históricos del deporte universitario.</b> .....	<b>25</b>
<b>1.4 El deporte universitario actual.</b> .....	<b>27</b>
Fig. 1.4.1 Miembros asociados a la Federación Internacional de Deporte Universitario .....	28
<b>1.5 El deporte universitario cubano.</b> .....	<b>29</b>
<b>1.6 La Industria del Software Deportivo.</b> .....	<b>29</b>
1.6.1 Sistemas de Gestión Deportiva en el mundo. ....	29
1.6.1.1 DeporWin. ....	30
1.6.1.2 Fútbol TRAIN.....	31
1.6.1.3 Voley TRAIN 2. ....	31
1.6.1.4 SportSPEED 2. ....	31
1.6.1.5 Video STAT 2.0.....	32
1.6.1.6 Minuto Final 2.....	32
1.6.2 Softwares deportivos en Cuba. ....	33
1.6.2.1 Ficha o Expediente del Deportista.....	33
1.6.2.2 Control de la Dotación.....	34
1.6.2.3 Especialistas y Técnicos contratados. ....	34
1.6.2.4 Base histórica de Resultados Deportivos. ....	34
1.6.2.5 Sistema de Gestión Documental (SIGED). ....	34
1.6.2.6 Sistema de Procesamiento Informacional (SIPROINF).....	34
1.6.2.7 Infodeporte.....	35
1.6.2.8 Sistema de Información de Control de competencias (SINFO). ....	35
1.6.2.9 CHESSUC: (Chees=ajedrez; UC=Universidad de Camagüey).....	35
1.6.2.10 EXPERTO V.1.0. ....	35
1.6.2.11 Pesoft Calificador 13-14 y Pesoft Calificador 15-16.....	36
1.6.3 Análisis crítico de los Sistemas de Gestión Deportiva existentes.....	36
<b>1.7 Tendencias y tecnologías actuales.</b> .....	<b>36</b>
1.7.1 Tecnología Cliente/Servidor. ....	36
1.7.2 Lenguajes de Programación. ....	37
1.7.2.1 PHP. ....	37
Fig. 1.7.2.1.1 Funcionamiento del PHP.....	38

1.7.2.2	Java	40
1.7.2.3	ASP.NET	40
1.7.2.4	Perl (Practical Extraction and Report Language)	40
1.7.3	Fundamentación del lenguaje a escoger	40
1.7.4	Sistemas Gestor de Base de Datos	41
1.7.4.1	Microsoft SQL Server	42
1.7.4.2	PostgreSQL	43
1.7.4.3	MySQL	44
1.7.5	¿Por qué usar MySQL?	47
1.7.6	Servidor Web Apache	48
1.7.7	Proceso de desarrollo de software	49
1.7.7.1	Lenguaje Unificado de Modelado	50
1.7.8	Metodologías de desarrollo de software	50
1.7.8.1	Racional Unified Process (RUP)	51
1.7.8.2	Extreme Programing (XP)	53
1.7.9	Herramientas CASE de Modelado con UML	55
1.7.9.1	Rational Rose	56
1.7.9.2	Enterprise Architect	57
1.7.10	Editores PHP	58
1.7.10.1	NuSphere PHPEd	58
1.7.10.2	Zend Studio	58
1.7.11	Smarty	59
<b>1.8</b>	<b>Conclusiones</b>	<b>60</b>
<b>Capítulo 2: Características del sistema</b>		<b>62</b>
<b>2.1</b>	<b>Introducción</b>	<b>62</b>
<b>2.2</b>	<b>Objeto de estudio</b>	<b>62</b>
2.2.1	Objetivos estratégicos de la organización	62
2.2.2	Flujo actual de los procesos de negocio	64
2.2.2.1	Profesor de Educación Física	64
2.2.2.2	Técnico del gimnasio	64
2.2.2.3	Entrenador	65
2.2.2.4	Organizador de competencia	65
2.2.3	Análisis crítico de la ejecución de los procesos	65
2.2.4	Procesos que serán objeto de automatización	66
2.2.4.1	Control de las competencias deportivas realizadas	67
2.2.4.2	Control de participantes en competencias deportivas	67
2.2.4.3	Control de los enfrentamientos que se realizan y sus resultados	68
2.2.4.4	Confeción de la tabla de posiciones de una competencia deportiva	68
2.2.4.5	Control de los préstamos de utensilios deportivos	68
2.2.4.6	Control de las visitas al gimnasio	68
2.2.4.7	Control de los datos físicos y personales del atleta	68
2.2.4.8	Control de los datos de un atleta en competencias	69
2.2.5	Sistemas automatizados similares existentes vinculados al campo de acción	69
2.2.6	Información que se maneja	70

<b>2.3 Propuesta de sistema.</b>	<b>70</b>
2.3.1 Descripción general de la propuesta de sistema.	70
Fig. 2.3.1.1 Propuesta de sistema	70
2.3.1.1 Módulo de Gestión de competencia.	71
2.3.1.2 Módulo de Gestión de atleta.	71
2.3.1.3 Módulo de pruebas de eficiencia física.	71
2.3.1.4 Módulo de Gestión de gimnasio.	71
2.3.1.5 Módulo de administración.	72
2.3.2 Análisis comparativo de otras soluciones existentes con la propuesta.	72
<b>2.4 Modelo de negocio.</b>	<b>73</b>
2.4.1 Modelo del negocio actual.	73
2.4.2 Reglas del negocio a considerar.	75
2.4.2.1 Reglas de Restricción.	75
2.4.2.2 Reglas del Modelo de Datos.	76
2.4.2.3 Reglas de Derivación.	76
2.4.3 Actores del negocio.	76
Tabla 2.4.3.1 Actores del negocio.	77
2.4.4 Diagrama de casos de uso del negocio.	77
Fig. 2.4.4.1 Diagrama de casos de uso del negocio.	78
2.4.5 Trabajadores del negocio.	78
Tabla 2.4.5.1 Trabajadores del negocio.	79
2.4.6 Especificación de los Casos de Uso del Negocio.	79
Tabla 2.4.6.1 Descripción textual del caso de uso "Solicitar préstamo de utensilio deportivo".	81
Fig. 2.4.6.1 Diagrama de actividades del caso de uso "Solicitar préstamo de utensilio deportivo".	82
2.4.7 Diagrama de clases del modelo de objetos.	82
Fig. 2.4.7.1 Diagrama de clases del Modelo de Objetos.	83
<b>2.5 Definición de los requisitos funcionales.</b>	<b>83</b>
RF 1. Gestionar Prueba de Eficiencia Física.	83
RF 2. Gestionar Atleta.	84
RF 3. Gestionar préstamo de utensilio deportivo.	84
RF 4. Gestionar visita al gimnasio.	85
RF 5. Gestionar competencia deportiva.	85
RF 6. Gestionar representación de competencia deportiva.	85
RF 7. Gestionar equipo de competencia deportiva.	86
RF 8. Gestionar enfrentamiento en competencia deportiva.	86
RF 9. Gestionar resultado de enfrentamiento en competencia deportiva.	86
RF 10. Generar tabla de posiciones de competencia deportiva.	86
RF 11. Gestionar lesión de atleta.	87
RF 12. Gestionar categoría de competencia.	87
RF 13. Gestionar fase de competencia.	87
RF 14. Gestionar sistema de competencia.	87
RF 15. Gestionar tipo de enfrentamiento.	87
RF 16. Gestionar unidad de resultado en enfrentamiento.	88
RF 17. Gestionar deporte.	88
RF 18. Gestionar indicador de eficiencia física.	88

RF 19. Gestionar utensilio deportivo.....	88
RF 20. Gestionar localización de lesión.....	88
RF 21. Gestionar tipo de lesión.....	89
RF 22. Gestionar rol de usuario.....	89
<b>2.6 Definición de los requisitos no funcionales.....</b>	<b>89</b>
2.6.1 Software.....	89
2.6.2 Hardware.....	89
2.6.3 Apariencia o interfaz externa.....	90
2.6.4 Usabilidad.....	90
2.6.5 Rendimiento.....	90
2.6.6 Soporte.....	90
2.6.7 Portabilidad.....	90
2.6.8 Seguridad.....	91
2.6.9 Legales.....	91
2.6.10 Ayuda y documentación en línea.....	91
2.6.11 Restricciones en el diseño y la implementación.....	91
<b>2.7 Modelo de sistema.....</b>	<b>91</b>
2.7.1 Actores del sistema.....	91
Tabla 2.7.1.1 Actores del sistema.....	92
2.7.2 Diagrama de casos de uso del sistema.....	92
Fig. 2.7.2.1 Diagrama de caso de uso del sistema del módulo Gestión de atleta.....	93
Fig. 2.7.2.2 Diagrama de caso de uso del sistema del módulo de Nivelación de pruebas de eficiencia física.....	93
Fig. 2.7.2.3 Diagrama de caso de uso del sistema del módulo de Administración.....	94
Fig. 2.7.2.4 Diagrama de caso de uso del sistema del módulo Gestión de Gimnasio.....	94
Diagrama de caso de uso del sistema del módulo Gestión de competencia.....	95
Fig. 2.7.2.5 Diagrama de caso de uso del sistema del módulo Gestión de competencia.....	95
2.7.3 Descripción de los casos de uso del sistema.....	95
Tabla 2.7.3.1 Descripción textual del CUS “Gestionar atleta”.....	96
<b>2.8 Conclusiones.....</b>	<b>96</b>
<b>Capítulo 3: Análisis y Diseño del sistema.....</b>	<b>98</b>
<b>3.1 Introducción.....</b>	<b>98</b>
<b>3.2 Análisis.....</b>	<b>98</b>
3.2.1 Diagramas de clases del análisis.....	98
Fig. 3.2.1.1 Diagrama de clases del análisis del CUS “Gestionar prueba de eficiencia física”.....	99
<b>3.3 Diseño.....</b>	<b>99</b>
3.3.1 Diagrama de clases del Diseño.....	99
3.3.1.1 Diagrama de clases Web para el caso de uso “Gestionar competencia”.....	100
Fig. 3.3.1.1 Diagrama de clases del diseño del caso de uso “Gestionar competencia” escenario “Crear competencia”.....	100
Fig. 3.3.1.2 Diagrama de clases del diseño del caso de uso “Gestionar competencia” escenario “Modificar competencia”.....	101
3.3.2 Descripción textual de las clases Web.....	101

3.3.2.1 Clases Controladoras .....	101
Tabla 3.3.2.1.1 Clase controladora “Catleta_Manager” .....	102
3.3.2.2 Clases Acceso a Datos.....	102
Tabla 3.3.2.2.1 Clase de acceso a datos “CAcceso_Datos” .....	104
3.3.2.3 Clases Entidades .....	104
Tabla 3.3.2.3.1 Clase entidad “Catleta” .....	104
3.3.3 Diagrama de clases persistentes.....	105
Fig. 3.3.3.1 Diagrama de clases persistentes .....	105
3.3.4 Modelo de datos. ....	105
3.3.4.1 Descripción de las tablas de la base de datos. ....	106
3.3.4.1.1 Descripción de la tabla de base de datos “tb_atleta” .....	106
3.3.5 Principios de diseño .....	106
3.3.6 Principios de diseño de interfaz. ....	108
3.3.7 Interfaz de usuario .....	109
Fig. 3.3.7.1 Interfaz de usuario .....	109
3.3.8 Formato de reportes .....	110
3.3.9 Ayuda .....	110
3.3.10 Tratamiento de errores. ....	110
3.3.11 Seguridad .....	110
<b>3.4 Conclusiones .....</b>	<b>111</b>
<b>Capítulo 4: Implementación y prueba del sistema. ....</b>	<b>112</b>
<b>4.1 Introducción .....</b>	<b>112</b>
<b>4.2 Implementación .....</b>	<b>112</b>
4.2.1 Diagrama de componentes .....	112
4.2.1.1 Diagrama de componentes del sistema y la Base de Datos. ....	112
Fig. 4.2.1.1.1 Diagrama de compontes del sistema y la Base de Datos. ....	113
4.2.1.2 Diagrama de componentes del sistema. ....	113
Fig. 4.2.1.2.1 Diagrama de componentes del sistema. ....	113
Fig. 4.2.1.2.2 Diagrama de componentes del caso de uso “Gestionar préstamo de utensilio deportivo”. ....	114
4.2.2 Diagrama de despliegue.....	114
Fig. 4.2.2.1 Diagrama de despliegue del sistema.....	115
<b>4.3 Prueba.....</b>	<b>115</b>
4.3.1 Prueba de caja negra.....	115
Tabla 4.3.1.1 Pruebas de Caja Negra para el Caso de uso “Autenticarse” .....	117
<b>4.4 Conclusiones .....</b>	<b>117</b>
<b>Conclusiones.....</b>	<b>119</b>
<b>Recomendaciones. ....</b>	<b>120</b>
<b>Referencias Bibliográficas .....</b>	<b>121</b>
<b>Bibliografía .....</b>	<b>124</b>

<b>Anexos .....</b>	<b>125</b>
<b>Anexo #1: Fases y flujos de trabajo de la metodología de desarrollo RUP.....</b>	<b>125</b>
Fig. A1.1 Rational Unified Process.....	125
<b>Anexo # 2: Descripción textual y diagrama de actividades de los casos de uso del negocio.....</b>	<b>125</b>
Tabla A2.1 Descripción textual del CUN “Visitar el gimnasio” .....	126
Fig. A2.1 Diagrama de actividades del caso de uso del negocio “Visitar el gimnasio” .....	127
Tabla A2.2 Descripción textual del caso de uso del negocio “Realizar prueba de eficiencia física”.....	128
Fig. A2.2 Diagrama de actividades del caso de uso del negocio “Realizar prueba de eficiencia física” .....	129
Tabla A2.3 Descripción textual del caso de uso del negocio “Competir en un enfrentamiento deportivo”.....	130
Fig. A2.3 Diagrama de actividades de los casos de uso “Competir en un enfrentamiento deportivo”.....	130
<b>Anexo # 3: Descripción textual en formato expandido de los casos de usos del sistema. ....</b>	<b>131</b>
Tabla A3.1 Descripción del caso de uso del sistema “Nivelar prueba de eficiencia física” .....	136
Tabla A3.2 Descripción del caso de uso del sistema “Gestionar atleta”.....	139
Tabla A3.3 Descripción del caso de uso del sistema “Gestionar lesión de atleta” .....	142
Tabla A3.4 Descripción del caso de uso del sistema “Gestionar préstamo de utensilio deportivo” .....	147
Tabla A3.5 Descripción del caso de uso del sistema “Gestionar visita al gimnasio” .....	151
Tabla A3.6 Descripción del caso de uso del sistema “Gestionar competencia deportiva”.....	157
Tabla A3.7 Descripción del caso de uso del sistema “Gestionar participantes en competencias deportivas” ..	164
Tabla A3.8 Descripción del caso de uso del sistema “Gestionar enfrentamiento de competencia deportiva”	169
Tabla A3.9 Descripción del caso de uso del sistema “Gestionar resultado de enfrentamiento de una competencia deportiva”.....	173
Tabla A3.10 Descripción del caso de uso del sistema “Generar tabla de posiciones de una competencia deportiva” .....	174
Tabla A3.11 Descripción del caso de uso del sistema “Gestionar categoría de competencia” .....	177
Tabla A3.12 Descripción del caso de uso del sistema “Gestionar fase de competencia” .....	179
Tabla A3.13 Descripción del caso de uso del sistema “Gestionar sistema de competencia” .....	182
Tabla A3.14 Descripción del caso de uso del sistema “Gestionar tipo de enfrentamiento” .....	185
Tabla A3.15 Descripción del caso de uso del sistema “Gestionar unidad de resultado de enfrentamiento” ...	188
Tabla A3.16 Descripción del caso de uso del sistema “Gestionar color de la piel” .....	190
Tabla A3.17 Descripción del caso de uso del sistema “Gestionar deporte” .....	193
Tabla A3.18 Descripción del caso de uso del sistema “Gestionar localización de lesión” .....	195
Tabla A3.19 Descripción del caso de uso del sistema “Gestionar tipo de lesión” .....	198
Tabla A3.20 Descripción del caso de uso del sistema “Gestionar utensilio deportivo” .....	200
Tabla A3.21 Descripción del caso de uso del sistema “Gestionar rol” .....	206
<b>Anexo # 4: Diagramas de clases del análisis.....</b>	<b>207</b>
Fig. A4.1 Diagrama de clases del análisis del CUS “Generar tabla posiciones competencia”.....	207
Fig. A4.2 Diagrama de clases de análisis del caso de uso “Gestiona prueba eficiencia física”. .....	208
Fig. A4.3 Diagrama de clases de análisis del caso de uso “Gestiona tipo enfrentamiento” .....	208
Fig. A4.4 Diagrama de clases de análisis del caso de uso “Gestionar atleta”. .....	209
Fig. A4.5 Diagrama de clases de análisis del caso de uso “Gestionar categoría competencia” .....	209
Fig. A4.6 Diagrama de clases de análisis del caso de uso “Gestionar color piel”.....	210
Fig. A4.7 Diagrama de clases de análisis del caso de uso “Gestionar competencia”.....	210
Fig. A4.8 Diagrama de clases de análisis del caso de uso “Gestionar deporte” .....	211
Fig. A4.9 Diagrama de clases de análisis del caso de uso “Gestionar enfrentamiento competencia” .....	211

Fig. A4.10 Diagrama de clases de análisis del caso de uso “Gestionar fase competencia”	212
Fig. A4.11 Diagrama de clases de análisis del caso de uso “Gestionar indicador eficiencia física”	212
Fig. A4.12 Diagrama de clases de análisis del caso de uso “Gestionar lesión atleta”	213
Fig. A4.13 Diagrama de clases de análisis del caso de uso “Gestionar localización lesión”	213
Fig. A4.14 Diagrama de clases de análisis del caso de uso “Gestionar participantes competencia”	214
Fig. A4.15 Diagrama de clases de análisis del caso de uso “Gestionar préstamo utensilio”	214
Fig. A4.16 Diagrama de clases de análisis del caso de uso “Gestionar rol”	215
Fig. A4.17 Diagrama de clases de análisis del caso de uso “Gestionar sistema competencia”	215
Fig. A4.18 Diagrama de clases de análisis del caso de uso “Gestionar tipo deporte”	216
Fig. A4.19 Diagrama de clases de análisis del caso de uso “Gestionar tipo lesión”	216
Fig. A4.20 Diagrama de clases de análisis del caso de uso “Gestionar unidad resultado”	217
Fig. A4.21 Diagrama de clases de análisis del caso de uso “Gestionar utensilio”	217
Fig. A4.22 Diagrama de clases de análisis del caso de uso “Gestionar visita gimnasio”	218

**Anexo # 5: Diagramas de clases Web del diseño.....219**

Fig. A5.1 Diagrama de clases del diseño para el caso de uso “Gestionar competencia” escenario “Crear competencia”	219
Fig. A5.2 Diagrama de clases del diseño para el caso de uso “Gestionar competencia” escenario “Modificar competencia”	220
Fig. A5.3 Diagrama de clases del diseño para el caso de uso “Gestionar enfrentamiento competencia” escenario “Gestionar enfrentamiento”	220
Fig. A5.4 Diagrama de clases del diseño para el caso de uso “Gestionar enfrentamiento competencia” escenario “Gestionar resultado enfrentamiento”	221
Fig. A5.5 Diagrama de clases del diseño para el caso de uso “Generar tabla posiciones competencia”	221
Fig. A5.6 Diagrama de clases del diseño para el caso de uso “Gestionar atleta”	222
Fig. A5.7 Diagrama de clases del diseño para el caso de uso “Gestionar categoría competencia”	222
Fig. A5.8 Diagrama de clases del diseño para el caso de uso “Gestionar color piel”	223
Fig. A5.9 Diagrama de clases del diseño para el caso de uso “Gestionar deporte”	223
Fig. A5.10 Diagrama de clases del diseño para el caso de uso “Gestionar fase competencia”	224
Fig. A5.11 Diagrama de clases del diseño para el caso de uso “Gestionar lesión atleta”	224
Fig. A5.12 Diagrama de clases del diseño para el caso de uso “Gestionar localización lesión”	225
Fig. A5.13 Diagrama de clases del diseño para el caso de uso “Gestionar participantes competencia”	225
Fig. A5.14 Diagrama de clases del diseño para el caso de uso “Gestionar préstamo utensilio”	226
Fig. A5.15 Diagrama de clases del diseño para el caso de uso “Gestionar prueba eficiencia”	226
Fig. A5.16 Diagrama de clases del diseño para el caso de uso “Gestionar rol”	227
Fig. A5.17 Diagrama de clases del diseño para el caso de uso “Gestionar sistema competencia”	227
Fig. A5.18 Diagrama de clases del diseño para el caso de uso “Gestionar tipo enfrentamiento”	228
Fig. A5.19 Diagrama de clases del diseño para el caso de uso “Gestionar tipo lesión”	228
Fig. A5.20 Diagrama de clases del diseño para el caso de uso “Gestionar unidad resultado”	229
Fig. A5.21 Diagrama de clases del diseño para el caso de uso “Gestionar utensilio”	229
Fig. A5.22 Diagrama de clases del diseño para el caso de uso “Gestionar visita gimnasio”	230
Fig. A5.23 Diagrama de clases del subsistema de Acceso a Datos	230

**Anexo # 6: Descripción textual de las clases Web.....231**

6.1 Clases entidades.....	231
Tabla A6.1 Clase entidad “Catleta”	231
Tabla A6.2 Clase entidad “Ccategoria_competencia”	232

Tabla A6.3	Clase entidad "Ccolor_piel"	232
Tabla A6.4	Clase entidad "Ccomision"	232
Tabla A6.5	Clase entidad "Ccomision_organizadora"	233
Tabla A6.6	Clase entidad "Ccompetencia"	234
Tabla A6.7	Clase entidad "Ccurso"	234
Tabla A6.8	Clase entidad "Cdeporte"	235
Tabla A6.9	Clase entidad "Cenfrentamiento"	235
Tabla A6.10	Clase entidad "Cenfrentamiento_duo_atleta"	236
Tabla A6.11	Clase entidad "Cenfrentamiento_duo_equipo"	236
Tabla A6.12	Clase entidad "Cenfrentamiento_multiple_atleta"	237
Tabla A6.13	Clase entidad "Cequipo"	237
Tabla A6.14	Clase entidad "Cestudiante"	237
Tabla A6.15	Clase entidad "Cevento"	238
Tabla A6.16	Clase entidad "Cfase_competencia"	238
Tabla A6.17	Clase entidad "Cgrupo"	239
Tabla A6.18	Clase entidad "Cindicador"	239
Tabla A6.19	Clase entidad "Clesion"	240
Tabla A6.20	Clase entidad "Clocalizacion_lesion"	240
Tabla A6.21	Clase entidad "Cnivel_general_est"	240
Tabla A6.22	Clase entidad "Cnivel_general_grupo"	241
Tabla A6.23	Clase entidad "Cnivel_grupo_indicador"	241
Tabla A6.24	Clase entidad "Cpersona"	242
Tabla A6.25	Clase entidad "Cprestamo"	243
Tabla A6.26	Clase entidad "Cprofesor_ef"	243
Tabla A6.27	Clase entidad "Crepresentacion"	244
Tabla A6.28	Clase entidad "Cresultado"	244
Tabla A6.29	Clase entidad "Croll"	245
Tabla A6.30	Clase entidad "Csistema_competencia"	245
Tabla A6.31	Clase entidad "Ctipo_deporte"	245
Tabla A6.32	Clase entidad "Ctipo_enfrentamiento"	246
Tabla A6.33	Clase entidad "Ctipo_lesion"	246
Tabla A6.34	Clase entidad "Cunidad_resultado_enf"	247
Tabla A6.35	Clase entidad "Cusuario"	247
Tabla A6.36	Clase entidad "Cutensilio"	247
Tabla A6.37	Clase entidad "Cvisita"	248
6.2	Clases de acceso a datos	248
Tabla A6.38	Clase de acceso a datos "CAcceso_Datos"	249
Tabla A6.39	Clase de acceso a datos "Catleta_DAO"	249
Tabla A6.40	Clase de acceso a datos "Ccategoria_competencia_DAO"	250
Tabla A6.41	Clase de acceso a datos "Ccomision_DAO"	250
Tabla A6.42	Clase de acceso a datos "Ccomision_organizadora_DAO"	250
Tabla A6.43	Clase de acceso a datos "Ccompetencia_DAO"	251
Tabla A6.44	Clase de acceso a datos "Cdeporte_DAO"	251
Tabla A6.45	Clase de acceso a datos "Cenfrentamiento_DAO"	252
Tabla A6.46	Clase de acceso a datos "Cenfrentamiento_duo_atleta_DAO"	252
Tabla A6.47	Clase de acceso a datos "Cenfrentamiento_duo_equipo_DAO"	252
Tabla A6.48	Clase de acceso a datos "Cenfrentamiento_multiple_atleta_DAO"	253

Tabla A6.49	Clase de acceso a datos "Cequipo_DAO" .....	253
Tabla A6.50	Clase de acceso a datos "Cestudiante_DAO" .....	254
Tabla A6.51	Clase de acceso a datos "Cevento_DAO" .....	254
Tabla A6.52	Clase de acceso a datos "Cfase_competencia_DAO" .....	254
Tabla A6.53	Clase de acceso a datos "Cgrupo_DAO" .....	255
Tabla A6.54	Clase de acceso a datos "Cindicador_DAO" .....	255
Tabla A6.55	Clase de acceso a datos "Clesion_DAO" .....	256
Tabla A6.56	Clase de acceso a datos "Clocalizacion_lesion_DAO" .....	256
Tabla A6.57	Clase de acceso a datos "Cnivel_general_est_DAO" .....	256
Tabla A6.58	Clase de acceso a datos "Cnivel_general_grupo_DAO" .....	257
Tabla A6.59	Clase de acceso a datos "Cnivel_grupo_indicador_DAO" .....	257
Tabla A6.60	Clase de acceso a datos "Cnormativa_DAO" .....	257
Tabla A6.61	Clase de acceso a datos "Cpersona_DAO" .....	258
Tabla A6.62	Clase de acceso a datos "Cprestamo_DAO" .....	258
Tabla A6.63	Clase de acceso a datos "Cprofesor_ef_DAO" .....	259
Tabla A6.64	Clase de acceso a datos "Cprueba_DAO" .....	259
Tabla A6.65	Clase de acceso a datos "Crepresentacion_DAO" .....	260
Tabla A6.66	Clase de acceso a datos "Cresultado_DAO" .....	260
Tabla A6.67	Clase de acceso a datos "Croll_DAO" .....	260
Tabla A6.68	Clase de acceso a datos "Csistema_competencia_DAO" .....	261
Tabla A6.69	Clase de acceso a datos "Ctipo_deporte_DAO" .....	261
Tabla A6.70	Clase de acceso a datos "Ctipo_enfrentamiento_DAO" .....	261
Tabla A6.71	Clase de acceso a datos "Ctipo_lesion_DAO" .....	262
Tabla A6.72	Clase de acceso a datos "Cunidad_resultado_enf_DAO" .....	262
Tabla A6.73	Clase de acceso a datos "Cusuario_DAO" .....	263
Tabla A6.74	Clase de acceso a datos "Cutensilio_DAO" .....	263
Tabla A6.75	Clase de acceso a datos "Cvisita_DAO" .....	263
6.3	Clase Fachada .....	264
Tabla A6.76	Clase de acceso a datos "CFachada" .....	271
6.4	Clases controladoras .....	271
Tabla A6.77	Clase controladora "Catleta_Manager" .....	271
Tabla A6.78	Clase controladora "Ccategoria_competencia_Manager" .....	271
Tabla A6.79	Clase controladora "Ccompetencia_Manager" .....	272
Tabla A6.80	Clase controladora "Cdeporte_Manager" .....	272
Tabla A6.81	Clase controladora "Ccomision_Manager" .....	273
Tabla A6.82	Clase controladora "Ccomision_organizadora_Manager" .....	273
Tabla A6.83	Clase controladora "Cenfrentamiento_duo_atleta_Manager" .....	273
Tabla A6.84	Clase controladora "Cenfrentamiento_duo_equipo_Manager" .....	274
Tabla A6.85	Clase controladora "Cenfrentamiento_Manager" .....	274
Tabla A6.86	Clase controladora "Cenfrentamiento_multiple_atleta_Manager" .....	275
Tabla A6.87	Clase controladora "Cequipo_Manager" .....	275
Tabla A6.88	Clase controladora "Cestudiante_Manager" .....	275
Tabla A6.89	Clase controladora "Cevento_Manager" .....	276
Tabla A6.90	Clase controladora "Cfase_competencia_Manager" .....	276
Tabla A6.91	Clase controladora "Cgrupo_Manager" .....	277
Tabla A6.92	Clase controladora "Cindicador_Manager" .....	277
Tabla A6.93	Clase controladora "Clesion_Manager" .....	277

Tabla A6.94 Clase controladora “Clocalizacion_lesion_Manager” .....	278
Tabla A6.95 Clase controladora “Cnivel_general_est_Manager” .....	278
Tabla A6.96 Clase controladora “Cnormativa_Manager” .....	279
Tabla A6.97 Clase controladora “Cpersona_Manager” .....	279
Tabla A6.98 Clase controladora “Cprestamo_Manager” .....	280
Tabla A6.99 Clase controladora “Cprofesor_ef_Manager” .....	280
Tabla A6.100 Clase controladora “Cusuario_Manager” .....	281
Tabla A6.101 Clase controladora “Cutensilio_Manager” .....	281

**Anexo # 7: Diseño de la base de datos.....282**

Fig. A7.1 Diseño de la base de datos del sistema SGD-UCI.....	282
---	-----

**Anexo # 8: Descripción textual de las tablas de la base de datos. ....283**

Tabla A8.1 Descripción textual de la tabla “cl_categoria_competencia” .....	283
Tabla A8.2 Descripción textual de la tabla “cl_tipo_deporte” .....	283
Tabla A8.3 Descripción textual de la tabla “cl_tipo_enfrentamiento” .....	283
Tabla A8.4 Descripción textual de la tabla “cl_tipo_lesion” .....	284
Tabla A8.5 Descripción textual de la tabla “no_fase” .....	284
Tabla A8.6 Descripción textual de la tabla “tb_atleta” .....	284
Tabla A8.7 Descripción textual de la tabla “no_indicador_ef” .....	284
Tabla A8.8 Descripción textual de la tabla “no_localizacion_lesion” .....	285
Tabla A8.9 Descripción textual de la tabla “no_rol” .....	285
Tabla A8.10 Descripción textual de la tabla “no_sexo” .....	285
Tabla A8.11 Descripción textual de la tabla “no_sistema_competencia” .....	286
Tabla A8.12 Descripción textual de la tabla “no_unidad_resultado” .....	286
Tabla A8.13 Descripción textual de la tabla “tb_comision” .....	286
Tabla A8.14 Descripción textual de la tabla “tb_comision_organizadora” .....	287
Tabla A8.15 Descripción textual de la tabla “tb_comision_tecnica” .....	287
Tabla A8.16 Descripción textual de la tabla “tb_competencia” .....	288
Tabla A8.17 Descripción textual de la tabla “tb_deporte” .....	288
Tabla A8.18 Descripción textual de la tabla “tb_deporte_competencia” .....	288
Tabla A8.19 Descripción textual de la tabla “tb_determinar_niveles” .....	289
Tabla A8.20 Descripción textual de la tabla “tb_normativa” .....	289
Tabla A8.21 Descripción textual de la tabla “tb_enfrentamiento” .....	290
Tabla A8.22 Descripción textual de la tabla “tb_enfrentamiento_duo_atleta” .....	290
Tabla A8.23 Descripción textual de la tabla “tb_enfrentamiento_duo_equipo” .....	291
Tabla A8.24 Descripción textual de la tabla “tb_enfrentamiento_multiple_atleta” .....	291
Tabla A8.25 Descripción textual de la tabla “tb_equipo” .....	292
Tabla A8.26 Descripción textual de la tabla “tb_estudinate” .....	292
Tabla A8.27 Descripción textual de la tabla “tb_evento” .....	293
Tabla A8.28 Descripción textual de la tabla “tb_grupo” .....	293
Tabla A8.29 Descripción textual de la tabla “tb_lesion” .....	293
Tabla A8.30 Descripción textual de la tabla “tb_nivel_general_est” .....	294
Tabla A8.31 Descripción textual de la tabla “tb_persona” .....	295
Tabla A8.32 Descripción textual de la tabla “tb_prestamo” .....	296
Tabla A8.33 Descripción textual de la tabla “tb_profesor_ef” .....	296
Tabla A8.34 Descripción textual de la tabla “tb_prueba_ef” .....	296

Tabla A8.35	Descripción textual de la tabla “tb_representacion” .....	297
Tabla A8.36	Descripción textual de la tabla “tb_resultado” .....	297
Tabla A8.37	Descripción textual de la tabla “tb_usuario” .....	298
Tabla A8.38	Descripción textual de la tabla “tb_usuario_roll” .....	298
Tabla A8.39	Descripción textual de la tabla “tb_utensilio” .....	298
Tabla A8.40	Descripción textual de la tabla “tb_visita” .....	299
<b>Anexo # 9: Diagrama de componentes de los casos de uso del sistema. ....</b>		<b>299</b>
Fig. A9.1	Diagrama de componente del caso de uso “Gestionar deporte” .....	299
Fig. A9.2	Diagrama de componentes del caso de uso “Gestionar préstamo utensilio”. .....	300
Fig. A9.3	Diagrama de componentes del caso de uso “Gestionar prueba”. .....	300
Fig. A9.4	Diagrama de componentes del caso de uso “Gestionar rol”. .....	301
Fig. A9.5	Diagrama de componentes del caso de uso “Gestionar utensilio” .....	301
Fig. A9.6	Diagrama de componentes del caso de uso “Gestionar visita”. .....	302
<b>Anexo # 10: Prueba de caja negra de los casos de uso del sistema.....</b>		<b>302</b>
Tabla A10.1	Pruebas de Caja Negra para el Caso de uso “Gestionar deporte” escenario “Entrar deporte” .....	303
Tabla A10.2	Pruebas de Caja Negra para el Caso de uso “Gestionar deporte” escenario “Eliminar deporte”. .....	305
Tabla A10.3	Pruebas de Caja Negra para el Caso de uso “Gestionar deporte” escenario “Entrar tipo de deporte”. .....	306
Tabla A10.4	Pruebas de Caja Negra para el Caso de uso “Gestionar deporte” escenario “Modificar tipo de deporte”. .....	307
Tabla A10.5	Pruebas de Caja Negra para el Caso de uso “Gestionar deporte” escenario “Eliminar tipo de deporte”. .....	308
Tabla A10.6	Pruebas de Caja Negra para el Caso de uso “Gestionar utensilio” escenario “Entrar utensilio” .....	309
Tabla A10.7	Pruebas de Caja Negra para el Caso de uso “Gestionar utensilio” escenario “Modificar utensilio”. .....	310
Tabla A10.8	Pruebas de Caja Negra para el Caso de uso “Gestionar utensilio” escenario “Eliminar utensilio”. .....	311
Tabla A10.9	Pruebas de Caja Negra para el Caso de uso “Gestionar préstamo de utensilio” escenario “Crear préstamo”. .....	312
Tabla A10.10	Pruebas de Caja Negra para el Caso de uso “Gestionar préstamo de utensilio” escenario “Terminar préstamo”. .....	314
Tabla A10.11	Pruebas de Caja Negra para el Caso de uso “Gestionar préstamo de utensilio” escenario “Modificar préstamo”. .....	315
Tabla A10.12	Pruebas de Caja Negra para el Caso de uso “Gestionar préstamo de utensilio” escenario “Reporte de préstamo por fecha”. .....	316
Tabla A10.13	Pruebas de Caja Negra para el Caso de uso “Gestionar préstamo de utensilio” escenario “Reporte de préstamo por usuario”. .....	317
Tabla A10.14	Pruebas de Caja Negra para el Caso de uso “Gestionar préstamo de utensilio” escenario “Reporte de préstamo pendientes”. .....	319
Tabla A10.15	Pruebas de Caja Negra para el Caso de uso “Gestionar préstamo de utensilio” escenario “Reporte de utensilio por deporte”. .....	320
Tabla A10.16	Pruebas de Caja Negra para el Caso de uso “Gestionar préstamo de utensilio” escenario “Reporte de utensilio más prestados”. .....	321
Tabla A10.17	Pruebas de Caja Negra para el Caso de uso “Gestionar visita al gimnasio” escenario “Registrar visita”. .....	322
Tabla A10.18	Pruebas de Caja Negra para el Caso de uso “Gestionar visita al gimnasio” escenario “Registrar salida”. .....	323

Tabla A10.19 Pruebas de Caja Negra para el Caso de uso “Gestionar visita al gimnasio” escenario “Reporte de visita por fecha” .....	324
Tabla A10.20 Pruebas de Caja Negra para el Caso de uso “Gestionar visita al gimnasio” escenario “Reporte de visita de los usuarios más visitantes” .....	325
Tabla A10.21 Pruebas de Caja Negra para el Caso de uso “Gestionar prueba de eficiencia física” escenario “Crear prueba de eficiencia física” .....	327
Tabla A10.22 Pruebas de Caja Negra para el Caso de uso “Gestionar prueba de eficiencia física” escenario “Modificar prueba de eficiencia física” .....	329
Tabla A10.23 Pruebas de Caja Negra para el Caso de uso “Gestionar prueba de eficiencia física” escenario “Reportes de nivelación” .....	330

## INTRODUCCIÓN

El deporte es uno de los fenómenos más importante de la sociedad contemporánea. Tanto como práctica activa, o como espectáculo de masas, ha pasado a formar parte esencial de la vida y la cultura del hombre moderno. El deporte educativo y la enseñanza de la Educación Física despierta un elevado interés y motivación en los estudiantes, ya que conforma un medio pedagógico de elevados valores educativos.

El deporte universitario constituye una actividad importante en el quehacer estudiantil en la actualidad mundial, donde se aglutina un conjunto de actividades físicas, recreativas y deportivas, que deben desarrollarse dentro de un orden, un sistema y un programa establecido que aseguran la autodisciplina y el optimismo, el fortalecimiento del carácter y la voluntad de la entrega responsable en el trabajo, la solidaridad en la comunidad universitaria, el orgullo y el tesón por el ejercicio, en cualquiera de sus expresiones. Esta práctica no sólo es competitiva, sino que frecuentemente se dirige hacia la realización de actividades de recreación deportiva para emplear el tiempo de ocio de forma saludable.

El deporte universitario en el mundo está dirigido de esta forma:

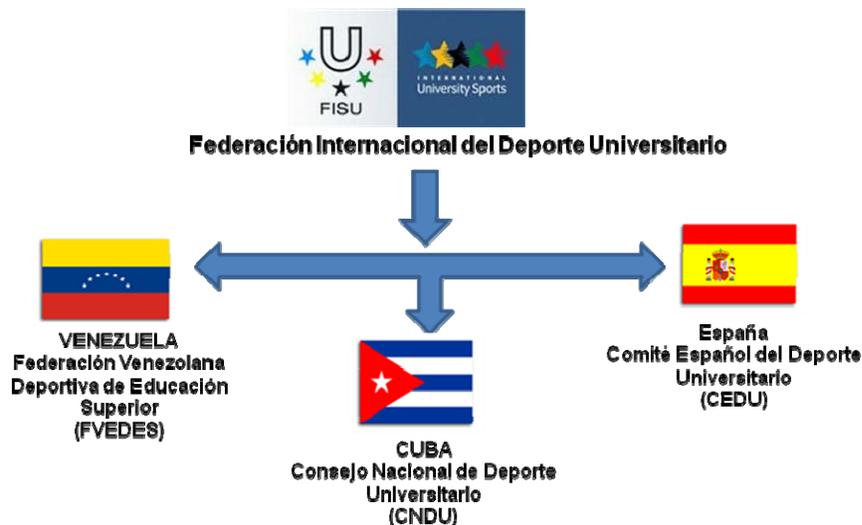


Fig. I.1 Organización del deporte universitario en el mundo

La Revolución Cubana desde sus inicios mantuvo una estratégica preocupación por el desarrollo de la cultura física y el deporte, como manifestación plena del mejoramiento físico y espiritual del pueblo

cubano. Esa estrategia ha permitido cambiar la situación imperante antes de 1959 y crear un sistema de cultura física y de deporte masivo. El 23 de Febrero de 1961 fue dictada la Ley 936 por acuerdo del Consejo de Ministros, creando así el Instituto Nacional de Deportes, Educación Física y Recreación (INDER) como rector de la actividad de cultura física y deportes en el país. El organismo tiene a su cargo la preparación competitiva de los atletas, la formación integral de los alumnos, mediante la Educación Física, y la recreación de los trabajadores y pueblo en general, por medio del deporte y la cultura física. (INDER)

El deporte, es una de las actividades de mayor arraigo en las Instituciones de la Enseñanza Superior (IES) en Cuba, caracterizado en la última década por privilegiar el deporte participativo, sin descuidar el competitivo o de alto rendimiento, todo ello sustentado en las clases de Educación Física que reciben los estudiantes de todas las carreras, como parte de su currículum, durante los dos primeros años de estudio en los cursos regulares diurnos. El deporte universitario cubano tiene como sello distintivo, que las competencias deportivas son convocadas por la Federación Estudiantil Universitaria (FEU), mediante una comisión organizadora en la que también participan las autoridades universitarias y los departamentos de deporte, de acuerdo al nivel y envergadura de la competencia.

El sistema competitivo universitario en el país se estructura de la siguiente forma:



**Fig. I.2 Sistema competitivo en Cuba.**

Todo este sistema se coordina por el Consejo Nacional del Deporte Universitario, que está conformado de la siguiente forma:



Fig. I.3 Organización del deporte universitario en Cuba

La Universidad de las Ciencias Informáticas fue creada en el año 2002, dedicada en su totalidad a la formación de Ingenieros en Ciencias Informáticas. La UCI, universidad cubana que inicia una nueva época en la enseñanza superior en nuestro país, ya que tiene como principio del proceso docente educativo, la formación desde la producción, basado en el modelo estudio-trabajo, dando a luz en Cuba al nuevo concepto de universidad productiva.

En la UCI, los estudiantes son los principales protagonistas del desarrollo productivo. Al incorporarse a un proyecto productivo el estudiante adquiere una gran responsabilidad, ya que debe cumplir con las tareas del proyecto sin descuidar la parte docente. Esto provoca gran cantidad de horas de trabajo y estudio, que pueden ser aliviadas con la práctica del deporte y la actividad física. Por todo esto, hace que la práctica del deporte y la actividad física jueguen un papel importante y decisivo en la formación física, intelectual, creativa y educacional del estudiante. En esta universidad a diferencia del resto de las IES, la Educación Física se imparte como asignatura curricular durante los cinco años de la carrera. La UCI, al igual que el resto de las IES organiza una serie de Juegos Deportivos para fomentar el deporte universitario.

En la UCI se desarrollan juegos inter-años, inter-facultades, y de ahí se seleccionan un grupo de atletas de alto rendimiento que conforman las representaciones de la universidad en eventos provinciales y nacionales. Además se desarrolla otra serie de juegos internos como lo constituyen el Mundialito de Fútbol, Mundialito de Baloncesto, entre otros.

La Universidad de las Ciencias Informáticas, es uno de esos centros universitarios en los que la cantidad de información que se maneja en todo momento es muy grande, debido a que la misma se divide en 10 Facultades y cada una de estas a su vez, está compuesta entre 1000 y 2000 estudiantes de todos los años de la carrera, además de contar con un número no pequeño de profesores y trabajadores, que provoca que la información que se necesite obtener de cada una de estas personas y de las actividades que realizan a cada momento, sea considerable, por lo que se requiere de que la misma esté lo más organizada posible para ser procesada de forma eficiente.

Los procesos deportivos en la universidad presentan un grupo de problemas surgidos por la dificultad que tienen los profesores y entrenadores de los diferentes equipos, al manejar una gran cantidad de información sobre los resultados en los juegos deportivos que convoca la universidad, tanto a nivel de equipos como a nivel individual, así como lo difícil y engorroso que se les hace gestionar una competencia deportiva, en cuanto a, equipos y atletas a participar, sistema de competencia, resultados de los enfrentamientos que se desarrolla, etc. Esto unido a la dificultad por parte de los profesores de Educación Física, de registrar y evaluar todos los datos que generan cada una de las pruebas de eficiencia física que se realizan a los estudiantes, teniendo en cuenta que en el curso 2006-2007 la matrícula de la universidad ascendió a 1034 estudiantes. Otras de las dificultades, es que la UCI cuenta con un gimnasio al que acceden todos los estudiantes y trabajadores del centro, en el cual se realizan préstamos de implementos deportivos. Los datos importantes de estos préstamos se recogen de forma manual, y al aumentar la matrícula y así el número de préstamos ha provocado descontrol y pérdida de utensilios deportivos en la instalación.

Toda la información que se genera de los procesos deportivos que se llevan a cabo en la universidad, se maneja de forma manual, asunto que se torna un tanto engorroso, lo que puede llevar a que se vea afectado en determinado momento la toma de decisiones, por no contar con toda la información necesaria o debidamente organizada para ello, además de esto se han identificado los siguientes problemas:

☞ La información se manipula y organiza de forma manual, lo cual provoca que existan demoras a la hora de realizar una búsqueda de la misma, teniendo en cuenta que en muchas ocasiones, la información que se está buscando aparece en más de un documento y en más de un lugar de almacenamiento, por lo que si se desea hacer una recopilación de la información proveniente de diferentes fuentes, el proceso se torna bastante lento.

El gran volumen de información que se manipula en la universidad relacionada con todos aquellos aspectos deportivos, se almacena en muchos casos, en papel, mientras que en otros se almacena en documentos Word y Excel, lo cual puede provocar con el paso del tiempo, pérdidas de información, ya sea por deterioro de los papeles o por la pérdida de algunos de los documentos anteriormente mencionados, debido a la aparición de imprevistos como: virus, problemas en el Sistema Operativo, entre otros, que traigan consigo la pérdida de información.

No existe una centralización de la información de manera que se pueda acceder a esta en todo momento y desde cualquier parte, lo que puede incidir notablemente en demoras a la hora de recopilar información proveniente de diversas personas.

Por ello se parte de un **problema científico**, “¿Cómo erradicar los problemas en los procesos de gestión de competencias, préstamos de utensilios deportivos, nivelación de pruebas de eficiencia física y recopilación de información de los atletas en la Universidad de las Ciencias Informáticas? De aquí que le **objeto de estudio** lo constituye “los procesos de actividad física y deportiva en las universidades”. Todo ello encaminado a cumplir el **objetivo general** de “realizar el análisis y diseño del Sistema de Gestión de la Deportiva para la Universidad de las Ciencias Informáticas y la implementación de los módulos de gestión del gimnasio, nivelación de pruebas de eficiencia física y administración”. El **campo de acción** abarca “el proceso de la actividad física y deportiva en la UCI”.

Por esta razón se plantea para este trabajo como **hipótesis** “sí se realiza un estudio del proceso de actividad física y deportiva en la UCI, el funcionamiento del deporte en la universidades mundiales y cubanas, así como un análisis de los software que gestionan actividad física y deportiva en el mundo y en Cuba se obtendrá un sistema informático que automatice los procesos deportivos en la UCI que responda a las características y necesidades del entorno”.

Los **Objetivos Específicos** que se persiguen son:

- ☞ Informatizar los procesos de préstamos de utensilios que se realizan en el gimnasio de la UCI.
- ☞ Informatizar el control de acceso al gimnasio.
- ☞ Informatizar la nivelación de la pruebas de eficiencia física que se realizan en la UCI.

- 🖥️ Proponer y desarrollar una versión del sistema, que de solución a los problemas existentes.
- 🖥️ Conformar la documentación del sistema con vista al desarrollo de posteriores versiones.

Para darle cumplimiento a los objetivos trazados se ha decidido desarrollar las siguientes **tareas de investigación**:

- 🖥️ Definir la importancia del deporte en las universidades
- 🖥️ Establecer una comparación con otros sistemas similares dentro y fuera del campo de acción del proyecto.
- 🖥️ Realizar un estudio de las tendencias y tecnologías actuales con el objetivo de seleccionar las más apropiadas para la elaboración del sistema.
- 🖥️ Modelar el negocio actual de la actividad deportiva en la UCI.
- 🖥️ Determinar y conformar los requisitos que debe cumplir el sistema.
- 🖥️ Proponer una solución que contenga las funcionalidades principales con el objetivo de eliminar en gran medida, los problemas existentes.
- 🖥️ Realizar el análisis y diseño del sistema propuesto.
- 🖥️ Llevar a cabo la implementación y las pruebas del sistema.

Este trabajo está estructurado en 4 capítulos, en los cuales se tratan las siguientes cuestiones:

🖥️ **Capítulo 1. Fundamentación Teórica:** Este capítulo trata acerca de algunos elementos teóricos necesarios que sirven de soporte para la realización de todo el trabajo en general, tales como: algunos sistemas similares existentes vinculados al campo de acción, además de una descripción y selección de las herramientas a utilizar.

🖥️ **Capítulo 2. Características del Sistema:** En dicho capítulo se realiza una descripción del objeto de estudio, además de que se describen todos los procesos, actores, casos de usos y trabajadores del negocio actual, también se muestra el diagrama de casos de usos del negocio y el modelo de objetos del negocio. Por otro lado, se definen cuáles son los requerimientos funcionales y no funcionales, a la vez que

se presentan los actores del sistema y diagrama de casos de uso del mismo, acompañado de la descripción textual de cada uno de estos.

 **Capítulo 3. Análisis y Diseño del Sistema:** En este capítulo se presentan los diagramas de clases del análisis, lo cual influye notablemente a la hora de concebir el diseño del sistema, además de los diagramas de clases Web, que reflejan de una forma más clara cómo va a funcionar dicho sistema y qué clases están presentes en el mismo, acompañado también de la descripción de cada una de estas.

 **Capítulo 4. Implementación y Prueba:** En este capítulo se describe como está implementado el sistema, a través de los diagramas de componentes y el diagrama de despliegue, además de que se exponen y detallan las diferentes pruebas que se le realizan al mismo.

Al final se podrán observar las conclusiones del trabajo, además de las recomendaciones, referencias bibliográficas y bibliografía citada a lo largo del trabajo. Por último se encuentran los anexos, los cuales contienen información de apoyo a algunos aspectos tratados durante el trabajo.

## Capítulo 1: Fundamentación Teórica

### 1.1 Introducción.

En este capítulo se realiza un análisis de todas las definiciones relacionadas con el deporte, la actividad física, juego, Educación Física, entre otras. Se lleva a cabo un estudio del estado del arte donde incluye, el deporte como fenómeno social, antecedentes históricos del deporte universitario y cómo se practica en la actualidad, las leyes y regulaciones que lo rigen. El deporte cubano antes y después del triunfo de la Revolución en 1959, el deporte universitario cubano, así como se desarrolla y fomenta el deporte en la Universidad de la Ciencias Informáticas. Se exponen algunos softwares deportivos en el mundo y el quehacer de Cuba en esta rama. Por último se detalla algunas tendencias y tecnologías actuales para el desarrollo de aplicaciones y servicios Web.

### 1.2 Definiciones y términos relacionados con el deporte.

La palabra deporte, con la que en la actualidad se define como algo tan claro y que evidencia un fenómeno sociocultural y educativo por todos conocidos, es sin embargo un de las palabras más polémicas, además de ser un término tan fascinante como complejo.

El deporte es un fenómeno que adquiere a diario un gran impacto dentro de las diferentes sociedades. Es una práctica humana tan significativa en la época como pocos fenómenos sociológicos, forma parte de la cultura contemporánea. Puede empezar a entenderse la importancia sociocultural de este término cuando se lee en Cagigal “El deporte es algo que existe; intrínseco a la naturaleza humana; que se manifiesta, que se ha manifestado siempre donde el hombre ha existido. El deporte es una realidad metafísica del hombre. Es decir, que donde quiera que se da el hombre se da el deporte y sólo en el hombre se puede éste concebir”. (Cagigal, 1966)

Según Cagigal y coincidiendo con Piernavieja, la palabra deporte es indudable origen latino: “Según Paul Adam nace de la expresión *de-portare*, referente a la salida fuera de las puertas (*porta*) de la ciudad, para dedicarse a juegos competitivos” (Cagigal, 1981).

La definición del vocablo deporte, se ofrecen en los diccionarios de forma esquematizada, rotunda y definitiva. Es lógico pensar que unas obras de carácter general no detallan exhaustivamente el sentido de

este término, por lo que es necesario realizar un recorrido por las definiciones realizadas por autores, que son autoridades en el tema del deporte y por las definiciones que han sido resultado de estudios o aproximaciones en reuniones o congresos a nivel internacional.

En sus “Memorias olímpicas”, Coubertin expresa con claridad su postura sobre el deporte: “El deporte no es un artículo de lujo, no es una ocupación para ociosos ni una compensación por el trabajo intelectual. El deporte es una fuente de perfeccionamiento interno para cada persona. La profesión no tiene nada que ver con ello. Antes bien, el deporte es un regalo irremplazable que les dado a todas las personas en igual medida. Desde una perspectiva étnica tampoco existe diferencia, ya que, por naturaleza, todas las razas disponen del deporte como de algo propio y en igualdad de derecho”. (Coubertin, 1959)

En la primera Reunión Internacional de Trabajo sobre Terminología del Ejercicio Físico, celebrada en octubre de 1963 en la ciudad de Strobl en Wolfgangse (Austria) se propuso la siguiente definición:

“Deporte es aquella forma de ejercicio físico (corporal) que se caracteriza preferentemente por la aspiración al rendimiento y por una ejecución y valoración convencionalmente acordada o sometida a una norma con vistas a un fin puntual. La práctica de los diferentes deportes, por regla general está siendo atendida por organizaciones especializadas según los deportes que cultivan o según otros criterios sociales o complejos, como ocurre, por ejemplo, en el caso del deporte juvenil, deporte de masas, deporte élite”. (Devís, 1996)

La definición que aportaba el Consejo Internacional de Educación Física y Deportes de la UNESCO, en una reunión celebrada en Bucarest en 1969 era:

“Actividad humana significativa que se manifiesta y se concreta en la práctica de los ejercicios físicos, bajo forma competitiva”. (Apuntes para un historia de la documentación deportiva, 2000)

En 1964 se elabora por el Consejo Internacional de Educación Física y Deportes (C.I.E.P.S.) en colaboración con la UNESCO el *Manifiesto sobre el deporte*. Dicho documento fue sometido al dictamen de los gobiernos y todas las instituciones privadas que se interesaba por el deporte, para darse a conocer en la Conferencia Internacional sobre el Deporte y la Educación que se celebró en México los días 7 al 9 de 1968. Dicho manifiesto consta de tres partes fundamentales en la que se agrupan un conjunto de definiciones del deporte:

- 1) Mensaje del director de la UNESCO en esa época, René Maheu, quien resaltó las siguientes características: “fenómeno social de planetarias dimensiones”, “práctica y espectáculo, aspiración y diversión, educación, higiene y cultura”, “respuesta al ocio para devolvernos a nosotros mismos liberados”, “elemento catártico como juego limpio”, “animado por la energía creadora del espíritu”, “contribución educativa a la formación del hombre moderno”, “esencial originalidad psicofisiológica”.
- 2) En la introducción del evento a cargo del presidente del C.I.E.P.S., Philip Noël Parker, quien enuncia unas notas que ensalzan la presencia del deporte en el mundo actual: “pujante medio de comprensión internacional”, “el mayor espectáculo del mundo”, “juego limpio”, “como valor pedagógico, como medio de esfuerzo y disfrute del colectivo”.
- 3) Durante el preámbulo se realiza una definición del deporte: “el deporte es toda actividad física con carácter de juego, que adopte forma de lucha consigo mismo o con los demás o constituya una confrontación con los elementos naturales. Si implica competencia habrá de realizarse juego limpio”.

El cuerpo del documento consta de tres capítulos en los que se desarrolla la doctrina del deporte:

- 📖 Versión educativa: deporte en la escuela: “deporte como vía de educación equilibrada”, “como ocasión para desarrollar una actividad libre”, como preparación de una actuación responsable.
- 📖 Versión ociosa: deporte en los ratos de ocio: “para la expansión de la personalidad del individuo”, “como ocasión de contactos sociales”, “posibilidad de una actividad deportiva conjunta en la familia”.
- 📖 Versión competitiva: deporte de alta competición: “contribuye al perfeccionamiento humano del campeón”, “factor para el avance social, y para la promoción profesional del campeón”. (Modelo educativo del deporte, 2007)

Al analizar este conjunto de definiciones vemos que en general se observa la tendencia al deporte como fenómeno social, educativo, competitivo y de recreación. En este trabajo se quiere enfocar el concepto de deporte para todos, como derecho del pueblo y a la vez, enfocar la definición a la parte educativa de este, por lo que se propone la siguiente definición, representativa del Barón Pierre de Coubertin y del Manifiesto del deporte, referido anteriormente:

Deporte: Actividad física que desarrolla el ser humano, sin tener en cuenta razas, sexo, ni clase social. Provoca un impacto en la educación, la cultura y la sociedad contemporánea, y establece un factor de

disciplina, tolerancia y creatividad en los ciudadanos. Constituye un excelente instrumento para equilibrar la formación y el desarrollo humano de la persona a cualquier edad.

### ***1.3 Antecedentes históricos del deporte universitario.***

Según plantea Roland Renson “el primer período del Deporte Universitario abarcó del siglo XIII hasta el siglo XVIII”. (Renson, 1999) Esto significa que los estudiantes practicaban los juegos y el deporte ya desde la fundación de las primeras universidades europeas (Bolonia, Oxford, Cambridge, Lisboa, Pécs, etc.) en el siglo XIII, pero el ejercicio físico no estaba integrado dentro del programa universitario. No fue aceptado por las autoridades académicas hasta el siglo XVII. Cuando se impuso en Europa la filosofía de la ilustración fue cuando algunas universidades se hicieron conscientes de la importancia de la actividad física.

El segundo período abarca el siglo XIX, hasta el estallido de la Primera Guerra Mundial. Los juegos y el deporte aparecieron en Gran Bretaña primero en las escuelas, y posteriormente en las universidades, y se extendieron por todas las colonias británicas. Durante aquellas décadas nacieron el atletismo de élite y el amateurismo, e inspiraron la fundación de los modernos Juegos Olímpicos a finales del siglo XIX por Pierre de Coubertin. (Coubertin, 1959)

A finales del siglo XIX y principios del XX, dio comienzo la institucionalización del deporte estudiantil a nivel local, regional y nacional. Muy al principio los estudiantes, fuera de las universidades, organizaron clubes deportivos, posteriormente sindicatos, asociaciones o federaciones. La primera federación nacional de deporte universitario se fundó en 1905 en los EEUU, y la segunda en Hungría en 1907.

El tercer período (desde 1919 hasta la actualidad) puede caracterizarse por la rápida diseminación del deporte estudiantil, su aceptación oficial en las universidades y su internacionalización (durante el siglo XX, el deporte estudiantil surgió en todos los continentes). Al principio, las universidades más prestigiosas ofrecieron programas deportivos y de ejercicio físico en Gran Bretaña y los EEUU, después fueron reconocidos los valores educacionales del deporte, que fueron admitidos de manera gradual en todo el mundo. El deporte y el juego se convirtieron en una parte integral tanto en el estilo de vida de los estudiantes como en sus programas educativos. En muchos países se crearon las asociaciones nacionales universitarias.

La internacionalización del deporte estudiantil comenzó relativamente pronto, pero la Federación Internacional del Deporte Universitario (FIDU) fue fundada relativamente tarde. La tercera era de la historia del deporte universitario puede subdividirse en tres periodos según Renson: “el periodo comprendido entre 1918 y 1940, los años de la guerra fría (1945-1959), y por ultimo el periodo de las Universiadas desde 1959 hasta nuestros días.” (Renson, 1999)

El primer organismo deportivo internacional para el deporte estudiantil se creó en 1923 con el nombre de Internacional Conuffissariat of University Sport (I.C.U.S.), en el marco de la Confederación Internacional de Estudiantes, sus siglas en inglés, (I.C.S.). En nombre de estas organizaciones, un francés, Jean Petitjean, a quien se considera como parte del movimiento del Deporte Universitario mundial, organizó los primeros Juegos Estudiantiles Mundiales en París en el mismo año, es decir, en 1923, que fueron seguidos por otros muchos similares juegos (de verano) y por los juegos universitarios de invierno. Un congreso internacional relacionado con estos juegos ya recomendó la creación de una asociación internacional de deporte universitario que finalmente no se fundó hasta dos décadas y media más tarde.

Tras la Segunda Guerra Mundial, en la atmósfera de la Guerra Fría, el deporte universitario internacional se dividió en dos grandes grupos. Los países del Este fundaron la Internacional Union of Students (I.U.S.) con el objetivo de continuar las tradiciones de la I.C.S., y organizar juegos universitarios mundiales. Debido a las diferencias fundamentales en los enfoques políticos, en 1949 algunos países occidentales fundaron una organización rival independiente, llamada Federation International the Sport Universitaire, la “FISU” (Federación Internacional del Deporte Universitario), y comenzaron a organizar sus propios eventos deportivos.

El deporte estudiantil internacional se volvió a unir 10 años más tarde, cuando la campaña de reconciliación de Petitjean se vio coronada por el éxito y los países miembros de la I.U.S. se adhirieron a la FISU. Desde 1959 los estudiantes han participado en los mismos juegos estudiantiles mundiales, denominados Universiadas, independientemente de la afiliación política de sus países. La filosofía de la FISU declara que esta persigue sus objetivos sin tener en cuenta ni discriminar por razones políticas, confesionales ni raciales.

#### **1.4 El deporte universitario actual.**

El deporte universitario constituye una actividad importante en el quehacer estudiantil en la actualidad mundial, donde se aglutina un conjunto de actividades físicas, recreativas y deportivas, que deben desarrollarse dentro de un orden, un sistema y un programa establecido que aseguran la autodisciplina y el optimismo, el fortalecimiento del carácter y la voluntad de la entrega responsable en el trabajo, la solidaridad en la comunidad universitaria, el orgullo y el tesón por el ejercicio, en cualquiera de sus expresiones. Esta práctica no sólo es competitiva, sino que frecuentemente se dirige hacia la realización de actividades de recreación deportiva para emplear el tiempo de ocio de forma saludable.

Las centros universitarios a nivel mundial organizan su actividad deportiva según las normas que el gobierno nacional ha dictado en esta materia, así como también se deben tener presentes las resoluciones que al respecto han expedido los organismos que rigen tanto la políticas universitarias como las políticas de bienestar y salud. Estas leyes, normas y resoluciones fomentan en primer lugar, la realización de programas de bienestar, entendidos como el conjunto de actividades que se orientan al desarrollo físico, psico – afectivo, espiritual y social de los estudiantes. Además establecen que las instituciones de educación superior garantizarán campos y escenarios deportivos, con el propósito de facilitar el desarrollo de estas actividades.

Entre las principales actividades que se realizan en el marco universitario internacional para llevar adelante un buen desarrollo de la actividad física y la práctica deportiva podemos mencionar: promover programas de capacitación de carácter técnico, encaminados a mejorar la calidad de instructores, monitores, entrenadores de las diferentes especialidades deportivas; organizar e impulsar programas de recreación, juegos deportivos, torneos universitarios e intercambios a nivel regional y nacional; promover la realización de programas de proyección deportiva hacia la comunidad, con carácter promocional, didáctico y formativo; propender por la creación de programas que mediante la coordinación interdisciplinaria, defina pautas para la formación físico- atlética integral del estudiante universitario.

La organización rectora de estas actividades es la Federación Internacional del Deporte Universitario, con siglas en inglés FISU, la cual fue fundada en 1949. La FISU posee como principal responsabilidad supervisar las Universiadas de verano e invierno, y los campeonatos universitarios a nivel mundial. Está compuesta por una Asamblea General que la conforman representantes de las 141 organizaciones

adscritas a ella. La FISU elige su Comité Ejecutivo por un periodo de 4 años, que es quien toma las decisiones de la federación.



Fig. 1.4.1 Miembros asociados a la Federación Internacional de Deporte Universitario

## **1.5 El deporte universitario cubano.**

El deporte, es una de las actividades de mayor arraigo en las Instituciones de la Enseñanza Superior (IES.), caracterizado en la última década por privilegiar el deporte participativo, sin descuidar el competitivo o de alto rendimiento, todo ello sustentado en las clases de Educación Física que reciben los estudiantes de todas las carreras, como parte de su currículum, durante los dos primeros años de estudio en los cursos regulares diurnos.

El sistema deportivo universitario desde 1989, en que se realizó la I Universiada Nacional, ha sido sometido a un perfeccionamiento constante, por lo que hasta el momento cada versión ha tenido sus características técnicas organizativas.

El deporte universitario tiene como sello distintivo, que las competencias deportivas son convocadas por la Federación Estudiantil Universitaria, mediante una comisión organizadora en la que también participan las autoridades universitarias y los departamentos de deporte, de acuerdo al nivel y envergadura de la competencia.

Todo este sistema se coordina por el Consejo Nacional del Deporte Universitario, que lo integran representaciones de los Organismos de la Administración Central del Estado (OACE) con IES adscritas, la Federación Estudiantil Universitaria (FEU) y el organismo deportivo. Esta misma estructura existe a nivel de provincia donde están constituidos los Consejos Provinciales del Deporte Universitario.

Hasta el momento se han realizado ocho Universiadas Nacionales (1989-2006) en diferentes territorios del país. El deporte universitario cubano en la arena internacional se ha hecho sentir, pues en las nueve ediciones realizadas de los Juegos Deportivos Universitarios del Caribe, los universitarios cubanos han salido victoriosos en ocho.

## **1.6 La Industria del Software Deportivo.**

### **1.6.1 Sistemas de Gestión Deportiva en el mundo.**

En los últimos años, el deporte se ha convertido no solo en una actividad de entretenimiento, cultura física y competencia, sino que también, ha pasado a jugar un papel importante en la esfera económica y lucrativa sobre todo en la Comunidad Europea y Estados Unidos. La informática se ha ido

progresivamente incorporando a diversos ámbitos, y el mundo del deporte no es ajeno a este fenómeno, presentándose como un campo en el que esta ciencia tiene enormes posibilidades de aplicación.

Debido a este impulso, la industria del software deportivo ha experimentado un amplio desarrollo. Son innumerables las empresas y compañías que se dedican en todo el mundo a la creación de software destinado al deporte. Las actuales tendencias en el mundo del deporte y del entrenamiento deportivo se orientan hacia la incorporación de nuevas tecnologías (sophisticado aparatos de medición y registro, aplicaciones informáticas, servicios en Internet, etc.) que asisten a los profesionales del ámbito deportivo en su trabajo cotidiano. La progresiva implantación de sistemas informáticos en el mundo deportivo es un hecho constatado.

Una de las compañías más destacadas en la rama es la nombrada Systemtronic, empresa de servicio 100% española cuya principal actividad es desarrollar y mantener sus aplicaciones informáticas, que permiten la gestión integral de centros deportivos, termolúdicos y balnearios. El producto que ha permitido que Systemtronic Software sea líder del mercado español, es DeporWin, por su fácil adaptación a la gestión propia de cada centro. DeporWin es la aplicación central del software de gestión 'DeporWin Suite'. Tras más de catorce años en el sector y cerca de 200 instalaciones en marcha, se ha consolidado como una de las mejores aplicaciones de gestión deportiva a nivel mundial.

#### 1.6.1.1 *DeporWin.*

Está diseñado como una aplicación Windows, con la comodidad de uso que ello implica y los reducidos costes de formación para poder empezar a trabajar con él. Está estructurado para ser utilizado por cuatro perfiles de usuario: Conserjería (control de acceso, venta de entradas y material), Recepción (atención al público), Administración (facturación, definición de servicios, analítica) y Administrador de Sistema (Seguridad, definición de listados). Permite trabajar con diferentes entornos de datos (Microsoft Access, Microsoft SQL Server Desktop Engine y SQL Server) permite que la aplicación pueda ir creciendo con las necesidades de cada centro. Gracias a su doble arquitectura (Servidor de datos y Servidor de aplicaciones) soporta conexiones locales de los clientes de la red local así como conexiones remotas de clientes conectados a la Intranet.

Otras de las entidades destacadas en productos de este tipo es la también empresa española Desarrollo de Software Deportivo S.L. (DSD) la cual fue creada en el año 1998. DSD está plenamente dedicada al

diseño y desarrollo de software y tecnología de aplicación específica en el ámbito del deporte, ofrece una extensa gama de servicios y productos informáticos, así como la posibilidad de diseñar y desarrollar software específico y personalizado que se adapte a las necesidades particulares de deportistas, entrenadores, clubes, instituciones deportivas, federaciones, etc. Entre algunos de los productos que ofertan al mercado podemos nombrar:

#### 1.6.1.2 Fútbol TRAIN.

Software que ha sido diseñado de forma específica para entrenadores de Fútbol. Esta herramienta informática incorpora una extensa base de datos con ejercicios/tareas de entrenamiento que pueden consultarse en función de los objetivos técnico-tácticos, implicaciones físicas, recursos materiales, etc., así como diversas utilidades de planificación de las cargas de entrenamiento que desarrolla un equipo a lo largo de una temporada.

#### 1.6.1.3 Voley TRAIN 2.

Constituye el primer software interactivo de aplicación específica en el ámbito del Voleibol, que ha sido diseñado para servir de herramienta de ayuda y medio de formación para los entrenadores de esta disciplina. Este programa informático integra una base de datos abierta, que contiene más de 3.500 ejercicios de entrenamiento que cada técnico puede consultar de forma personalizada, ofreciendo la posibilidad de que cada usuario incorpore nuevos ejercicios. Además, Voley TRAIN 2 posee utilidades de ayuda al técnico en la planificación del entrenamiento y control de las cargas de trabajo aplicadas. Adicionalmente, esta herramienta informática incluye un sistema de documentación en-línea, de tipo hipertextual, que permite localizar con suma facilidad numerosa información de consulta relacionada con el entrenamiento y la planificación.

#### 1.6.1.4 SportSPEED 2.

Software de cronometraje encaminado a la medición de la velocidad de carrera del deportista con aplicación tanto en el Atletismo como en aquellas otras modalidades en las que la velocidad de desplazamiento sea un factor de rendimiento (Fútbol, Baloncesto, Balonmano, Tenis, etc.). Este software, integrado en el DSD Láser System permite registrar en tiempo real datos parciales y acumulados de velocidades, aceleraciones, distancias recorridas, índices de mejora, en cualquier test de desplazamiento

(lineal, entrecortado, etc.) que defina el entrenador, almacenando los registros de forma permanente y ofreciendo distintas posibilidades gráficas para su análisis e interpretación.

Informática & Deportes, es otra de los proyectos destinados al desarrollo de software deportivo. Esta empresa argentina posee más de 12 años de trayectoria en el rubro de informática aplicada al deporte y la actividad física. Sus sistemas son desarrollados por un grupo de profesionales y entrenadores de élite, bajo el asesoramiento de las más importantes personalidades en las áreas de Fisiología del Ejercicio, Planificación, Periodización del Entrenamiento y Medicina del Deporte. Esta exquisita combinación de profesionales da como resultado las mejores aplicaciones orientadas al entrenamiento y la actividad física, las más avanzadas en su género y por sobre todas las cosas, las más innovadoras, dado que siempre están creando nuevas tecnologías. Dentro de sus productos se encuentran:

#### 1.6.1.5 Video STAT 2.0.

Sistema informático que permite realizar el análisis estadístico de cualquier partido de cualquier deporte, teniendo en cuenta cualquier tipo de situaciones de juego y el lugar de la cancha donde éstas han ocurrido. Como elemento adicional de análisis, permite capturar tanto video como imágenes fijas desde una capturadora de video, desde cualquier archivo de video o bien desde una cámara digital, para luego terminar confeccionando un informe resumido acerca de las diferentes situaciones del partido o bien separar guardar las partes más importantes del video.

#### 1.6.1.6 Minuto Final 2.

Herramienta que le brinda al técnico un completo análisis del rendimiento de los equipos de fútbol y de cada uno de los jugadores. Facilita al cuerpo técnico el planteamiento de la mejor estrategia de acuerdo al rival, capitalizando las debilidades del oponente y aprovechando las virtudes propias. Proporciona datos estadísticos precisos referentes a las distintas acciones de juego que transcurren en un partido.

Después de haber analizado las funcionalidades y características de cada uno de los productos informáticos que se destacan a nivel mundial en el mercado del software deportivo, podemos concluir que la tendencia mundial se basa fundamentalmente en el desarrollo de herramientas que faciliten el entrenamiento de atletas de alto rendimiento, teniendo en cuenta un deporte en específico, además se observa la relevancia que toman en estos productos la esfera económica. Siendo muchos de ellos controladores de gastos y ganancias de una instalación deportiva privatizada. Otros de los aspectos a

señalar es que la mayoría de estos software son aplicaciones de escritorio, siguiendo como estándares aplicaciones Windows, y utilizan Sistemas Gestores de Base de Datos como SQL Server, Microsoft Access y Microsoft SQL Server Desktop Engine, donde todos estos productos son patentizados por la empresa Microsoft, imposibles de obtener por Cuba debido a la situación de bloqueo impuesta por los Estados Unidos de América.

### **1.6.2 Softwares deportivos en Cuba.**

El desarrollo del software deportivo en Cuba se inicia con la creación en 1979 del Centro de Investigación e Informática del Deporte (CINID), con su red de centros provinciales, pertenece al Sistema de Cultura Física y Deportes (SCFD), y constituye una unidad de servicios científico-técnicos. (INDER)

Su objetivo supremo es facilitar a los cuadros de dirección, entrenadores, investigadores, deportistas y demás usuarios, acceso a la información especializada en temas científicos, tecnológicos, económicos, etc., así como de sistemas informáticos; que se requieran para el mejor desempeño de sus funciones.

El CINID realiza una importante y compleja actividad mediante la aplicación de los distintos tipos de procesamiento de la literatura y la informatización, así como con el diseño e implementación de programas de aplicaciones orientados a la gestión de documentos, datos e información.

En el área de diseño, creación y programación de software es importante el lugar alcanzado por los especialistas pertenecientes al CINID, lo cual es reconocido por diferentes autoridades deportivas internacionales y el apoyo que desde el punto de vista informático nuestros especialistas con los software creados, brindan a la realización de exitosa de disímiles eventos deportivos de carácter regional. Entre los productos que ofrece esta institución podemos citar:

#### **1.6.2.1 Ficha o Expediente del Deportista.**

Incluye datos personales, socioeconómicos y específicos de su trayectoria como atleta. Reportes con información específica de un atleta. Posibilidad de emisión de un amplio número de listados y reportes por deportes y otros criterios de recuperación.

**1.6.2.2 Control de la Dotación.**

Herramienta para el control y la administración de los recursos. Registro sistemático de las entregas, dotaciones, ropa y calzado deportivo, implementos deportivos y medios de entrenamiento, etc. que se realiza a deportistas y federaciones. Listado de artículos entregados a deportistas y federaciones totales o en período de tiempo seleccionados. Registro de tallas de ropa y calzado de atletas. Listados de distribución de tallas de ropa y calzado para grupos de atletas seleccionados (por deportes o general), de utilidad para la preparación de vestuario para delegaciones.

**1.6.2.3 Especialistas y Técnicos contratados.**

Base de datos con información de interés sobre técnicos del deporte. Emisión de reportes variados para la caracterización de los recursos humanos disponibles en el país y contratados.

**1.6.2.4 Base histórica de Resultados Deportivos.**

Software para la compilación estadística exhaustiva de resultados de juegos regionales, campeonatos mundiales, eventos internacionales, entre otros. Es el almacén de datos ideal para realizar consultas, seguimiento de atletas, estudio de adversarios e inferencias. Posibilidades de configuración para incluir medallistas, finalistas o todos los participantes en una competencia.

**1.6.2.5 Sistema de Gestión Documental (SIGED).**

Software para la creación y mantenimiento de un catálogo bibliográfico electrónico que incluye libros, publicaciones seriadas, tesis, sitios Web, entre otras fuentes documentales. Incluye además un vocabulario de términos deportivos y un motor de búsqueda por los distintos atributos de los documentos que permite a través de búsquedas sucesivas, un fácil acceso a la bibliografía deseada.

**1.6.2.6 Sistema de Procesamiento Informativo (SIPROINF).**

Permite la aplicación de métodos y técnicas de análisis de información a fin de convertir los datos en información para potenciar el nivel de informatización de directivos y técnicos del deporte en la toma de decisiones fundamentadas para servicios de valor agregado y dispone de un potente convertidor de formatos.

**1.6.2.7 Infodeporte.**

Sistema de bases de datos de tipo Bibliográfica con resúmenes, Texto completo. Contiene más de 22 mil registros documentales, de artículos de revistas, libros y traducciones al español. Los artículos son procesados de revistas especializadas y de Internet que constituyen el núcleo de interés para el desarrollo del deporte, seleccionadas por especialistas con vasta experiencia informacional deportiva. Para optimizar la búsqueda y recuperación de información dispone de un vocabulario controlado especializado para el deporte que incluye disciplinas relativas y otras afines al mismo, desarrollado por especialistas cubanos para la organización en español de la información científico-técnica, de interés para el deporte, y las ciencias aplicadas.

**1.6.2.8 Sistema de Información de Control de competencias (SINFO).**

Desarrollado para más de 40 deportes que cumplen los requisitos de las Federaciones Internacionales, que incluye, entre otros servicios: gestión de la competencia, acreditación, control del medallero, control del programa de competencia, respaldo, seguridad, sistema de consulta para la prensa internacional, etc.

En muchos centros provinciales también se han desarrollado muchos proyectos de menos alcance pero que no dejan de jugar un papel primordial en el desarrollo de la rama del software deportivo en el país. Podemos mencionar:

**1.6.2.9 CHESSUC: (Chees=ajedrez; UC=Universidad de Camagüey).**

Software creado por el Doctor en Ciencias Pedagógicas Lázaro Bueno Pérez, de la Universidad de Camagüey. Este constituye un novedoso programa de enseñanza del ajedrez. (INDER)

**1.6.2.10 EXPERTO V.1.0.**

Diseñado por Juan Carlos Fernández Campos profesor del Departamento de Informática del ISCF. Facultad Camagüey, dicho instrumento fue avalado por el Consejo Científico de la facultad antes señalada; validándose su efectividad a través de la utilización, en trabajos de diploma y otras investigaciones desarrolladas por estudiantes y profesores de la institución señalada. (INDER)

**1.6.2.11 Pesoft Calificador 13-14 y Pesoft Calificador 15-16.**

Aplicaciones que forman parte del paquete Pesoft: Software para el deporte Levantamiento de Pesas desarrollado por INDERMATICA del CPID-HOLGUIN. (INDER).

**1.6.3 *Análisis crítico de los Sistemas de Gestión Deportiva existentes.***

Al observar las características de cada uno de los productos informáticos destinados a los procesos deportivos realizados en Cuba, se puede observar al igual que las tendencias mundiales, al deporte de alto rendimiento, sin embargo, la parte lucrativa del deporte, y la privatización, y el enriquecimiento por parte de grandes multinacionales, es punto obviado en estos productos, por la características de la sociedad cubana, donde el deporte es derecho de todo el pueblo sin distinciones de razas, sexos, ni posición social. Sin embargo, es mínima la creación de sistemas de gestión deportiva encaminados a competencias escolares y universitarias, así como la informatización de procesos deportivos relevantes en las universidades cubanas, como lo son: el desarrollo de pruebas de eficiencia física a los estudiantes, y el control de los utensilios deportivos y acceso a las aéreas deportivas con que cuentan las universidades.

Por lo que se observa la imperiosa necesidad de desarrollar un sistema de gestión deportiva que automatice procesos tales como: realización de competencias deportivas universitarias, control de los utensilios deportivos de la universidad, a través de una gestión adecuada de los datos de cada uno y los préstamos que se realizan de ellos, el control del acceso a las instalaciones deportivas, y la gestión de resultados de las pruebas de eficiencia física que se le realizan a los estudiantes.

**1.7 *Tendencias y tecnologías actuales.***

A continuación se describen algunas de las tendencias y tecnologías actuales posibles a utilizar para darle solución a los problemas planteados anteriormente de manera eficiente, teniendo en cuenta las necesidades existentes y el entorno donde se aplicará el sistema que se va a construir.

**1.7.1 *Tecnología Cliente/Servidor.***

La Tecnología Cliente/Servidor, es un modelo que implica productos y servicios enmarcados en el uso de la tecnología de punta, y que permite la distribución de la información en forma ágil y eficaz a las diversas áreas de una organización (empresa o institución pública o privada), así como también fuera de ella.

La tecnología denominada Cliente/Servidor es utilizada por todas las aplicaciones de Internet/Intranet. Un cliente funciona en su ordenador local, se comunica con el servidor remoto, y pide a éste información. El servidor envía la información solicitada. Un único servidor típicamente sirve a una multitud de clientes, ahorrando a cada uno de ellos el problema de tener la información instalada y almacenada localmente. Los Servidores Web también utilizan la tecnología Cliente/Servidor, aunque añaden aspectos nuevos y propios a la misma.

### 1.7.2 Lenguajes de Programación.

Un lenguaje de programación es un lenguaje que puede ser utilizado para controlar el comportamiento de una máquina, particularmente una computadora. Consiste en un conjunto de reglas sintácticas y semánticas que definen su estructura y el significado de sus elementos, respectivamente.

Un lenguaje de programación permite a un programador especificar de *manera precisa*: sobre qué datos una computadora debe operar, cómo deben ser estos almacenados y transmitidos y qué acciones debe tomar bajo una variada gama de circunstancias. Todo esto, a través de un lenguaje que intenta estar *relativamente* próximo al lenguaje humano o natural, tal como sucede con el lenguaje léxico. (Programación)

#### 1.7.2.1 PHP.

Es un lenguaje de programación usado generalmente para la creación de contenido para sitios Web. PHP es un acrónimo recurrente que significa "PHP Hypertext Pre-processor" (inicialmente PHP Tools, o, Personal Home Page Tools), y se trata de un lenguaje interpretado usado para la creación de aplicaciones para servidores, o creación de contenido dinámico para sitios Web. (PHP)



**Fig. 1.7.2.1.1 Funcionamiento del PHP**

El fácil uso y la similitud con los lenguajes más comunes de programación estructurada, como C y Perl, permiten a la mayoría de los programadores experimentados crear aplicaciones complejas con una curva de aprendizaje muy suave. También les permite involucrarse con aplicaciones de contenido dinámico sin tener que aprender todo un nuevo grupo de funciones y prácticas. (Programación)

Su interpretación y ejecución se da en el servidor Web, en el cual se encuentra almacenado el script, y el cliente sólo recibe el resultado de la ejecución. Cuando el cliente hace una petición al servidor para que le envíe una página Web, generada por un script PHP, el servidor ejecuta el intérprete de PHP, el cual procesa el script solicitado que generará el contenido de manera dinámica, pudiendo modificar el contenido a enviar, y regresa el resultado al servidor, el cual se encarga de regresárselo al cliente. Además es posible utilizar PHP para generar archivos PDF, Flash, así como imágenes en diferentes formatos, entre otras cosas. (PHP)

Permite la conexión a diferentes tipos de servidores de base de datos tales como: MySQL, PostgreSQL, Oracle, ODBC, DB2, Microsoft SQL Server, Firebird, SQLite; lo cual permite la creación de aplicaciones Web muy robustas. (PHP)

PHP también tiene la capacidad de ser ejecutado en la mayoría de los sistemas operativos tales como UNIX (y de ese tipo, como Linux), Windows y Mac, y puede interactuar con los servidores de Web más populares ya que existe en versión CGI, módulo para Apache, e ISAPI. (PHP)

Los principales usos del PHP son los siguientes:

-  Programación de páginas Web dinámicas, habitualmente en combinación con el motor de base de datos MySQL, aunque cuenta con soporte nativo para otros motores, incluyendo el estándar ODBC, lo que amplía en gran medida sus posibilidades de conexión.
-  Programación en consola, al estilo de Perl o Shell scripting.
-  Creación de aplicaciones gráficas independientes del navegador, por medio de la combinación de PHP y GTK (GIMP Tool Kit), lo que permite desarrollar aplicaciones de escritorio en los sistemas operativos en los que está soportado. (PHP)

Ventajas de PHP.

- 🖥️ Es un lenguaje multiplataforma.
- 🖥️ Capacidad de conexión con la mayoría de los manejadores de base de datos que se utilizan en la actualidad, destaca su conectividad con MySQL.
- 🖥️ Leer y manipular datos desde diversas fuentes, incluyendo datos que pueden ingresar los usuarios desde formularios HTML.
- 🖥️ Capacidad de expandir su potencial utilizando la enorme cantidad de módulos (llamados extensiones).
- 🖥️ Posee una amplia documentación en su página oficial, entre la cual se destaca que todas las funciones del sistema están explicadas y ejemplificadas en un único archivo de ayuda.
- 🖥️ Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.
- 🖥️ Permite las técnicas de Programación Orientada a Objetos.
- 🖥️ Permite crear los formularios para la Web.
- 🖥️ Biblioteca nativa de funciones sumamente amplia e incluida.
- 🖥️ No requiere definición de tipos de variables ni manejo detallado del bajo nivel. (PHP)

Otros lenguajes como Perl (Practical Extraction and Report Language), ASP (Active Server Pages) y JSP (Java Server Pages) tienen características similares al PHP aunque poseen rasgos que los marcan y por ello los distingue, entre ellos podemos encontrar:

- 🖥️ Características multiplataformas: Menos el ASP, que es solamente soportado por la plataforma Windows, los demás lenguajes están soportados en múltiples plataformas.
- 🖥️ Velocidad de ejecución: la velocidad es mayor en PHP, seguidos por PERL y JSP.
- 🖥️ Disponibilidad de recursos: actualmente los más utilizados en la Internet son el PHP y el JSP, siendo más utilizado en la publicación de artículos y códigos de ejemplos. PHP tiene una de las comunidades más grandes en Internet, al igual que la de Java.
- 🖥️ Familiaridad con el lenguaje: En las universidades los lenguajes más utilizados por los programadores es el ASP y el PHP. (M. Ricote, junio 2006)

### 1.7.2.2 Java.

Supone un significativo avance en el mundo de los entornos software, y esto viene avalado por tres elementos claves que diferencian a este lenguaje desde un punto de vista tecnológico:

- ☞ Es un lenguaje de programación que ofrece la potencia del diseño orientado a objetos con una sintaxis fácilmente accesible y un entorno robusto y agradable.
- ☞ Proporciona un conjunto de clases potente y flexible.
- ☞ Pone al alcance de cualquiera la utilización de aplicaciones que se pueden incluir directamente en páginas Web (aplicaciones denominadas *applets*).

Java aporta a la Web una interactividad que se había buscado durante mucho tiempo entre usuario y aplicación. (JAVA)

### 1.7.2.3 ASP.NET.

Es un conjunto de tecnologías de desarrollo de aplicaciones Web comercializado por Microsoft. Es usado por programadores para construir sitios Web domésticos, aplicaciones Web y servicios XML. Forma parte de la plataforma .NET de Microsoft y es la tecnología sucesora de la tecnología *Active Server Pages* (ASP). (Programación)

### 1.7.2.4 Perl (Practical Extraction and Report Language).

Lenguaje interpretado que tiene varias utilidades, pero está principalmente orientado a la búsqueda, extracción y formateado de ficheros de tipo texto. También es muy usado para manejo y gestión de procesos (estado de procesos, conteo y extracción de parámetros característicos, etc.).

Es una combinación de las características de los lenguajes más usados por los programadores de sistemas, como son los *shell* del sistema operativo UNIX, los utilidad (que incluye un lenguaje interpretado propio) AWK para formateo y tratamiento de texto e incluso características de Pascal, aunque su potencia se basa en la similitud con las mejores características del lenguaje estructurado C. (Programación)

## **1.7.3 Fundamentación del lenguaje a escoger.**

Por qué utilizar PHP y no otras opciones: (PHP)

- ☞ El fácil uso y la similitud con los lenguajes más comunes de programación estructurada, como C y Perl, permiten a la mayoría de los programadores experimentados crear aplicaciones complejas con una curva de aprendizaje muy suave. También les permite involucrarse con aplicaciones de contenido dinámico sin tener que aprender todo un nuevo grupo de funciones y prácticas.
- ☞ Debido al diseño de PHP, también es posible crear aplicaciones con una interfaz gráfica para el usuario (también llamada GUI), utilizando la extensión PHP-GTK. También puede ser usado desde la línea de órdenes, de la misma manera como Perl o Python pueden hacerlo, esta versión de PHP se llama PHP CLI (*Command Line Interface*).
- ☞ Su interpretación y ejecución se da en el servidor, en el cual se encuentra almacenado el script, y el cliente sólo recibe el resultado de la ejecución. Cuando el cliente hace una petición al servidor para que le envíe una página web, generada por un script PHP, el servidor ejecuta el intérprete de PHP, el cual procesa el script solicitado que generará el contenido de manera dinámica, pudiendo modificar el contenido a enviar, y regresa el resultado al servidor, el cual se encarga de regresárselo al cliente. Además es posible utilizar PHP para generar archivos PDF, Flash, así como imágenes en diferentes formatos, entre otras cosas.
- ☞ Permite la conexión a diferentes tipos de servidores de bases de datos tales como MySQL, PostgreSQL, Oracle, ODBC, DB2, Microsoft SQL Server, Firebird y SQLite; lo cual permite la creación de aplicaciones web muy robustas.
- ☞ PHP también tiene la capacidad de ser ejecutado en la mayoría de los sistemas operativos tales como UNIX (y de ese tipo, como Linux), Window y Mac OS X, y puede interactuar con los servidores web más populares ya que existe en versión CGI, módulo para Apache, e ISAPI.
- ☞ El modelo PHP puede ser visto como una alternativa al sistema de Microsoft que utiliza ASP.NET/C#/VB.NET, a ColdFusion de la compañía Macromedia, a JSP/Java de Sun Microsystems, y al famoso CGI/Perl. Aunque su creación y desarrollo se da en el ámbito de los sistemas libres, bajo la licencia GNU, existe además un IDE comercial llamado Zend Optimizar.

#### **1.7.4 Sistemas Gestor de Base de Datos.**

Los sistemas gestores de base de datos (SGBD) son los encargados de manipular la información almacenada en las bases de datos y definir las estructuras para su almacenamiento. Se compone de un

lenguaje de definición de datos, de un lenguaje de manipulación de datos y de un lenguaje de consulta. En estos sistemas existe sólo una copia de los datos para que todos los programas trabajen con ella. Otra de sus características es la capacidad de interactuar en un ambiente cliente/servidor donde los usuarios (ya sea de una intranet o desde Internet) trabajan con un conjunto único de datos alojados en un servidor y al mismo tiempo.

En la actualidad existen muchos SGBD cuyos lenguajes son en general, diferentes. Sin embargo, todos ellos tienen embebido un lenguaje único conocido como SQL (siglas de Structured Query Language). Este lenguaje permite definir relaciones, vistas, índices; controlar los accesos a la base de datos; realizar consultas, así como insertar, suprimir y modificar registros. (PostgreSQL)

#### 1.7.4.1 Microsoft SQL Server.

Es un SGBD relacionales basada en el lenguaje SQL, capaz de poner a disposición de muchos usuarios grandes cantidades de datos de manera simultánea. Así de tener unas ventajas que a continuación se pueden describir.

Entre sus características figuran:

- 🖥 Soporte de transacciones.
- 🖥 Gran estabilidad.
- 🖥 Gran seguridad.
- 🖥 Escalabilidad.
- 🖥 Soporta procedimientos almacenados.
- 🖥 Incluye también un potente entorno gráfico de administración, que permite el uso de comandos DDL y DML gráficamente.
- 🖥 Permite trabajar en modo cliente-servidor donde la información y datos se alojan en el servidor y las terminales o clientes de la red sólo acceden a la información.
- 🖥 Además permite administrar información de otros servidores de datos

Este sistema incluye una versión reducida, llamada Microsoft SQL Server Desktop Engine (MSDE) con el mismo motor de base de datos pero orientado a proyectos más pequeños. Microsoft SQL Server

constituye la alternativa de Microsoft a otros potentes sistemas gestores de bases de datos como son Oracle o Sybase ASE.

Es común desarrollar completos proyectos complementando Microsoft SQL Server y Microsoft Access a través de los llamados ADP (Access Data Project). De esta forma se completa una potente base de datos (Microsoft SQL Server) con un entorno de desarrollo cómodo y de alto rendimiento a través de la implementación de aplicaciones de dos capas mediante el uso de formularios Windows.

Para el desarrollo de aplicaciones más complejas (tres o más capas), Microsoft SQL Server incluye interfaces de acceso para la mayoría de las plataformas de desarrollo, incluyendo .NET.

Microsoft SQL Server, al contrario de su más cercana competencia, no es multiplataforma, ya que sólo está disponible en Sistemas Operativos de Microsoft.

#### 1.7.4.2 PostgreSQL

Es un SGBD libre, liberado bajo la licencia BSD. (Berkeley Software Distribution).



Alta concurrencia.

Mediante un sistema denominado MVCC (Acceso concurrente multiversión) PostgreSQL permite que mientras un proceso escribe en una tabla, otros accedan a la misma tabla sin necesidad de bloqueos. Cada usuario obtiene una visión consistente de lo último a lo que se le hizo commit. Esta estrategia es superior al uso de bloqueos por tabla o por filas común en otras bases, eliminando la necesidad del uso de bloqueos explícitos.

PostgreSQL provee nativamente soporte para: (PostgreSQL)



Números de precisión arbitraria.



Texto de largo ilimitado.



Figuras geométricas (con una variedad de funciones asociadas)



Direcciones IP (IPv4 e IPv6).



Bloques de direcciones estilo CIDR. (Classless Inter-Domain Routing)



Direcciones MAC.

🖥️ Arrays.

Adicionalmente los usuarios pueden crear sus propios tipos de datos, los que pueden ser por completo indexables gracias a la infraestructura GiST de PostgreSQL. Algunos ejemplos son los tipos de datos GIS creados por el proyecto PostGIS.

🖥️ Claves ajenas también denominadas Llaves ajenas o Llaves Foráneas (foreign keys).

🖥️ Disparadores (triggers).

🖥️ Vistas.

🖥️ Integridad transaccional.

🖥️ Herencia de tablas.

🖥️ Tipos de datos y operaciones geométricas.

Bloques de código que se ejecutan en el servidor. Pueden ser escritos en varios lenguajes, con la potencia que cada uno de ellos da, desde las operaciones básicas de programación, tales como bifurcaciones y bucles, hasta las complejidades de la programación orientación a objetos o la programación funcional. Los disparadores (triggers en inglés) son funciones enlazadas a operaciones sobre los datos.

PostgreSQL soporta funciones que retornan "filas", donde la salida puede tratarse como un conjunto de valores que pueden ser tratados igual a una fila retornada por una consulta (query en inglés).

Las funciones pueden ser definidas para ejecutarse con los derechos del usuario ejecutor o con los derechos de un usuario previamente definido. El concepto de funciones, en otros DBMS, son muchas veces referidas como "procedimientos almacenados" (stored procedures en inglés). (PostgreSQL)

#### 1.7.4.3 MySQL

MySQL es el servidor de bases de datos relacionales más popular, desarrollado y proporcionado por MySQL AB. MySQL AB es una empresa cuyo negocio consiste en proporcionar servicios en torno al servidor de bases de datos MySQL. Una base de datos es una colección estructurada de datos. La información que puede almacenar una base de datos puede ser tan simple como la de una agenda, un contador, o un libro de visitas, ó tan vasta como la de una tienda en línea, un sistema de noticias, un

portal, o la información generada en una red corporativa. Para agregar, acceder, y procesar los datos almacenados en una base de datos, se necesita un sistema de administración de bases de datos, tal como MySQL.

MySQL es muy utilizado en aplicaciones web como MediaWiki o Drupal, en plataformas (Linux/Windows-Apache-MySQL-PHP/Perl/Python), y por herramientas de seguimiento de errores como Bugzilla. Su popularidad como aplicación web está muy ligada a PHP, que a menudo aparece en combinación con MySQL. MySQL es un SGBD muy rápido en la lectura cuando utiliza el motor no transaccional MyISAM, pero puede provocar problemas de integridad en entornos de alta concurrencia en la modificación. En aplicaciones web hay baja concurrencia en la modificación de datos y en cambio el entorno es intensivo en lectura de datos, lo que hace a MySQL ideal para este tipo de aplicaciones.

#### Principales características de MySQL: (MySQL-PostgreSQL)

- MySQL es un sistema de administración de bases de datos relacionales, una base de datos relacional almacena los datos en tablas separadas en lugar de poner todos los datos en un solo lugar. Esto agrega velocidad y flexibilidad. Las tablas son enlazadas al definir relaciones que hacen posible combinar datos de varias tablas cuando se necesitan consultar datos. La parte SQL de "MySQL" significa "Lenguaje Estructurado de Consulta", y es el lenguaje más usado y estandarizado para acceder a bases de datos relacionales.

- MySQL es Open Source, esto significa que la persona que quiera puede usar y modificar MySQL. Cualquiera puede descargar el software de MySQL de Internet y usarlo sin pagar por ello. Inclusive, cualquiera que lo necesite puede estudiar el código fuente y cambiarlo de acuerdo a sus necesidades. MySQL usa la licencia GPL (Licencia Pública General GNU), para definir qué es lo que se puede y no se puede hacer con el software para diferentes situaciones. Sin embargo, si uno está incómodo con la licencia GPL o tiene la necesidad de incorporar código de MySQL en una aplicación comercial es posible comprar una versión de MySQL con una licencia comercial.

- MySQL funciona sobre múltiples plataformas, incluyendo AIX, BSD, FreeBSD, HP-UX, GNU/Linux, Mac OS X, NetBSD, Novell Netware, OpenBSD, OS/2 Warp, QNX, SGI IRIX, Solaris, SunOS, SCO OpenServer, SCO UnixWare, Tru64, Windows 95, Windows 98, Windows NT, Windows 2000, Windows XP, Windows Vista y otras versiones de Windows.

#### Características adicionales. (MySQL-PostgreSQL)

- 🖥️ Usa GNU Automake, Autoconf, y Libtool para portabilidad
- 🖥️ Uso de multihilos mediante hilos del kernel.
- 🖥️ Usa tablas en disco b-tree para búsquedas rápidas con compresión de índice
- 🖥️ Tablas hash en memoria temporales
- 🖥️ El código MySQL se prueba con Purify (un detector de memoria perdida comercial) así como con Valgrind, una herramienta GPL.
- 🖥️ Completo soporte para operadores y funciones en cláusulas select y where.
- 🖥️ Completo soporte para cláusulas group by y order by, soporte de funciones de agrupación
- 🖥️ Seguridad: ofrece un sistema de contraseñas y privilegios seguro mediante verificación basada en el host y el tráfico de contraseñas está encriptado al conectarse a un servidor.
- 🖥️ Soporta gran cantidad de datos. MySQL Server tiene bases de datos de hasta 50 millones de registros.
- 🖥️ Se permiten hasta 64 índices por tabla (32 antes de MySQL 4.1.2). Cada índice puede consistir desde 1 hasta 16 columnas o partes de columnas. El máximo ancho de límite son 1000 bytes (500 antes de MySQL 4.1.2).
- 🖥️ Los clientes se conectan al servidor MySQL usando sockets TCP/IP en cualquier plataforma. En sistemas Windows se pueden conectar usando named pipes y en sistemas Unix usando ficheros socket Unix.
- 🖥️ En MySQL 5.0, los clientes y servidores Windows se pueden conectar usando memoria compartida.
- 🖥️ MySQL contiene su propio paquete de pruebas de rendimiento proporcionado con el código fuente de la distribución de MySQL

#### Características distintivas. (MySQL-PostgreSQL)

☞ Múltiples motores de almacenamiento (MyISAM, Merge, InnoDB, BDB, Memory/heap, MySQL Cluster, Federated, Archive, CSV, Blackhole y Example en 5.x), permitiendo al usuario escoger la que sea más adecuada para cada tabla de la base de datos.

☞ Agrupación de transacciones, reuniendo múltiples transacciones de varias conexiones para incrementar el número de transacciones por segundo.

### **1.7.5 ¿Por qué usar MySQL?**

El servidor MySQL fue desarrollado originalmente para manejar grandes bases de datos mucho más rápido que las soluciones existentes y ha estado siendo usado exitosamente en ambientes de producción sumamente exigentes por varios años. Aunque se encuentra en desarrollo constante, el servidor MySQL ofrece hoy un conjunto rico y útil de funciones. Su conectividad, velocidad, y seguridad hacen de MySQL un servidor bastante apropiado para acceder a bases de datos en aplicaciones Web.

#### Comparación MySQL vs. PostgreSQL.

MySQL. Ventajas: Su principal objetivo de diseño fue la velocidad. Se sacrificaron algunas características esenciales en sistemas más "serios" con este fin. Otra característica importante es que consume muy pocos recursos, tanto de CPU como de memoria. Licencia GPL a partir de la versión 3.23.19.

Inconvenientes: No soporta transacciones, rollbacks ni subselects. No considera las claves ajenas. Ignora la integridad referencial, dejándola en manos del programador de la aplicación.

PostgreSQL. Ventajas: PostgreSQL intenta ser un sistema de bases de datos de mayor nivel que MySQL, a la altura de Oracle, Sybase o Interbase. Licencia BSD. (MySQL-PostgreSQL)

Inconvenientes: Consume bastantes más recursos y carga más el sistema. Límite del tamaño de cada fila de las tablas a 8k (se puede ampliar a 32k recompilando, pero con un coste añadido en el rendimiento). Es de 2 a 3 veces más lenta que MySQL. Menos funciones en PHP. En cuanto a consideraciones de estabilidad del servidor, cada comparativa da datos contradictorios. En general MySQL es más estable y Postgres tiende a desperdiciar memoria y sobrecargar bastante el sistema.

Comparación de Rendimiento: Las pruebas han sido ejecutadas en una máquina Pentium IV con 256 Mb de RAM. Estas pruebas han sido realizadas para analizar el rendimiento independientemente del tráfico de una red y otros factores que pueden influir en el tiempo en responder el servidor a la petición. Hemos

hecho una consulta para extraer todos los registros de una tabla que contiene 3000 datos y estos han sido los tiempos medios en devolvernos la página completa. En general la primera petición siempre tarda un poco más que las siguientes: (MySQL-PostgreSQL)

🖥️ PHP a MySQL: Tiempo medio de respuesta: 2.14 segundos.

🖥️ PHP a Postgresql: Tiempo medio de respuesta: 2.50 segundos.

Comparación en Prestaciones: En general MySQL parece algo más rápido que PostgreSQL, si bien es una diferencia muy pequeña, sin embargo PostgreSQL tiene más funcionalidades: soporta transacciones (esto quiere decir que podemos hacer, por ejemplo, un montón de sentencias "INSERT" y sí, por alguna razón, algo falla, cancelar todas estas sentencias con la instrucción "ROLLBACK", o por el contrario, hacerlas efectivas con un "COMMIT"). Hasta ahora esto no era soportado por MySQL (en las versiones más modernas ya se ha introducido). La comprobación de datos es más efectiva (se pueden comparar claves foráneas y otras funcionalidades avanzadas). En conclusión, PostgreSQL parece más completa y avanzada, pero esto tiene un precio en consumo de recursos del sistema. Por otra parte, en muchísimas situaciones no se necesitan estas funcionalidades, como por ejemplo en aplicaciones Web en las que sólo se leen datos. En la práctica, en entornos de Internet se suele usar MySQL preferentemente por su sencillez y rapidez.

### **1.7.6 Servidor Web Apache.**

Un servidor de páginas Web es un programa que permite acceder a páginas Web alojadas en un ordenador. Hoy en día Apache es el servidor Web más utilizado del mundo, encontrándose muy por encima de sus competidores, tanto gratuitos como comerciales. Es un software de código abierto que funciona sobre cualquier plataforma. Desde su origen ha evolucionado hasta convertirse en uno de los mejores servidores en términos de eficiencia, funcionalidad y velocidad, surgió en abril de 1996 y ya en julio del 2002 era utilizado por el 57% de los sitios Web de Internet.

Tiene capacidad para servir páginas tanto de contenido estático, para lo que nos serviría sencillamente un viejo ordenador 486, como de contenido dinámico a través de otras herramientas soportadas que facilitan la actualización de los contenidos mediante bases de datos, ficheros u otras fuentes de información, es muy potente y altamente configurable. (M. Ricote, junio 2006)

Los servidores Web suministran páginas Web a los navegadores que lo solicitan. En términos más técnicos, los servidores Web soportan el Protocolo de Transferencia de Hipertexto como HTTP (HyperText Transfer Protocol), el estándar de Internet para comunicaciones Web. Usando HTTP, un servidor Web envía páginas Web en HTML y Common Gateway Interface (CGI), así como otros tipos de scripts a los navegadores o browsers cuando éstos los requieren. Cuando un usuario hace clic sobre un enlace a una página Web, se envía una solicitud al servidor Web para localizar los datos nombrados por ese enlace. El servidor Web recibe esta solicitud y suministra los datos que le han sido solicitados o bien devuelve un mensaje de error. (M. Ricote, junio 2006)

El servidor Apache es un software que esta estructurado en módulos, es decir, está dividido en muchas porciones de código que hacen referencia a diferentes aspectos o funcionalidades del servidor Web. Esta modularidad es intencionada ya que la configuración de cada módulo se hace mediante la configuración de las directivas que están contenidas dentro del módulo. Los módulos del Apache se pueden clasificar en tres categorías: (M. Ricote, junio 2006)

🖥️ Módulos Base: Módulo con las funciones básicas del Apache.

🖥️ Módulos Multiproceso: Son los responsables de la unión con los puertos de la máquina, aceptando las peticiones y enviando a los hijos a atender a las peticiones.

🖥️ Módulos Adicionales: Cualquier otro módulo que le añada una funcionalidad al servidor.

Las funcionalidades más elementales se encuentran en el módulo base, siendo necesario un módulo multiproceso para manejar las peticiones. Se han diseñado varios módulos multiprocesos para cada uno de los sistemas operativos sobre los que se ejecuta el Apache, optimizando el rendimiento y rapidez del código. (M. Ricote, junio 2006)

El resto de funcionalidades del servidor se consigue por medio de módulos adicionales que se pueden cargar. Para añadir un conjunto de utilidades al servidor, simplemente hay que añadirle un módulo, de forma que no es necesario volver a instalar el software. (M. Ricote, junio 2006)

### **1.7.7 Proceso de desarrollo de software.**

Con el de cursar de los tiempos la producción de software ha ido creciendo en cuanto a cantidad, calidad y complejidad, debido a que cada vez son mayores las exigencias de los usuarios, por lo que cada vez se

hace mayor la necesidad de un proceso que integre todas las facetas del desarrollo del software y que sea perfectamente entendible por todos aquellos que participan en su elaboración.

#### 1.7.7.1 Lenguaje Unificado de Modelado.

UML (Unified Modeling Language) es un lenguaje para visualizar, especificar, construir y documentar los artefactos de un sistema de software. (OMG, 2003) Sus creadores pretendieron con este lenguaje, unificar las experiencias acumuladas sobre técnicas de modelado e incorporar las mejores prácticas en un acercamiento estándar.

El UML está compuesto por diversos elementos gráficos que se combinan para conformar diagramas y proporciona un estándar que permite al analista de sistemas generar un anteproyecto de varias facetas que sean comprensibles para los clientes, desarrolladores y todos aquellos que estén involucrados en el proceso de desarrollo. Un modelo UML indica que es lo que supuestamente hará el sistema pero no como lo hará (OMG, 2003)

El UML permite a los creadores de sistemas generar diseños que capturen sus ideas en una forma convencional y fácil de comprender para comunicarlas a otras personas que estén involucradas en el proceso de desarrollo de los sistemas, esto se lleva a cabo mediante un conjunto de símbolos y diagramas. (OMG, 2003)

Existen varias herramientas CASE (Computer-Aided Systems Engineering), que dan asistencia a analistas, ingenieros de software y desarrolladores durante el ciclo de vida de desarrollo de un software, pero es Rational Rose líder en el modelado del desarrollo de los proyectos. La herramienta fue desarrollada por los creadores de UML, utilizando la notación estándar en la arquitectura de software. Esta herramienta integra todos los elementos que propone la metodología RUP para cubrir el ciclo de vida de un proyecto. (M. Ricote, junio 2006)

#### **1.7.8 Metodologías de desarrollo de software.**

Las metodologías de desarrollo de software son un conjunto de procedimientos, técnicas y pasos a seguir para construir un software. Por otra parte una metodología de desarrollo de software define quién debe hacer qué, cuándo y cómo para alcanzar un determinado objetivo. Una metodología es un proceso, y en su modelación se definen como elementos principales los siguientes:

- 📖 Trabajadores (quién): Define el comportamiento y responsabilidades (rol) de un individuo, grupo de individuos, sistema automatizado o máquina, que trabajan en conjunto como un equipo. Ellos realizan las actividades y son propietarios de elementos.
- 📖 Actividades (como): Es una tarea que tiene un propósito claro, es realizada por un trabajador y manipula elementos.
- 📖 Artefactos (qué): Productos tangibles del proyecto que son producidos, modificados y usados por las actividades. Pueden ser modelos, elementos dentro del modelo, código fuente y ejecutables.
- 📖 Flujo de actividades (cuando): Secuencia de actividades realizadas por trabajadores y que produce un resultado de valor observable.

#### 1.7.8.1 Racional Unified Process (RUP):

La metodología RUP, llamada así por sus siglas en inglés Rational Unified Process, divide en 4 fases el desarrollo del software (*Ver Anexo 1*):

- 📖 Inicio (El objetivo de esta etapa es determinar la visión del proyecto)
- 📖 Elaboración (En esta etapa el objetivo es determinar la arquitectura)
- 📖 Construcción (En esta etapa el objetivo es llegar a alcanzar una funcionalidad operativa)
- 📖 Transición (El objetivo de esta etapa es llegar a obtener el release del producto)

En cada una de estas fases tienen lugar iteraciones de varios flujos de trabajos (*Ver Anexo 1*), que son: (I. Jacobson, 2000)

- 📖 Modelo del negocio: Describe los procesos de negocio, identificando quiénes participan y las actividades que requieren automatización.
- 📖 Requerimientos: Define qué es lo que el sistema debe hacer, para lo cual se identifican las funcionalidades requeridas y las restricciones que se imponen.
- 📖 Análisis y Diseño: Describe cómo el sistema será realizado a partir de la funcionalidad prevista y las restricciones impuestas (requerimientos), por lo que indica con precisión lo que se debe programar.

- 📄 **Implementación:** Define cómo se organizan las clases y objetos en componentes, cuáles nodos se utilizarán y la ubicación en ellos de los componentes y la estructura de capas de la aplicación.
- 📄 **Prueba (Testeo):** Busca los defectos a lo largo del ciclo de vida.
- 📄 **Instalación o despliegue:** Produce release del producto y realiza actividades (empaquete, instalación, asistencia a usuarios, etc.) para entregar el software a los usuarios finales.
- 📄 **Administración del proyecto:** Involucra actividades con las que se busca producir un producto que satisfaga las necesidades de los clientes.
- 📄 **Administración de configuración y cambios:** Describe cómo controlar los elementos producidos por todos los integrantes del equipo de proyecto en cuanto a: utilización/actualización concurrente de elementos, control de versiones, etc.
- 📄 **Ambiente:** Contiene actividades que describen los procesos y herramientas que soportarán el equipo de trabajo del proyecto; así como el procedimiento para implementar el proceso en una organización.

#### Características de RUP: (I. Jacobson, 2000)

- 📄 Creado por Jacobson, Rumbaugh y Booch.
- 📄 Unifica los mejores elementos de metodologías anteriores.
- 📄 Preparado para desarrollar grandes y complejos proyectos.
- 📄 Orientado a Objetos.
- 📄 Utiliza el UML como lenguaje de representación visual.

#### El ciclo de vida de RUP se caracteriza por: (I. Jacobson, 2000)

1. **Dirigido por casos de uso:** Los casos de uso reflejan lo que los usuarios futuros necesitan y desean, lo cual se capta cuando se modela el negocio y se representa a través de los requerimientos. A partir de aquí los casos de uso guían el proceso de desarrollo ya que los modelos que se obtienen, como resultado de los diferentes flujos de trabajo, representan la realización de los casos de uso (cómo se llevan a cabo).
2. **Centrado en la arquitectura:** La arquitectura muestra la visión común del sistema completo en la que el equipo de proyecto y los usuarios deben estar de acuerdo, por lo que describe los elementos del modelo que son más importantes para su construcción, los cimientos del sistema que son necesarios

como base para comprenderlo, desarrollarlo y producirlo económicamente. RUP se desarrolla mediante iteraciones, comenzando por los CU relevantes desde el punto de vista de la arquitectura. El modelo de arquitectura se representa a través de vistas en las que se incluyen los diagramas de UML (Vista de Casos de Uso, Vista Lógica, Vista de Procesos, Vista de Implementación, Vista de Despliegue).

3. **Iterativo e Incremental!**: Aunque el *Anexo 1* puede sugerir que los flujos de trabajo se desarrollan en cascada, la “lectura” de este gráfico tiene que ser vertical y horizontal. RUP propone que cada fase se desarrolle en iteraciones. Una iteración involucra actividades de todos los flujos de trabajo, aunque desarrolla fundamentalmente algunos más que otros. Por ejemplo, una iteración de elaboración centra su atención en el análisis y diseño, aunque refina los requerimientos y obtiene un producto con un determinado nivel, pero que irá creciendo incrementalmente en cada iteración.

#### 1.7.8.2 Extreme Programming (XP).

Es una de las metodologías de desarrollo de software más exitosas en la actualidad utilizada para proyectos de corto plazo y equipo de desarrollo pequeño.

La metodología consiste en una programación rápida o extrema, cuya particularidad es tener como parte del equipo, al usuario final, pues es uno de los requisitos para llegar al éxito del proyecto. (Software)

Esta metodología de desarrollo de software se basa en: (Software)

📖 **Pruebas Unitarias**: se basa en las pruebas realizadas a los principales procesos, de tal manera que adelantándonos en algo hacia el futuro, podamos hacer pruebas de las fallas que pudieran ocurrir. Es como si nos adelantáramos a obtener los posibles errores.

📖 **Refabricación**: se basa en la reutilización de código, para lo cual se crean patrones o modelos estándares, siendo más flexible al cambio.

📖 **Programación en pares**: una particularidad de esta metodología es que propone la programación en pares, la cual consiste en que dos desarrolladores participen en un proyecto en una misma estación de trabajo. Cada miembro lleva a cabo la acción que el otro no está haciendo en ese momento. Es como el chofer y el copiloto: mientras uno conduce, el otro consulta el mapa.

#### ¿Qué es lo que propone XP?

📖 Empieza en pequeño y añade funcionalidad con retroalimentación continua.

- 🖥 El manejo del cambio se convierte en parte sustancial del proceso.
- 🖥 El costo del cambio no depende de la fase o etapa.
- 🖥 No introduce funcionalidades antes que sean necesarias.
- 🖥 El cliente o el usuario se convierte en miembro del equipo. (Software)

#### Derechos del Cliente

- 🖥 Decidir que se implementa.
- 🖥 Saber el estado real y el progreso del proyecto.
- 🖥 Añadir, cambiar o quitar requerimientos en cualquier momento.
- 🖥 Obtener lo máximo de cada semana de trabajo.
- 🖥 Obtener un sistema funcionando cada 3 o 4 meses. (Software)

#### Derechos del Desarrollador

- 🖥 Decidir como se implementan los procesos.
- 🖥 Crear el sistema con la mejor calidad posible.
- 🖥 Pedir al cliente en cualquier momento aclaraciones de los requerimientos.
- 🖥 Estimar el esfuerzo para implementar el sistema.
- 🖥 Cambiar los requerimientos en base a nuevos descubrimientos. (Software)

#### Lo fundamental en este tipo de metodología es:

- 🖥 La comunicación, entre los usuarios y los desarrolladores.
- 🖥 La simplicidad, al desarrollar y codificar los módulos del sistema.
- 🖥 La retroalimentación, concreta y frecuente del equipo de desarrollo, el cliente y los usuarios finales. (Software)

### **1.7.9 Herramientas CASE de Modelado con UML**

A medida que los sistemas que hoy se construyen se tornan más y más complejos, las herramientas de modelado con UML ofrecen muchos beneficios para todos los involucrados en un proyecto, por ejemplo, administrador del proyecto, analistas, arquitectos, desarrolladores y otros. Las herramientas CASE (**Computer Aided Software Engineering**, Ingeniería de Software Asistida por Ordenador) de modelado con UML nos permiten aplicar la metodología de análisis y diseño orientados a objetos y abstraernos del código fuente, en un nivel donde la arquitectura y el diseño se tornan más obvios y más fáciles de entender y modificar. Cuanto más grande es un proyecto, es más importante utilizar una herramienta CASE. Al usar las herramientas CASE:

- ☞ Los Analistas de Negocio/ Sistemas pueden capturar los requisitos del negocio/sistema con un modelo de casos de uso
- ☞ Los Diseñadores/Arquitectos pueden producir el modelo de diseño para articular la interacción entre los objetos o los subsistemas de la misma o de diferentes capas (los diagramas UML típicos que se crean son los de clases y los de interacción)
- ☞ Los Desarrolladores pueden transformar rápidamente los modelos en una aplicación funcionando, y buscar un subconjunto de clases y métodos y asimilar el entendimiento de cómo lograr interfaces con ellos. (Rose)

El modelo actúa como el plano y guiará finalmente la construcción del sistema. De manera semejante, la administración es capaz de ver, puntualmente y desde un alto nivel, una representación del diseño y comprender lo que está sucediendo.

Por estas razones, las herramientas CASE de UML acompañadas con metodologías, nos brindan una forma de representar sistemas demasiados complejos para comprenderlos a través de su código fuente subyacente y nos permiten desarrollar la solución de software correcta más rápido y más económicamente.

Sin embargo, las herramientas CASE varían con respecto a las capacidades de modelado con UML, el soporte del ciclo de vida del proyecto, las ingenierías directa y reversa, el modelado de datos, la performance, el precio, el soporte, la facilidad de uso, etc. (Rose)

### 1.7.9.1 Rational Rose.

Rational Rose es la herramienta CASE desarrollada por los creadores de UML (Booch, Rumbaugh y Jacobson), que cubre todo el ciclo de vida de un proyecto: Concepción y formalización del modelo, construcción de los componentes, transición a los usuarios y certificación de las distintas fases y entregables. El navegador UML de Rational Rose nos permite establecer una trazabilidad real entre el modelo (análisis y diseño) y el código ejecutable.

Facilita el desarrollo de un proceso cooperativo en el que todos los agentes tienen sus propias vistas de información (vista de Casos de Uso, vista Lógica, vista de Componentes y vista de Despliegue), pero utilizan un lenguaje común para comprender y comunicar la estructura y la funcionalidad del sistema en construcción. (Rose)

Es la herramienta CASE que comercializan los desarrolladores de UML y que soporta de forma completa la especificación del UML. Esta herramienta propone la utilización de cuatro tipos de modelos para realizar el diseño del sistema, utilizando una vista estática y otra dinámica de los modelos del sistema, uno lógico y otro físico. Permite crear y refinar estas vistas creando de esta forma un modelo completo que representa el dominio del problema y el sistema de software.

Rational Rose utiliza un proceso de desarrollo iterativo controlado (controlled iterative process development), donde el desarrollo se lleva a cabo en una secuencia de iteraciones. Cada iteración comienza con una primera aproximación del análisis, diseño e implementación para identificar los riesgos del diseño, los cuales se utilizan para conducir la iteración, primero se identifican los riesgos y después se prueba la aplicación para que estos se hagan mínimos.

Cuando la implementación pasa todas pruebas que se determinan en el proceso, esta se revisa y se añaden los elementos modificados al modelo de análisis y diseño. Una vez que la actualización del modelo se ha modificado, se realiza la siguiente iteración.

Rational Rose permite que haya varias personas trabajando a la vez en el proceso iterativo controlado, para ello posibilita que cada desarrollador opere en un espacio de trabajo privado que contiene el modelo completo y tenga un control exclusivo sobre la propagación de los cambios en ese espacio de trabajo.

También es posible descomponer el modelo en unidades controladas e integrarlas con un sistema para realizar el control de proyectos que permite mantener la integridad de dichas unidades. (Rose)

### 1.7.9.2 Enterprise Architect:

Enterprise Architect es una herramienta comprensible de [diseño y análisis UML](#), cubriendo [el desarrollo de software](#) desde el paso de los requerimientos a través de las etapas del análisis, modelos de diseño, pruebas y mantenimiento. EA es una herramienta multi-usuario, basada en Windows, diseñada para ayudar a construir software robusto y fácil de mantener. Ofrece salida de documentación flexible y de alta calidad. (Architect)

Algunas de las principales características de EA son las siguientes: (Architect)

Velocidad, estabilidad y rendimiento: El Lenguaje Unificado de Modelado (UML) provee beneficios significativos para ayudar a construir modelos de sistemas de software rigurosos y donde es posible mantener la trazabilidad de manera consistente. Enterprise Architect soporta este proceso en un ambiente fácil de usar, rápido y flexible.

Trazabilidad de extremo a extremo: Enterprise Architect provee trazabilidad completa desde el análisis de requerimientos hasta los artefactos de análisis y diseño, a través de la implementación y el despliegue. Combinados con la ubicación de recursos y tareas incorporados, los equipos de administradores de proyectos y calidad están equipados con la información que ellos necesitan para ayudarles a entregar proyectos en tiempo.

Construido sobre UML 2.1: Las bases de Enterprise Architect están construidas sobre la especificación de UML 2.0 - pero no se detiene ahí Usa Perfiles UML para extender el dominio de modelado, mientras que la validación del modelo asegura integridad.

Por otra parte EA soporta generación e ingeniería inversa de código fuente para muchos lenguajes populares, incluyendo C++, C#, Java, Delphi, VB.Net, Visual Basic y PHP. Con un editor de código fuente con "resaltador de sintaxis" incorporado, EA le permite navegar y explorar su modelo de código fuente en el mismo ambiente. Las plantillas de generación de código le permiten personalizar el código fuente generado a las especificaciones de su compañía.

### **1.7.10 Editores PHP**

#### **1.7.10.1 NuSphere PHPEd.**

NuSphere PHPEd es un editor para programadores con soporte para múltiples formatos, similar a otras aplicaciones como PHP Coder.

PHPEd facilita tu trabajo de programación con numerosas características de gran utilidad entre las que destacan: (PHPEd)

- 📖 Completo sistema de ayuda
- 📖 Plantillas de documentos y de fragmentos de código frecuentes.
- 📖 Código de colores para comandos en PHP, Perl, Javascript, SQL, HTML y más.

Además, esta herramienta incluye un cliente de FTP y un servidor Web integrados, totalmente configurables según tus necesidades de trabajo.

#### **1.7.10.2 Zend Studio.**

Se trata de un programa de la casa Zend, impulsores de la tecnología de servidor PHP, orientado a desarrollar aplicaciones Web, en lenguaje PHP. El programa, además de servir de editor de texto para páginas PHP, proporciona una serie de ayudas que pasan desde la creación y gestión de proyectos hasta la depuración de código. (Studio)

El programa entero está escrito en Java, lo que a veces supone que no funcione tan rápido como otras aplicaciones de uso diario. Sin embargo, esto ha permitido a Zend lanzar con relativa facilidad y rapidez versiones del producto para Windows, Linux y MacOS, aunque el desarrollo de las versiones de este último sistema se retrase un poco más.

Zend Studio consta de dos partes en las que se dividen las funcionalidades de parte del cliente y las del servidor. Las dos partes se instalan por separado, la del cliente contiene el interfaz de edición y la ayuda. Permite además hacer depuraciones simples de scripts, aunque para disfrutar de toda la potencia de la herramienta de depuración habrá que disponer de la parte del servidor, que instala Apache y el módulo PHP o, en caso de que estén instalados, los configura para trabajar juntos en depuración. (Studio)

Lo más destacable de este editor es que contiene una ayuda contextual con todas las librerías de funciones del lenguaje que asiste en todo momento ofreciendo nombres de las funciones y parámetros que deben recibir. Aunque esta ayuda contextual no solo se queda en las funciones definidas en el lenguaje, sino que también reporta ayudas con las funciones que vayamos creando nosotros, incluso en páginas que tengamos incluidas con la función `include()`. (Studio)

También dispone de herramientas para gestionar los proyectos, muy útiles para mejorar la productividad en la programación. Los proyectos permiten guardar mucha más información al programa sobre los archivos, discos, servidores, etc. que se gestionen en nuestras aplicaciones PHP. (Studio)

Sin duda, más de una vez los programadores de PHP se han visto en un duro problema por no encontrar un error en algún script que está dando resultados inesperados. Zend Studio dispone de una herramienta muy interesante de debug o depuración. Gracias a ella se pueden ejecutar páginas y conocer en todo momento el contenido de las variables de la aplicación y las variables del entorno, como las cookies; las recibidas por formulario o en la sesión. Se pueden colocar puntos de parada de los scripts y realizar las acciones típicas de depuración.

Además de la ventana para visualizar el contenido de las variables, dispone de otras donde muestra la salida del script según se va generando, y otra donde se pueden ver las alertas y errores. Las posibilidades se completan con distintos tipos de depuración, en local, en remoto o a partir de una URL. (Studio)

### **1.7.11 Smarty**

Smarty es un motor de plantillas para PHP, cuyo objetivo es separar el contenido de la presentación en una página Web, se encuentra bajo la licencia GPL por lo que puede ser usado libremente. (Smarty)

Es común que en grandes proyectos el rol de diseñador gráfico y el de programador sean cubiertos por personas distintas, sin embargo la programación en PHP tiene la tendencia a combinar estas dos labores en una persona y dentro del mismo código lo que trae consigo grandes dificultades a la hora de cambiar alguna parte del diseño de la página, pues se tiene que escarbar entre los scripts para modificar la presentación del contenido, Smarty tiene como objetivo solucionar este problema. (Smarty)

Existen más sistemas de plantillas para PHP, pero éste parece ser el más avanzado y con más frecuencia de desarrollo. También hay detractores de estas técnicas que alegan que las mismas hacen en cierta

medida un grado más complejo el desarrollo Web, por la necesidad de aprender un lenguaje más. (Smarty)

## **1.8 Conclusiones**

Teniendo en cuenta las necesidades que posee la UCI, a partir de los problemas existentes en la gestión de la información que generan los procesos de actividad física y deportiva, y habiendo hecho un previo análisis de las principales características de las tecnologías y herramientas más usadas en el mundo en la actualidad, se decidió seleccionar, para la construcción y desarrollo del sistema propuesto, las siguientes herramientas y tecnologías:

Como lenguaje de programación para implementar el sistema, el PHP, debido a que, a pesar de poseer muchas ventajas al igual que Java, este lenguaje posee algunas características que lo distinguen, como es el caso de la simplicidad en su código. Además de que este lenguaje, posee una de las comunidades más grandes de desarrolladores en el mundo y aquí en la propia UCI, lo cual es muy beneficioso a la hora de consultar alguna duda.

Por otro parte, el hecho de usar PHP, justifica el uso del servidor Web Apache, los cuales poseen una gran compatibilidad, además de que este último, nos brinda algunas ventajas, como son: una gran modularidad, es gratuito y multiplataforma al igual que el PHP, además de que es muy configurable. Como editor de PHP se decidió utilizar el NuSphere PHPEd, debido a que, es de fácil manejo a la hora de trazar el código y no necesita configuración.

Se decidió también, el uso de Smarty, debido a que es un motor de plantillas para PHP, que tiene como finalidad, separar la capa de presentación de la capa lógica, lo cual trae grandes ventajas, a la hora de cambiar en cualquier momento el diseño de las páginas Web sin que afecte la lógica del negocio.

En lo que respecta a la metodología de software a utilizar, se decidió escoger RUP, debido a que la misma abarca todo el ciclo de vida del software de manera organizada, dividiéndolo en fases, en las cuales, se realizan varias iteraciones que traen consigo que se vaya desarrollando el producto de forma incremental, además de que utiliza UML para la representación visual, el cual constituye un estándar a nivel internacional, que trae consigo un entendimiento común del sistema por parte de los diferentes desarrolladores. El hecho de haber escogido la metodología de desarrollo de software RUP, justifica en gran medida la decisión de seleccionar como herramienta CASE a Rational Rose, debido a que esta

integra todos los elementos que propone dicha metodología para cubrir el ciclo de vida de un proyecto, además de que es la herramienta líder del mundo en este sentido.

## *Capítulo 2: Características del sistema*

### **2.1 Introducción.**

En el presente capítulo se enuncia y describe el objeto de estudio, además de que se dan a conocer cuáles son los procesos que serán automatizados. Por otra parte se exponen las reglas que debe cumplir el negocio en cuestión, además de se describen los actores y trabajadores que intervienen en el mismo, así como se muestra el modelo de objetos. También se identifican cuáles son los requisitos funcionales y no funcionales del sistema que darán solución a los problemas existentes; a la vez que se presentan cuáles son las funcionalidades que brinda este sistema y quienes son los actores que interactúan con el mismo.

### **2.2 Objeto de estudio.**

La actividad física y deportiva en la Universidad de las Ciencias Informáticas es uno de los procesos más importantes en la formación de un profesional integral. Por lo que el **objeto de estudio** de este trabajo son los procesos de actividad física y deportiva en las universidades.

#### **2.2.1 Objetivos estratégicos de la organización.**

La UCI es un centro universitario de reciente creación, que tiene como objetivo fundamental darles una formación integral a los jóvenes que cursan estudios en la misma. La premisa central de esta formación debe ser la de formar jóvenes altamente comprometidos con la patria y portadores de los ideales más puros que se necesitan de un revolucionario de estos tiempos, a la vez que se les da una sólida preparación académica para que una vez graduados, sean profesionales competentes que se pronuncien por resolver los problemas existentes en su entorno laboral, así como participar en la construcción de productos de software que impulsen el desarrollo económico del país.

Por otro lado, esta institución pretende ser un espacio no solo para fomentar la creación de sistemas informáticos, sino también para promover la cultura en todas sus manifestaciones, así como la práctica

frecuente del deporte como complemento sano al gran cúmulo de actividades docentes y productivas que allí se realizan durante todo el curso.

La Dirección de Deporte y Educación Física en la universidad tiene como objetivos fundamentales:

- 🖨 Desarrollar habilidades y capacidades físicas vinculadas a la formación profesional del estudiante y elevar la calidad de vida.
- 🖨 Reafirmar valores éticos, estéticos, humanistas y ambientales, a través de la labor educativa.
- 🖨 Garantizar el vínculo de la Dirección con la Extensión Universitaria, sobre la base de promover tanto la recreación, la cultura nacional como la universal.

Y como objetivos específicos:

- 🖨 Fomentar el deporte para la comunidad universitaria.
- 🖨 Adquirir medios e implementos deportivos para garantizar la calidad de la docencia y el deporte participativo.
- 🖨 Realizar ofertas deportivas que propicien una vida sana.
- 🖨 Establecer un fuerte vínculo entre las direcciones de deporte y recreación.
- 🖨 Mejorar y aumentar las áreas deportivas en función del deporte participativo.
- 🖨 Teniendo en cuenta la cantidad de estudiantes, profesores y trabajadores preparar los juegos deportivos inter-facultades y los juegos de trabajadores.
- 🖨 Mejorar el funcionamiento del gimnasio.
- 🖨 Revisar los convenios existentes con el INDER en aras de perfeccionar el trabajo entre la universidad y este instituto.
- 🖨 Crear grupos de activistas especialistas en deportes que fomenten la práctica de ejercicios físicos sistemáticos.
- 🖨 Aumentar y sistematizar la visita de glorias deportivas a la universidad

Por ello, la ejecución de todos los procesos deportivos que tiene lugar en la universidad, revisten una importancia crucial, debido a que repercuten en la formación integral que se quiere dar a estos jóvenes, y a inculcarles una cultura física de alto nivel, que ayude a sus actividades profesionales futuras.

### ***2.2.2 Flujo actual de los procesos de negocio.***

Analizar el flujo actual de los procesos, trae consigo que se pueda conocer como funciona en realidad el negocio en cuestión y de ahí obtener los posibles resultados. Estos últimos pueden ser: una información determinada, un servicio, un producto o combinaciones de ellos. El análisis de cómo se llevan a cabo estos procesos permite identificar un grupo de problemas que tienen lugar en la entidad donde se desarrollan y que dificultan el buen desenvolvimiento de las actividades que allí se realizan, como por ejemplo: demoras en la búsqueda de información, pérdida de información, duplicación de información o incorrecto formato en que se presenta la misma, entre otras.

Los procesos deportivos que tienen lugar en la universidad, referente a las competencias deportivas, pruebas de eficiencia física y servicios en las áreas deportivas se describen a continuación:

#### ***2.2.2.1 Profesor de Educación Física.***

Es el encargado de realizarle a sus estudiantes un conjunto de pruebas de eficiencia física durante el curso escolar, con el objetivo de conocer la capacidad física de cada uno de los estudiantes a través de un nivel, que se le otorga según los resultados obtenidos en la realización de un conjunto de ejercicios, en los cuales el profesor de Educación Física, le mide a cada estudiante un conjunto de indicadores (fuerza abdominal, tracción, rapidez, resistencia). Estos resultados son registrados por el profesor de Educación Física, el cual es el encargado de consultar la tabla de Normativas del Plan de Eficiencia Física de la UCI para nivelar al estudiante en cada uno de los indicadores medidos, y la tabla de Determinación de Niveles de la UCI para obtener el nivel general de cada estudiante, a través de la conformación de un número con los niveles obtenidos en cada indicador.

#### ***2.2.2.2 Técnico del gimnasio.***

El gimnasio es una de las áreas deportivas de la universidad. El técnico del gimnasio es el encargado de controlar la actividad física que se realiza en el interior de la instalación. Entre estas actividades se encuentran la solicitud de préstamos de utensilios deportivos, donde el solicitante entrega su solapín y el

técnico del gimnasio registra el utensilio deportivo solicitado, la hora de inicio del préstamo y la fecha. El solapín no es devuelto a su dueño hasta que este no devuelve el utensilio deportivo solicitado en préstamo, como medida para contrarrestar la pérdida de utensilios deportivos.

#### **2.2.2.3 Entrenador.**

El entrenador es el encargado del entrenamiento de los atletas de alto rendimiento en la universidad. El entrenador tiene la responsabilidad de conocer todos los datos personales y físicos de cada atleta que entrena, así como el resultado obtenido de cada uno de ellos en las competencias en que ha participado, ya sea competencias organizadas en el marco de la universidad o fuera de ella.

#### **2.2.2.4 Organizador de competencia.**

Es el responsable de organizar todas las competencias deportivas que se realicen en la universidad. Son los encargados de registrar los datos de cada competencia que se realice, las representaciones y equipos que participan, así como los enfrentamientos que se llevan a cabo y sus resultados correspondientes. Este rol se desempeña por un conjunto de personas, el asesor de la Dirección de Deporte y Educación Física de la universidad, el responsable de deporte de la FEU a nivel de universidad, o un profesor de Educación Física que se le asigne la responsabilidad por la Dirección.

### ***2.2.3 Análisis crítico de la ejecución de los procesos.***

En la actualidad, los procesos deportivos que se llevan a cabo en la universidad, no se realizan de forma eficiente. En ello influyen algunos factores, como son:

-  No seguimiento de la información.
-  Información en formato duro.
-  Gestión de la información de forma manual.
-  No almacenamiento de la información.
-  Insuficiencia a la hora de recopilar información de fuentes diferentes.
-  No centralización de la información.
-  Pérdidas de información.

🖥️ No contar con la información que se necesita en tiempo.

Estos factores provocan lo siguiente: el profesor de Educación Física, al realizar las prueba de eficiencia física, gestiona un gran número de datos, al este proceso realizarlo de forma manual provoca demora en la obtención de la información; además cuando el profesor necesita la información generada en la realización de las pruebas de eficiencia física, está no se encuentra disponible, ni almacenada en ningún lugar, lo que provoca que no se lleve a cabo un seguimiento de la información.

Por otro lado, el técnico del gimnasio a la hora de registrar los datos de los préstamos también realiza el proceso de forma manual, lo que conlleva que cuando desea acceder a la información se le dificulta debido a que esta puede encontrarse en distintas fuentes, o sea registrada por varios técnicos en distintos documentos. El organizador de competencia es un rol conformado por varias personas, las cuales brindan información acerca del desarrollo y organización de una competencia.

Esta información, al no encontrarse centralizada y disponible para todos, y no contar con ella en un tiempo prudencial, provoca grandes problemas en la organización del evento multidisciplinario. Esta información de competencias también es necesaria para los entrenadores de cada atleta, para llevar un seguimiento de ella, que le sea útil para la toma de decisiones. Al no estar la información centralizada y disponible provoca que el entrenador no pueda realizar con la mayor calidad su función.

#### ***2.2.4 Procesos que serán objeto de automatización.***

Los procesos que se desean automatizar son aquellos relacionados con la actividad física y deportiva que se realiza en la universidad, que de alguna forma u otra contribuirán a resolver los problemas existentes en los mismos.

Se quiere automatizar los siguientes procesos:

##### **Competencias deportivas:**

- 🖥️ Control de las competencias deportivas realizadas.
- 🖥️ Control de los participantes en competencias.
- 🖥️ Control de los enfrentamientos que se realizan y sus resultados.
- 🖥️ Confeción de la tabla de posiciones de una competencia deportiva.

##### **Gimnasio:**

- 🖨 Control de los préstamos de utensilios deportivos.
- 🖨 Control de las visitas al gimnasio.

**Pruebas de eficiencia física:**

- 🖨 Control de los resultados de la pruebas de eficiencia física.
- 🖨 Control de los niveles de cada estudiante.

**Atleta.**

- 🖨 Control de los datos físicos y personales del atleta.
- 🖨 Control de los datos de un atleta en competencias.

A continuación se describen brevemente cada uno de los procesos que serán objeto de automatización:

**2.2.4.1 Control de las competencias deportivas realizadas.**

El proceso de realización de una competencia deportiva se inicia desde el mismo instante en que el organizador de competencia decide el período de realización de la competencia. A continuación se registran las representaciones, los equipos y los atletas que participan. Después que se definen estos datos se pasa a la etapa de enfrentamiento, donde cada equipo o atleta compite con distintos adversarios, en dependencia de la fase y el sistema de competencia que se utilice. Estos enfrentamientos general resultados que adicionan puntos a las representaciones ganadoras. Al finalizar la etapa de enfrentamientos se confecciona la tabla de posiciones de la competencia, que define los lugares obtenidos por cada representación.

**2.2.4.2 Control de participantes en competencias deportivas.**

Los participantes en una competencia siempre son registrados por sus representantes de la FEU, por ejemplo en los Juegos Inter-facultades, las inscripciones de los equipos y atletas de cada facultad es responsabilidad del jefe de deporte de cada facultad, en los Juegos inter-años, son los coordinares de años. Ellos días antes de iniciada la etapa de enfrentamiento registran en nombre de todos los atletas y los equipos que participan en la competencia.

#### 2.2.4.3 Control de los enfrentamientos que se realizan y sus resultados.

Los enfrentamientos deportivos que se realizan en una competencia deportiva están identificados por la fase de la competencia en que ocurren (eliminatórias, 4to de finales, 8vo de finales, semifinales y final), el deporte y el evento en que se compite, así como los equipos o los atletas oponentes. El organizador de competencia es el encargado de registrar los resultados generados de estos enfrentamientos, ya que aportan a la confección de la tabla de posiciones de la competencia.

#### 2.2.4.4 Confección de la tabla de posiciones de una competencia deportiva.

La tabla de posiciones se puede confeccionar al finalizar el día teniendo en cuenta el resultado de cada uno de los enfrentamientos realizados en esa jornada. Y al finalizar la etapa de enfrentamiento, se confecciona la tabla de posiciones final de la competencia.

#### 2.2.4.5 Control de los préstamos de utensilios deportivos.

Un préstamo de utensilio deportivo se inicia cuando una persona solicita el préstamo de un utensilio deportivo. El técnico del gimnasio recoge el número de solapín y registra la fecha y la hora de inicio del préstamo. Al devolver el utensilio deportivo el técnico entrega el solapín al solicitante y registra la fecha y la hora de fin del préstamo.

#### 2.2.4.6 Control de las visitas al gimnasio.

Una visita al gimnasio se inicia cuando una persona entra al interior de la instalación a realizar ejercicios físicos. El proceso de registrar al visitante con la fecha y la hora de entrada, y la salida no se realiza en la actualidad. A través de este trabajo se propone que se lleve a cabo este registro, ya que permitiría el control de cierta información como pudiera ser: cantidad de personal que se encuentra actualmente en el gimnasio realizando ejercicios físicos, el horario de más visitas, los grupos, facultades o personal que más frecuenta la instalación.

#### 2.2.4.7 Control de los datos físicos y personales del atleta.

El control de los datos de un atleta es un proceso que se lleva a cabo parcialmente, solo contando con la información que brinda el directorio de la intranet de la universidad. A través de este trabajo se propone

que este control sea más estricto y completo, para que cada entrenador conozca todos los datos personales y físicos de cada uno de sus atletas, así como las lesiones que han presentado.

#### **2.2.4.8 Control de los datos de un atleta en competencias.**

Este control no se lleva independientemente del control de competencias deportivas en estos momentos. A través de la informatización de este proceso se logrará la independencia con respecto a las competencias, ya que no solo se quiere que se expongas los resultados de un atleta en una competencia deportiva, sino en todas en la que ha participado.

#### **2.2.5 Sistemas automatizados similares existentes vinculados al campo de acción.**

En la actualidad la UCI cuenta con un conjunto de sitios informativos relacionados con la actividad física y la práctica del deporte. Entre ellos se puede mencionar <http://beisbol.uci.cu/> que brinda información acerca de los resultados de la Serie Nacional en tiempo real, y un historial de los resultados en series anteriores. Además permite durante el desarrollo de la Serie Nacional, crear equipos virtuales con los jugadores de los 16 equipos de la serie, y a partir de los resultados de los integrantes del equipo virtual, en la serie, brinda puntos a los equipos virtuales. En la UCI también se han desarrollado un conjunto de aplicaciones por parte de algunos estudiantes que organizan competencias deportivas independientes. Por ejemplo, tenemos el caso de la Facultad 8, que al desarrollar su mundialito de fútbol, diseñó una aplicación para gestionar los resultados de cada partido de fútbol, y confeccionar la tabla de posiciones del mundialito.

Encontramos además la plataforma de aprendizaje Moodle, en la que está publicada una gran cantidad de información referente a la realización de ejercicios físicos y a la práctica del deporte, y un grupo de evaluaciones acerca de estos temas, que les sirven de apoyo a los estudiantes para el aprendizaje de la asignatura de Educación Física.

En general, en la UCI no se cuenta con un sistema que facilite la gestión de la actividad física y deportiva de la universidad, y que brinde la posibilidad de contar con algunas de las funcionalidades como: controlar el desarrollo de competencias deportivas, sus participantes y el resultado de los enfrentamientos que en ella se desarrolla, además un control sobre los préstamos de utensilios deportivos, las visitas al gimnasio, los datos personales y físicos del atleta y su participación en competencias deportivas.

### 2.2.6 Información que se maneja.

En el presente trabajo se maneja información de corte deportivo. Entre la que podemos encontrar tiempos, como son los resultados de los enfrentamientos de los eventos de marcas, entre los que podemos mencionar, los eventos de atletismo que miden rapidez en una distancia determinada, como 400 metros, 200 metros y 5000 metros, en esa misma situación encontramos los eventos de natación.

También se maneja estadísticas deportivas como average, cantidad de hits, cantidad de carreras permitidas, en el caso del beisbol, y otras a mencionar, como cantidad de tiros a portería, cantidad de canastas, cantidad de goles, etc.

Otra de los puntos a mencionar es el conjunto de términos deportivos que se utilizan como indicadores de eficiencia física, categoría de la competencia, tipo de enfrentamiento, clasificación de los deportes, entre otros.

## 2.3 Propuesta de sistema.

### 2.3.1 Descripción general de la propuesta de sistema.

El Sistema de Gestión Deportiva para la UCI, va a estar conformado por cinco módulos: Módulo de competencia, Módulo de gimnasio, Módulo de atleta, Módulo de pruebas de eficiencia física y Módulo de administración. A continuación se describe las funcionalidades que permitirá cada módulo.

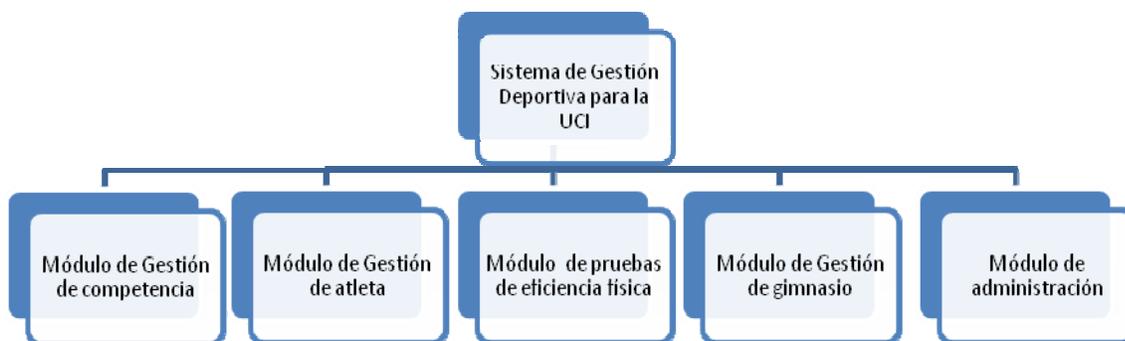


Fig. 2.3.1.1 Propuesta de sistema

#### 2.3.1.1 Módulo de Gestión de competencia.

El módulo de competencia permitirá crear, modificar y eliminar una competencia deportiva, donde al crear una competencia, permite insertar los datos propios de una competencia, la comisión organizadora y los jefes de comisiones técnicas de la competencia. Este módulo permite además gestionar los participantes en una competencia deportiva, o sea las representaciones, los equipos y los atletas. Otras de sus funcionalidades es el control de los enfrentamientos que se llevan a cabo en las competencias y los resultados que generan cada uno, así como la confección de la tabla de posiciones de la competencia deportiva.

#### 2.3.1.2 Módulo de Gestión de atleta.

Este módulo permite adicionar, modificar y eliminar un atleta, teniendo en cuenta sus datos personales y sus datos físicos. Además muestra información acerca de las competencias en que ha participado y los resultados de un atleta determinado.

#### 2.3.1.3 Módulo de pruebas de eficiencia física.

El módulo de pruebas de eficiencia física permite crear, modificar y eliminar una prueba de eficiencia física, teniendo en cuenta que una prueba de eficiencia física está identificada por su fecha de realización. Además permitirá insertar y modificar los resultados obtenidos por cada estudiante en cada uno de los indicadores evaluados y asignarle una nivelación al consultar la tabla de Normativas del Plan de Eficiencia Física de la UCI para nivelar al estudiante en cada uno de los indicadores medidos, y la tabla de Determinación de Niveles de la UCI para obtener un nivel general del estudiante.

#### 2.3.1.4 Módulo de Gestión de gimnasio.

Este módulo permitirá las opciones de crear, modificar, eliminar y terminar un préstamo de utensilio deportivo, teniendo en cuenta que solo se pueden modificar préstamos que no han sido terminados o sea pendientes, y solo se pueden eliminar préstamos terminados. El módulo contiene la funcionalidad además de adicionar, modificar, eliminar y terminar una visita al gimnasio, teniendo en cuenta condiciones parecidas a los préstamos de utensilio, o sea, una visita no puede ser modificada si ya se terminó, y una visita no puede ser eliminada si está pendiente.

#### 2.3.1.5 Módulo de administración.

Este módulo gestiona toda la información necesaria por los módulos anteriores, como pueden ser los deportes, los indicadores de eficiencia física, sistema de competencia, clasificación de competencia, deporte y enfrentamiento, y la parte de seguridad del sistema a través de la asignación de roles a cada uno de los usuarios del sistema.

#### **2.3.2 Análisis comparativo de otras soluciones existentes con la propuesta.**

Para llevar a cabo un estudio de los sistemas existentes, se ha realizado una comparación entre ellos para determinar las ventajas y desventajas de cada uno. En los sistemas de gestión deportiva a nivel mundial se observa una tendencia a que los productos constituyan software propietarios, no disponible para la mayoría de las personas y empresas. Además son aplicaciones de escritorios, que utilizan tecnologías patentizadas en el mercado. Sus funcionalidades van dirigidas fundamentalmente a un deporte en específico y centran mucho la atención en la parte de gestión económica, tanto en instalaciones como en competencias deportivas.

Los software deportivos nacionales van encaminados fundamentalmente al control de los atletas de alto rendimiento, sus estadísticas, sus datos personales, sociales y físicos, además tratan aspectos como el control de recursos e implementos deportivos, y el procesamiento de información relacionada con el deporte.

Cabe destacar la presencia de un Sistema de Información de Control de competencias, que gestiona todo lo referente a una competencia deportiva, aunque va encaminado a competencias de alto nivel, donde se incluyen una serie de aspectos tales como prensa internacional, acreditación, entre otros.

El sistema propuesto cumple con funcionalidades propias, que no están presentes en ninguno de los software analizados. Por ejemplo, la nivelación de los estudiantes según sus resultados en pruebas de eficiencia física, no ha sido un proceso automatizado en ningunos de los sistemas nacionales, ni internacionales. Se observa también una tendencia a informatizar los procesos de deporte de alto nivel, no teniendo en cuenta el deporte universitario, como forjador de profesionales sanos y preparados físicamente. Por lo que se llega a la conclusión de la necesidad de realizar el sistema propuesto por la carencia de estas funcionalidades en los software existentes.

## **2.4 Modelo de negocio.**

El modelo del negocio brinda la posibilidad de que tanto el cliente como los desarrolladores, tengan una visión común de los procesos que tienen lugar en la entidad donde se desea poner en marcha el sistema. Además de que trae consigo que se tenga una mayor comprensión de todo lo que acontece en la misma por parte de los desarrolladores, con el fin de tener bien identificados los problemas a resolver.

### **2.4.1 Modelo del negocio actual**

En la Universidad de las Ciencias Informáticas (UCI) se llevan a cabo un conjunto de procesos relacionados con la actividad física y deportiva. Los profesores de Educación Física, unido a estudiantes y trabajadores son los principales protagonistas de estas tareas. Entre uno de estos procesos podemos citar la realización de pruebas de eficiencia física a los estudiantes con el objetivo de conocer su rendimiento físico. Entre estas pruebas encontramos la medición de indicadores tales como resistencia, salto, velocidad, fuerza abdominal y tracción. La periodicidad con que se realizan es la siguiente: en el primer y segundo año de la carrera, al inicio y final de cada semestre; y en 3ro y 4to año al inicio y fin de cada curso escolar. En 5to año no se realizan.

El proceso inicia cuando el profesor le comunica al estudiante que en ese turno va a realizar un conjunto de pruebas de eficiencia física para nivelarlos según su rendimiento físico. Los estudiantes comienzan a realizar cada una de las pruebas y el profesor registra cada uno de los resultados de los indicadores. Al terminar el turno de clase, el profesor consultando la Tabla de Percentil de la Población Cubana, nivela cada uno de sus estudiantes, en cada indicador, y al finalizar realiza un consenso entre todos los niveles de cada indicador para asignar un nivel general al estudiante.

Este proceso se realiza de forma manual, lo que provoca demora en la nivelación, teniendo en cuenta el gran porcentaje de estudiante que cada profesor atiende, debido al claustro pobre que presenta la Dirección de Deportes en la universidad. Unas de las mejoras que se propone es que no solamente se nivele a un estudiante en un indicador, y de forma general, sino que también se pueda nivelar a todo un grupo en un indicador específico, y obtener la nivelación general del grupo. Esto aportaría información importante en futuros estudios científicos acerca del rendimiento físico de los estudiantes universitarios, así como permitiría comparar el nivel de un grupo en pruebas realizadas anteriormente o en futuras pruebas.

Otro de los procesos es el préstamo de utensilios deportivos. Un estudiante, trabajador o profesor solicita un préstamo de un utensilio deportivo (pelota, suiza, mascota, etc.). El técnico del gimnasio atiende la petición y le exige al solicitante que entregue su identificación de la universidad (solapín), para recogerle sus datos personales (nombre y # de solapín), y registrar la fecha y hora de inicio del préstamo. Al solicitante entregar el utensilio, se registra la fecha y la hora de fin del préstamo, y se le entrega la identificación.

Este proceso se realiza manualmente, y como no se registra el estado en que se entrega el utensilio, y el estado en que se recoge, provoca que muchos utensilios se encuentren en mal estado, sin conocer el responsable. Además en el cambio de turno del técnico del gimnasio, no se entrega la información correcta acerca de todos los préstamos que se encuentran pendientes de devolución, lo que en muchos casos ha provocado la pérdida del utensilio.

El proceso de visitar el gimnasio está estrechamente relacionado con el campo de acción de la actividad física y deportiva. Este proceso se inicia cuando un estudiante, profesor o trabajador entra a la instalación deportiva a realizar distintos tipos de ejercicios físicos, ya sea del sexo femenino o masculino. En la actualidad este acceso al gimnasio no se controla de ninguna manera, lo que provoca que no exista datos importantes para la Dirección de Deporte de la universidad como: promedio de visitas diarias al gimnasio, horario de más visitas, sexo, grupo, o facultad que más visita realiza, etc.

El entrenamiento deportivo es un proceso que realizan los estudiantes de alto rendimiento deportivo, los atletas de la universidad. Estos, a los menos tres días a la semana, realizan sesiones de entrenamiento con duración de 2 a 3 horas. En estos entrenamientos, el profesor-entrenador indica los ejercicios a realizar y registra el resultado de sus atletas en indicadores que son de su interés llevar un seguimiento.

Estos resultados pueden ser útiles para tomar decisiones como la participación de un atleta en un evento deportivo, si está lesionado, si su preparación física ha disminuido o aumentado en cierto período de tiempo, etc. Este proceso se realiza en estos momentos de forma manual por parte de cada entrenador, lo que le dificulta el trabajo y sobretodo le es difícil confeccionar reportes generales sobre cierta información o indicador.

El proceso de competir en enfrentamientos deportivos es un proceso protagonizado por estudiantes y trabajadores del centro. Todo comienza cuando por parte de la Dirección de Deporte, la FEU y otras organizaciones de masa se convoca a la realización de distintas competencias deportivas, que son todo

un acontecimiento en la universidad. En estas competencias participan un conjunto de representaciones la cuales pueden ser, facultades, grupos, años, y dentro de cada representación encontramos los equipos a participar en la competencia, así como los atletas individuales. Estos equipos y los atletas participan en un conjunto de enfrentamientos deportivos en correspondencia al deporte. Los resultados de estos enfrentamientos son recogidos por los árbitros, así como las estadísticas específicas de cada enfrentamiento. Con esto se confecciona una tabla de posiciones donde se le asigna una puntuación a cada representación por los resultados de los enfrentamientos en que participó en el día, y se muestra un total, donde se puede observar la representación que marcha al frente de la competencia.

Una parte de este proceso se encuentra automatizado, sin embargo hay muchas cosas que aún se realizan de forma manual, como es la realización de los pareos competitivos según el sistema de competencia, los datos recogidos en cada enfrentamientos no se almacenan, lo que provoca no poder efectuar una comparación de los indicadores con competencias anteriores. Además la inscripción de cada representación, con sus integrantes se entrega en formato duro por parte de los responsables de deportes de cada representación a la Dirección de Deporte de la Universidad, lo que ha provocado tardanza y desorganización en esto datos.

## **2.4.2 Reglas del negocio a considerar.**

### **2.4.2.1 Reglas de Restricción.**

-  Un atleta puede participar en varios enfrentamientos, siempre y cuando sean enfrentamientos pertenecientes a un mismo deporte.
-  Un utensilio deportivo solo presenta dos estado: buen o mal estado.
-  Un atleta no puede pertenecer a más de una representación en una misma competencia.
-  Un atleta lesionado recientemente no puede participar en ningún enfrentamiento deportivo.
-  Una competencia deportiva solo puede presentar 5 fases, eliminatorias, 4tos de finales, 8vos de finales, semifinales y final.
-  Los tipos de enfrentamiento son tres: enfrentamiento dúo atleta, enfrentamiento dúo equipo, enfrentamiento atleta múltiple.

#### 2.4.2.2 Reglas del Modelo de Datos.

- 📄 El número que identifica a cada grupo se conforma con el número de la facultad, el número del año, y los dos últimos números representan el grupo.
- 📄 Un deporte solamente puede ser clasificado en: individual, colectivo o de enfrentamiento.
- 📄 El número que identifica un apartamento está conformado por el número del edificio, el número del paso de escalera, y el número del apartamento.

#### 2.4.2.3 Reglas de Derivación.

- 📄 Si un solicitante de préstamo entrega un utensilio en mal estado se convierte en un solicitante irresponsable.
- 📄 Si un solicitante es declarado irresponsable por tercera vez, no puede solicitar más préstamos.
- 📄 Si un visitante al gimnasio no presenta su solapín no puede acceder al área deportiva.

#### **2.4.3 Actores del negocio.**

Un actor del negocio es cualquier individuo, grupo, organización, máquina o sistema de información externo que interactúa con el negocio. El término actor significa el rol que algo o alguien juega cuando interactúa con el negocio para beneficiarse de sus resultados. De acuerdo con esta idea un actor del negocio representa un tipo particular de usuario del negocio más que un usuario físico, ya que varios usuarios físicos pueden realizar el mismo papel en relación al negocio, o sea, ser instancias de un mismo actor.

A continuación se muestra en la tabla, con los actores del negocio y la justificación de los mismos:

<b>Actores del Negocio</b>	<b>Justificación</b>
----------------------------	----------------------

---

Estudiante	Constituye el principal protagonista de las actividades deportivas en la universidad. Realiza pruebas eficiencia física, solicita préstamos de utensilios deportivos y asiste al gimnasio de la universidad.
Atleta	El atleta es un estudiante con alto rendimiento físico. Realiza entrenamiento físico y participa como competidor en eventos deportivos.
Usuario	Persona que solicita el préstamo de un utensilio deportivo y visita las áreas del gimnasio.

**Tabla 2.4.3.1 Actores del negocio.**

#### ***2.4.4 Diagrama de casos de uso del negocio.***

El diagrama de casos de uso del negocio representa gráficamente los procesos del negocio y los actores que intervienen en el mismo, además de las relaciones que se establecen entre estos.

A continuación se muestra la figura correspondiente al diagrama de casos de uso del negocio.

**Diagrama de casos de uso del negocio**

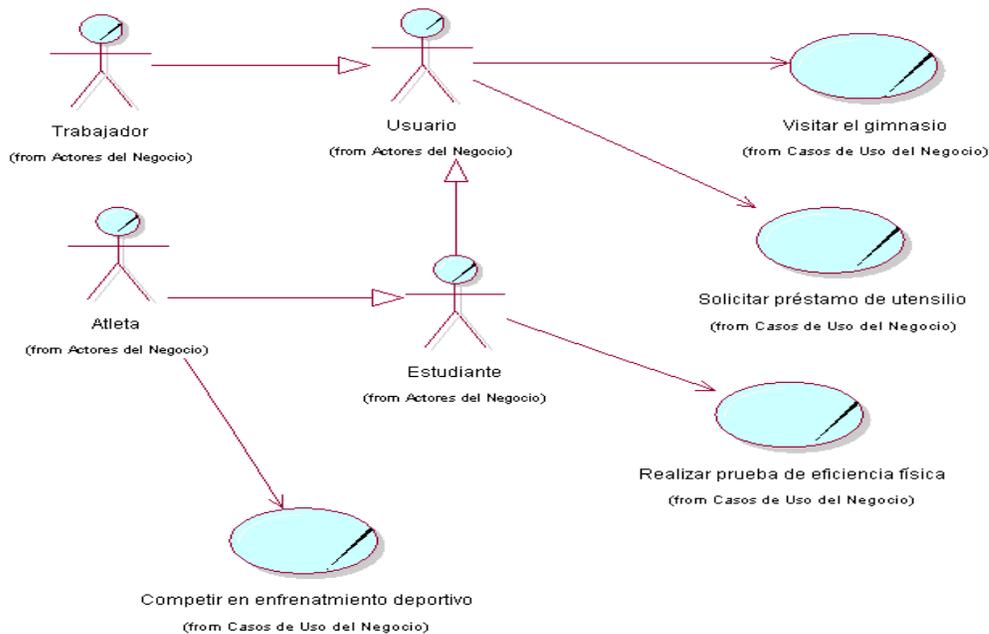


Fig. 2.4.4.1 Diagrama de casos de uso del negocio.

### 2.4.5 Trabajadores del negocio.

Define el comportamiento y responsabilidades (rol) de un individuo, grupo de individuos, sistema automatizado o máquina, que trabajan en conjunto como un equipo. Ellos realizan las actividades y son propietarios de elementos.

A continuación se muestran los trabajadores del negocio y su correspondiente justificación:

Trabajadores del Negocio	Justificación
Técnico del Gimnasio.	Es el encargado de llevar el control de los utensilios que se solicitan en préstamo a través de un registro de los datos del solicitante, el nombre del utensilio que se solicitó, y la hora de

---

	inicio y fin del préstamo.
Profesor de Educación Física.	Registra los resultados de cada uno de sus estudiantes en los indicadores de las pruebas de eficiencia física que se le realizan.
Organizador de competencias.	Es el principal responsable del registro de una competencia con todos sus datos, así como los datos de las representaciones, equipos y atletas que participan. Además es el encargado de registrar los resultados de cada uno de los enfrentamientos deportivos que se desarrolla en cada fase.
Árbitro	Controla el desarrollo de un enfrentamiento, brinda los resultados al organizador de competencia

**Tabla 2.4.5.1 Trabajadores del negocio**

### **2.4.6 Especificación de los Casos de Uso del Negocio.**

La especificación de los Casos de Uso del Negocio, propicia una mayor comprensión de los procesos del negocio, tanto por parte del cliente como de los desarrolladores. Se **describe de forma textual** como se llevan a cabo las actividades dentro del negocio y quienes las realizan. Además de que se muestra el flujo de los procesos de manera gráfica, por medio de los **diagramas de actividades**, en los cuales, las actividades representan alguna acción que, el actor o los trabajadores realizan, y las entidades no son más que la información con las cuales estos interactúan.

A continuación se realiza la descripción textual de los Casos de Uso del Negocio y se muestran los diagramas de actividades de cada uno:

<b>Caso de Uso</b>		<b>Solicitar préstamo de utensilio deportivo.</b>	
<b>Actores</b>	Usuario (inicia)		
<b>Propósito</b>	Solicitar un préstamo de un utensilio deportivo al técnico del gimnasio para la realización de actividades físicas y deportivas.		
<b>Resumen</b>	El CUN se inicia cuando el usuario solicita un préstamo de utensilio deportivo al técnico del gimnasio para realizar actividades físicas y deportivas. El técnico registra sus datos, y le entrega el utensilio. Al terminar sus actividad el usuario devuelve el utensilio, el técnico cierra el préstamo y termina el CUN.		
<b>Curso normal de los eventos</b>			
<b>Acción del actor</b>		<b>Respuesta del Negocio</b>	
1. El usuario solicita en préstamo un utensilio deportivo al técnico del gimnasio.		1.1 El técnico solicita el solapín al usuario	
2. El usuario le hace entrega de su solapín al técnico del gimnasio.		2.1 Si el usuario le entrega el solapín, el técnico registra los datos del usuario y la fecha, la hora y el nombre del utensilio solicitado en préstamo.	
		2.2 El técnico del gimnasio devuelve el solapín al usuario.	
		2.3 El técnico del gimnasio le hace entrega del utensilio en préstamo al usuario que solicitó el utensilio.	

<p>3. El usuario utiliza el utensilio en el desarrollo de su actividad física.</p>	
<p>4. Al finalizar el usuario le hace entrega al técnico del gimnasio el utensilio prestado.</p>	<p>4.1 El técnico del gimnasio revisa el estado del utensilio.</p>
	<p>4.2 Si el utensilio se encuentra en buen estado el técnico registra la devolución del utensilio y termina el CUN.</p>
<p><b>Curso alternativo de los eventos.</b></p>	
<p><b>Acción 2.1</b></p>	<p>Si el usuario no hace entrega de su solapín al técnico del gimnasio, este le informa que no puede realizar el préstamo del utensilio.</p>
<p><b>Acción 4.2</b></p>	<p>Si el utensilio deportivo se encuentra en mal estado, el técnico registra al usuario como irresponsable y finaliza el CUN.</p>

Tabla 2.4.6.1 Descripción textual del caso de uso “Solicitar préstamo de utensilio deportivo”.

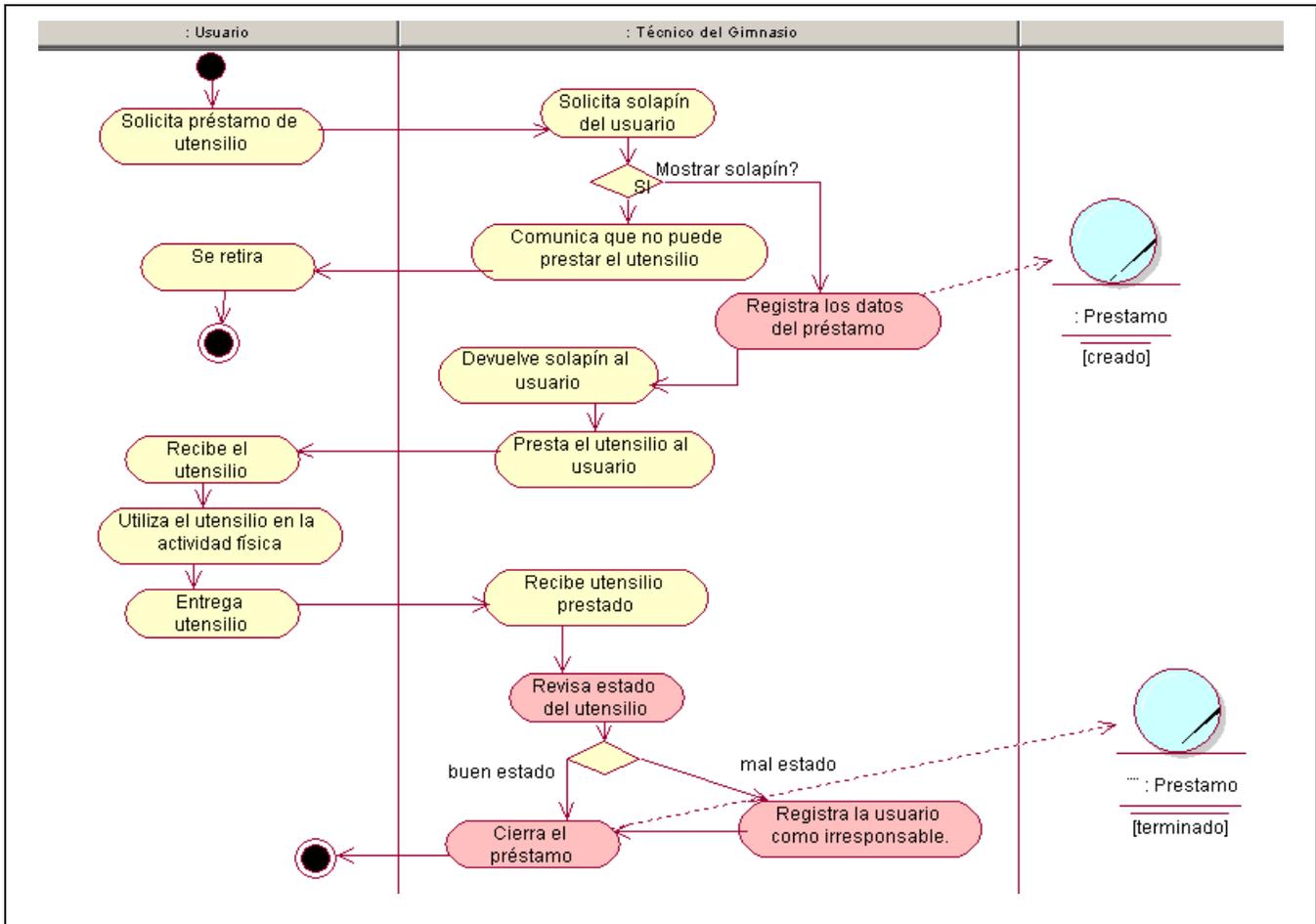


Fig. 2.4.6.1 Diagrama de actividades del caso de uso “Solicitar préstamo de utensilio deportivo”.

El resto de las descripciones textuales y diagrama de actividades de los casos de uso: “Visitar el gimnasio”, “Realizar prueba de eficiencia física” y “Competir en enfrentamiento deportivo” se puede apreciar en el Anexo # 2.

### 2.4.7 Diagrama de clases del modelo de objetos.

Un modelo de objetos del negocio es un modelo interno a un negocio. Describe cómo cada caso de uso del negocio es llevado a cabo por parte de un conjunto de trabajadores que utilizan un conjunto de entidades del negocio y unidades de trabajo.

A continuación se muestra en la figura el Diagrama de clases del modelo de objetos del negocio:

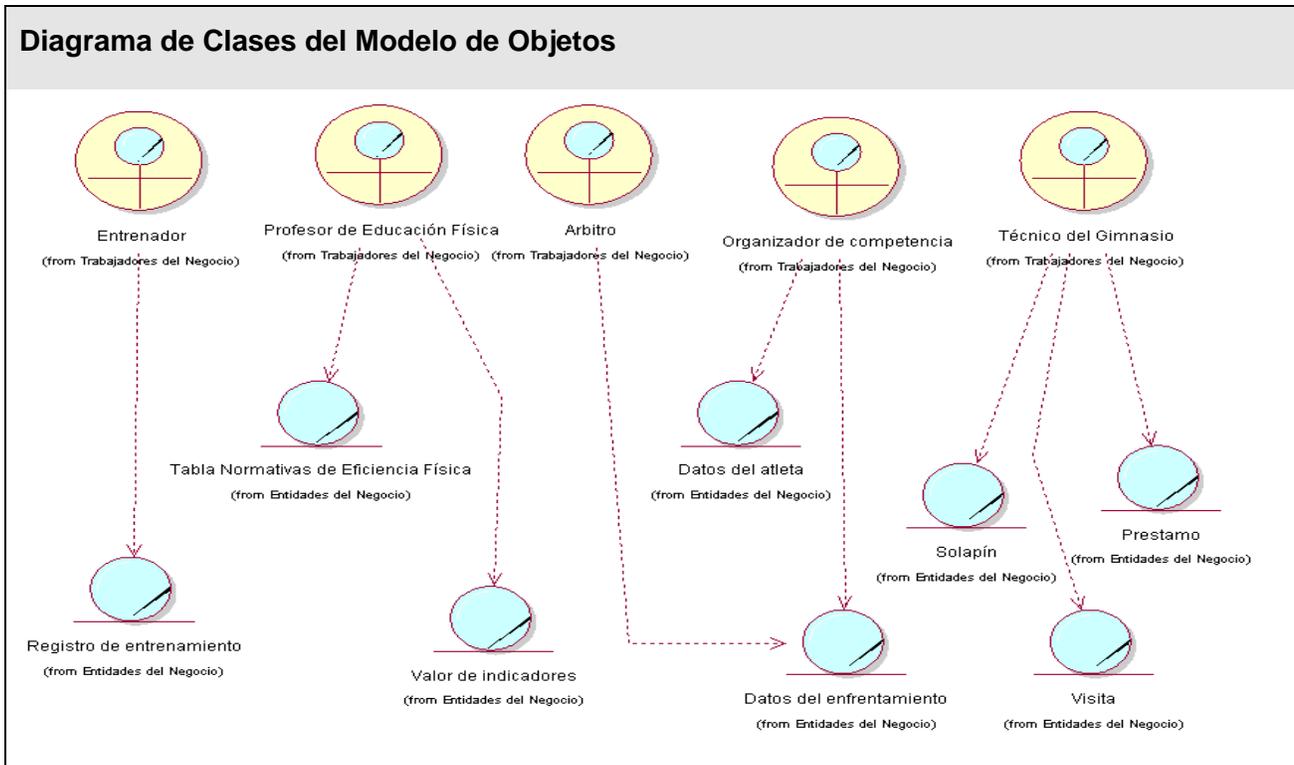


Fig. 2.4.7.1 Diagrama de clases del Modelo de Objetos.

### 2.5 Definición de los requisitos funcionales.

Los requerimientos o requisitos funcionales, son capacidades o condiciones que el sistema debe cumplir, es decir, define qué es lo que el sistema debe hacer, para lo cual se identifican las funcionalidades requeridas.

A continuación se presentan las funcionalidades que debe cumplir el sistema:

#### RF 1. *Gestionar Prueba de Eficiencia Física.*

1.1 Adicionar prueba de eficiencia física.

1.1.1 Los datos a insertar son: fecha, los indicadores a medir, el curso en que se desarrolla, nombre y apellidos de estudiante que la realiza.

1.2 Modificar prueba de eficiencia física.

1.3 Eliminar prueba de eficiencia física.

1.4 Mostrar nivelación de estudiante.

1.4.1 Los datos a mostrar son: el nivel de cada estudiante en cada indicador, y el nivel general del estudiante.

**RF 2. Gestionar Atleta.**

1.5 Adicionar atleta.

1.5.1 Los datos a insertar son: nombre y apellidos, dirección particular, sexo, edad, municipio y provincia de nacimiento, grupo, deporte que practica, peso, talla, color de la piel, calzado, lesiones.

1.6 Modificar atleta.

1.7 Eliminar atleta.

1.8 Mostrar resultado del atleta en competencias.

1.8.1 Los datos a mostrar son: nombre de la competencia, deporte en que compitió, cantidad de eventos que participó el atleta y cuales ganó, fase a la que llegó en la competencia, lugar obtenido en la competencia. (Estos datos son individuales del atleta o del equipo donde el atleta compitió).

**RF 3. Gestionar préstamo de utensilio deportivo.**

1.9 Adicionar préstamo de utensilio deportivo.

1.9.1 Los datos a insertar son: utensilio prestado, deporte del utensilio, fecha y hora de inicio del préstamo y estado inicial del utensilio.

1.10 Modificar préstamo de utensilio deportivo.

1.11 Eliminar préstamo de utensilio deportivo.

1.12 Terminar préstamo de utensilio deportivo.

1.12.1 Los datos a insertar son: estado final del utensilio, fecha y hora de fin del préstamo.

**RF 4. *Gestionar visita al gimnasio.***

1.13 Adicionar visita al gimnasio.

1.13.1 Los datos a insertar son: el número de solapín del visitante, la fecha y la hora de entrada al gimnasio.

1.14 Modificar visita al gimnasio.

1.15 Eliminar visita al gimnasio.

1.16 Terminar visita al gimnasio.

1.16.1 Los datos a insertar son: el número de solapín del visitante y la hora de salida.

**RF 5. *Gestionar competencia deportiva.***

1.17 Crear competencia deportiva.

1.17.1 Adicionar datos de una competencia deportiva.

1.17.2 Adicionar comisión organizadora de la competencia.

1.17.3 Adicionar jefes de comisiones técnicas.

1.18 Modificar competencia deportiva.

1.18.1 Modificar datos de una competencia deportiva.

1.18.2 Modificar comisión organizadora de la competencia.

1.18.3 Modificar jefes de comisiones técnicas.

**RF 6. *Gestionar representación de competencia deportiva.***

1.19 Adicionar representación de competencia deportiva.

1.19.1 Los datos a insertar son: competencia en la que participa, el nombre de la representación y la mascota.

1.20 Modificar representación de competencia deportiva.

1.21 Eliminar representación de competencia deportiva.

**RF 7. *Gestionar equipo de competencia deportiva.***

1.22 Adicionar equipo de competencia deportiva.

1.22.1 Los datos a insertar son: competencia en la que participa, deporte, sexo, solapín del capitán del equipo.

1.23 Modificar equipo de competencia deportiva.

1.24 Eliminar equipo de competencia deportiva.

**RF 8. *Gestionar enfrentamiento en competencia deportiva.***

1.25 Adicionar enfrentamiento de competencia deportiva.

1.25.1 Los datos a insertar son: competencia en la que se realiza, representaciones que se enfrentan, deporte y evento en el que se compite, lugar, fecha, hora de inicio, hora de fin, cantidad de árbitros, cantidad de personal médico, los equipos oponentes, y en caso de ser individual, los atletas oponentes.

1.26 Modificar enfrentamiento de competencia deportiva.

1.27 Eliminar enfrentamiento de competencia deportiva.

**RF 9. *Gestionar resultado de enfrentamiento en competencia deportiva.***

1.28 Adicionar resultado de enfrentamiento en competencia deportiva.

1.28.1 Los datos a insertar son: competencia en la que participa, deporte, sexo, solapín del capitán del equipo.

1.29 Modificar resultado de enfrentamiento en competencia deportiva.

1.30 Eliminar resultado de enfrentamiento en competencia deportiva.

**RF 10. *Generar tabla de posiciones de competencia deportiva.***

1.31 Seleccionar competencia.

1.32 Generar tabla de posiciones.

**RF 11. *Gestionar lesión de atleta.***

1.33 Adicionar lesión deportiva de atleta.

1.33.1 Los datos a insertar son: nombre y apellidos del atleta lesionado, tipo de lesión, localización de la lesión, y equipo al que pertenece el atleta.

1.34 Modificar lesión deportiva de atleta.

1.35 Eliminar lesión deportiva de un atleta.

**RF 12. *Gestionar categoría de competencia.***

1.36 Adicionar categoría de competencia.

1.37 Modificar categoría de competencia.

1.38 Eliminar categoría de competencia.

**RF 13. *Gestionar fase de competencia.***

1.39 Adicionar fase de competencia.

1.40 Modificar fase de competencia.

1.41 Eliminar fase de competencia.

**RF 14. *Gestionar sistema de competencia.***

1.42 Adicionar sistema de competencia.

1.43 Modificar sistema de competencia.

1.44 Eliminar sistema de competencia.

**RF 15. *Gestionar tipo de enfrentamiento.***

1.45 Adicionar tipo de enfrentamiento.

1.46 Modificar tipo de enfrentamiento.

1.47 Eliminar tipo de enfrentamiento.

**RF 16. *Gestionar unidad de resultado en enfrentamiento.***

1.48 Adicionar unidad de resultado en enfrentamiento.

1.49 Modificar unidad de resultado en enfrentamiento.

1.50 Eliminar unidad de resultado en enfrentamiento.

**RF 17. *Gestionar deporte.***

1.51 Adicionar deporte.

1.52 Modificar deporte.

1.53 Eliminar deporte.

1.54 Adicionar tipo de deporte.

1.55 Modificar tipo de deporte.

1.56 Eliminar tipo de deporte

**RF 18. *Gestionar indicador de eficiencia física.***

1.57 Adicionar indicador de eficiencia física.

1.58 Modificar indicador de eficiencia física.

1.59 Eliminar indicador de eficiencia física.

**RF 19. *Gestionar utensilio deportivo.***

1.60 Adicionar utensilio deportivo.

1.61 Modificar utensilio deportivo.

1.62 Eliminar utensilio deportivo.

**RF 20. *Gestionar localización de lesión.***

1.63 Adicionar localización de lesión.

1.64 Modificar localización de lesión.

1.65 Eliminar localización de lesión.

**RF 21. Gestionar tipo de lesión.**

1.66 Adicionar tipo de lesión.

1.67 Modificar tipo de lesión.

1.68 Eliminar tipo de lesión.

**RF 22. Gestionar rol de usuario.**

1.69 Asignar rol a usuario.

1.70 Modificar rol de usuario.

1.71 Eliminar rol de usuario.

**2.6 Definición de los requisitos no funcionales.**

Los requerimientos no funcionales son propiedades o cualidades que el producto debe tener, que posibilitan que este sea mas atractivo, usable, rápido, confiable, entre otras. También, puede ser alguna restricción que este debe tener para cumplir una determinada funcionalidad.

A continuación se presentan los requisitos no funcionales:

**2.6.1 Software.**

**RNF 1.** Servidor Web Apache v2.x o superior.

**RNF 2.** SGBD: MySQL preferiblemente v5.x en adelante.

**RNF 3.** Navegador Internet Explorer v4.0 o superior.

**2.6.2 Hardware.**

**RNF 5.** Tarjeta de red.

**RNF 6.** Para los servidores tanto Web como SGBD: PENTIUM II o superior con 256 MB de RAM o más.

**RNF 7.** Capacidad de disco duro en Gigabyte, preferiblemente mayor a los 10 GB.

### ***2.6.3 Apariencia o interfaz externa.***

**RNF 8.** La interfaz no debe contener muchas imágenes que demoren las respuestas al usuario.

**RNF 9.** El diseño de la interfaz debe ser sencillo y claro. Debe contener elementos visibles que identifiquen cada una de sus acciones.

**RNF 10.** La navegabilidad debe ser sencilla.

### ***2.6.4 Usabilidad.***

**RNF 11.** El sistema podrá ser usado por cualquier persona que acceda a él que tenga algún conocimiento básico de computación y trabajo en la Web.

**RNF 12.** Rápido acceso de búsqueda de la información, en tiempos relativamente cortos.

### ***2.6.5 Rendimiento.***

**RNF 13.** El sistema deberá ser capaz de gestionar toda la información y dar respuesta a las solicitudes lo más rápido posible.

**RNF 14.** Debe ser eficiente a la hora de gestionar las solicitudes logrando que sin mucha navegación por el sitio se obtenga los resultados deseados.

**RNF 15.** Debe estar disponible las 24 horas del día.

### ***2.6.6 Soporte.***

**RNF 16.** El sistema debe ser de fácil instalación y configuración, con vista a poder darle un mantenimiento asequible en caso de fallos.

### ***2.6.7 Portabilidad.***

**RNF 17.** Multiplataforma. El sistema se podrá montar sobre Unix, Linux, Windows, etc. Así mismo podrá usar una serie de SGBD como PostgreSQL, MySQL, Oracle, entre otros, aunque preferiblemente se desea la portabilidad sobre software libre.

### **2.6.8 Seguridad.**

- RNF 18.** La seguridad estará garantizada de acuerdo a los roles que contendrá el mismo.
- RNF 19.** El acceso al mismo será a través de autenticación donde solo se podrá acceder si el usuario se encuentra registrado en algún rol del sistema.
- RNF 20.** Las informaciones tendrán también acceso en dependencia del rol por lo que será vista por el personal al cual se le haya dado acceso a la misma logrando confidencialidad respecto a ellas.
- RNF 21.** Chequeo de seguridad sobre las acciones tales como verificación de borrado, de modificado, etc.

### **2.6.9 Legales.**

- RNF 22.** Reconocido y autorizado por instancias superiores tales como la directiva de la UCI.
- RNF 23.** Documentación legal de uso como Declaración de Autoría.

### **2.6.10 Ayuda y documentación en línea.**

- RNF 25.** Documentación de ayuda para uso del sistema, la cual estará asequible desde cualquier parte del mismo para satisfacer cualquier duda que el usuario presente con el manejo y uso de la aplicación.

### **2.6.11 Restricciones en el diseño y la implementación.**

- RNF 26.** Para la programación en PHP se recomienda el editor Nusphere PHPEd.
- RNF 27.** Se recomienda el uso de la arquitectura de n-capas con el fin de tener una mayor organización del código, además de hacer el software más extensible.

## **2.7 Modelo de sistema.**

### **2.7.1 Actores del sistema.**

Los actores del sistema definen el comportamiento y responsabilidades (rol) de un individuo, grupo de individuos, sistema automatizado o máquina, que interactúan con el mismo, intercambiando información con este.

A continuación se detallan los actores del sistema:

<b>Actores del sistema</b>	<b>Justificación</b>
Profesor de Educación Física.	Registra los resultados de cada uno de sus estudiantes en los indicadores de las pruebas de eficiencia física que se le realizan.
Técnico del Gimnasio.	Es el encargado de llevar el control de los utensilios que se solicitan en préstamo a través de un registro de los datos del solicitante, el nombre del utensilio que se solicitó, y la hora de inicio y fin del préstamo. Además controla los datos de los asistentes al gimnasio.
Entrenador.	El entrenador es un profesor de la asignatura de Educación Física, que posee conocimientos y experiencias en un deporte determinado y toma la responsabilidad del equipo de atletas en ese deporte. Registra los datos personales y físicos de cada uno de sus atletas, así como las lesiones de cada uno.
Organizador de competencias.	Es el principal responsable del registro de una competencia con todos sus datos, así como los datos de las representaciones, equipos y atletas que participan. Además es el encargado de registrar los resultados de cada uno de los eventos deportivos que se desarrolla en cada fase.
Administrador del sistema	Es el encargado de insertar los datos de las tablas nomencladoras y clasificadoras de la base de datos.

**Tabla 2.7.1.1 Actores del sistema.**

### **2.7.2 Diagrama de casos de uso del sistema.**

Los Casos de Uso del Sistema (CUS) son un conjunto de secuencia de acciones que un sistema ejecuta y que produce un resultado observable para un actor. Con otras palabras, son “fragmentos” de funcionalidad que el sistema ofrece a los actores que interactúan con el mismo, reportándole algunos que otro beneficio.

Por la amplitud del Diagrama de Casos de Uso del Sistema se dividió en subdiagramas, representativos de cada uno de los módulos del sistema.

**Diagrama de caso de uso del sistema del módulo Gestión de atleta.**

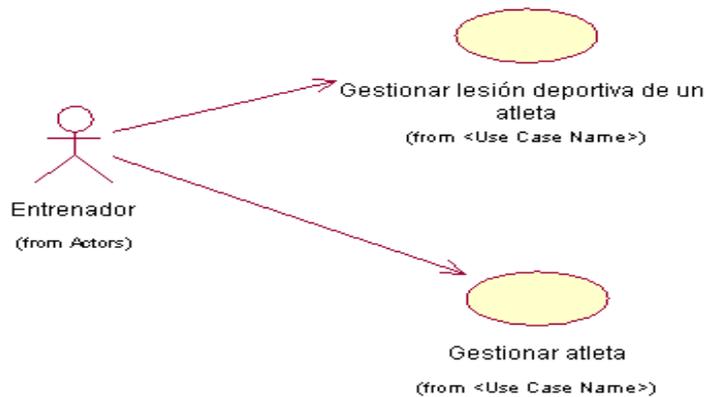


Fig. 2.7.2.1 Diagrama de caso de uso del sistema del módulo Gestión de atleta.

**Diagrama de casos de uso del sistema del módulo de Nivelación de pruebas de eficiencia física.**

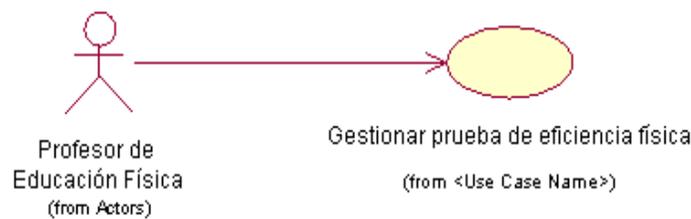


Fig. 2.7.2.2 Diagrama de caso de uso del sistema del módulo de Nivelación de pruebas de eficiencia física.

**Diagrama de caso de uso del sistema del módulo de Administración.**

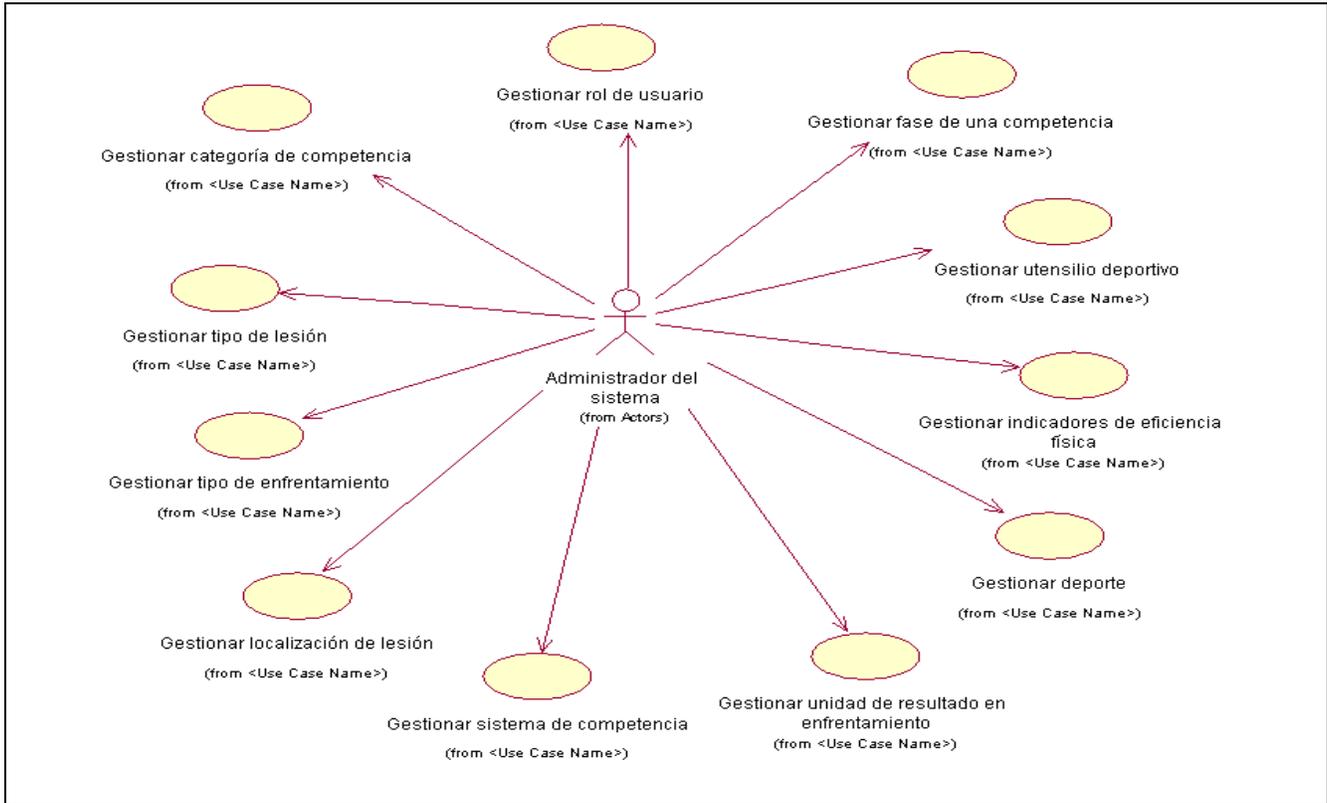


Fig. 2.7.2.3 Diagrama de caso de uso del sistema del módulo de Administración

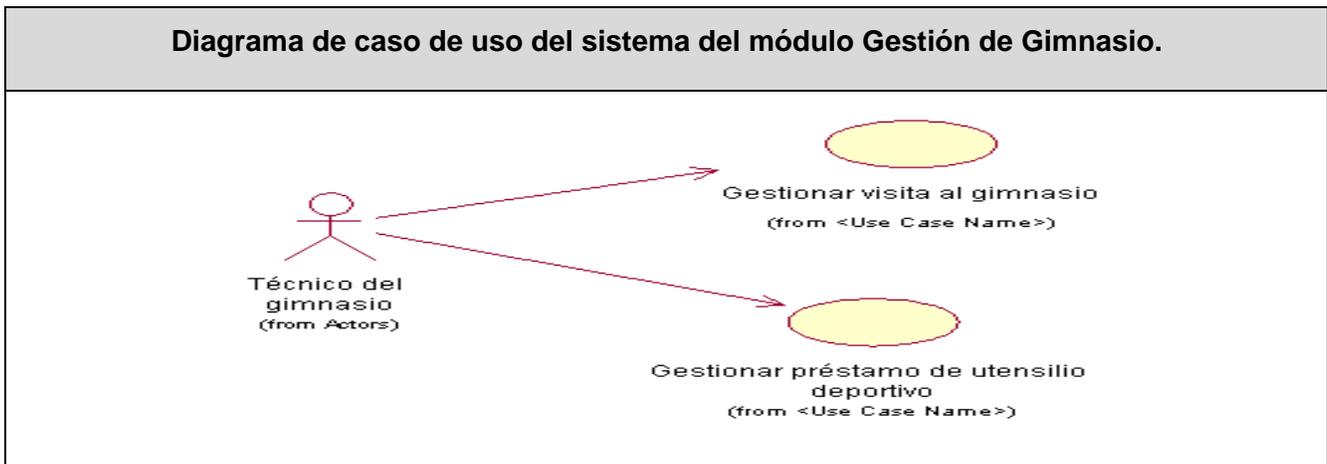


Fig. 2.7.2.4 Diagrama de caso de uso del sistema del módulo Gestión de Gimnasio

**Diagrama de caso de uso del sistema del módulo Gestión de competencia.**

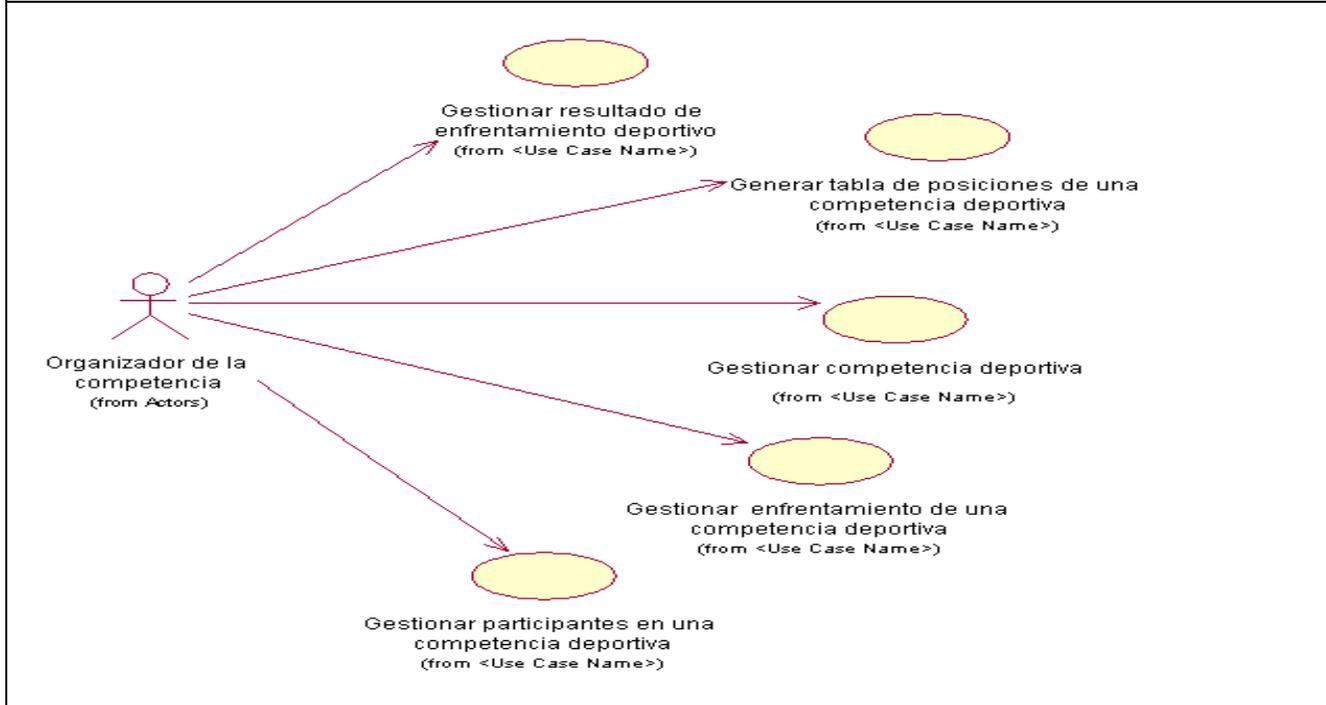


Fig. 2.7.2.5 Diagrama de caso de uso del sistema del módulo Gestión de competencia.

**2.7.3 Descripción de los casos de uso del sistema.**

La descripción de los casos de uso del sistema, detalla las acciones que tienen lugar durante la interacción actor-sistema, es decir, describe el flujo de actividades que realiza el actor al hacer uso del sistema y las correspondientes respuestas del mismo. Por lo tanto, establece un acuerdo entre clientes y desarrolladores sobre que es lo que el sistema debe hacer.

<b>Caso de Uso</b>	<b>Gestionar atleta</b>
<b>Actores</b>	Entrenador (inicia).
<b>Propósito</b>	Le permite al entrenador adicionar, modificar y eliminar un atleta y mostrar los datos del atleta en las competencias que ha participado.

<b>Resumen</b>	El CUS se inicia cuando el entrenador r selecciona la opción Gestionar atleta. El sistema muestra las opciones, Adicionar atleta, Modificar atleta, Eliminar atleta y Mostrar resultados de atleta en competencia. El entrenador selecciona la opción que desea realizar e introduce al sistema los datos requeridos. El sistema muestra el resultado de la acción del entrenador .Estos datos se registran en la Base de Datos si son correctos y termina el CUS.
<b>Referencias</b>	RF 2
<b>Prioridad</b>	<b>Crítico</b>

Tabla 2.7.3.1 Descripción textual del CUS “Gestionar atleta”.

El resto de las descripciones textuales de cada CUS se pueden apreciar en el Anexo # 3.

## **2.8 Conclusiones.**

En este capítulo se han abordado aspectos relacionados con el objeto de estudio, con el objetivo de profundizar en el conocimiento de estos temas y fundamentar la necesidad de dar una solución informática a la situación existente.

Se realizó la modelación del negocio a fin de lograr un mejor entendimiento de cómo se llevan a cabo los procesos del negocio y se logra que los usuarios finales y desarrolladores tengan un entendimiento común. Se obtuvo el diagrama de casos de uso del negocio, así como la descripción de cada uno de ellos en formato expandido y los diagramas de actividades donde se sombrearon las actividades que se van a automatizar en esta primera versión del sistema. Se identificaron y describieron los actores y trabajadores del negocio, se obtuvo además el modelo de objetos del negocio.

Se identificaron los requerimientos del sistema, obteniéndose así las funcionalidades que debe tener el mismo, además se obtuvo un listado de requerimientos no funcionales a tener en cuenta para el desarrollo de la aplicación.

Se definieron los casos de uso del sistema, mostrándose los diagramas de casos de uso. Se describió además cada caso de uso, así como los actores del sistema que se van a relacionar con ellos, dando de esta forma una vista global de cómo está concebido y cómo va a funcionar el sistema.

## *Capítulo 3: Análisis y Diseño del sistema.*

### **3.1 Introducción**

En este capítulo se expone la concepción general referente al análisis y diseño del sistema propuesto, donde el análisis tiene como objetivo mantener un modelo eficiente de la solución propuesta que sirva de base para el diseño. Este último tiene a su vez, como objetivo presentar como es que está construido el sistema, lo cual se realiza en este caso a partir de los diagramas de clases Web, que tienen la finalidad de describir la interacción entre las distintas páginas de la aplicación. Por otro lado se presenta el diagrama de clases persistentes de la base de datos del sistema, además de los estándares de diseño y programación seguidos.

### **3.2 Análisis**

#### **3.2.1 Diagramas de clases del análisis**

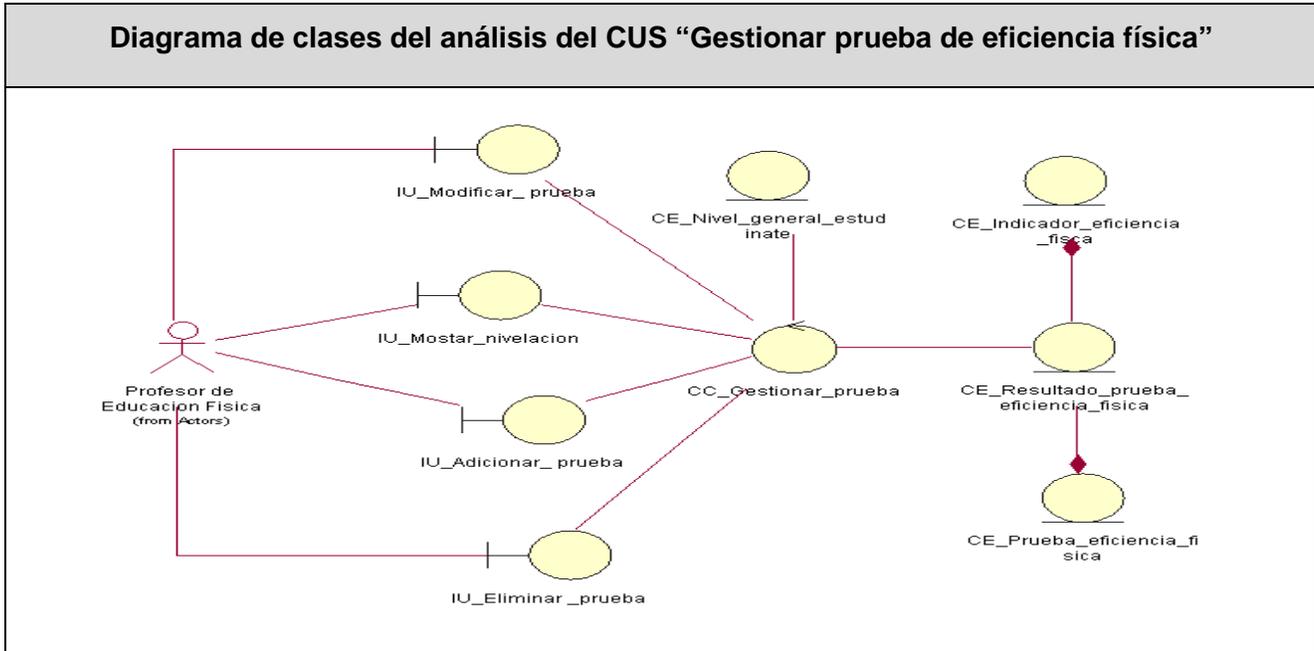
En el análisis se presentan los siguientes estereotipos de clases:

**Clase de frontera o interfaz:** Modela la interfaz del sistema, y manejan la comunicación entre el entorno y el interior del mismo. Durante el diseño, estas clases son refinadas para tomar en consideración los mecanismos de interfaz seleccionados o implementados, además de para facilitar la comunicación con otros sistemas, etc.

**Clases de entidad o sistema:** Representan la información manejada en el caso de uso, además de que modelan información y comportamiento asociado que generalmente es de larga duración. Reflejan entidades del mundo real, que resultan necesarias para realizar tareas internas del sistema.

**Clases de control o software:** Coordinan los eventos necesarios para la realización o especificación del caso de uso, con otras palabras, son las que ejecutan el caso de uso. Usualmente son dependientes de la aplicación, además de tener un control sobre todas las acciones a realizar.

A continuación se muestran los diagramas de clases del análisis para cada uno de los casos de uso del sistema:



**Fig. 3.2.1.1 Diagrama de clases del análisis del CUS “Gestionar prueba de eficiencia física”**

El resto de los diagramas de clases del análisis de los CUS se pueden apreciar en el Anexo # 4.

### 3.3 Diseño

#### 3.3.1 Diagrama de clases del Diseño.

Un diagrama de clases presenta las clases del sistema con sus relaciones estructurales y de herencia. En el caso de las aplicaciones Web, el diagrama de clases representa las colaboraciones que ocurren entre las páginas, donde cada página lógica puede ser representada como una clase.

A continuación se muestran los diagramas de clases Web para cada uno de los casos de uso del sistema:

3.3.1.1 Diagrama de clases Web para el caso de uso “Gestionar competencia”.

Un caso de uso puede ser dividido en **escenarios**, los cuales representan funcionalidades específicas dentro del caso de uso y se utilizan a menudo cuando se tienen casos de uso que poseen varias funcionalidades, para lograr con ello una mayor comprensión del mismo.

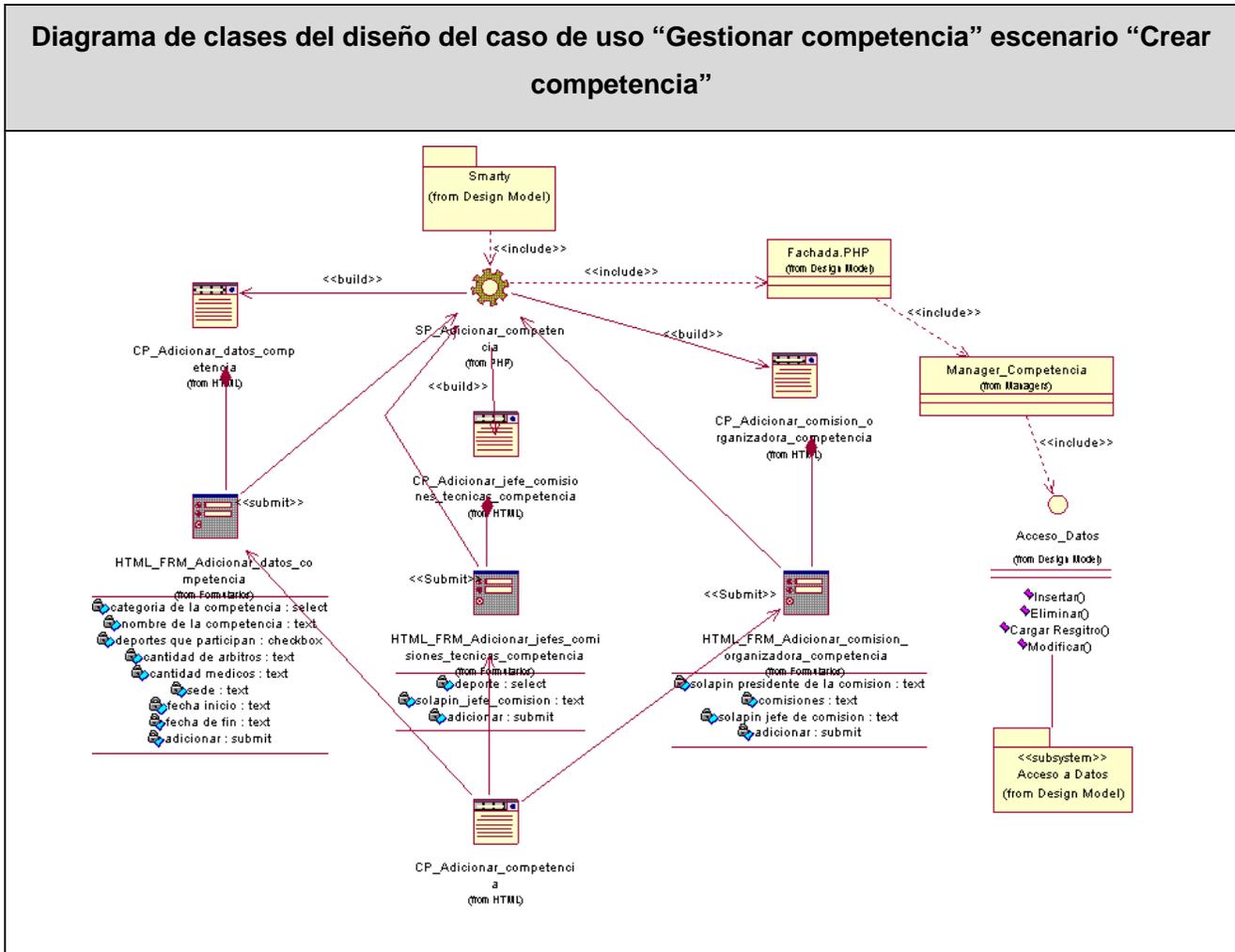


Fig. 3.3.1.1 Diagrama de clases del diseño del caso de uso “Gestionar competencia” escenario “Crear competencia”

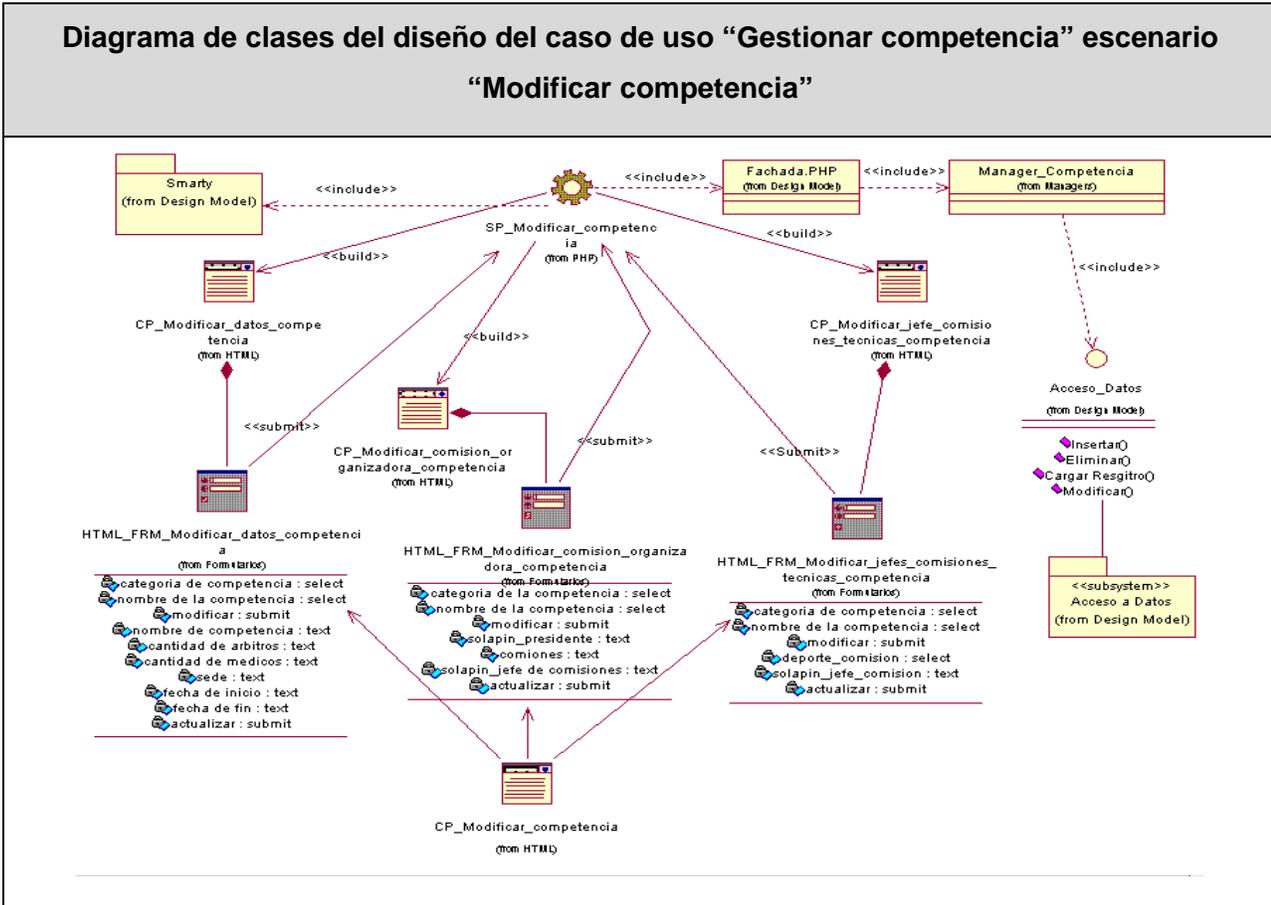


Fig. 3.3.1.2 Diagrama de clases del diseño del caso de uso “Gestionar competencia” escenario “Modificar competencia”

Los diagramas de clases del diseño del resto de los casos de uso se puede apreciar en el Anexo # 5.

### 3.3.2 Descripción textual de las clases Web.

La descripción textual de las clases del diseño ayudan a comprender el alcance y la responsabilidad de cada una de estas dentro del sistema, es decir, con dicha descripción se puede conocer qué funcionalidad específica realiza cada clase, además de la información que cada una de esta maneja.

#### 3.3.2.1 Clases Controladoras

<b>Nombre:</b>	Catleta_Manager
----------------	-----------------

<b>Tipo de clase:</b>	Controladora
<b>Responsabilidad:</b>	
<b>Nombre:</b>	Insertar_atleta(\$asolapin)
	Modificar_atleta(\$asolapin)
	Eliminar_atleta(\$asolapin)
	Listar_atletas()
	Listar_atleta_equipo()
	Listar_atleta_bajo_peso()
	Listar_atleta_competencia()
	Listar_atleta_deporte()
<b>Descripción:</b>	Controla la lógica de todo el proceso de las actividades de los atletas del sistema.

Tabla 3.3.2.1.1 Clase controladora “Catleta\_Manager”

El resto de las descripciones de las clases controladoras se pueden apreciar en el Anexo # 6.4.

### 3.3.2.2 Clases Acceso a Datos

<b>Nombre:</b>	CAcceso_Datos	
<b>Tipo de clase:</b>	DAO	
<b>Atributo</b>	<b>Tipo</b>	
servidor	string	

usuario	string
password	string
basedato	string
conexión	string
error	string
Ado_DB	ADONewConnection
<b>Responsabilidad:</b>	
<b>Nombre:</b>	Base_Datos(\$s=SERVIDOR, \$u=USUARIO, \$p=PASSWORD, \$bd = BASEDATOS)
	Abrir_Conexion()
	Cerrar_Conexion()
	Ejecutar_Consulta(sql)
	Cargar_Registro(sql)
	Empezar_Transaccion()
	Terminar_Transaccion()
	Ocurrio_Error_Transaccion()
	Forzar_Transaccion()
<b>Descripción:</b>	Clase encargada de utilizar la herramienta adodb para realizar las operaciones de acceso a la base de datos.

Tabla 3.3.2.2.1 Clase de acceso a datos “CAcceso\_Datos”

El resto de las descripciones de las clases de acceso a datos se pueden apreciar en el Anexo # 6.2.

3.3.2.3 Clases Entidades

<b>Nombre:</b>	Catleta	
<b>Tipo de clase:</b>	Entidad	
<b>Atributo</b>	<b>Tipo</b>	
solapin	varchar	
id_equipo	int	
calzado	float	
estatura	float	
peso	float	
id_color_piel	Int	
<b>Responsabilidad:</b>		
<b>Nombre:</b>	construct(\$solapin=NULL,\$aid_equipo=NULL,\$acalzado=NULL,\$aestatura=NUL L,\$apeso=NULL,\$aid_colo_piel=NULL)	
	Set()	
	Get()	
<b>Descripción:</b>	Clase entidad que representa la tabla atleta de la base de datos.	

Tabla 3.3.2.3.1 Clase entidad “Catleta”



3.3.4.1 Descripción de las tablas de la base de datos.

<b>Nombre:</b>	tb_atleta	
<b>Descripción:</b>	Esta tabla almacena a los atletas y algunos datos adicionales de ellos	
<b>Atributo</b>	<b>Tipo</b>	<b>Descripción</b>
solapin	Varchar	Es un atributo que hace referencia a la tabla (tb_persona)
id_equipo	Int	Es un atributo que hace referencia a la tabla (tb_equipo).
calzado	Varchar	Es un atributo que tiene el tamaño del pie.
estatura	Float	Es un atributo que tiene la estatura.
peso	Float	Es un atributo que tiene el peso.
id_color_piel	Int	Es un atributo que hace referencia a la tabla (no_color_piel).

3.3.4.1.1 Descripción de la tabla de base de datos “tb\_atleta”

El resto de las descripciones de las tablas de la base de datos se pueden observar en el Anexo # 8.

**3.3.5 Principios de diseño**

Un patrón de diseño es una descripción de clases y objetos comunicándose entre sí adaptada para resolver un problema de diseño general en un contexto particular. Normalmente los patrones están especificados en gráficos de UML y utilizan la semántica de la orientación a objetos.

El patrón seguido en la aplicación propuesta es el patrón n capas.

Existen muchas razones por las que usar el desarrollo de aplicaciones bajo la arquitectura de n-Capas:

- 🖥 **Abstracción total acerca del origen de datos.** Las distintas capas se especializan absolutamente en la funcionalidad que deben brindar (procesamiento en las reglas de negocios o presentación de datos en la capa cliente) sin importar cual es el origen de los datos procesados.
- 🖥 **Bajo costo de desarrollo y mantenimiento de las aplicaciones.** Si bien al momento del diseño podemos observar una mayor carga de complejidad, la utilización de esta arquitectura nos brinda un control más cercano de cada componente, así como también la posibilidad de una verdadera reutilización del código.
- 🖥 **Estandarización de las reglas de negocio.** Las reglas de negocio se encuentran encapsuladas en un set de rutinas comunes y pueden ser llamadas desde diversas aplicaciones sin necesidad de saber cómo esta funciona o ha sido diseñada.
- 🖥 **Mejor calidad en las aplicaciones.** Como las aplicaciones son construidas en unidades separadas, estas pueden ser testeadas independientemente y con mucho mas detalle, esto conduce a obtener un producto mucho mas sólido.
- 🖥 **Reutilización de código.** La concepción natural de un sistema desarrollado con esta arquitectura, promueve la reutilización de sus componentes en varias partes del propio desarrollo y de futuros sistemas. (Web)

En la capa de acceso a datos se utiliza el patrón de acceso a datos DAO.

El patrón DAO (Data Access Object) es una solución al problema del diferencial de impedancia entre un programa de aplicación orientado a objetos y una base de datos relacional, empleando únicamente la interfaz de programación (API) nativa del manejador de base de datos, o algún otro sustituto como el ODBC, DBI, etcétera. El propósito del patrón DAO es abstraer y encapsular todos los accesos a la fuente de datos. Con esto se obtiene varias ventajas:

- 🖥 Se tiene un paliativo al problema del diferencial de impedancia (transparencia)
- 🖥 Se baja en nivel de acoplamiento entre clases, reduciendo la complejidad de realizar cambios
- 🖥 Se aísla las conexiones a la fuente de datos en una capa fácilmente identificable y de fácil mantenimiento. (DAO, 2007)

En la capa presentación se utiliza el patrón fachada.

El patrón Fachada proporciona una interfaz unificada de alto nivel para un subsistema, denominada “fachada”, que oculta las interfaces de bajo nivel de las clases que lo implementan. Con esto se consiguen dos objetivos fundamentales: hacer el subsistema más fácil de usar y desacoplar a los clientes de las clases del subsistema. Esta clase “fachada” representa al subsistema completo y muchas veces ambos tienen el mismo nombre. El valor que añade esta clase es el ofrecer a los clientes una forma única y simplificada de acceder a los servicios más generales del subsistema. Para ello, los clientes enviarán mensajes solo a la fachada, y esta se encargará de poner en funcionamiento la maquinaria del subsistema para conseguir el objetivo pretendido y devolver al cliente los resultados. (Fachada, 2007)

### **3.3.6 Principios de diseño de interfaz.**

El diseño de una aplicación debe estar orientado a los usuarios finales. El diseño no solo se encarga de la visualización de la interfaces con las que el usuario interactúa, está vinculado también a la navegabilidad que presente la aplicación, al grado de dificultad para obtener la información requerida (usabilidad) y a como están organizados los contenidos con los que trata la aplicación.

Con el fin de llevar a cabo estos objetivos en la aplicación propuesta: SGD-UCI, se han seguido ciertos principios generales de diseño:

- 🖥️ Uso de la aplicación con un mínimo nivel de aprendizaje, bajo conocimiento de computación para utilizar el sistema.
- 🖥️ Minimizar el esfuerzo por parte del usuario para obtener la información requerida.
- 🖥️ Operaciones sencillas con el objetivo de garantizar no comprensión de las potencialidades que el sistema brinda.
- 🖥️ Permitir que el usuario realice solo las operaciones aplicables a su rol.
- 🖥️ Mostrar mensajes de confirmación al realizar alguna operación de importancia en el sistema.
- 🖥️ No cargar la interfaz con imágenes que puedan ralentizar la comunicación entre el servidor y el cliente.
- 🖥️ Garantizar la estabilidad y uniformidad en el diseño gráfico, usando una tipografía seria y formal en los textos, así como un color que contraste con el fondo, además de un tamaño de letra capaz de visualizarse en resoluciones altas, con el objetivo de lograr la legibilidad de estos.

Prever los errores que puedan ser inducidos por mala operación del sistema por parte del usuario, así como garantizar la integridad de la información que se maneje en caso de ocurrir caídas inesperadas de la aplicación.

### 3.3.7 Interfaz de usuario

La confección del diseño de interfaz de usuario de un sistema es una tarea de importancia en el desarrollo del mismo. En muchos casos el éxito de un sistema se ha visto comprometido por una mala calidad de la interfaz. Por tales motivos para lograr una consistencia en el diseño de la interfaz del sistema se sigue el esquema de la imagen siguiente:



Fig. 3.3.7.1 Interfaz de usuario

Se recurre al uso de etiquetas (div), tablas y plantilla para el diseño de la aplicación. Otro elemento importante es el uso de CSS para guardar las configuraciones de los elementos utilizados en las páginas del sistema. De manera general se utiliza el tipo de fuente Verdana con tamaño variable entre 12 y 14 px, una combinación de colores suaves como fondo, así como el formato de los controles de los formularios.

### **3.3.8 Formato de reportes**

La información mostrada en los reportes se presenta en forma de tablas ordenadas siguiendo algunos criterios de organización, según convenga en cada uno de los reportes. La información recogida en estas tablas de los elementos es información general, con la opción de seleccionar un elemento de la tabla para ver sus datos con más detalle.

### **3.3.9 Ayuda**

El sistema SGD-UCI esta orientadO a usuarios, los cuales no todos están calificados para el manejo de herramientas sofisticadas de organización, como por ejemplo, los profesores de Educación Física, por tales razones se propone la creación de un manual de ayuda para el estudio de las funcionalidades del sistema. Al mismo se accede a través de un vínculo existente en el sistema para poder descargarlo de la aplicación, vínculo que estará ubicado en la parte superior derecha de la zona de navegación.

### **3.3.10 Tratamiento de errores.**

En el sistema pueden ocurrir dos tipos de errores: los que origina el usuario por la entrada incorrecta de algún dato y los que pueden generarse durante alguna operación de conexión con la base de datos.

Para el primer grupo se realiza una serie de funciones de validación de entrada de datos por parte del usuario, dichas funciones son implementadas en el lado del servidor con el objetivo de no utilizar funciones de javascript que luego no pudieran ejecutarse debido a la incompatibilidad que posee este lenguaje con algunos navegadores. Para la solución del segundo grupo de errores, las operaciones de conexión a la base de datos siempre se realizan a través de transacciones con el objetivo de mantener la integridad de la información en la base de datos.

### **3.3.11 Seguridad**

Para la seguridad del sistema se implementó un clase de autenticación del usuario con el objetivo de controlar que usuario está haciendo que operación y de mantener un sistema de permisos a los diferentes usuarios del sistema para realizar las operaciones que este brinda. Existe además un método de cambio de contraseña del usuario para disminuir los ataques que se realizan para descifrar las contraseñas de dichos usuarios. El sistema cuenta con un usuario encargado de la administración del sistema el cual puede en algún momento dado realizar modificaciones a los distintos usuarios del sistema y sus

contraseñas. Cada vez que se necesite realizar una transferencia entre el cliente y el servidor de información importante, se utiliza un canal seguro (SSL) el cual se encarga de transportar la información de una estación a otra.

### **3.4 Conclusiones**

En este capítulo se han presentado los diagramas de clases del análisis y del diseño de los casos de uso principales del sistema. Se hace referencia al modelo de datos con su diagrama de clases persistentes que le dio origen.

De esta manera se han logrado modelar todos los procesos que han sido objeto de estudio durante todo el transcurso del trabajo investigativo, proporcionando una idea completa de lo que realmente es el software y se materializan con precisión los requerimientos del cliente.

Los diagramas y especificaciones de diseño que se proponen constituyen una guía que puede ser fácilmente leída y comprendida por aquellos que construirán el código, lo probarán y le darán mantenimiento; lo único que resta es implementar la aplicación que se ha diseñado.

---

## *Capítulo 4: Implementación y prueba del sistema.*

### **4.1 Introducción**

En este capítulo se hará referencia a algunos elementos del modelo de implementación como son el Diagrama de Despliegue y el Diagrama de componentes. Además en este capítulo se realizarán los casos de prueba de todos los escenarios implementados en el sistema utilizando el método de pruebas de caja negra.

### **4.2 Implementación**

El modelo de implementación describe como las clases del diseño se implementan en términos de componentes, como ficheros de código fuente, ejecutables, scripts y ficheros de código binario y cómo estos se organizan de acuerdo a los nodos específicos.

#### **4.2.1 Diagrama de componentes**

Los diagramas de componentes son usados para estructurar el modelo de implementación en términos de subsistemas de implementación y mostrar las relaciones entre los elementos de implementación. También se utilizan para mostrar las dependencias de compilación de los ficheros de código, relaciones de derivación entre ficheros de código fuente y ficheros que son resultados de la compilación, dependencias entre elementos de implementación y los correspondientes elementos de diseños que son implementados.

##### **4.2.1.1 Diagrama de componentes del sistema y la Base de Datos.**

En el siguiente diagrama se muestra la relación del sistema completo como un componente y la base de datos.

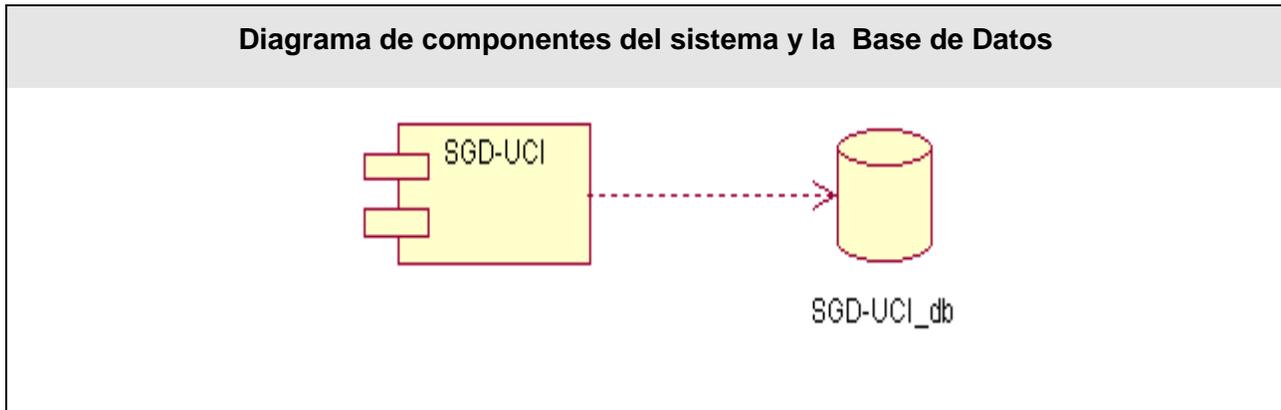


Fig. 4.2.1.1.1 Diagrama de compontes del sistema y la Base de Datos.

4.2.1.2 Diagrama de componentes del sistema.

El siguiente diagrama muestra la relación que existe entre todos los paquetes de componentes del sistema.

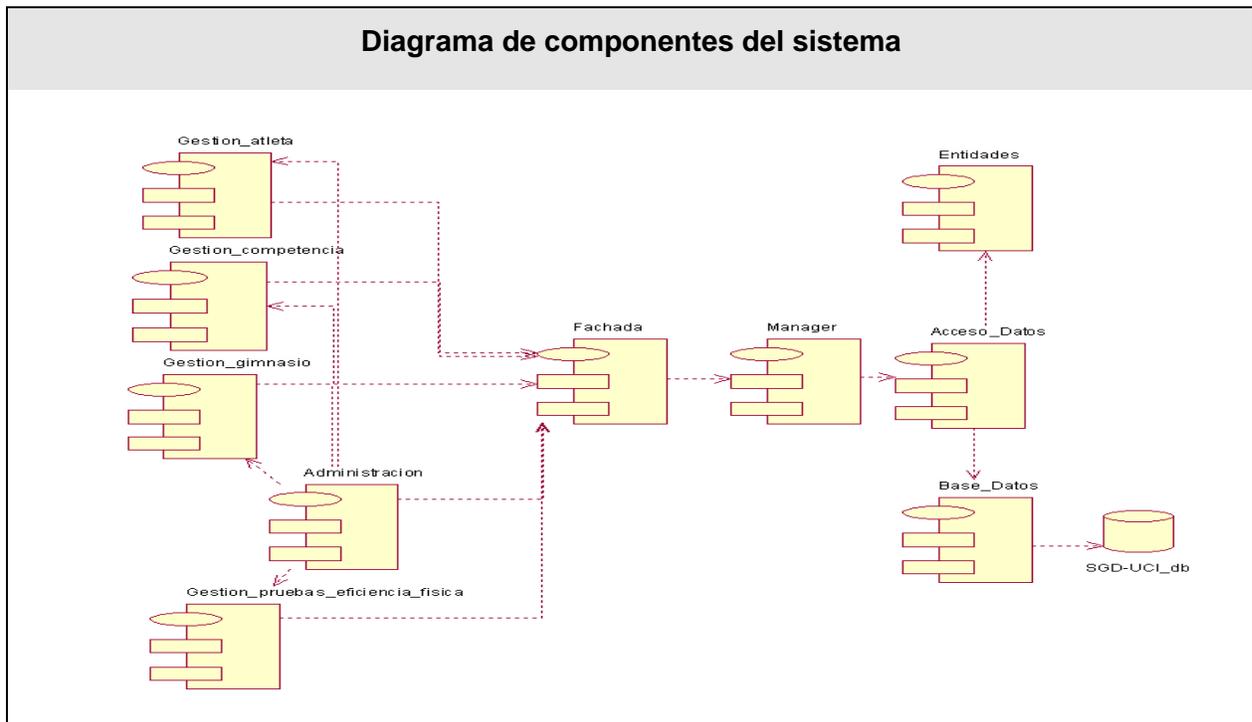


Fig. 4.2.1.2.1 Diagrama de componentes del sistema.

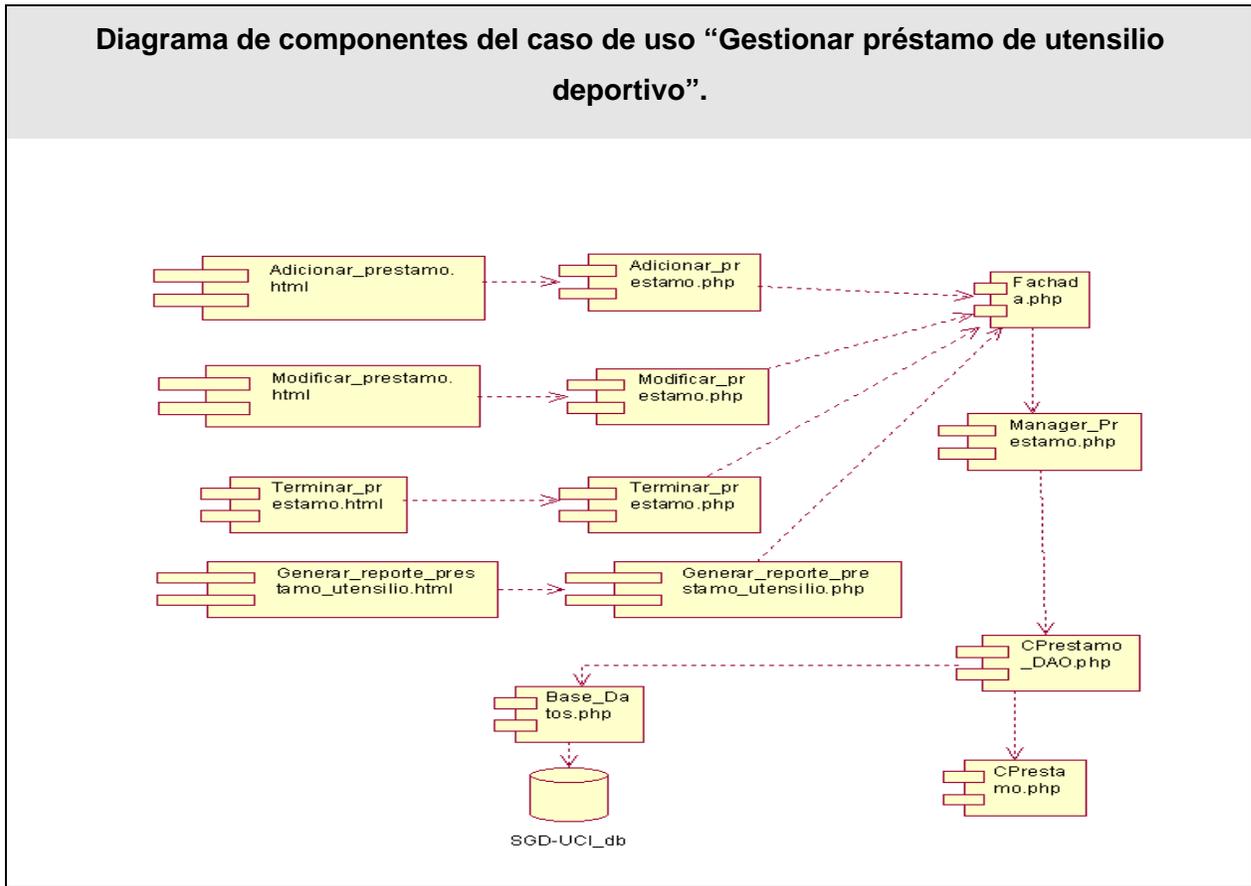


Fig. 4.2.1.2.2 Diagrama de componentes del caso de uso “Gestionar préstamo de utensilio deportivo”.

Los diagramas de componentes del resto de los casos de uso se observan en el Anexo # 9.

#### 4.2.2 Diagrama de despliegue

El diagrama de despliegue es un modelo de objetos que describe la distribución física del sistema en términos de cómo se distribuye la funcionalidad entre los nodos de cómputo. Es una colección de nodos y arcos; donde cada nodo representa un recurso de cómputo, normalmente un procesador o un dispositivo de hardware similar.

El diagrama de despliegue de la solución propuesta se puede apreciar a continuación:

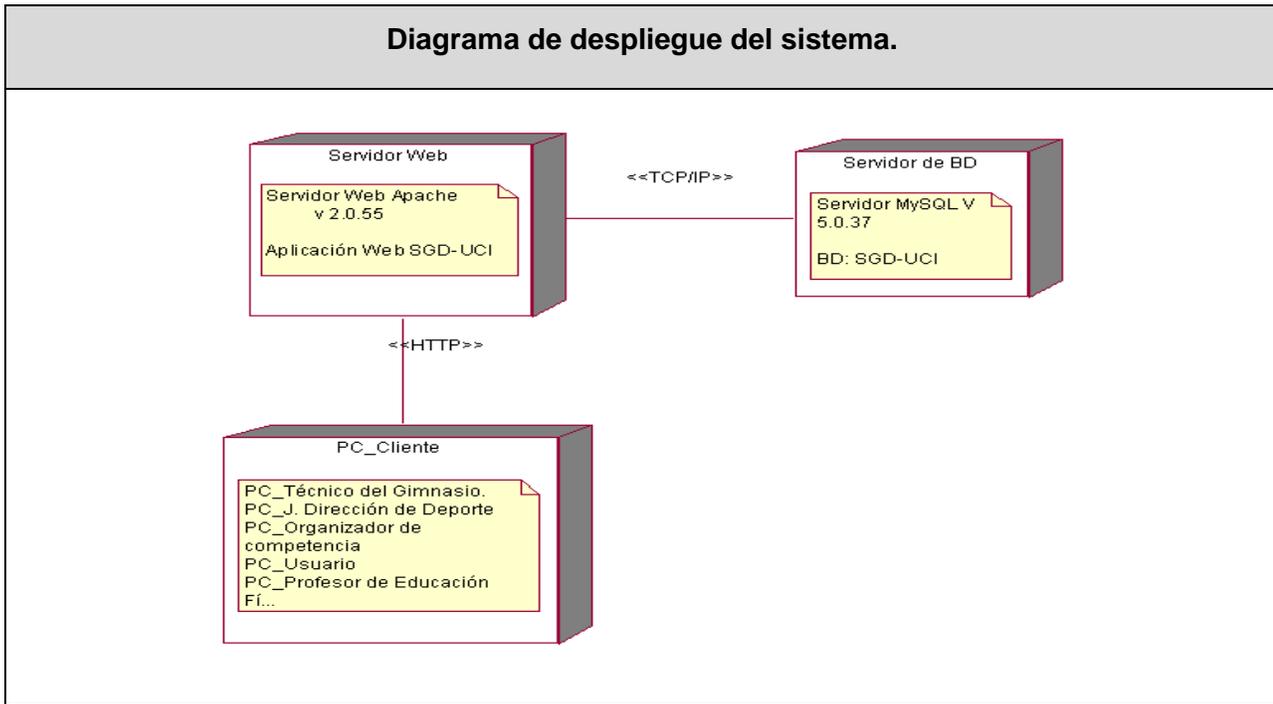


Fig. 4.2.2.1 Diagrama de despliegue del sistema.

### 4.3 Prueba

Las pruebas son una actividad en la cual un sistema o componente es ejecutado bajo unas condiciones o requerimientos específicos, los resultados son observados y registrados, y una evaluación es hecha de algún aspecto del sistema o componente.

#### 4.3.1 Prueba de caja negra

- 🖥️ Verifican las especificaciones funcionales y no consideran la estructura interna del programa.
- 🖥️ Es hecha sin el conocimiento interno del producto.
- 🖥️ No validan funciones ocultas (por ejemplo funciones implementadas pero no descritas en las especificaciones funcionales del diseño) por tanto los errores asociados a ellas no serán encontrados.

En otras palabras, la prueba de la caja negra se refiere a las pruebas que se llevan a cabo sobre la interfaz del software. O sea los casos de prueba pretenden:

- 1) Demostrar que las funciones del software son operativas.
- 2) Que las entradas se aceptan de la forma adecuada y que se produce el resultado correcto.

Condición de Entrada	Casos Válidos	Casos no Válidos
Nombre de usuario	Letras	No introducir el nombre usuario Introducir nombre de usuario incorrecto
Contraseña	Letras, números y caracteres extraños.	Introducir contraseña incorrecta

*Caso válido*

<b>Caso de uso:</b>	Autenticarse
<b>Caso de prueba:</b>	Autenticación de un usuario entrando correctamente los datos.
<b>Entrada:</b>	El usuario decide entrar al sistema para entrar a los módulos correspondientes a los que tiene permiso, para eso debe autenticarse, introduciendo su nombre de usuario y contraseña.  Nombre usuario: efouces  Contraseña: erich
<b>Resultado:</b>	El sistema comprueba la valides de la información entrada y el usuario accede al sistema para trabajar.

<b>Condiciones:</b>	Los datos introducidos deben ser compatibles con algún usuario de la base de datos.
---------------------	---

*Caso inválido*

<b>Caso de uso:</b>	Autenticarse
<b>Caso de prueba:</b>	Autenticación de un usuario entrando algún dato erróneo.
<b>Entrada:</b>	El usuario decide entrar al sistema para entrar a los módulos correspondientes a los que tiene permiso, para eso debe autenticarse, introduciendo su nombre de usuario y contraseña.  Nombre usuario: efoucesss  Contraseña: erich
<b>Resultado:</b>	El sistema comprueba la validez de la información entrada, muestra un mensaje notificando que el usuario o contraseña no es válido.
<b>Condiciones:</b>	Los datos introducidos deben ser compatibles con algún usuario de la base de datos.

**Tabla 4.3.1.1 Pruebas de Caja Negra para el Caso de uso “Autenticarse”**

Las pruebas de caja negra para el resto de los casos de uso las podemos encontrar en el Anexo # 10.

#### **4.4 Conclusiones**

En este capítulo fue presentado cómo está construido el sistema a partir de los diagramas de componentes de cada caso de uso, así como, los diagramas de componentes de todo el sistema en general. También fue mostrado el diagrama de despliegue, el cual ilustra cuáles serán los nodos que serán usados para la implantación de la aplicación. Por último, fueron detalladas un conjunto de pruebas que se le realizaron al sistema, donde se especifica qué parámetros fueron utilizados durante la realización de las mismas, además del resultado obtenido durante la ejecución de cada una de ellas.



## Conclusiones

Una vez concluida la investigación realizada, se dieron cumplimiento a los objetivos planteados, obteniéndose los siguientes resultados:

-Se logró seleccionar la metodología y herramientas adecuadas para el desarrollo del sistema informático, según las características particulares del mismo.

-Se realizó un estudio detallado de los procesos de competencia deportivo, seguimiento de atleta, realización de pruebas de eficiencia física y préstamos de utensilios deportivos en la Universidad de las Ciencias Informáticas.

-Se realizó el análisis y diseño del Sistema de Gestión Deportiva para la UCI. (SGD-UCI)

- Se llevó a cabo la implementación y prueba de los módulos de Gestión de préstamos de utensilios y Control de Acceso al Gimnasio, el módulo de Nivelación de pruebas de eficiencia física y Administración del sistema de Gestión Deportiva para la UCI.

Con el uso del sistema diseñado en el presente trabajo de diploma los beneficios serán notorios:

🖥️ Facilitará el trabajo de los profesores de Educación Física en la nivelación de las pruebas de eficiencia física que se le realizan a los estudiantes en la UCI.

🖥️ Permitirá tener por parte de la Dirección de Deporte de la UCI un control de acceso al gimnasio y de los préstamos de los utensilios deportivos.

🖥️ El sistema brindará información necesaria para próximas investigaciones sobre rendimiento físico de estudiantes universitarios en las diferentes edades de este período.

## **Recomendaciones.**

Luego de la presentación del estudio realizado que culmina con la implementación y prueba de los módulos de Gestión de préstamos de utensilios y Control de Acceso al Gimnasio, el módulo de Nivelación de pruebas de eficiencia física y Administración del sistema de Gestión Deportiva para la UCI, se listan a continuación una serie de recomendaciones para la ampliación, modificación, mejora y construcción de nuevas versiones de este sistema:

- 🖥️ Continuar con la implementación y prueba de los módulos de Gestión de competencias deportivas y Gestión de atleta, basándose en el análisis y diseño de estos módulos propuesto en el presente trabajo.
- 🖥️ Modelar el sistema en iteraciones posteriores de forma tal que, pueda ser extendido a todas las Instituciones de la Enseñanza Superior en el país.
- 🖥️ Realizar un estudio de la factibilidad del sistema con el objetivo de definir la viabilidad económica del mismo.
- 🖥️ Realizar un Glosario de Términos que permita un mayor esclarecimiento acerca de vocablos citados en el presente trabajo relacionados con el deporte, la informática y la ingeniería de software.

## Referencias Bibliográficas

*Apuntes para un historia de la documentación deportiva.* **Vegas, José Aquesolo. 2000.** Barcelona : Revista General de Información y Documentación, 2000.

**Architect, Enterprise.** Enterprise Architect. [En línea]  
<http://www.sparxsystems.com.ar/products/ea.html>.

**Cagigal, José María. 1981.** *!OH Deporte. Anatomía de un gigante.* Valladolid, España : Editorial Miñón, 1981.

—. **1966.** *Deporte, Pedagogía y Humanismo.* Madrid : Publicaciones del Comité Olímpico de España, 1966.

**Coubertin, Pierre de. 1959.** *Memorias Olímpicas.* Frankfurt, Limpert : s.n., 1959.

**DAO. 2007.** Patrón Acceso a Datos. [En línea] 20 de junio de 2007. <http://www.proactiva-calidad.com/java/patrones/DAO.html>.

**Devís, José Devís. 1996.** *Educación Físoca, deporte y Curriculum. Investigación y desarrollo curricular.* Aprendizaje visor. Madrid : s.n., 1996.

**Fachada. 2007.** Patrón Fachada. *Arquitectura n-Capas.* [En línea] 2007. [Citado el: 26 de junio de 2007.] <http://zarza.fis.usal.es/~fgarcia/docencia/poo/03-04/Trabajos/Fachada.pdf>.

*Finding the righth balance for sport.* **Towley, Stephen. enero de 1998.** s.l. : SPORTVISION magazine de la GAIFS, enero de 1998.

**I. Jacobson, G. Booch, J. Rumbaugh. 2000.** *El Proceso Unificado de Desarrollo de software.* s.l. : Addison-Wesley, 2000.

**INDER, Fuente.** Sitio Web Oficial del Instituto Nacional de Deporte, Educación Física y Recreación. [En línea] <http://www.inder.co.cu>.

**JAVA.** Características de JAVA. [En línea] [http://pisuerga.inf.ubu.es/lsi/Invest/Java/Tuto/I\\_3.htm](http://pisuerga.inf.ubu.es/lsi/Invest/Java/Tuto/I_3.htm).

**M. Ricote, A y Figueredo. junio 2006.** *SGIF: Sistema de Gestión de la Información para la facultad.* Instituto Superior Politécnico “José Antonio Echeverría”, Ciudad de la Habana : Trabajo de Diploma para optar por el título de Ingeniero Informático, junio 2006.

*Modelo educativo del deporte.* **Alcaraz, Ramón Cantó.** 2007. 006, Madrid, España : Revista general de Información y Documentación, 2007, Vol. III.

**MySQL-PostgreSQL.** Comparativa. [En línea]

[http://www.netpecos.org/docs/mysql\\_postgres/x108.html](http://www.netpecos.org/docs/mysql_postgres/x108.html).

—. Comparativa. [En línea] [http://www.netpecos.org/docs/mysql\\_postgres/x57.html](http://www.netpecos.org/docs/mysql_postgres/x57.html).

**OMG.** 2003. *OMG Unified Modeling Language Specification*. . 2003.

**PHP.** Wikipedia. [En línea] [http://es.wikipedia.org/wiki/PHP#Ventajas\\_de\\_PHP](http://es.wikipedia.org/wiki/PHP#Ventajas_de_PHP).

**PHPEd, NuSphere.** NuSphere PHPEd. [En línea] <http://nusphere-phped.softonic.com/>.

**PostgreSQL.** NETPECOS. [En línea] [http://www.netpecos.org/docs/mysql\\_postgres/x15.html](http://www.netpecos.org/docs/mysql_postgres/x15.html).

**Programación, Lenguaje de.** Wikipedia. [En línea] [Citado el: 16 de abril de 2007.]

[http://es.wikipedia.org/wiki/Lenguaje\\_de\\_programaci%C3%B3n](http://es.wikipedia.org/wiki/Lenguaje_de_programaci%C3%B3n).

**Renson, Roland.** 1999. *Palin et hockeyspel va de Mapuche-Indianen uit Chili*. 1999.

**Rose, Enterprise Architect y Rational.** Comparación de Herramientas de Modelado UML. [En línea] <http://www.apexnet.com.ar/index.php/news/main/38/event=view>.

**Rose, Rational.** Procedimientos básicos para desarrollar un proyecto con UML. [En línea]

<http://www.vico.org/TallerRationalRose.pdf>.

**Schmuller, J.** 2005. *Aprendiendo UML en 24 horas*. s.l. : Prentice Hall, 2005.

**Smarty.** Wikipedia. [En línea] <http://es.wikipedia.org/wiki/Smarty>.

**Software, Metodologías de Desarrollo de.** Metodologías de Desarrollo de Software. [En línea]

[http://www.informatizate.net/articulos/metodologias\\_de\\_desarrollo\\_de\\_software\\_07062004.htm](http://www.informatizate.net/articulos/metodologias_de_desarrollo_de_software_07062004.htm).

**Studio, Zend.** Editores para PHP . [En línea]

<http://www.adrformacion.com/cursos/php/leccion1/tutorial3.html>.

**Web, Arquitectura.** Arquitectura Web. [En línea] [Citado el: 22 de junio de 2007.]

<http://www.di.uniovi.es/~dflanvin/docencia/dasdi/teoria/Transparencias/06.%20Arquitectura%20Web.pdf>.



## Bibliografía

*Apuntes para un historia de la documentación deportiva.* **Vegas, José Aquesolo. 2000.** Barcelona : Revista General de Información y Documentación, 2000.

**Cagigal, José María. 1981.** *!OH Deporte. Anatomía de un gigante.* Valladolid, España : Editorial Miñón, 1981.

—. **1966.** *Deporte, Pedagogía y Humanismo.* Madrid : Publicaciones del Comité Olímpico de España, 1966.

**Coubertin, Pierre de. 1959.** *Memorias Olímpicas.* Frankfurt, Limpert : s.n., 1959.

**Devís, José Devís. 1996.** *Educación Físoca, deporte y Curriculum. Investigación y desarrollo curricular.Aprendizaje visor.* Madrid : s.n., 1996.

*Finding the rigth balance for sport.* **Towley, Stephen. enero de 1998.** s.l. : SPORTVISION magazine de la GAIFS, enero de 1998.

**I. Jacobson, G. Booch, J. Rumbaugh. 2000.** *El Proceso Unificado de Desarrollo de software.* s.l. : Addison-Wesley, 2000.

**M. Ricote, A y Figueredo. junio 2006.** *SGIF: Sistema de Gestión de la Información para la facultad.* Instituto Superior Politécnico “José Antonio Echeverría”, Ciudad de la Habana : Trabajo de Diploma para optar por el título de Ingeniero Informático, junio 2006.

*Modelo educativo del deporte.* **Alcaraz, Ramón Cantó. 2007.** 006, Madrid, España : Revista general de Información y Documentación, 2007, Vol. III.

**OMG. 2003.** *OMG Unified Modeling Language Specification.* . 2003.

**Renson, Roland. 1999.** *Palin et hockeyspel va de Mapuche-Indianen uit Chili.* 1999.

## Anexos

### Anexo #1: Fases y flujos de trabajo de la metodología de desarrollo RUP.

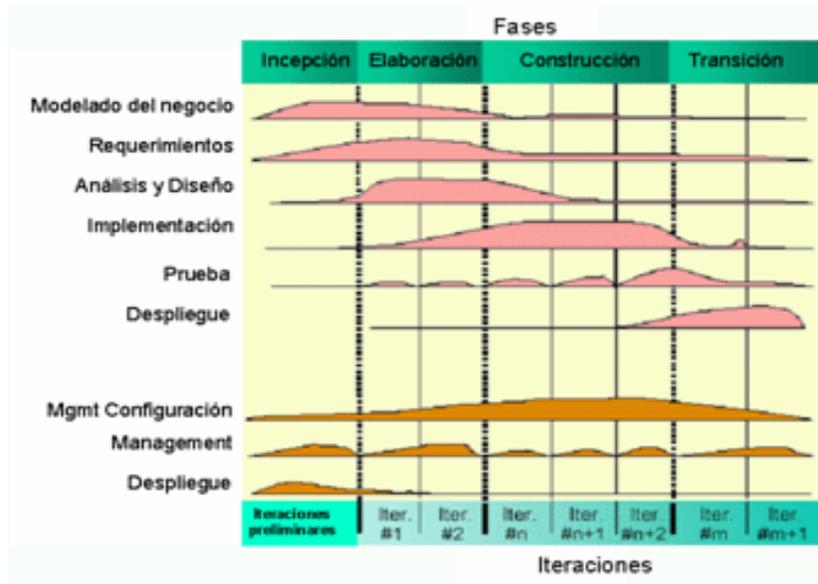


Fig. A1.1 Rational Unified Process.

### Anexo # 2: Descripción textual y diagrama de actividades de los casos de uso del negocio.

<b>Caso de Uso</b>	Visitar el gimnasio.
<b>Actores</b>	Usuario (inicia)
<b>Propósito</b>	Visitar el gimnasio deportivo de la universidad para realizar actividades físicas y deportivas
<b>Resumen</b>	El CUN se inicia cuando el usuario solicita la entrada al gimnasio. El técnico del gimnasio solicita su solapín de la universidad, registra sus datos y le permite el acceso. El usuario entra a la instalación y realiza ejercicios físicos. El usuario comunica que se retira del área y el técnico del gimnasio registra la hora de salida

	del usuario y termina el CUN.
<b>Curso normal de los eventos</b>	
<b>Acción del actor</b>	<b>Respuesta del Negocio</b>
1. El usuario solicita la entrada al gimnasio	1.1 El técnico solicita el solapín al usuario
2. El usuario le hace entrega de su solapín al técnico del gimnasio.	2.1 Si el usuario le entrega el solapín, el técnico registra los datos del usuario y le permite la entrada a la instalación.
3. El usuario realiza ejercicios físicos.	
4. Al finalizar el usuario le comunica al técnico del gimnasio que desea retirarse del gimnasio.	4.1 El técnico del gimnasio registra los datos de terminación de la visita.
5. El usuario se retira de la instalación y termina el CUN.	
<b>Curso alterno de los eventos.</b>	
<b>Acción 2.1</b>	Si el usuario no hace entrega de su solapín al técnico del gimnasio, este le informa que no puede acceder al interior del gimnasio.

Tabla A2.1 Descripción textual del CUN "Visitar el gimnasio".

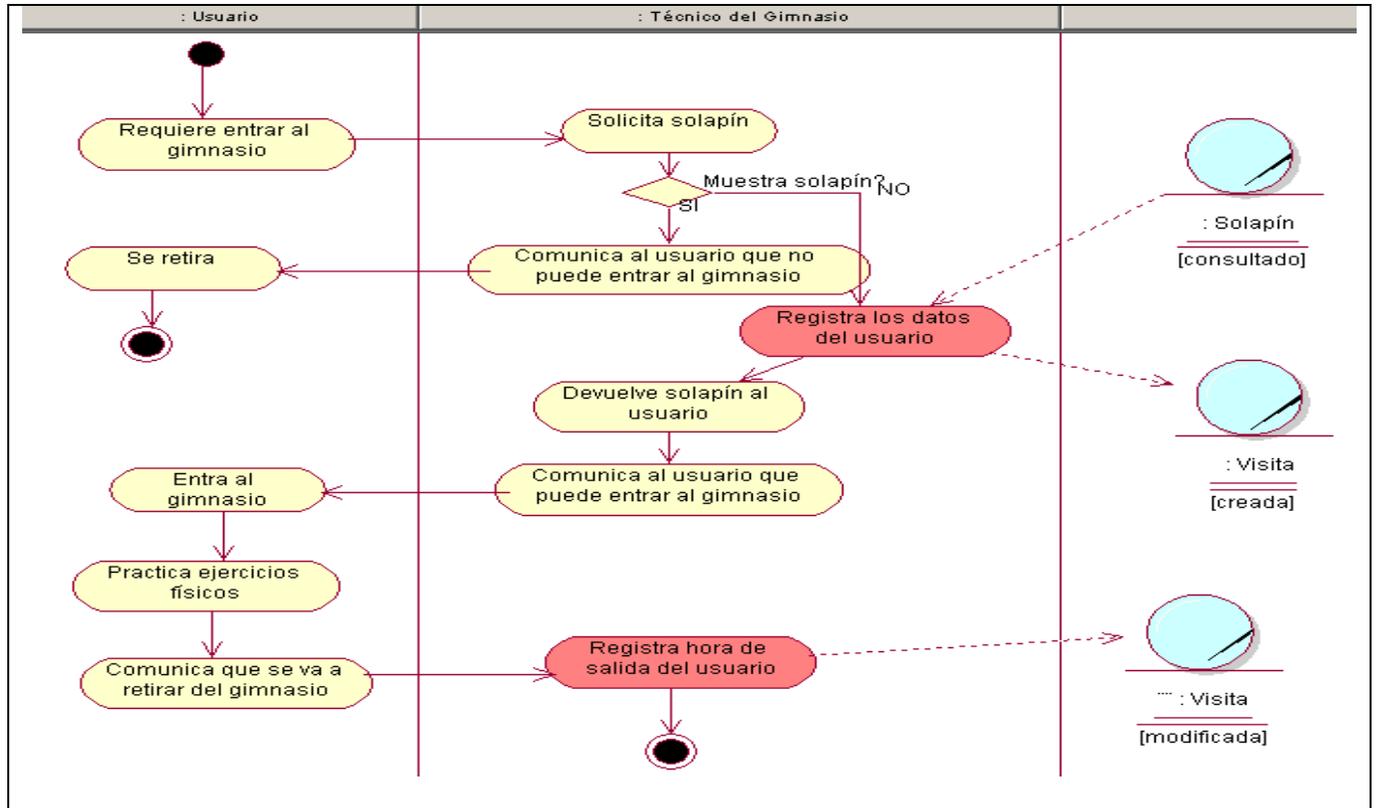


Fig. A2.1 Diagrama de actividades del caso de uso del negocio “Visitar el gimnasio”

<b>Caso de Uso</b>	<b>Realizar prueba de eficiencia física.</b>
<b>Actores</b>	Estudiante. (inicia)
<b>Propósito</b>	Le permite al profesor de Educación Física nivelar a sus estudiantes según los resultados obtenidos en cada indicador medido en la prueba.
<b>Resumen</b>	El CUN de uso se inicia cuando el estudiante asiste al turno de Educación Física. El profesor le comunica la realización de la prueba de eficiencia física. El profesor de Educación Física realiza la prueba y registra los resultados obtenidos por cada estudiante en cada indicador. El profesor consulta la tabla Normativas del Plan de Eficiencia Física de la UCI y la tabla para determinar niveles de Física, nivela a

	sus estudiantes, le comunica el nivel alcanzado y termina el CUN.
<b>Curso normal de los eventos</b>	
<b>Acción del actor</b>	<b>Respuesta del Negocio</b>
1. EL estudiante asiste al turno de Educación Física.	1.1 El profesor le comunica a sus estudiantes la realización de la pruebas de eficiencia física.
	1.2 El profesor menciona los indicadores a medir.
2. El estudiante realiza el conjunto de ejercicios para medir los indicadores.	2.1 El profesor de Educación Física registra el resultado de cada estudiante en cada indicador.
	2.2 El profesor de Educación Física consulta la tabla de Normativas del Plan de Eficiencia Física para nivelar al estudiante en cada indicador.
	2.3 El profesor de Educación Física consulta la tabla para determinar niveles en Física para asignarle el nivel general a cada estudiante.
	2.4 El profesor le comunica la estudiante el nivel alcanzado en la prueba de eficiencia física y termina el CUN

Tabla A2.2 Descripción textual del caso de uso del negocio “Realizar prueba de eficiencia física”.

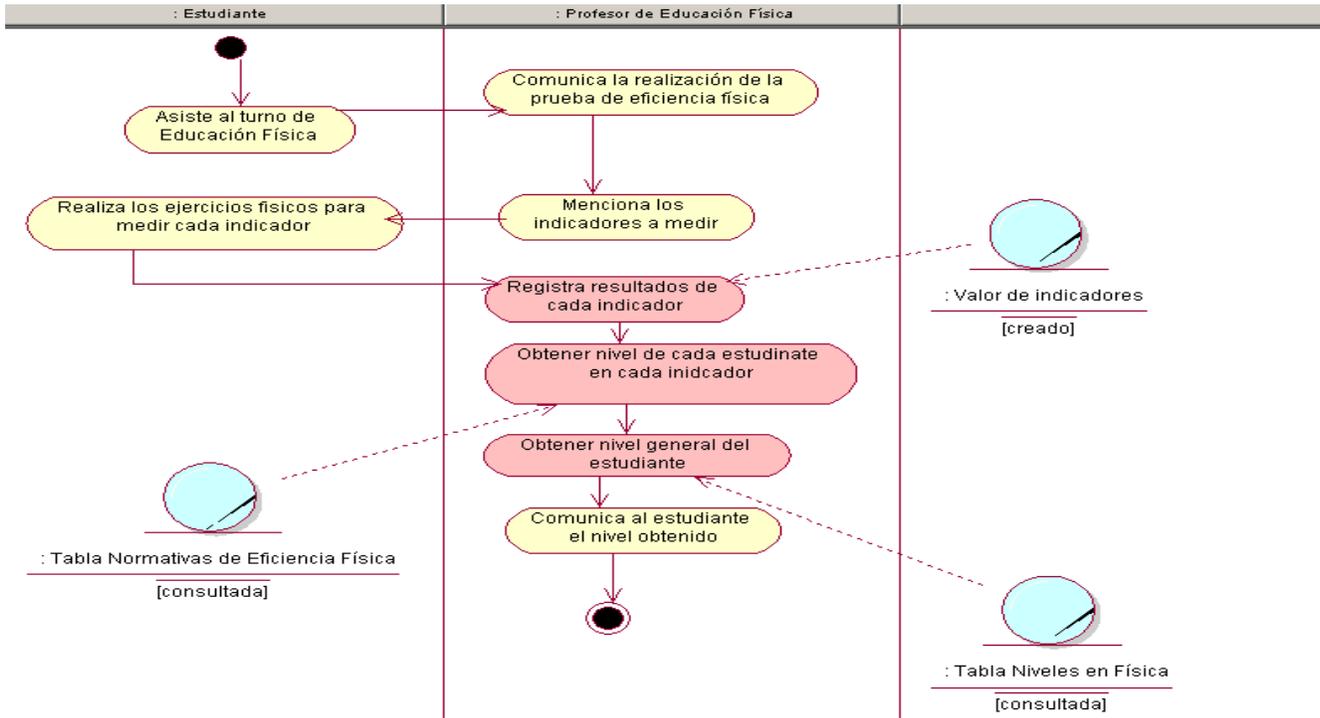


Fig. A2.2 Diagrama de actividades del caso de uso del negocio “Realizar prueba de eficiencia física”

<b>Caso de Uso</b>	<b>Competir en un enfrentamiento deportivo.</b>	
<b>Actores</b>	Atleta. (inicia)	
<b>Propósito</b>	Le permite al organizador de la competencia en que participa el atleta, conocer los datos personales del atleta y sus resultados en el enfrentamiento deportivo.	
<b>Resumen</b>	El CUN de uso se inicia el atleta decide participar en un enfrentamiento deportivo. El organizador de la competencia le registra sus datos personales y le comunica los datos de su oponente en el enfrentamiento. El atleta realiza el enfrentamiento. El árbitro del enfrentamiento controla el curso del enfrentamiento y le trasmite los resultados al organizador de competencia. Este registra los resultados y termina el CUN.	
<b>Curso normal de los eventos</b>		
<b>Acción del actor</b>	<b>Respuesta del Negocio</b>	

1. El atleta decide competir en un enfrentamiento deportivo.	1.1 El organizador de competencia registra los datos personales y físicos del atleta.
	1.2 El organizador de competencia le comunica los datos de su oponente.
2. Realiza el enfrentamiento deportivo.	2.1 El árbitro del enfrentamiento controla el curso del enfrentamiento deportivo.
	2.2 El árbitro le trasmite los resultados del enfrentamiento al organizador de competencia.
	2.3 El organizador de competencia registra los resultados del enfrentamiento y termina el CUN.

Tabla A2.3 Descripción textual del caso de uso del negocio “Competir en un enfrentamiento deportivo”.

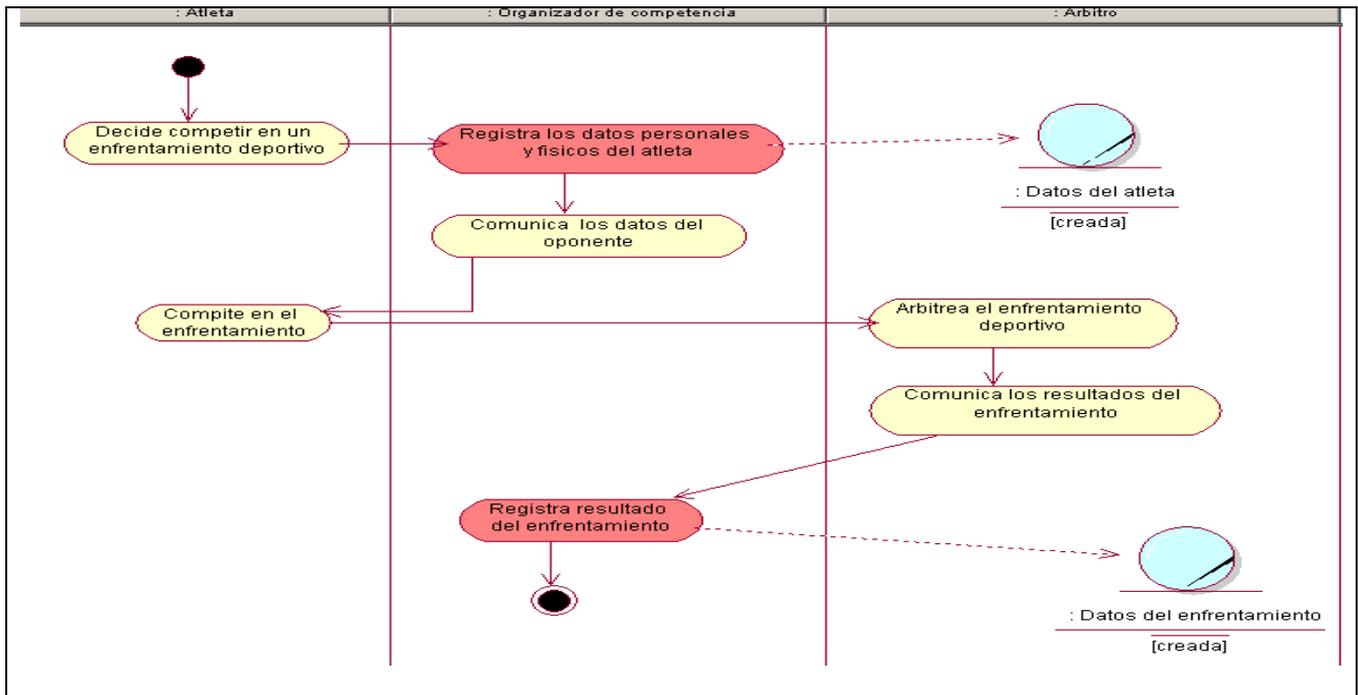


Fig. A2.3 Diagrama de actividades de los casos de uso “Competir en un enfrentamiento deportivo”.

**Anexo # 3: Descripción textual en formato expandido de los casos de usos del sistema.**

<b>Caso de Uso</b>	<b>Gestionar prueba de eficiencia física.</b>	
<b>Actores</b>	Profesor de Educación Física (inicia).	
<b>Propósito</b>	Le permite al profesor de Educación Física adicionar, modificar y eliminar prueba de eficiencia física de un estudiante en el actual curso escolar y mostrar el nivel del estudiante en cada indicador y el general.	
<b>Resumen</b>	El CUS se inicia cuando el profesor de Educación Física selecciona la opción de Gestionar prueba de eficiencia física. El sistema muestra las opciones Adicionar prueba de eficiencia física, Modificar prueba de eficiencia física, Eliminar prueba de eficiencia física y Mostrar nivel del estudiante. El profesor de educación física sigue las instrucciones del sistema e introduce los datos necesarios para la realización de estas acciones. Estos datos se registran en la Base de Datos si son correctos y termina el CUS.	
<b>Referencias</b>	<b>RF 1</b>	
<b>Precondiciones</b>	1. Indicadores de prueba de eficiencia física almacenados en la Base de Datos.	
<b>Poscondiciones</b>	<ol style="list-style-type: none"> <li>1. Adicionada prueba de eficiencia física a la Base de Datos.</li> <li>2. Información de la prueba de eficiencia física, modificada en la Base de Datos.</li> <li>3. Prueba de eficiencia física eliminada de la Base de Datos.</li> </ol>	
<b>Curso normal de los eventos</b>		
<b>Acción del actor</b>	<b>Respuesta del sistema</b>	
1. El profesor de Educación Física selecciona la opción de Gestionar prueba de eficiencia física.	1.1 El sistema muestra las opciones: Adicionar prueba de eficiencia física, Modificar prueba de eficiencia física, Eliminar prueba de eficiencia	

	física y Mostrar nivel del estudiante.
<b>Escenario 1: Adicionar prueba de eficiencia física.</b>	
1. El profesor de Educación Física selecciona la opción de Adicionar prueba de eficiencia física.	1.1 El sistema muestra un formulario para insertar el número de solapín del estudiante que va a realizar la prueba de eficiencia física.
2. El profesor de Educación Física introduce los datos solicitados por el sistema y selecciona la opción Comprobar.	2.1 El sistema verifica los datos introducidos por el profesor de Educación Física.
	2.2 Si los datos introducidos son correctos, el sistema muestra un listado con los indicadores de prueba de eficiencia física existentes en la Base de Datos.
3. El profesor de Educación Física selecciona los indicadores a medir en la prueba de eficiencia física.	3.1 El sistema muestra un formulario para adicionar los datos correspondientes a la prueba de eficiencia.
4. El profesor de Educación Física introduce los datos solicitados por el sistema y selecciona la opción Adicionar.	4.1 El sistema verifica los datos introducidos por el profesor de Educación Física.
	4.2 Si los datos introducidos son correctos, el sistema adiciona la prueba de eficiencia física a la Base de Datos y termina el CUS.
<b>Curso alterno de los eventos</b>	
<b>Acción 2.2</b>	Si los datos introducidos por el profesor de Educación Física son incorrectos, el sistema muestra un mensaje de error indicando que el usuario no existe en la Base de Datos e indica al usuario retornar a la acción 2.

<p><b>Acción 4.2</b></p>	<p>Si los datos introducidos por el profesor de Educación Física son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato erróneo e indica al usuario retornar a la acción 4.</p>
<p><b>Escenario 2: Modificar prueba de eficiencia física.</b></p>	
<p>1. El profesor de Educación Física selecciona la opción de Modificar prueba de eficiencia física.</p>	<p>1.1 El sistema muestra un formulario para insertar el número de solapín del estudiante que realizó la prueba de eficiencia física que se desea modificar.</p>
<p>2. El profesor de Educación Física introduce los datos solicitados por el sistema y selecciona la opción Comprobar.</p>	<p>2.1 El sistema verifica los datos introducidos por el profesor de Educación Física.</p>
	<p>2.2 Si los datos introducidos son correctos, el sistema muestra un listado con las fechas de las pruebas de eficiencia física realizadas por el estudiante en el actual curso escolar existentes en la Base de Datos.</p>
<p>3. El profesor de Educación Física selecciona la fecha de la prueba que desea modificar sus datos y selecciona la opción Modificar.</p>	<p>8.1 El sistema localiza los datos de la prueba de eficiencia física y los muestra listos para ser modificados.</p>
<p>4. El profesor de Educación Física realiza los cambios necesarios a los datos y selecciona la opción Actualizar.</p>	<p>4.1 El sistema verifica los datos modificados por el profesor de Educación Física.</p>
	<p>4.2 Si los datos están correctos el sistema actualiza los datos de la prueba de eficiencia física en la Base de Datos y termina el CUS.</p>

<b>Curso alternativo de los eventos.</b>	
<b>Acción 2.2</b>	Si los datos introducidos por el profesor de Educación Física son incorrectos, el sistema muestra un mensaje de error indicando que el usuario no existe en la Base de Datos e indica al usuario retornar a la acción 2.
<b>Acción 4.2</b>	Si los datos introducidos por el profesor de Educación Física son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato erróneo e indica al usuario retornar a la acción 4.
<b>Sección 3: Eliminar prueba de eficiencia física.</b>	
1. El profesor de Educación Física selecciona la opción de Eliminar prueba de eficiencia física.	1.1 El sistema muestra un formulario para insertar el número de solapín del estudiante que realizó la prueba de eficiencia física que se desea eliminar.
2. El profesor de Educación Física introduce los datos solicitados por el sistema y selecciona la opción Comprobar.	2.1 El sistema verifica los datos introducidos por el profesor de Educación Física.
	2.2 Si los datos introducidos son correctos, el sistema muestra un listado con las fechas de las pruebas de eficiencia física realizadas por el estudiante en el actual curso escolar existentes en la Base de Datos.
3. El profesor de Educación Física selecciona la o las pruebas de eficiencia físicas que desea eliminar y selecciona la opción Eliminar.	4.1 El sistema muestra un mensaje de advertencia para la acción a realizar.
4. El profesor de Educación Física	5.1 Si el profesor de Educación Física acepta, el

confirma si desea o no eliminar la o las pruebas de eficiencia físicas seleccionadas.	sistema elimina la o las pruebas de eficiencia física seleccionada en la Base de Datos y termina el CUS.
<b>Curso alternativo de los eventos.</b>	
<b>Acción 2.2</b>	Si los datos introducidos por el profesor de Educación Física son incorrectos, el sistema muestra un mensaje de error indicando que el usuario no existe en la Base de Datos e indica al usuario retornar a la acción 2.
<b>Acción 4.1</b>	Si el profesor de Educación Física cancela la acción, se culmina el CUS sin ejecutar ninguna acción
<b>Escenario 4: Mostrar nivel de un estudiante.</b>	
3. El profesor de Educación Física selecciona la opción de Mostrar nivel de un estudiante.	1.2 El sistema muestra un formulario para insertar el número de solapín del estudiante que se desea conocer su nivel.
4. El profesor de Educación Física introduce los datos solicitados por el sistema y selecciona la opción Comprobar.	2.3 El sistema verifica los datos introducidos por el profesor de Educación Física.
	2.4 Si los datos introducidos son correctos, el sistema muestra un listado con las fechas de las pruebas de eficiencia física realizadas por el estudiante en el actual curso escolar existentes en la Base de Datos.
5. El profesor de Educación Física selecciona la fecha de la prueba que se desea saber el nivel del estudiante y selecciona Mostrar.	8.2 El sistema muestra el nivel del estudiante en cada indicador y el nivel general en la prueba seleccionada.

Curso alternativo de los eventos.	
<b>Acción 2.2</b>	Si los datos introducidos por el profesor de Educación Física son incorrectos, el sistema muestra un mensaje de error indicando que el usuario no existe en la Base de Datos e indica al usuario retornar a la acción 2.
<b>Prioridad</b>	Crítico

Tabla A3.1 Descripción del caso de uso del sistema “Nivelar prueba de eficiencia física”.

Caso de Uso	Gestionar de atleta
<b>Actores</b>	Entrenador (inicia).
<b>Propósito</b>	Le permite al entrenador adicionar, modificar y eliminar un atleta y mostrar los datos del atleta en las competencias que ha participado.
<b>Resumen</b>	El CUS se inicia cuando el entrenador selecciona la opción Gestionar atleta. El sistema muestra las opciones, Adicionar atleta, Modificar atleta, Eliminar atleta y Mostrar resultados de atleta en competencia. El entrenador selecciona la opción que desea realizar e introduce al sistema los datos requeridos. El sistema muestra el resultado de la acción del entrenador. Estos datos se registran en la Base de Datos si son correctos y termina el CUS.
<b>Referencias</b>	<b>RF 2</b>
<b>Precondiciones</b>	1 Color de piel almacenado en la Base de Datos.
<b>Poscondiciones</b>	1. Atleta adicionado a la Base de Datos. 2. Información del atleta, modificada en la Base de Datos. 3. Atleta eliminado de la Base de Datos.
Curso normal de los eventos	
<b>Acción del actor</b>	<b>Respuesta del sistema</b>

1. El entrenador selecciona la opción de Gestionar atleta.	1.1 El sistema muestra las opciones: Adicionar atleta, Modificar atleta, Eliminar atleta y Mostrar resultado de atleta en competencia.
<b>Escenario 1: Adicionar Atleta.</b>	
1. El entrenador selecciona la opción de Adicionar atleta.	1.2 El sistema muestra el formulario a completar para la adición de un nuevo atleta.
2. El entrenador introduce los datos solicitados por el sistema y selecciona la opción Adicionar.	2.1 El sistema verifica los datos introducidos por el entrenador.
	4.1 Si los datos introducidos son correctos, el sistema adiciona dicho atleta a la Base de Datos y termina el CUS.
<b>Curso alterno de los eventos</b>	
<b>Acción 2.2</b>	Si los datos introducidos por el entrenador son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato erróneo e indica al usuario retornar a la acción 2.
<b>Escenario 2: Modificar atleta.</b>	
1. El entrenador selecciona la opción de Modificar atleta.	1.1 El sistema muestra un listado con los equipos existentes en la Base de Datos.
2. El entrenador selecciona el equipo al que pertenece el atleta que se desea modificar.	2.1 El sistema muestra un listado con los atletas existentes en la Base de Datos que pertenecen al equipo seleccionado.
3. El entrenador selecciona el atleta que será modificado y selecciona la opción Modificar.	3.1 El sistema localiza los datos del atleta seleccionado y los muestra, listos para ser modificados.
4. El entrenador realiza los cambios necesarios a los datos y selecciona la	4.1 El sistema verifica los datos modificados por el entrenador.

opción Actualizar	
	4.2 Si los datos están correctos el sistema actualiza los datos del atleta en la Base de Datos y termina el CUS.
<b>Curso alternativo de los eventos</b>	
<b>Acción 4.2</b>	Si los datos introducidos por el entrenador son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato erróneo e indica al usuario retornar a la acción 4.
<b>Escenario 3: Eliminar atleta.</b>	
1. El entrenador selecciona la opción de Eliminar atleta.	1.1 El sistema muestra un listado con los equipos existentes en la Base de Datos.
2. El entrenador selecciona el equipo al que pertenece el atleta que se desea eliminar.	2.1 El sistema muestra un listado con los atletas existentes en la Base de Datos que pertenecen al equipo seleccionado.
3. El entrenador selecciona el o los atletas que serán eliminados y selecciona la opción Eliminar.	3.1 El sistema muestra un mensaje de advertencia para la acción a realizar.
4. El entrenador confirma si desea o no eliminar el o los atletas seleccionados.	4.1 Si el entrenador acepta, el sistema elimina los datos correspondientes al atleta o los atletas seleccionados en la Base de Datos y termina el CUS.
<b>Curso alternativo de los eventos</b>	
<b>Acción 4.1</b>	Si el entrenador cancela la acción, se culmina el CUS sin ejecutar ninguna acción.
<b>Escenario 2: Mostrar resultado de Atleta en competencia</b>	
1. El entrenador selecciona la opción de Mostrar resultado de atleta en competencia.	1.1 El sistema muestra un listado con los equipos existentes en la Base de Datos.
2. El entrenador selecciona el equipo al que	2.1 El sistema muestra un listado con los

pertenece el atleta que se desea mostrar sus resultados en competencia.	atletas existentes en la Base de Datos que pertenecen al equipo seleccionado.
3. El entrenador selecciona el atleta que se desea mostrar sus resultados en competencia.	3.1 El sistema muestra un listado con las competencias realizadas por este el atleta, existentes en la Base de Datos.
4. El entrenador selecciona la competencia que desea ver los resultados del atleta y selecciona la opción Mostrar.	4.1 El sistema localiza los resultados del atleta seleccionado en la competencia escogida, los muestra y termina el CUS.
<b>Prioridad</b>	Crítico

Tabla A3.2 Descripción del caso de uso del sistema “Gestionar atleta”

<b>Caso de Uso</b>		<b>Gestionar lesión deportiva de un atleta.</b>
<b>Actores</b>	Entrenador (inicia).	
<b>Propósito</b>	Permite al entrenador adicionar, modificar y eliminar una lesión deportiva de una atleta.	
<b>Resumen</b>	El CUS se inicia cuando el entrenador selecciona la opción de Gestionar lesión deportiva de un atleta. El sistema muestra las opciones: Adicionar lesión deportiva de un atleta, Modificar lesión deportiva de un atleta y Eliminar una lesión deportiva de un atleta. El entrenador selecciona la opción que desea realizar e introduce al sistema los datos requeridos. El sistema muestra el resultado de la acción del entrenador. Estos datos se registran en la Base de Datos si son correctos y termina el CUS.	
<b>Referencias</b>	<b>RF 3</b>	
<b>Precondiciones</b>	-	
<b>Poscondiciones</b>	<ol style="list-style-type: none"> <li>1. Lesión adicionada en la Base de Datos.</li> <li>2. Información de la lesión, modificada en la Base de Datos.</li> <li>3. Lesión eliminada de la Base de Datos.</li> </ol>	
<b>Curso normal de los eventos</b>		

Acción del actor	Respuesta del sistema
1. El entrenador selecciona la opción de Gestionar lesión deportiva de atleta.	1.1 El sistema muestra las opciones: Adicionar lesión deportiva de atleta, Modificar lesión deportiva de atleta, Eliminar deportiva lesión de atleta.
<b>Escenario 1: Adicionar lesión deportiva de atleta.</b>	
1. El entrenador selecciona la opción de Adicionar lesión de atleta.	1.1 El sistema muestra una lista de los equipos existentes en la base de datos.
2. El entrenador selecciona el equipo al que pertenece el atleta que se lesionó.	2.1 El sistema muestra un listado con los atletas existentes en la Base de Datos que pertenecen al equipo seleccionado.
3. El entrenador selecciona el atleta que se lesionó.	3.1 El sistema muestra un listado con los tipos de lesión existentes en la Base de Datos.
4. El entrenador selecciona el tipo de la lesión ocurrida al atleta.	4.1 El sistema muestra un listado con las localizaciones de una lesión deportiva, existentes en la Base de Datos.
5. El entrenador selecciona la localización de la lesión del atleta.	5.1 El sistema muestra el formulario para la adición de una nueva lesión.
6. El entrenador introduce los datos solicitados por el sistema y selecciona la opción Adicionar.	6.1 El sistema verifica los datos introducidos por el entrenador.
	6.2 Si los datos introducidos son correctos, el sistema adiciona dicha lesión a la Base de Datos y termina el CUS.

<b>Curso alterno de los eventos</b>	
<b>Acción 6.2</b>	Si los datos introducidos por el entrenador son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato erróneo e indica al usuario retornar a la acción 6.
<b>Escenario 2: Modificar lesión de atleta.</b>	
1. El entrenador selecciona la opción de Modificar lesión de atleta.	1.1 El sistema muestra un listado con los equipos existentes en la Base de Datos.
2. El entrenador selecciona el equipo al que pertenece el atleta, que presentó la lesión a modificar.	2.1 El sistema muestra un listado con los atletas existentes en la Base de Datos que pertenecen al equipo seleccionado.
3. El entrenador selecciona el atleta que presentó la lesión que se desea modificar.	3.1 El sistema muestra un listado con las lesiones existentes en la Base de Datos que pertenecen al atleta seleccionado.
4. El entrenador selecciona la lesión a modificar y selecciona la opción Modificar.	4.1 El sistema localiza los datos de la lesión seleccionada y los muestra, listos para ser modificados.
5. El entrenador realiza los cambios necesarios a los datos y selecciona la opción Actualizar.	5.1 El sistema verifica los datos introducidos por el entrenador.
	5.2 Si los datos introducidos son correctos, el sistema actualiza los datos de dicha lesión a la Base de Datos y termina el CUS.
<b>Curso alterno de los eventos</b>	
<b>Acción 5.2</b>	Si los datos introducidos por el entrenador son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato erróneo e indica al usuario retornar a la acción 5.
<b>Escenario 3: Eliminar lesión de atleta.</b>	

1. El entrenador selecciona la opción de Eliminar lesión de atleta.	1.1 El sistema muestra un listado con los equipos existentes en la Base de Datos.
2. El entrenador selecciona el equipo al que pertenece el atleta, que presentó la lesión a eliminar.	2.1 El sistema muestra un listado con los atletas existentes en la Base de Datos que pertenecen al equipo seleccionado.
3. El entrenador selecciona el atleta que presentó la lesión que se desea eliminar.	3.1 El sistema muestra un listado con las lesiones existentes en la Base de Datos que pertenecen al atleta seleccionado.
4. El entrenador selecciona la o las lesiones que serán eliminadas y selecciona la opción Eliminar.	4.1 El sistema muestra un mensaje de advertencia para la acción a realizar.
5. El entrenador confirma si desea o no eliminar la o las lesiones seleccionadas.	5.1 Si el entrenador acepta, el sistema elimina los datos correspondientes a la lesión o las lesiones seleccionadas, en la Base de Datos y termina el CUS.
<b>Curso alternativo de los eventos</b>	
<b>Acción 5.1</b>	Si el entrenador cancela la acción, se culmina el CUS sin ejecutar ninguna acción.
<b>Prioridad</b>	Opcional

Tabla A3.3 Descripción del caso de uso del sistema “Gestionar lesión de atleta”

<b>Caso de Uso</b>	<b>Gestionar préstamo de utensilio deportivo.</b>
<b>Actores</b>	Técnico del gimnasio (inicia).
<b>Propósito</b>	Permite al técnico del gimnasio adicionar, modificar y eliminar un préstamo de utensilio deportivo.
<b>Resumen</b>	El CUS se inicia cuando el técnico del gimnasio selecciona la opción Gestionar préstamo de utensilio deportivo. El sistema muestra las opciones, Adicionar préstamo de utensilio deportivo, Modificar préstamo de utensilio

	deportivo, Terminar préstamo de utensilio deportivo y Eliminar préstamo de utensilio deportivo. El técnico del gimnasio selecciona la opción que desea realizar e introduce al sistema los datos requeridos. El sistema muestra el resultado de la acción del técnico del gimnasio. Estos datos son registrados en la Base de Datos si están correctos y termina el CUS.	
<b>Referencias</b>	<b>RF 4</b>	
<b>Precondiciones</b>	Utensilio deportivo almacenado en la Base de Datos.	
<b>Poscondiciones</b>	<ol style="list-style-type: none"> <li>1. Préstamo de utensilio deportivo adicionado a la Base de Datos.</li> <li>2. Información del préstamo de utensilio deportivo, modificada en la Base de Datos.</li> <li>3. Préstamo de utensilio deportivo eliminado de la Base de Datos.</li> </ol>	
<b>Curso normal de los eventos</b>		
<b>Acción del actor</b>	<b>Respuesta del sistema</b>	
1. El técnico del gimnasio selecciona la opción de Gestionar préstamo de utensilio deportivo.	1.2 El sistema muestra las opciones: Adicionar préstamo de utensilio deportivo, Modificar préstamo de utensilio deportivo, Eliminar préstamo de utensilio deportivo y Terminar préstamo de utensilio deportivo.	
<b>Escenario 1: Adicionar préstamo de utensilio deportivo.</b>		
3. El técnico del gimnasio selecciona la opción de Adicionar préstamo de utensilio deportivo.	1.3 El sistema muestra un formulario para insertar el número de solapín del usuario que solicita el préstamo de utensilio deportivo.	
4. El técnico del gimnasio introduce los datos solicitados por el sistema y selecciona la opción Comprobar.	2.3 El sistema verifica los datos introducidos por el técnico del gimnasio.	
	2.4 Si los datos introducidos son correctos, el sistema muestra un formulario para la	

	adición de los datos del préstamo de utensilio deportivo.
5. El técnico del gimnasio introduce los datos solicitados por el sistema y selecciona la opción Adicionar.	4.1 El sistema verifica los datos introducidos por el técnico del gimnasio.
	4.2 Si los datos introducidos son correctos, el sistema adiciona dicho préstamo utensilio deportivo a la Base de Datos y termina el CUS.
<b>Curso alterno de los eventos</b>	
<b>Acción 2.2</b>	Si los datos introducidos por el técnico del gimnasio son incorrectos, el sistema muestra un mensaje de error indicando que el usuario no existe en la Base de Datos e indica al usuario retornar a la acción 2.
<b>Acción 4.2</b>	Si los datos introducidos por el técnico del gimnasio son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato erróneo e indica al usuario retornar a la acción 4.
<b>Escenario 2: Modificar préstamo de utensilio deportivo.</b>	
1. El técnico del gimnasio selecciona la opción de Modificar préstamo de utensilio deportivo.	1.1 El sistema muestra un formulario para insertar el número de solapín del usuario que solicitó el préstamo de utensilio deportivo.
2. El técnico del gimnasio introduce los datos solicitados por el sistema y selecciona la opción Comprobar.	2.1 El sistema verifica los datos introducidos por el técnico del gimnasio.
	2.2 Si los datos introducidos son correctos, el sistema muestra los préstamos de utensilios deportivos pendientes del usuario, existentes en la Base de Datos.
3. El técnico del gimnasio selecciona el	3.1 El sistema localiza los datos del préstamo

préstamo que será modificado y selecciona la opción Modificar.	seleccionado y los muestra, listos para ser modificados.
4. El técnico del gimnasio realiza los cambios necesarios a los datos y selecciona la opción Actualizar.	4.1 El sistema verifica los datos modificados por el técnico del gimnasio.
	4.2 Si los datos están correctos el sistema actualiza los datos del préstamo de utensilio deportivo en la Base de Datos y termina el CUS.
<b>Curso alterno de los eventos</b>	
<b>Acción 2.2</b>	Si los datos introducidos por el técnico del gimnasio son incorrectos, el sistema muestra un mensaje de error indicando que el usuario no existe en la Base de Datos e indica al usuario retornar a la acción 2.
<b>Acción 4.2</b>	Si los datos introducidos por el técnico del gimnasio son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato erróneo e indica al usuario retornar a la acción 4.
<b>Escenario 3: Eliminar préstamo de utensilio deportivo.</b>	
1. El técnico del gimnasio selecciona la opción de Eliminar préstamo de utensilio deportivo.	2.1 El sistema muestra un formulario para insertar el número de solapín del usuario que solicitó el préstamo de utensilio deportivo.
2. El técnico del gimnasio introduce los datos solicitados por el sistema y selecciona la opción Comprobar.	2.1 El sistema verifica los datos introducidos por el técnico del gimnasio.
	2.2 Si los datos introducidos son correctos, el sistema muestra los préstamos de utensilios deportivos terminados del usuario, existentes en la Base de Datos.
3. El técnico del gimnasio	3.1 El sistema muestra un mensaje de

selecciona el préstamo de utensilio deportivo que será eliminado y selecciona la opción Eliminar.	advertencia para la acción a realizar.
4. El técnico del gimnasio confirma si desea o no eliminar el o los préstamos de utensilios deportivos seleccionados.	4.1 Si el técnico del gimnasio acepta, el sistema elimina los datos correspondientes al préstamo o los préstamos de utensilio seleccionados en la Base de Datos y termina el CUS.
<b>Curso alterno de los eventos</b>	
<b>Acción 2.2</b>	Si los datos introducidos por el técnico del gimnasio son incorrectos, el sistema muestra un mensaje de error indicando que el usuario no existe en la Base de Datos e indica al usuario retornar a la acción 2.
<b>Acción 4.1</b>	Si el técnico del gimnasio cancela la acción, se culmina el CUS sin ejecutar ninguna acción.
<b>Escenario 4: Terminar préstamo de utensilio.</b>	
1. El técnico del gimnasio selecciona la opción de Terminar préstamo de utensilio deportivo.	1.1 El sistema muestra un formulario para insertar el número de solapín del usuario que solicitó el préstamo de utensilio deportivo.
2. El técnico del gimnasio introduce los datos solicitados por el sistema y selecciona la opción Comprobar.	2.1 El sistema verifica los datos introducidos por el técnico del gimnasio.
	2.2 Si los datos introducidos son correctos, el sistema muestra los préstamos de utensilio deportivo pendientes del usuario, existentes en la Base de Datos.
3. El técnico del gimnasio selecciona el préstamo que será terminado y selecciona la opción Terminar.	3.1 El sistema muestra un formulario para insertar los datos de terminación del préstamo de utensilio deportivo.

4.	El técnico del gimnasio introduce los datos solicitados por el sistema y selecciona la opción Terminar.	4.1 El sistema verifica los datos introducidos por el técnico del gimnasio.
		4.2 Si los datos introducidos son correctos, el sistema termina dicho préstamo de utensilio a la Base de Datos y termina el CUS.
<b>Curso alternativo de los eventos</b>		
<b>Acción 2.2</b>		Si los datos introducidos por el técnico del gimnasio son incorrectos, el sistema muestra un mensaje de error indicando que el usuario no existe en la Base de Datos e indica al usuario retornar a la acción 2.
<b>Acción 4.2</b>		Si los datos introducidos por el técnico del gimnasio son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato erróneo e indica al usuario retornar a la acción 4.
<b>Prioridad</b>	Crítico.	

Tabla A3.4 Descripción del caso de uso del sistema “Gestionar préstamo de utensilio deportivo”

<b>Caso de Uso</b>	<b>Gestionar visita al gimnasio.</b>
<b>Actores</b>	Técnico del Gimnasio (inicia).
<b>Propósito</b>	Le permite al técnico del gimnasio adicionar, modificar, eliminar y terminar una visita al gimnasio.
<b>Resumen</b>	El CUS se inicia cuando el técnico del gimnasio selecciona la opción Gestionar visita al gimnasio. El sistema muestra cuatro opciones Adicionar una visita, Modificar una visita, Eliminar una visita y Terminar una visita. El técnico del gimnasio introduce los datos necesarios para la realización de estas acciones. Si son correctos el sistema los introduce en la Base de Datos y termina el CUS.

<b>Referencias</b>	<b>RF 5</b>	
<b>Precondiciones</b>	1. Usuarios almacenados en la Base de Datos.	
<b>Poscondiciones</b>	4. Adicionada visita a la Base de Datos. 5. Información de la visita, modificada en la Base de Datos. 6. Visita eliminada de la Base de Datos.	
<b>Curso normal de los eventos</b>		
<b>Acción del actor</b>	<b>Respuesta del sistema</b>	
1. El técnico del gimnasio selecciona la opción de Gestionar visita al gimnasio.	1.1 El sistema muestra las opciones: Adicionar visita al gimnasio, Modificar visita al gimnasio, Eliminar visita al gimnasio y Terminar visita al gimnasio.	
<b>Escenario 1: Adicionar visita al gimnasio.</b>		
1. El técnico del gimnasio selecciona la opción Adicionar visita al gimnasio.	1.1 El sistema muestra un formulario para ingresar el número de solapín del usuario.	
2. El técnico del gimnasio introduce los datos solicitados por el sistema.	2.1 El sistema verifica los datos introducidos por el técnico del gimnasio.	
	2.2 Si los datos introducidos son correctos, el sistema muestra un formulario para ingresar los datos de la visita.	
3. El técnico del gimnasio introduce los datos solicitados por el sistema y selecciona la opción Adicionar.	4.1 El sistema verifica los datos introducidos por el técnico del gimnasio.	
	4.2 Si los datos introducidos son correctos, el sistema adiciona la visita a la Base de Datos y termina el CUS.	
<b>Curso alterno de los eventos</b>		

<b>Acción 2.2</b>	Si los datos introducidos por el técnico del gimnasio son incorrectos, el sistema muestra un mensaje de error indicando que el usuario no existe en la Base de Datos e indica al usuario retornar a la acción 2.
<b>Acción 4.2</b>	Si los datos introducidos por el técnico del gimnasio son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato erróneo e indica al usuario retornar a la acción 3.
<b>Sección 2: Modificar visita al gimnasio.</b>	
1. El técnico del gimnasio selecciona la opción Modificar visita al gimnasio.	1.1 El sistema muestra un formulario para ingresar el número de solapín del usuario que visitó al gimnasio.
2. El técnico del gimnasio introduce los datos solicitados por el sistema.	2.1 El sistema verifica los datos introducidos por el técnico del gimnasio.
	2.2 Si los datos introducidos son correctos, el sistema muestra un listado de las visitas pendientes de este usuario, existentes en la Base de Datos.
3. El técnico del gimnasio selecciona la visita a modificar y selecciona la opción Modificar.	3.1 El sistema localiza los datos de la visita seleccionado y los muestra, listos para modificarlos.
4. El técnico del gimnasio realiza los cambios necesarios a los datos y selecciona la opción Actualizar.	4.1 El sistema verifica los datos introducidos por el técnico del gimnasio.
	4.2 Si los datos introducidos son correctos, el sistema actualiza la visita en la Base de Datos y termina el CUS.
<b>Curso alternativo de los eventos.</b>	
<b>Acción 2.2</b>	Si los datos introducidos por el técnico del gimnasio

	son incorrectos, el sistema muestra un mensaje de error indicando que el usuario no existe en la Base de Datos e indica al usuario retornar a la acción 2.
<b>Acción 4.2</b>	Si los datos introducidos por el técnico del gimnasio son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato erróneo e indica al usuario retornar a la acción 4.
<b>Sección 3: Eliminar visita al gimnasio.</b>	
1. El técnico del gimnasio selecciona la opción Eliminar visita al gimnasio.	1.1 El sistema muestra un formulario para ingresar el número de solapín del usuario que visitó al gimnasio.
2. El técnico del gimnasio introduce los datos solicitados por el sistema.	2.1 El sistema verifica los datos introducidos por el técnico del gimnasio.
	2.2 Si los datos introducidos son correctos, el sistema muestra un listado de las visitas terminadas de este usuario, existentes en la Base de Datos.
3. El técnico del gimnasio selecciona la visita a eliminar y selecciona la opción Eliminar.	3.1 El sistema muestra un mensaje de advertencia para la acción a realizar.
4. El técnico del gimnasio confirma si desea o no eliminar la o las visitas seleccionadas.	4.1 Si el técnico del gimnasio acepta, el sistema elimina los datos de la o las visitas seleccionadas en la Base de Datos y termina el CUS.
<b>Curso alterno de los eventos.</b>	
<b>Acción 2.2</b>	Si los datos introducidos por el técnico del gimnasio son incorrectos, el sistema muestra un mensaje de error indicando que el usuario no existe en la Base de Datos e indica al usuario retornar a la acción 2.
<b>Acción 4.1</b>	Si el técnico del gimnasio cancela la acción, se culmina el CUS sin ejecutar ninguna acción.

<b>Escenario 4: Terminar visita al gimnasio.</b>	
3. El técnico del gimnasio selecciona la opción Terminar visita al gimnasio.	1.2 El sistema muestra un formulario para ingresar el número de solapín del usuario que visitó al gimnasio.
3. El técnico del gimnasio introduce los datos solicitados por el sistema.	2.3 El sistema verifica los datos introducidos por el técnico del gimnasio.
	2.4 Si los datos introducidos son correctos, el sistema muestra un listado de las visitas pendientes de este usuario, existentes en la Base de Datos.
5. El técnico del gimnasio selecciona la visita que será terminada y selecciona la opción Terminar.	3.1 El sistema muestra un formulario para insertar los datos de terminación de la visita.
6. El técnico del gimnasio introduce los datos solicitados por el sistema y selecciona la opción Terminar.	4.2 El sistema verifica los datos introducidos por el técnico del gimnasio.
	4.3 Si los datos introducidos son correctos, el sistema termina la visita en la Base de Datos y termina el CUS.
<b>Prioridad</b>	Crítico.

Tabla A3.5 Descripción del caso de uso del sistema “Gestionar visita al gimnasio”

<b>Caso de Uso</b>	<b>Gestionar competencia deportiva.</b>
<b>Actores</b>	Organizador de competencia (inicia).
<b>Propósito</b>	Le permite al organizador de competencia crear, modificar y eliminar una competencia deportiva.
<b>Resumen</b>	El CUS se inicia cuando el organizador de competencia selecciona la opción de Gestionar competencia deportiva. El sistema muestra las opciones Crear competencia deportiva, Modificar competencia deportiva,

	Eliminar competencia deportiva. El organizador de competencia sigue las instrucciones del sistema e introduce los datos necesarios para la realización de estas acciones. Estos datos se registran en la Base de Datos si son correctos y termina el CUS.	
<b>Referencias</b>	<b>RF 6</b>	
<b>Precondiciones</b>	2. La categoría de las competencias deportivas están almacenadas en la Base de Datos. 3. Los deportes están almacenados en la Base de Datos.	
<b>Poscondiciones</b>	7. Adicionada competencia deportiva a la Base de Datos. 8. Información de la competencia deportiva, modificada en la Base de Datos. 9. Competencia deportiva eliminada de la Base de Datos.	
<b>Curso normal de los eventos</b>		
<b>Acción del actor</b>	<b>Respuesta del sistema</b>	
1. El organizador de competencia selecciona la opción de Gestionar competencia deportiva.	1.1 El sistema muestra las opciones: Crear competencia deportiva, Modificar competencia deportiva, Eliminar competencia deportiva.	
<b>Escenario 1: Crear competencia deportiva.</b>		
1. El organizador de competencia selecciona la opción de Crear competencia deportiva.	1.1 El sistema muestra las opciones: Adicionar datos de la competencia, Adicionar comisión organizadora de la competencia, Adicionar jefes de comisiones técnicas de la competencia.	
<b>Sección 1: Adicionar datos de competencia deportiva.</b>		
1. El organizador de competencia selecciona la opción Adicionar datos de competencia deportiva.	2.1 El sistema muestra un formulario para adicionar los datos de una competencia deportiva.	
2. El organizador de competencia introduce los datos solicitados por el sistema y selecciona la opción	5.1 El sistema verifica los datos introducidos por el organizador de competencia.	

Adicionar.	
	5.1 Si los datos introducidos son correctos, el sistema adiciona los datos de la competencia a la Base de Datos.
<b>Curso alterno de los eventos</b>	
<b>Acción 2.2</b>	Si los datos introducidos por el organizador de competencia son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato erróneo e indica al usuario retornar a la acción 2.
<b>Sección 2: Adicionar comisión organizadora de la competencia.</b>	
1. El organizador de competencia selecciona la opción Adicionar comisión organizadora de la competencia.	3.1 El sistema muestra un formulario para adicionar los datos de la comisión organizadora de la competencia deportiva.
2. El organizador de competencia introduce los datos solicitados por el sistema y selecciona la opción Adicionar.	5.1 El sistema verifica los datos introducidos por el organizador de competencia.
	5.1 Si los datos introducidos son correctos, el sistema adiciona los datos de la comisión organizadora de la competencia a la Base de Datos.
<b>Curso alterno de los eventos</b>	
<b>Acción 2.2</b>	Si los datos introducidos por el organizador de competencia son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato erróneo e indica al usuario retornar a la acción 2.
<b>Sección 1: Adicionar jefes de comisiones técnicas de la competencia.</b>	
1. El organizador de competencia	1.1 El sistema muestra un formulario para adicionar

selecciona la opción Adicionar jefes de comisiones técnicas de la competencia.	los jefes de comisiones técnicas de la competencia.
2. El organizador de competencia introduce los datos solicitados por el sistema y selecciona la opción Adicionar.	2.1 El sistema verifica los datos introducidos por el organizador de competencia.
	2.2 Si los datos introducidos son correctos, el sistema adiciona los jefes de comisiones técnicas de la competencia a la Base de Datos y termina el CUS.
<b>Curso alternativo de los eventos</b>	
<b>Acción 2.2</b>	Si los datos introducidos por el organizador de competencia son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato erróneo e indica al usuario retornar a la acción 2.
<b>Escenario 2: Modificar competencia deportiva.</b>	
1. El organizador de competencia selecciona la opción de Modificar competencia deportiva.	1.1 El sistema muestra las opciones: Modificar datos de la competencia, Modificar comisión organizadora de la competencia, Modificar jefes de comisiones técnicas de la competencia.
<b>Sección 1: Modificar datos de competencia.</b>	
1. El organizador de competencia selecciona la opción Modificar datos de una competencia deportiva.	1.1 El sistema muestra un listado de las categorías de competencias existentes en la Base de Datos.
2. El organizador de competencia selecciona la categoría de la competencia a la que desea modificar los datos.	2.1 El sistema muestra un listado de las competencias deportivas que pertenecen a esa categoría existente en la Base de Datos.
3. El organizador de competencia	3.1 El sistema localiza los datos de la competencia

selecciona la competencia deportiva que desea modificarle los datos y selecciona la opción Modificar.	deportiva seleccionada y los muestra, listos para ser modificados.
4. El organizador de competencia realiza los cambios necesarios a los datos y selecciona la opción Actualizar.	4.1 El sistema verifica los datos modificados por el organizador de competencia.
	4.2 Si los datos están correctos el sistema actualiza los datos de la competencia deportiva en la Base de Datos y termina el CUS.
<b>Curso alterno de los eventos.</b>	
<b>Acción 4.2</b>	Si los datos introducidos por el organizador de competencia son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato erróneo e indica al usuario retornar a la acción 4.
<b>Sección 2: Modificar comisión organizadora de la competencia.</b>	
1. El organizador de competencia selecciona la opción Modificar comisión organizadora de la competencia.	1.1 El sistema muestra un listado de las categorías de competencias existentes en la Base de Datos.
2. El organizador de competencia selecciona la categoría de la competencia a la que desea modificar la comisión organizadora.	2.1 El sistema muestra un listado de las competencias deportivas que pertenecen a esa categoría existente en la Base de Datos.
3. El organizador de competencia selecciona la competencia deportiva que desea modificarle la comisión organizadora y selecciona la opción Modificar.	3.1 El sistema localiza los datos de la competencia deportiva seleccionada y los muestra, listos para ser modificados.
4. El organizador de competencia realiza los cambios necesarios a los datos y	4.1 El sistema verifica los datos modificados por el organizador

selecciona la opción Actualizar.	de competencia.
	4.2 Si los datos están correctos el sistema actualiza los datos de la comisión organizadora de la competencia deportiva en la Base de Datos y termina el CUS.
<b>Curso alterno de los eventos.</b>	
<b>Acción 4.2</b>	Si el organizador de competencia cancela la acción, se culmina el CUS sin ejecutar ninguna acción
<b>Sección 3: Modificar jefes de comisiones técnicas de una competencia.</b>	
1. El organizador de competencia selecciona la opción Modificar jefes de comisiones técnicas de una competencia.	1.1 El sistema muestra un listado de las categorías de competencias existentes en la Base de Datos.
2. El organizador de competencia selecciona la categoría de la competencia a la que desea modificar los jefes de las comisiones técnicas.	2.1 El sistema muestra un listado de las competencias deportivas que pertenecen a esa categoría existente en la Base de Datos.
3. El organizador de competencia selecciona la competencia deportiva que desea modificarle los jefes de las comisiones técnicas y selecciona la opción Modificar.	3.1 El sistema localiza los datos de la competencia deportiva seleccionada y los muestra, listos para ser modificados.
4. El organizador de competencia realiza los cambios necesarios a los datos y selecciona la opción Actualizar.	4.1 El sistema verifica los datos modificados por el organizador de competencia.
	4.2 Si los datos están correctos el sistema actualiza los datos de los jefes de las comisiones técnicas de la competencia en la Base de Datos y

	termina el CUS.
<b>Curso alternativo de los eventos.</b>	
<b>Acción 4.2</b>	Si el organizador de competencia cancela la acción, se culmina el CUS sin ejecutar ninguna acción
<b>Prioridad</b>	Crítico

Tabla A3.6 Descripción del caso de uso del sistema “Gestionar competencia deportiva”.

<b>Caso de Uso</b>	<b>Gestionar participantes en competencias deportivas.</b>
<b>Actores</b>	Organizador de competencia (inicia).
<b>Propósito</b>	Le permite al organizador de competencia adicionar, modificar y eliminar los participantes de una competencia deportiva.
<b>Resumen</b>	El CUS se inicia cuando el organizador de competencia selecciona la opción de Gestionar participantes en competencias deportivas. El sistema muestra las opciones Adicionar participantes en competencias deportivas, Modificar participantes en competencias deportivas, Eliminar participantes en competencias deportivas. El organizador de competencia sigue las instrucciones del sistema e introduce los datos necesarios para la realización de estas acciones. Si los datos introducidos por el organizador son correctos se registran en la Base de Datos y termina el CUS.
<b>Referencias</b>	<b>RF 7, RF 8</b>
<b>Precondiciones</b>	<ol style="list-style-type: none"> <li>1. La categoría de las competencias deportivas están almacenadas en la Base de Datos.</li> <li>2. Las competencias están almacenadas en la Base de Datos.</li> <li>3. Los atletas están almacenados en la Base de Datos.</li> </ol>
<b>Poscondiciones</b>	<ol style="list-style-type: none"> <li>1. Adicionada representación a la Base de Datos.</li> <li>2. Información de la representación, modificada en la Base de Datos.</li> <li>3. Representación eliminada de la Base de Datos.</li> </ol>

	<p>4. Adicionado equipo a la Base de Datos.</p> <p>5. Información del equipo, modificada en la Base de Datos.</p> <p>6. Equipo eliminado de la Base de Datos.</p>
<b>Curso normal de los eventos</b>	
<b>Acción del actor</b>	<b>Respuesta del sistema</b>
2. El organizador de competencia selecciona la opción de Gestionar participantes en competencias deportivas.	1.1 El sistema muestra las opciones: Adicionar participantes en una competencia deportiva, Modificar participantes en una competencia deportiva, Eliminar participantes en una competencia deportiva.
<b>Escenario 1: Adicionar participantes en una competencia deportiva.</b>	
1. El organizador de competencia selecciona la opción de Adicionar participantes de una competencia deportiva.	1.1 El sistema muestra las opciones: Adicionar representación de una competencia deportiva, Adicionar equipo de una competencia deportiva.
<b>Sección 1: Adicionar representación de una competencia deportiva.</b>	
1. El organizador de competencia selecciona la opción Adicionar representación de una competencia deportiva.	1.1 El sistema muestra un listado con las categorías de competencia existentes en la Base de Datos.
2. El organizador de competencia selecciona la categoría de la competencia en que participa la representación que se desea adicionar.	2.1 El sistema muestra un listado con las competencias pertenecientes a esa categoría existentes en la Base de Datos.
3. El organizador de competencia selecciona la competencia en que participa la representación que se desea adicionar.	3.1 El sistema muestra un formulario para adicionar los datos de la representación.
4. El organizador de competencia introduce los datos solicitados por el sistema y selecciona la opción Adicionar.	4.1 El sistema verifica los datos introducidos por el organizador de competencia.

	4.2 Si los datos introducidos son correctos, el sistema adiciona la representación de la competencia a la Base de Datos.
<b>Curso alternativo de los eventos</b>	
<b>Acción 4.2</b>	Si los datos introducidos por el organizador de competencia son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato erróneo e indica al usuario retornar a la acción 4.
<b>Sección 2: Adicionar equipo de una competencia deportiva.</b>	
1. El organizador de competencia selecciona la opción Adicionar equipo de una competencia deportiva.	1.1 El sistema muestra un listado con las categorías de competencia existentes en la Base de Datos.
2. El organizador de competencia selecciona la categoría de la competencia en que participa el equipo que se desea adicionar.	2.1 El sistema muestra un listado con las competencias pertenecientes a esa categoría existentes en la Base de Datos.
3. El organizador de competencia selecciona la competencia en que participa el equipo que se desea adicionar.	3.1 El sistema muestra un listado con todas las representaciones que compiten en la competencia seleccionada.
4. El organizador de competencia selecciona la representación a la que pertenece el equipo que se desea adicionar.	4.1 El sistema muestra un formulario para adicionar los datos del equipo.
5. El organizador de competencia introduce los datos solicitados por el sistema y selecciona la opción Adicionar.	5.1 El sistema verifica los datos introducidos por el organizador de competencia.
	5.2 Si los datos introducidos son correctos, el sistema adiciona los datos del equipo a la Base de Datos.

<b>Curso alternativo de los eventos</b>	
<b>Acción 5.2</b>	Si los datos introducidos por el organizador de competencia son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato erróneo e indica al usuario retornar a la acción 5.
<b>Escenario 1: Modificar participantes en una competencia deportiva.</b>	
1. El organizador de competencia selecciona la opción de Modificar participantes de una competencia deportiva.	1.1 El sistema muestra las opciones: Modificar representación de una competencia deportiva, Modificar equipo de una competencia deportiva.
<b>Sección 1: Modificar representación de una competencia deportiva.</b>	
1. El organizador de competencia selecciona la opción Modificar representación de una competencia deportiva.	1.1 El sistema muestra un listado con las categorías de competencia existentes en la Base de Datos.
2. El organizador de competencia selecciona la categoría de la competencia en que participa la representación que se desea modificar.	2.1 El sistema muestra un listado con las competencias pertenecientes a esa categoría existentes en la Base de Datos.
3. El organizador de competencia selecciona la competencia en que participa la representación que se desea modificar.	3.1 El sistema muestra un listado con todas las representaciones que participaron en la competencia seleccionada.
4. El organizador de competencia selecciona la representación que desea modificar y selecciona la opción Modificar.	4.1 El sistema localiza los datos de la representación seleccionada y los muestra, listos para ser modificados.
5. El organizador de competencia realiza los cambios necesarios a los datos y selecciona la opción Actualizar.	5.1 El sistema verifica los datos introducidos por el organizador de competencia.

	5.2 Si los datos introducidos son correctos, el sistema actualiza la representación en la Base de Datos.
<b>Curso alterno de los eventos</b>	
<b>Acción 5.2</b>	Si los datos introducidos por el organizador de competencia son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato erróneo e indica al usuario retornar a la acción 4.
<b>Sección 2: Modificar equipo de una competencia deportiva.</b>	
1. El organizador de competencia selecciona la opción Modificar equipo de una competencia deportiva.	1.1 El sistema muestra un listado con las categorías de competencia existentes en la Base de Datos.
2. El organizador de competencia selecciona la categoría de la competencia en que participa el equipo que se desea modificar.	2.1 El sistema muestra un listado con las competencias pertenecientes a esa categoría existentes en la Base de Datos.
3. El organizador de competencia selecciona la competencia en que participa el equipo que se desea modificar.	3.1 El sistema muestra un listado con todas las representaciones que participaron en la competencia seleccionada, existentes en la Base de Datos.
4. El organizador de competencia selecciona la representación a la que pertenece el equipo que se desea modificar.	4.1 El sistema muestra un listado con todos los equipos pertenecientes a la representación seleccionada existentes en la Base de Datos.

<p>5. El organizador de competencia selecciona el grupo que desea modificar y selecciona la opción Modificar.</p>	<p>5.1 El sistema localiza los datos pertenecientes al equipo seleccionado y los muestra listos para ser modificados.</p>
<p>6. El organizador de competencia realiza los cambios necesarios a los datos y selecciona la opción Actualizar.</p>	<p>6.1 El sistema verifica los datos modificados por el organizador de competencia.</p>
	<p>6.2 Si los datos están correctos el sistema actualiza los datos del equipo en la Base de Datos y termina el CUS.</p>
<p><b>Curso alternativo de los eventos.</b></p>	
<p><b>Acción 6.2</b></p>	<p>Si los datos introducidos por el organizador de competencia son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato erróneo e indica al usuario retornar a la acción 6.</p>
<p><b>Escenario 3: Eliminar participantes en una competencia deportiva.</b></p>	
<p>1. El organizador de competencia selecciona la opción Eliminar participantes en una competencia deportiva.</p>	<p>1.1 El sistema muestra las opciones: Eliminar representación de una competencia deportiva y Eliminar un equipo de una competencia deportiva.</p>
<p><b>Sección 1: Eliminar representación de una competencia deportiva.</b></p>	
<p>1. El organizador de competencia selecciona la opción Eliminar representación de una competencia deportiva.</p>	<p>1.1 El sistema muestra un listado con las categorías de competencia existentes en la Base de Datos.</p>

<p>2. El organizador de competencia selecciona la categoría de la competencia en que participa la representación que se desea eliminar.</p>	<p>2.1 El sistema muestra un listado con las competencias pertenecientes a esa categoría existentes en la Base de Datos.</p>
<p>3. El organizador de competencia selecciona la competencia en que participa la representación que se desea eliminar.</p>	<p>3.1 El sistema muestra un listado con todas las representaciones que participaron en la competencia seleccionada.</p>
<p>4. El organizador de competencia selecciona la o las representaciones que desea eliminar y selecciona la opción Eliminar.</p>	<p>4.1 El sistema muestra un mensaje de advertencia para la acción a realizar.</p>
<p>5. El organizador de competencia confirma si desea o no eliminar la o las representaciones seleccionadas.</p>	<p>5.1 Si el organizador de competencia acepta, el sistema elimina los datos correspondientes a la o las representaciones seleccionada en la Base de Datos y termina el CUS.</p>
<p><b>Curso alterno de los eventos</b></p>	
<p><b>Acción 5.1</b></p>	<p>Si los datos introducidos por el organizador de competencia son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato erróneo e indica al usuario retornar a la acción 5.</p>
<p><b>Escenario 2: Eliminar equipo de una competencia deportiva.</b></p>	
<p>1. El organizador de competencia selecciona la opción Eliminar equipo de una competencia deportiva.</p>	<p>1.1 El sistema muestra un listado con las categorías de competencia existentes en la Base de Datos.</p>
<p>2. El organizador de competencia selecciona la categoría de la competencia en que participa el equipo que se desea eliminar.</p>	<p>2.1 El sistema muestra un listado con las competencias pertenecientes a esa categoría existentes en la Base de Datos.</p>

3. El organizador de competencia selecciona la competencia en que participa el equipo que se desea eliminar.	3.1 El sistema muestra un listado con todas las representaciones que participaron en la competencia seleccionada, existentes en la Base de Datos.
4. El organizador de competencia selecciona la representación a la que pertenece el equipo que se desea eliminar.	4.1 El sistema muestra un listado con todos los equipos pertenecientes a la representación seleccionada existentes en la Base de Datos.
5. El organizador de competencia selecciona el o los equipos que desea eliminar y selecciona la opción Eliminar.	5.1 El sistema muestra un mensaje de advertencia para la acción a realizar.
6. El organizador de competencia confirma si desea o no eliminar el o los equipos seleccionados.	6.1 Si el organizador de competencia acepta, el sistema elimina los datos correspondientes al equipo o los equipos seleccionados en la Base de Datos y termina el CUS.
<b>Curso alternativo de los eventos</b>	
<b>Acción 6.1</b>	Si los datos introducidos por el organizador de competencia son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato erróneo e indica al usuario retornar a la acción 2.
<b>Prioridad</b>	Crítico

Tabla A3.7 Descripción del caso de uso del sistema “Gestionar participantes en competencias deportivas”

<b>Caso de Uso</b>	<b>Gestionar enfrentamiento de una competencia deportiva.</b>
<b>Actores</b>	Organizador de competencia (inicia).
<b>Propósito</b>	Permite al organizador de competencia adicionar, modificar y eliminar un enfrentamiento de una competencia deportiva.
<b>Resumen</b>	El CUS se inicia cuando el organizador de competencia selecciona la

	<p>opción Gestionar enfrentamiento de una competencia deportiva. El sistema muestra las opciones, adicionar enfrentamiento de una competencia deportiva, modificar enfrentamiento de una competencia deportiva y eliminar enfrentamiento de una competencia deportiva. El organizador de competencia sigue las instrucciones del sistema e introduce los datos necesarios para la realización de estas acciones. Estos datos se registran en la Base de Datos si son correctos y termina el CUS.</p>	
<b>Referencias</b>	<b>RF 9</b>	
<b>Precondiciones</b>	<ol style="list-style-type: none"> <li>1. Los deportes están almacenados en la Base de Datos.</li> <li>2. Los eventos por deporte están almacenados en la Base de Datos.</li> <li>3. Las competencias están almacenadas en la Base de Datos.</li> <li>4. Las representaciones están almacenadas en la Base de Datos.</li> <li>5. Los equipos están almacenados en la Base de Datos.</li> <li>6. Los atletas están almacenados en la Base de Datos.</li> </ol>	
<b>Poscondiciones</b>	<ol style="list-style-type: none"> <li>1. Adicionado enfrentamiento a la Base de Datos.</li> <li>2. Información del enfrentamiento, modificada en la Base de Datos.</li> <li>3. Eliminado enfrentamiento de la Base de Datos.</li> </ol>	
<b>Curso normal de los eventos</b>		
<b>Acción del actor</b>	<b>Respuesta del sistema</b>	
<ol style="list-style-type: none"> <li>1. El organizador de competencia selecciona la opción de Gestionar enfrentamiento de una competencia deportiva.</li> </ol>	<ol style="list-style-type: none"> <li>1.1 El sistema muestra las opciones: Adicionar enfrentamiento de una competencia deportiva, Modificar enfrentamiento de una competencia deportiva y Eliminar enfrentamiento de una competencia deportiva</li> </ol>	
<b>Escenario 1: Adicionar enfrentamiento de una competencia deportiva.</b>		

<p>1. El organizador de competencia selecciona la opción Adicionar enfrentamiento de una competencia.</p>	<p>1.1 El sistema muestra una lista con los deportes existentes en la Base de Datos.</p>
<p>2. El organizador de competencia selecciona el deporte en que se compitió en el enfrentamiento que se desea adicionar.</p>	<p>2.1 El sistema muestra un listado con los eventos correspondientes al deporte seleccionado existentes en la Base de Datos.</p>
<p>3. El organizador de competencia selecciona el evento correspondiente al enfrentamiento que se desea adicionar.</p>	<p>3.1 El sistema muestra las categorías de una competencia existentes en la Base de Datos.</p>
<p>4. El organizador de competencia selecciona la categoría de la competencia en que se desarrolló el enfrentamiento que se desea adicionar.</p>	<p>4.1 El sistema muestra un listado con las competencias correspondiente a la categoría seleccionada existentes en la Base de Datos.</p>
<p>5. El organizador de competencia selecciona la competencia en que se desarrolló el enfrentamiento que se desea adicionar.</p>	<p>5.1 El sistema muestra un listado con los tipos de enfrentamientos existentes en la Base de Datos.</p>
<p>6. El organizador de competencia selecciona el tipo de enfrentamiento del enfrentamiento que se desea adicionar.</p>	<p>6.1 En correspondencia con el tipo de enfrentamiento el sistema muestra un formulario para adicionar los datos del enfrentamiento.</p>
<p>7. El organizador de competencia introduce los datos solicitados por el sistema y selecciona la opción Adicionar.</p>	<p>7.1 El sistema verifica los datos introducidos por el organizador de competencia.</p>
	<p>7.2 Si los datos introducidos son correctos, el sistema adiciona el enfrentamiento a la Base de Datos y termina el CUS.</p>
<p><b>Curso alterno de los eventos</b></p>	

<p><b>Acción 7.2</b></p>	<p>Si los datos introducidos por el organizador de competencia son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato erróneo e indica al usuario retornar a la acción 7.</p>
<p><b>Escenario 2: Modificar enfrentamiento de una competencia deportiva.</b></p>	
<p>1. El organizador de competencia selecciona la opción Modificar enfrentamiento de una competencia.</p>	<p>1.1 El sistema muestra una lista con los deportes existentes en la Base de Datos.</p>
<p>2. El organizador de competencia selecciona el deporte en que se compitió en el enfrentamiento que se desea modificar.</p>	<p>2.1 El sistema muestra un listado con los eventos correspondientes al deporte seleccionado existentes en la Base de Datos.</p>
<p>3. El organizador de competencia selecciona el evento correspondiente al enfrentamiento que se desea modificar.</p>	<p>3.1 El sistema muestra las categorías de una competencia existentes en la Base de Datos.</p>
<p>4. El organizador de competencia selecciona la categoría de la competencia en que se desarrolló el enfrentamiento que se desea modificar.</p>	<p>4.1 El sistema muestra un listado con las competencias correspondiente a la categoría seleccionada existentes en la Base de Datos.</p>
<p>5. El organizador de competencia selecciona la competencia en que se desarrolló el enfrentamiento que se desea modificar.</p>	<p>5.1 El sistema muestra un listado con los tipos de enfrentamientos existentes en la Base de Datos.</p>
<p>6. El organizador de competencia selecciona el tipo de enfrentamiento del enfrentamiento que se desea modificar.</p>	<p>6.1 El sistema muestra un listado con los enfrentamientos existentes en la Base de Datos del tipo seleccionado.</p>
<p>7. El organizador de competencia selecciona el enfrentamiento que desea modificar y selecciona la opción Modificar.</p>	<p>7.1 El sistema localiza los datos del enfrentamiento seleccionado y los muestra listos para ser modificados.</p>

8. El organizador de competencia realiza los cambios necesarios a los datos y selecciona la opción Actualizar.	8.1 El sistema verifica los datos modificados por el organizador de competencia.
	8.2 Si los datos están correctos el sistema actualiza los datos del enfrentamiento en la Base de Datos y termina el CUS.
<b>Curso alterno de los eventos.</b>	
<b>Acción 8.2</b>	Si los datos introducidos por el organizador de competencia son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato erróneo e indica al usuario retornar a la acción 8.
<b>Escenario 3: Eliminar enfrentamiento de una competencia.</b>	
1. El organizador de competencia selecciona la opción Eliminar enfrentamiento de una competencia.	1.1 El sistema muestra una lista con los deportes existentes en la Base de Datos.
2. El organizador de competencia selecciona el deporte en que se compitió en el enfrentamiento que se desea eliminar.	2.1 El sistema muestra un listado con los eventos correspondientes al deporte seleccionado existentes en la Base de Datos.
3. El organizador de competencia selecciona el evento correspondiente al enfrentamiento que se desea eliminar.	3.1 El sistema muestra las categorías de una competencia existentes en la Base de Datos.
4. El organizador de competencia selecciona la categoría de la competencia en que se desarrolló el enfrentamiento que se desea eliminar.	4.1 El sistema muestra un listado con las competencias correspondiente a la categoría seleccionada existentes en la Base de Datos.
5. El organizador de competencia selecciona la competencia en que se desarrolló el enfrentamiento que se desea eliminar.	5.1 El sistema muestra un listado con los tipos de enfrentamientos existentes en la Base de Datos.

6. El organizador de competencia selecciona el tipo de enfrentamiento del enfrentamiento que se desea eliminar.	6.1 El sistema muestra un listado con los enfrentamientos existentes en la Base de Datos del tipo seleccionado.
7. El organizador de competencia selecciona el o los enfrentamientos enfrentamiento que desea eliminar y selecciona la opción Eliminar.	7.1 El sistema muestra un mensaje de advertencia para la acción a realizar.
8. El organizador de competencia confirma si desea o no eliminar el o los enfrentamientos seleccionados.	8.1 Si el organizador de competencia acepta, el sistema elimina el o los enfrentamientos seleccionado en la Base de Datos y termina el CUS.
<b>Curso alternativo de los eventos.</b>	
<b>Acción 8.1</b>	Si el organizador de competencia cancela la acción, se culmina el CUS sin ejecutar ninguna acción
<b>Prioridad</b>	Crítico

Tabla A3.8 Descripción del caso de uso del sistema “Gestionar enfrentamiento de competencia deportiva”

<b>Caso de Uso</b>	<b>Gestionar resultado de enfrentamiento de una competencia deportiva.</b>
<b>Actores</b>	Organizador de competencia (inicia).
<b>Propósito</b>	Permite al organizador de competencia adicionar, modificar y eliminar un enfrentamiento de una competencia deportiva.
<b>Resumen</b>	El CUS se inicia cuando el organizador de competencia selecciona la opción Gestionar resultado de enfrentamiento de una competencia deportiva. El sistema muestra las opciones, adicionar resultado de enfrentamiento de una competencia deportiva, modificar resultado de enfrentamiento de una competencia deportiva y eliminar resultado de enfrentamiento de una competencia deportiva. El organizador de competencia sigue las instrucciones del sistema e introduce los datos necesarios para la realización de estas acciones. Estos datos se registran

	en la Base de Datos si son correctos y termina el CUS.	
<b>Referencias</b>	<b>RF 9</b>	
<b>Precondiciones</b>	<ol style="list-style-type: none"> <li>1. Los deportes están almacenados en la Base de Datos.</li> <li>2. Los eventos por deporte están almacenados en la Base de Datos.</li> <li>3. Las competencias están almacenadas en la Base de Datos.</li> <li>4. Las representaciones están almacenadas en la Base de Datos.</li> <li>5. Los equipos están almacenados en la Base de Datos.</li> <li>6. Los atletas están almacenados en la Base de Datos.</li> <li>7. Los enfrentamientos están almacenados en la Base de Datos.</li> </ol>	
<b>Poscondiciones</b>	<ol style="list-style-type: none"> <li>4. Adicionado resultado de enfrentamiento a la Base de Datos.</li> <li>5. Información del resultado de enfrentamiento, modificada en la Base de Datos.</li> </ol>	
<b>Curso normal de los eventos</b>		
1. El organizador de competencia selecciona la opción de Gestionar resultado de un enfrentamiento de una competencia deportiva.	1.1	El sistema muestra las opciones: Adicionar resultado de un enfrentamiento de una competencia deportiva, Modificar resultado de un enfrentamiento de una competencia deportiva.
<b>Escenario 1: Adicionar resultados enfrentamiento.</b>		
1. El organizador de competencia selecciona la opción Adicionar resultado de enfrentamiento de una competencia.	1.1	El sistema muestra una lista con los deportes existentes en la Base de Datos.
2. El organizador de competencia selecciona el deporte en que se compitió en el enfrentamiento que se desea adicionar su resultado.	2.1	El sistema muestra un listado con los eventos correspondientes al deporte seleccionado existentes en la Base de Datos.
3. El organizador de competencia selecciona el evento correspondiente al enfrentamiento que se desea adicionar su resultado.	3.1	El sistema muestra las categorías de la competencia existentes en la Base de Datos.

4. El organizador de competencia selecciona la categoría de la competencia en que se desarrolló el enfrentamiento que se desea adicionar su resultado.	4.1 El sistema muestra un listado con las competencias correspondiente a la categoría seleccionada existentes en la Base de Datos.
5. El organizador de competencia selecciona la competencia en que se desarrolló el enfrentamiento que se desea adicionar su resultado.	5.1 El sistema muestra un listado con los tipos de enfrentamientos existentes en la Base de Datos.
6. El organizador de competencia selecciona el tipo de enfrentamiento del enfrentamiento que se desea adicionar su resultado.	6.1 El sistema muestra un listado con los enfrentamientos existentes en la Base de Datos del tipo seleccionado.
7. El organizador de competencia selecciona el enfrentamiento que desea adicionar su resultado.	7.1 El sistema muestra un formulario para adicionar los datos del resultado del enfrentamiento seleccionado.
8. El organizador de competencia introduce los datos solicitados por el sistema y selecciona la opción Adicionar.	8.1 El sistema verifica los datos introducidos por el organizador de competencia.
	8.2 Si los datos introducidos son correctos, el sistema adiciona los resultados del enfrentamiento seleccionado a la Base de Datos y termina el CUS.
<b>Curso alterno de los eventos</b>	
<b>Acción 8.2</b>	Si los datos introducidos por el organizador de competencia son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato erróneo e indica al usuario retornar a la acción 9.
<b>Escenario 2: Modificar resultados de enfrentamiento de competencias deportivas.</b>	
1. El organizador de competencia selecciona la opción Modificar resultado de	1.1 El sistema muestra una lista con los deportes existentes en la Base de Datos.

enfrentamiento de una competencia.	
2. El organizador de competencia selecciona el deporte en que se compitió en el enfrentamiento que se desea modificar su resultado.	2.1 El sistema muestra un listado con los eventos correspondientes al deporte seleccionado existentes en la Base de Datos.
3. El organizador de competencia selecciona el evento correspondiente al enfrentamiento que se desea modificar su resultado.	3.1 El sistema muestra las categorías de la competencia existentes en la Base de Datos.
4. El organizador de competencia selecciona la categoría de la competencia en que se desarrolló el enfrentamiento que se desea modificar su resultado.	4.1 El sistema muestra un listado con las competencias correspondiente a la categoría seleccionada existentes en la Base de Datos.
5. El organizador de competencia selecciona la competencia en que se desarrolló el enfrentamiento que se desea modificar su resultado.	5.1 El sistema muestra un listado con los tipos de enfrentamientos existentes en la Base de Datos.
6. El organizador de competencia selecciona el tipo de enfrentamiento del enfrentamiento que se desea modificar su resultado.	6.1 El sistema muestra un listado con los enfrentamientos existentes en la Base de Datos del tipo seleccionado.
7. El organizador de competencia selecciona el enfrentamiento que desea modificar su resultado y selecciona la opción Modificar.	7.1 El sistema localiza los datos del resultado del enfrentamiento seleccionado y los muestra listos para ser modificados.
8. El organizador de competencia realiza los cambios necesarios a los resultados del enfrentamiento y selecciona la opción Actualizar.	8.1 El sistema verifica los datos modificados por el organizador de competencia.

	8.1 Si los datos están correctos el sistema actualiza los resultados del enfrentamiento seleccionado en la Base de Datos y termina el CUS.
<b>Curso alterno de los eventos.</b>	
<b>Acción 8.2</b>	Si los datos introducidos por el organizador de competencia son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato erróneo e indica al usuario retornar a la acción 4.

Tabla A3.9 Descripción del caso de uso del sistema “Gestionar resultado de enfrentamiento de una competencia deportiva”

<b>Caso de Uso</b>	<b>Generar tabla de posiciones de una competencia deportiva.</b>	
<b>Actores</b>	Organizador de competencia (inicia).	
<b>Propósito</b>	El permite al organizador de competencia conocer el lugar en que quedó cada una de las representaciones en la competencia.	
<b>Resumen</b>	El CUS se inicia cuando el organizador de competencia selecciona la opción general tabla de posiciones de una competencia deportiva. El organizador selecciona la competencia donde desea generar la tabla de posiciones y el sistema muestra la tabla con las representaciones que participaron en la competencia en sus lugares correspondientes obtenidos en la competencia y termina el CUS.	
<b>Referencias</b>	<b>RF 10</b>	
<b>Precondiciones</b>	1. La competencia está almacenada en la base de datos.	
<b>Poscondiciones</b>		
<b>Curso normal de los eventos</b>		
<b>Acción del actor</b>	<b>Respuesta del sistema</b>	
2. El organizador de competencia	1.2 El sistema muestra un listado con las	

selecciona la opción Generar tabla de posiciones.	categorías de competencia existentes en la Base de Datos.
3. El organizador de competencia selecciona la categoría de la competencia que desea generar la tabla de posiciones.	1.3 El sistema muestra las competencias correspondientes a la categoría seleccionada existentes en la Base de Datos.
4. El organizador de competencia selecciona la competencia a la que le desea generar la tabla de posiciones y selecciona la opción Generar.	1.4 El sistema muestra la tabla de posiciones correspondiente a la competencia seleccionada.
<b>Prioridad</b>	Crítico

Tabla A3.10 Descripción del caso de uso del sistema “Generar tabla de posiciones de una competencia deportiva”

<b>Caso de Uso</b>	<b>Gestionar categoría de competencia</b>
<b>Actores</b>	Administrador de competencia (inicia).
<b>Propósito</b>	Permite al administrador de competencia adicionar, modificar y eliminar una categoría de competencia.
<b>Resumen</b>	El CUS se inicia cuando el administrador de competencia selecciona la opción Gestionar categoría de competencia. El sistema muestra las opciones Adicionar categoría de competencia, Modificar categoría de competencia y Eliminar categoría de competencia. El administrador de competencia sigue las instrucciones del sistema e introduce los datos necesarios para la realización de estas acciones. Estos datos se registran en la Base de Datos si son correctos y termina el CUS.
<b>Referencias</b>	<b>RF 11</b>
<b>Precondiciones</b>	-
<b>Poscondiciones</b>	10. Categoría de competencia adicionada a la Base de Datos. 11. Información de la categoría de competencia modificada en la Base de Datos.

	12. Categoría de competencia eliminada de la Base de Datos.
<b>Curso normal de los eventos</b>	
<b>Acción del actor</b>	<b>Respuesta del sistema</b>
1. El administrador de competencia selecciona la opción Gestionar categoría de competencia.	1.1 El sistema muestra las opciones: Adicionar categoría de competencia, Modificar categoría de competencia y Eliminar categoría de competencia.
<b>Escenario 1: Adicionar categoría de competencia.</b>	
1. El administrador de competencia selecciona la opción de Adicionar categoría de competencia.	1.1 El sistema muestra el formulario a completar para la adición de una nueva categoría de competencia.
2. El administrador de competencia introduce los datos solicitados por el sistema y selecciona la opción Adicionar.	2.1 El sistema verifica los datos introducidos por el administrador de competencia.
	2.2 Si los datos introducidos son correctos, el sistema adiciona dicha categoría de competencia a la Base de Datos y termina el CUS.
<b>Curso alterno de los eventos</b>	
<b>Acción 2.2</b>	Si los datos introducidos por el administrador de competencia son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato erróneo e indica al usuario retornar a la acción 2.
<b>Escenario 2: Modificar categoría de competencia.</b>	
1. El administrador de competencia selecciona la opción de Modificar categoría de competencia.	1.1 El sistema muestra un listado con las categorías de competencia existentes en la Base de Datos.
2. El administrador de competencia selecciona la categoría de competencia que	2.1 El sistema localiza los datos de la categoría de competencia seleccionada y los muestra, listos

será modificada y selecciona la opción Modificar.	para ser modificados.
3. El administrador de competencia realiza los cambios necesarios a los datos y selecciona la opción Actualizar.	3.1 El sistema verifica los datos modificados por el administrador de competencia.
	3.2 Si los datos están correctos el sistema actualiza los datos de la categoría de competencia seleccionada en la Base de Datos y termina el CUS.
<b>Curso alterno de los eventos</b>	
<b>Acción 3.2</b>	Si los datos introducidos por el administrador de competencia son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato erróneo e indica al usuario retornar a la acción 3.
<b>Escenario 3: Eliminar categoría de competencia.</b>	
1 El administrador de competencia selecciona la opción de Eliminar categoría de competencia.	1.1 El sistema muestra un listado con las categorías de competencia, existentes en la Base de Datos.
2 El administrador de competencia selecciona la o las categorías de competencia que serán eliminadas y selecciona la opción Eliminar.	2.1 El sistema muestra un mensaje de advertencia para la acción a realizar.
3 El administrador de competencia confirma si desea o no eliminar la o las categorías de competencia seleccionadas.	3.1 Si el administrador de competencia acepta, el sistema elimina los datos correspondientes a la categoría de competencia, o las categorías de competencia seleccionadas en la Base de Datos y termina el CUS.
<b>Curso alterno de los eventos</b>	
<b>Acción 3.1</b>	Si el administrador de competencia cancela la acción, se culmina el CUS sin ejecutar ninguna

	acción.
<b>Prioridad</b>	Crítico

Tabla A3.11 Descripción del caso de uso del sistema “Gestionar categoría de competencia”

<b>Caso de Uso</b>	<b>Gestionar fase de competencia.</b>	
<b>Actores</b>	Administrador de competencia (inicia).	
<b>Propósito</b>	Permite al administrador de competencia adicionar, modificar y eliminar una fase de competencia.	
<b>Resumen</b>	El CUS se inicia cuando el administrador de competencia selecciona la opción Gestionar fase de competencia. El sistema muestra las opciones Adicionar fase de competencia, Modificar fase de competencia y Eliminar fase de competencia. El administrador de competencia sigue las instrucciones del sistema e introduce los datos necesarios para la realización de estas acciones. Estos datos se registran en la Base de Datos si son correctos y termina el CUS.	
<b>Referencias</b>	<b>RF 12</b>	
<b>Precondiciones</b>	-	
<b>Poscondiciones</b>	<ol style="list-style-type: none"> <li>1. Fase de competencia adicionada a la Base de Datos.</li> <li>2. Información de la fase de competencia modificada en la Base de Datos.</li> <li>3. Fase de competencia eliminada de la Base de Datos.</li> </ol>	
<b>Curso normal de los eventos</b>		
<b>Acción del actor</b>	<b>Respuesta del sistema</b>	
2. El administrador de competencia selecciona la opción Gestionar fase de competencia.	1.1 El sistema muestra las opciones: Adicionar fase de competencia, Modificar fase de competencia y Eliminar fase de competencia.	
<b>Escenario 1: Adicionar fase de competencia.</b>		

1. El administrador de competencia selecciona la opción de Adicionar fase de competencia.	1.1 El sistema muestra el formulario a completar para la adición de una nueva fase de competencia.
2. El administrador de competencia introduce los datos solicitados por el sistema y selecciona la opción Adicionar.	2.1 El sistema verifica los datos introducidos por el administrador de competencia.
	2.2 Si los datos introducidos son correctos, el sistema adiciona dicha fase de competencia a la Base de Datos y termina el CUS.
<b>Curso alterno de los eventos</b>	
<b>Acción 2.2</b>	Si los datos introducidos por el administrador de competencia son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato erróneo e indica al usuario retornar a la acción 2.
<b>Escenario 2: Modificar fase de competencia.</b>	
1. El administrador de competencia selecciona la opción de Modificar fase de competencia.	1.2 El sistema muestra un listado con las fases de competencia existentes en la Base de Datos.
2. El administrador de competencia selecciona la fase de competencia que será modificada y selecciona la opción Modificar.	2.2 El sistema localiza los datos de la fase de competencia seleccionada y los muestra, listos para ser modificados.
3. El administrador de competencia realiza los cambios necesarios a los datos y selecciona la opción Actualizar.	3.1 El sistema verifica los datos modificados por el administrador de competencia.
	3.2 Si los datos están correctos el sistema actualiza los datos de la fase de competencia en la Base de Datos y termina el CUS.
<b>Curso alterno de los eventos</b>	

<b>Acción 3.2</b>	Si los datos introducidos por el administrador de competencia son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato erróneo e indica al usuario retornar a la acción 3.
<b>Escenario 3: Eliminar fase de competencia.</b>	
1 El administrador de competencia selecciona la opción de Eliminar fase de competencia.	1.2 El sistema muestra un listado con las fases de competencia existentes en la Base de Datos.
2 El administrador de competencia selecciona la o las fases de competencia que serán eliminadas y selecciona la opción Eliminar.	2.1 El sistema muestra un mensaje de advertencia para la acción a realizar.
3 El administrador de competencia confirma si desea o no eliminar la o las fases de competencia seleccionadas.	3.1 Si el administrador de competencia acepta, el sistema elimina los datos correspondientes a la fase de competencia, o las fases de competencia seleccionadas en la Base de Datos y termina el CUS.
<b>Curso alterno de los eventos</b>	
<b>Acción 3.1</b>	Si el administrador de competencia cancela la acción, se culmina el CUS sin ejecutar ninguna acción.
<b>Prioridad</b>	Crítico

Tabla A3.12 Descripción del caso de uso del sistema “Gestionar fase de competencia”

<b>Caso de Uso</b>	<b>Gestionar sistema de competencia.</b>
<b>Actores</b>	Administrador de competencia (inicia).
<b>Propósito</b>	Permite al administrador de competencia adicionar, modificar y eliminar un sistema de competencia.

<b>Resumen</b>	El CUS se inicia cuando el administrador de competencia selecciona la opción Gestionar sistema de competencia. El sistema muestra las opciones Adicionar sistema de competencia, Modificar sistema de competencia y Eliminar sistema de competencia. El administrador de competencia sigue las instrucciones del sistema e introduce los datos necesarios para la realización de estas acciones. Estos datos se registran en la Base de Datos si son correctos y termina el CUS.	
<b>Referencias</b>	<b>RF 13</b>	
<b>Precondiciones</b>	-	
<b>Poscondiciones</b>	<p>4. Sistema de competencia adicionado a la Base de Datos.</p> <p>5. Información del sistema de competencia modificado en la Base de Datos.</p> <p>6. Sistema de competencia eliminado de la Base de Datos.</p>	
<b>Curso normal de los eventos</b>		
<b>Acción del actor</b>	<b>Respuesta del sistema</b>	
3. El administrador de competencia selecciona la opción Gestionar sistema de competencia.	1.1 El sistema muestra las opciones: Adicionar sistema de competencia, Modificar sistema de competencia y Eliminar sistema de competencia.	
<b>Escenario 1: Adicionar sistema de competencia.</b>		
1. El administrador de competencia selecciona la opción de Adicionar sistema de competencia.	1.1 El sistema muestra el formulario a completar para la adición de un nuevo sistema de competencia.	
2. El administrador de competencia introduce los datos solicitados por el sistema y selecciona la opción Adicionar.	2.1 El sistema verifica los datos introducidos por el administrador de competencia.	
	2.2 Si los datos introducidos son correctos, el sistema adiciona dicho sistema de competencia a la Base de Datos y termina el CUS.	

<b>Curso alterno de los eventos</b>	
<b>Acción 2.2</b>	Si los datos introducidos por el administrador de competencia son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato erróneo e indica al usuario retornar a la acción 2.
<b>Escenario 2: Modificar sistema de competencia.</b>	
1. El administrador de competencia selecciona la opción de Modificar sistema de competencia.	1.1 El sistema muestra un listado con los sistemas de competencia existentes en la Base de Datos.
2. El administrador de competencia selecciona el sistema de competencia que será modificado y selecciona la opción Modificar.	2.1 El sistema localiza los datos del sistema de competencia seleccionado y los muestra, listos para ser modificados.
3. El administrador de competencia realiza los cambios necesarios a los datos y selecciona la opción Actualizar.	3.1 El sistema verifica los datos modificados por el administrador de competencia.
	3.2 Si los datos están correctos el sistema actualiza los datos del sistema de competencia en la Base de Datos y termina el CUS.
<b>Curso alterno de los eventos</b>	
<b>Acción 3.2</b>	Si los datos introducidos por el administrador de competencia son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato erróneo e indica al usuario retornar a la acción 3.
<b>Escenario 3: Eliminar sistema de competencia.</b>	
1 El administrador de competencia selecciona la opción de Eliminar sistema de competencia.	1.1 El sistema muestra un listado con los sistemas de competencia existentes en la Base de Datos.
2 El administrador de competencia	2.1 El sistema muestra un mensaje de

selecciona el o los sistemas de competencia que serán eliminados y selecciona la opción Eliminar.	advertencia para la acción a realizar.
3 El administrador de competencia confirma si desea o no eliminar el o los sistemas de competencia seleccionados.	3.1 Si el administrador de competencia acepta, el sistema elimina los datos correspondientes al sistema de competencia, o los sistemas de competencia seleccionados en la Base de Datos y termina el CUS.
<b>Curso alternativo de los eventos</b>	
<b>Acción 3.1</b>	Si el administrador de competencia cancela la acción, se culmina el CUS sin ejecutar ninguna acción.
<b>Prioridad</b>	Crítico

Tabla A3.13 Descripción del caso de uso del sistema “Gestionar sistema de competencia”

<b>Caso de Uso</b>	<b>Gestionar tipo de enfrentamiento.</b>
<b>Actores</b>	Administrador de competencia (inicia).
<b>Propósito</b>	Permite al administrador de competencia adicionar, modificar y eliminar un tipo de enfrentamiento.
<b>Resumen</b>	El CUS se inicia cuando el administrador de competencia selecciona la opción Gestionar tipo de enfrentamiento. El sistema muestra las opciones Adicionar tipo de enfrentamiento, Modificar tipo de enfrentamiento y Eliminar tipo de enfrentamiento. El administrador de competencia sigue las instrucciones del sistema e introduce los datos necesarios para la realización de estas acciones. Estos datos se registran en la Base de Datos si son correctos y termina el CUS.
<b>Referencias</b>	RF 13
<b>Precondiciones</b>	-

<b>Poscondiciones</b>	<ol style="list-style-type: none"> <li>1. Tipo de enfrentamiento adicionado a la Base de Datos.</li> <li>2. Información del tipo de enfrentamiento modificado en la Base de Datos.</li> <li>3. Tipo de enfrentamiento eliminado de la Base de Datos.</li> </ol>
<b>Curso normal de los eventos</b>	
<b>Acción del actor</b>	<b>Respuesta del sistema</b>
4. El administrador de competencia selecciona la opción Gestionar tipo de enfrentamiento.	1.1 El sistema muestra las opciones: Adicionar tipo de enfrentamiento, Modificar tipo de enfrentamiento y Eliminar tipo de enfrentamiento.
<b>Escenario 1: Adicionar tipo de enfrentamiento</b>	
1. El administrador de competencia selecciona la opción de Adicionar tipo de enfrentamiento.	1.1 El sistema muestra el formulario a completar para la adición de un nuevo tipo de enfrentamiento.
2. El administrador de competencia introduce los datos solicitados por el sistema y selecciona la opción Adicionar.	2.1 El sistema verifica los datos introducidos por el administrador de competencia.
	2.2 Si los datos introducidos son correctos, el sistema adiciona dicho tipo de enfrentamiento a la Base de Datos y termina el CUS.
<b>Curso alterno de los eventos</b>	
<b>Acción 2.2</b>	Si los datos introducidos por el administrador de competencia son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato erróneo e indica al usuario retornar a la acción 2.
<b>Escenario 2: Modificar tipo de enfrentamiento.</b>	
1. El administrador de competencia selecciona la opción de Modificar tipo de enfrentamiento.	1.1 El sistema muestra un listado con los tipos de enfrentamiento existentes en la Base de Datos.

<p>2. El administrador de competencia selecciona el tipo de enfrentamiento que será modificado y selecciona la opción Modificar.</p>	<p>2.1 El sistema localiza los datos del tipo de enfrentamiento seleccionado y los muestra, listos para ser modificados.</p>
<p>3. El administrador de competencia realiza los cambios necesarios a los datos y selecciona la opción Actualizar.</p>	<p>3.1 El sistema verifica los datos modificados por el administrador de competencia.</p>
	<p>3.2 Si los datos están correctos el sistema actualiza los datos del tipo de enfrentamiento en la Base de Datos y termina el CUS.</p>
<p><b>Curso alterno de los eventos</b></p>	
<p><b>Acción 3.2</b></p>	<p>Si los datos introducidos por el administrador de competencia son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato erróneo e indica al usuario retornar a la acción 3.</p>
<p><b>Escenario 3: Eliminar tipo de enfrentamiento</b></p>	
<p>1 El administrador de competencia selecciona la opción de Eliminar tipo de enfrentamiento.</p>	<p>1.1 El sistema muestra un listado con los tipos de enfrentamiento existentes en la Base de Datos.</p>
<p>2 El administrador de competencia selecciona el o los tipos de enfrentamiento que serán eliminados y selecciona la opción Eliminar.</p>	<p>2.1 El sistema muestra un mensaje de advertencia para la acción a realizar.</p>
<p>3 El administrador de competencia confirma si desea o no eliminar el o los tipos de enfrentamiento seleccionados.</p>	<p>3.1 Si el administrador de competencia acepta, el sistema elimina los datos correspondientes al tipo de enfrentamiento, o los tipos de enfrentamiento seleccionados en la Base de Datos y termina el CUS.</p>
<p><b>Curso alterno de los eventos</b></p>	

<b>Acción 3.1</b>	Si el administrador de competencia cancela la acción, se culmina el CUS sin ejecutar ninguna acción.
<b>Prioridad</b>	Crítico

Tabla A3.14 Descripción del caso de uso del sistema “Gestionar tipo de enfrentamiento”

<b>Caso de Uso</b>	<b>Gestionar unidad de resultado de enfrentamiento.</b>	
<b>Actores</b>	Administrador de competencia (inicia).	
<b>Propósito</b>	Permite al administrador de competencia adicionar, modificar y eliminar una unidad de resultado de enfrentamiento.	
<b>Resumen</b>	El CUS se inicia cuando el administrador de competencia selecciona la opción Gestionar unidad de resultado de enfrentamiento. El sistema muestra las opciones Adicionar unidad de resultado de enfrentamiento, Modificar unidad de resultado de enfrentamiento y Eliminar unidad de resultado de enfrentamiento. El administrador de competencia sigue las instrucciones del sistema e introduce los datos necesarios para la realización de estas acciones. Estos datos se registran en la Base de Datos si son correctos y termina el CUS.	
<b>Referencias</b>	RF 13	
<b>Precondiciones</b>	-	
<b>Poscondiciones</b>	<ol style="list-style-type: none"> <li>1. Unidad de resultado de enfrentamiento adicionada a la Base de Datos.</li> <li>2. Información de la unidad de resultado de enfrentamiento modificado en la Base de Datos.</li> <li>3. Unidad de resultado de enfrentamiento eliminada de la Base de Datos.</li> </ol>	
<b>Curso normal de los eventos</b>		
<b>Acción del actor</b>	<b>Respuesta del sistema</b>	

<p>1 El administrador de competencia selecciona la opción Gestionar unidad de resultado de enfrentamiento.</p>	<p>1.1 El sistema muestra las opciones: Adicionar unidad de resultado de enfrentamiento, Modificar unidad de resultado de enfrentamiento y Eliminar unidad de resultado de enfrentamiento.</p>
<p><b>Escenario 1: Adicionar unidad de resultado de enfrentamiento</b></p>	
<p>1. El administrador de competencia selecciona la opción de Adicionar unidad de resultado de enfrentamiento.</p>	<p>1.1 El sistema muestra el formulario a completar para la adición de una nueva unidad de resultado de enfrentamiento.</p>
<p>2. El administrador de competencia introduce los datos solicitados por el sistema y selecciona la opción Adicionar.</p>	<p>2.1 El sistema verifica los datos introducidos por el administrador de competencia.</p>
	<p>2.2 Si los datos introducidos son correctos, el sistema adiciona dicha unidad de resultado de enfrentamiento a la Base de Datos y termina el CUS.</p>
<p><b>Curso alterno de los eventos</b></p>	
<p><b>Acción 2.2</b></p>	<p>Si los datos introducidos por el administrador de competencia son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato erróneo e indica al usuario retornar a la acción 2.</p>
<p><b>Escenario 2: Modificar unidad de resultado de enfrentamiento.</b></p>	
<p>1. El administrador de competencia selecciona la opción de Modificar unidad de resultado de enfrentamiento.</p>	<p>1.1 El sistema muestra un listado con las unidades de resultado de enfrentamiento existentes en la Base de Datos.</p>
<p>2. El administrador de competencia selecciona la unidad de resultado de enfrentamiento que será modificado y selecciona la opción Modificar.</p>	<p>2.1 El sistema localiza los datos de la unidad de resultado de enfrentamiento seleccionada y los muestra, listos para ser modificados.</p>

<p>3. El administrador de competencia realiza los cambios necesarios a los datos y selecciona la opción Actualizar.</p>	<p>3.1 El sistema verifica los datos modificados por el administrador de competencia.</p>
	<p>3.2 Si los datos están correctos el sistema actualiza los datos de la unidad de resultado de enfrentamiento en la Base de Datos y termina el CUS.</p>
<p><b>Curso alterno de los eventos</b></p>	
<p><b>Acción 3.2</b></p>	<p>Si los datos introducidos por el administrador de competencia son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato erróneo e indica al usuario retornar a la acción 3.</p>
<p><b>Escenario 3: Eliminar unidad de resultado de enfrentamiento</b></p>	
<p>1 El administrador de competencia selecciona la opción de Eliminar unidad de resultado de enfrentamiento.</p>	<p>1.1 El sistema muestra un listado con las unidades de resultado de enfrentamiento existentes en la Base de Datos.</p>
<p>2 El administrador de competencia selecciona la o las unidades de resultado de enfrentamiento que serán eliminadas y selecciona la opción Eliminar.</p>	<p>2.1 El sistema muestra un mensaje de advertencia para la acción a realizar.</p>
<p>3 El administrador de competencia confirma si desea o no eliminar la o las unidades de resultado de enfrentamiento seleccionadas.</p>	<p>3.1 Si el administrador de competencia acepta, el sistema elimina los datos correspondientes a la unidad de resultado de enfrentamiento, o las unidades de resultado de enfrentamiento seleccionados en la Base de Datos y termina el CUS.</p>
<p><b>Curso alterno de los eventos</b></p>	
<p><b>Acción 3.1</b></p>	<p>Si el administrador de competencia cancela la acción, se culmina el CUS sin ejecutar ninguna</p>

	acción.
<b>Prioridad</b>	Opcional

Tabla A3.15 Descripción del caso de uso del sistema “Gestionar unidad de resultado de enfrentamiento”

<b>Caso de Uso</b>	<b>Gestionar color de piel.</b>	
<b>Actores</b>	Administrador general (inicia).	
<b>Propósito</b>	Permite al administrador general adicionar, modificar y eliminar un color de piel.	
<b>Resumen</b>	El CUS se inicia cuando el administrador general selecciona la opción Gestionar color de piel. El sistema muestra las opciones Adicionar color de piel, Modificar color de piel y Eliminar color de piel. El administrador general sigue las instrucciones del sistema e introduce los datos necesarios para la realización de estas acciones. Estos datos se registran en la Base de Datos si son correctos y termina el CUS.	
<b>Referencias</b>	RF 13	
<b>Precondiciones</b>	-	
<b>Poscondiciones</b>	<ol style="list-style-type: none"> <li>1 Color de piel adicionado a la Base de Datos.</li> <li>2 Información del color de piel modificado en la Base de Datos.</li> <li>3 Color de piel eliminado de la Base de Datos.</li> </ol>	
<b>Curso normal de los eventos</b>		
<b>Acción del actor</b>	<b>Respuesta del sistema</b>	
1 El administrador general selecciona la opción Gestionar color de piel.	1.1 El sistema muestra las opciones: Adicionar color de piel, Modificar color de piel y Eliminar color de piel.	
<b>Escenario 1: Adicionar color de piel</b>		

1. El administrador general selecciona la opción de Adicionar color de piel.	1.1 El sistema muestra el formulario a completar para la adición de un nuevo color de piel.
2. El administrador general introduce los datos solicitados por el sistema y selecciona la opción Adicionar.	2.1 El sistema verifica los datos introducidos por el administrador general.
	2.2 Si los datos introducidos son correctos, el sistema adiciona dicho color de piel a la Base de Datos y termina el CUS.
<b>Curso alternativo de los eventos</b>	
<b>Acción 2.2</b>	Si los datos introducidos por el administrador general son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato erróneo e indica al administrador general retornar a la acción 2.
<b>Escenario 2: Modificar color de piel.</b>	
1. El administrador general selecciona la opción de Modificar color de piel.	1.1 El sistema muestra un listado con los colores de piel existentes en la Base de Datos.
2. El administrador general selecciona el color de piel que será modificado y selecciona la opción Modificar.	2.1 El sistema localiza los datos del color de piel seleccionado y los muestra, listos para ser modificados.
3. El administrador general realiza los cambios necesarios a los datos y selecciona la opción Actualizar.	3.1 El sistema verifica los datos modificados por el administrador general.
	3.2 Si los datos están correctos el sistema actualiza los datos del color de piel en la Base de Datos y termina el CUS.
<b>Curso alternativo de los eventos</b>	

<b>Acción 3.2</b>	Si los datos introducidos por el administrador general son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato erróneo e indica al usuario retornar a la acción 3.
<b>Escenario 3: Eliminar color de piel.</b>	
1 El administrador general selecciona la opción de Eliminar color de piel.	1.1 El sistema muestra un listado con los colores de piel existentes en la Base de Datos.
2 El administrador general selecciona el o los colores de piel que serán eliminados y selecciona la opción Eliminar.	2.1 El sistema muestra un mensaje de advertencia para la acción a realizar.
3 El administrador general confirma si desea o no eliminar el o los colores de piel seleccionados.	3.1 Si el administrador general acepta, el sistema elimina los datos correspondientes al color de piel, o los colores de piel seleccionados en la Base de Datos y termina el CUS.
<b>Curso alterno de los eventos</b>	
<b>Acción 3.1</b>	Si el administrador general cancela la acción, se culmina el CUS sin ejecutar ninguna acción.
<b>Prioridad</b>	Crítico

Tabla A3.16 Descripción del caso de uso del sistema “Gestionar color de la piel”

<b>Caso de Uso</b>	<b>Gestionar deporte.</b>
<b>Actores</b>	Administrador general (inicia).
<b>Propósito</b>	Permite al administrador general adicionar, modificar y eliminar deporte.
<b>Resumen</b>	El CUS se inicia cuando el administrador general selecciona la opción Gestionar deporte. El sistema muestra las opciones Adicionar deporte, Modificar deporte y Eliminar deporte. El administrador general sigue las instrucciones del sistema e introduce los datos necesarios para la realización de estas acciones. Estos datos se registran en la Base de Datos

	si son correctos y termina el CUS.	
<b>Referencias</b>	RF 13	
<b>Precondiciones</b>	Tipo de deporte almacenado en la Base de Datos.	
<b>Poscondiciones</b>	4	Deporte adicionado a la Base de Datos.
	5	Información del deporte, modificada en la Base de Datos.
	6	Deporte eliminado de la Base de Datos.
<b>Curso normal de los eventos</b>		
<b>Acción del actor</b>	<b>Respuesta del sistema</b>	
2 El administrador general selecciona la opción Gestionar deporte.	1.2 El sistema muestra las opciones: Adicionar deporte, Modificar deporte y Eliminar deporte.	
<b>Escenario 1: Adicionar deporte</b>		
3. El administrador general selecciona la opción de Adicionar deporte.	1.1 El sistema muestra los tipos de deporte existentes en la Base de Datos.	
4. El administrador general selecciona el tipo de deporte al que pertenece el deporte que se desea adicionar.	1.1 El sistema muestra el formulario a completar para la adición de un nuevo deporte.	
5. El administrador general introduce los datos solicitados por el sistema y selecciona la opción Adicionar.	2.1 El sistema verifica los datos introducidos por el administrador general.	
	2.2 Si los datos introducidos son correctos, el sistema adiciona dicho deporte a la Base de Datos y termina el CUS.	
<b>Curso alterno de los eventos</b>		
<b>Acción 2.2</b>	Si los datos introducidos por el administrador general son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato erróneo e indica al usuario retornar a la acción 2.	

<b>Escenario 2: Modificar deporte.</b>	
4. El administrado general selecciona la opción de Modificar deporte.	1.2 El sistema muestra un listado con los tipos de deporte existentes en la Base de Datos.
5. El administrador de competencia selecciona el tipo de deporte correspondiente al deporte que se desea modificar.	1.3 El sistema muestra un listado con los deportes pertenecientes al tipo de deporte seleccionado existentes en la Base de Datos.
6. El administrador general selecciona el deporte que será modificado y selecciona la opción Modificar.	2.2 El sistema localiza los datos del deporte seleccionado y los muestra, listos para ser modificados.
7. El administrador general realiza los cambios necesarios a los datos y selecciona la opción Actualizar.	3.3 El sistema verifica los datos modificados por el administrador general.
	3.4 Si los datos están correctos el sistema actualiza los datos del deporte en la Base de Datos y termina el CUS.
<b>Curso alterno de los eventos</b>	
<b>Acción 3.2</b>	Si los datos introducidos por el administrador general son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato erróneo e indica al usuario retornar a la acción 3.
<b>Escenario 3: Eliminar deporte.</b>	
4 El administrador general selecciona la opción de Eliminar deporte.	1.1 .El sistema muestra un listado con los tipos de deporte existentes en la Base de Datos.
5 El administrador general selecciona el tipo de deporte correspondiente al deporte que se desea eliminar.	1.2 El sistema muestra un listado con los deportes pertenecientes al tipo de deporte seleccionado existentes en la Base de Datos
6 El administrador general selecciona el o los deportes que serán eliminados y	2.2 El sistema muestra un mensaje de advertencia para la acción a realizar.

selecciona la opción Eliminar.		
7	El administrador general confirma si desea o no eliminar el o los deportes seleccionados.	3.2 Si el administrador general acepta, el sistema elimina los datos correspondientes al deporte, o los deportes seleccionados en la Base de Datos y termina el CUS.
<b>Curso alternativo de los eventos</b>		
<b>Acción 3.1</b>		Si el administrador general cancela la acción, se culmina el CUS sin ejecutar ninguna acción.
<b>Prioridad</b>	Crítico	

Tabla A3.17 Descripción del caso de uso del sistema “Gestionar deporte”

Caso de Uso	Gestionar localización de lesión.
<b>Actores</b>	Administrador de atleta (inicia).
<b>Propósito</b>	Permite al administrador de atleta adicionar, modificar y eliminar una localización de lesión.
<b>Resumen</b>	El CUS se inicia cuando el administrador de atleta selecciona la opción Gestionar localización de lesión. El sistema muestra las opciones Adicionar localización de lesión, Modificar localización de lesión y Eliminar localización de lesión. El administrador de atleta sigue las instrucciones del sistema e introduce los datos necesarios para la realización de estas acciones. Estos datos se registran en la Base de Datos si son correctos y termina el CUS.
<b>Referencias</b>	RF 13
<b>Precondiciones</b>	-
<b>Poscondiciones</b>	7 Localización de lesión adicionada a la Base de Datos. 8 Información de la localización de lesión, modificada en la Base de Datos. 9 Localización de lesión eliminada de la Base de Datos.

<b>Curso normal de los eventos</b>	
<b>Acción del actor</b>	<b>Respuesta del sistema</b>
3 El administrador de atleta selecciona la opción Gestionar localización de lesión.	1.3 El sistema muestra las opciones: Adicionar localización de lesión, Modificar localización de lesión y Eliminar localización de lesión.
<b>Escenario 1: Adicionar ocupación</b>	
6. El administrador de atleta selecciona la opción de Adicionar localización de lesión.	1.1 El sistema muestra el formulario a completar para la adición de una nueva localización de lesión.
7. El administrador de atleta introduce los datos solicitados por el sistema y selecciona la opción Adicionar.	2.1 El sistema verifica los datos introducidos por el administrador de atleta.
	2.2 Si los datos introducidos son correctos, el sistema adiciona dicha localización de lesión a la Base de Datos y termina el CUS.
<b>Curso alterno de los eventos</b>	
<b>Acción 2.2</b>	Si los datos introducidos por el administrador de atleta son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato erróneo e indica al usuario retornar a la acción 2.
<b>Escenario 2: Modificar localización de lesión.</b>	
8. El administrador de atleta selecciona la opción de Modificar localización de lesión.	1.4 El sistema muestra un listado con las localizaciones de lesión existentes en la Base de Datos.
9. El administrador de atleta selecciona la localización de lesión que será modificada y selecciona la opción Modificar.	El sistema localiza los datos de la localización de lesión seleccionada y los muestra, listos para ser modificados.
10. El administrador de atleta realiza los	3.5 El sistema verifica los datos modificados por el

cambios necesarios a los datos y selecciona la opción Actualizar.	administrador de atleta.
	3.6 Si los datos están correctos el sistema actualiza los datos de la localización de lesión en la Base de Datos y termina el CUS.
<b>Curso alterno de los eventos</b>	
<b>Acción 3.2</b>	Si los datos introducidos por el administrador de atleta son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato erróneo e indica al usuario retornar a la acción 3.
<b>Escenario 3: Eliminar localización de lesión.</b>	
8 El administrador de atleta selecciona la opción de Eliminar localización de lesión.	1.2 El sistema muestra un listado con las localizaciones de lesión existentes en la Base de Datos.
9 El administrador de atleta selecciona la o las localizaciones de lesión que serán eliminadas y selecciona la opción Eliminar.	El sistema muestra un mensaje de advertencia para la acción a realizar.
10 El administrador de atleta confirma si desea o no eliminar la o las localizaciones de lesión seleccionadas.	3.3 Si el administrador de atleta acepta, el sistema elimina los datos correspondientes a la localización de lesión, o las localizaciones de lesión seleccionadas en la Base de Datos y termina el CUS.
<b>Curso alterno de los eventos</b>	
<b>Acción 3.1</b>	Si el administrador de atleta cancela la acción, se culmina el CUS sin ejecutar ninguna acción.
<b>Prioridad</b>	Crítico

Tabla A3.18 Descripción del caso de uso del sistema “Gestionar localización de lesión”

<b>Caso de Uso</b>	<b>Gestionar tipo de lesión.</b>	
<b>Actores</b>	Administrador de atleta (inicia).	
<b>Propósito</b>	Permite al administrador de atleta adicionar, modificar y eliminar un tipo de lesión.	
<b>Resumen</b>	El CUS se inicia cuando el administrador de atleta selecciona la opción Gestionar tipo de lesión. El sistema muestra las opciones Adicionar tipo de lesión, Modificar tipo de lesión y Eliminar tipo de lesión. El administrador de atleta sigue las instrucciones del sistema e introduce los datos necesarios para la realización de estas acciones. Estos datos se registran en la Base de Datos si son correctos y termina el CUS.	
<b>Referencias</b>	RF 13	
<b>Precondiciones</b>	-	
<b>Poscondiciones</b>	10 Tipo de lesión adicionada a la Base de Datos. 11 Información de la tipo de lesión, modificada en la Base de Datos. 12 Tipo de lesión eliminada de la Base de Datos.	
<b>Curso normal de los eventos</b>		
<b>Acción del actor</b>	<b>Respuesta del sistema</b>	
4 El administrador de atleta selecciona la opción Gestionar tipo de lesión.	1.4 El sistema muestra las opciones: Adicionar tipo de lesión, Modificar tipo de lesión y Eliminar tipo de lesión.	
<b>Escenario 1: Adicionar tipo de lesión.</b>		
8. El administrador de atleta selecciona la opción de Adicionar tipo de lesión.	1.1 El sistema muestra el formulario a completar para la adición de un nuevo tipo de lesión.	
9. El administrador de atleta introduce los datos solicitados por el sistema y selecciona la opción Adicionar.	2.1 El sistema verifica los datos introducidos por el administrador de atleta.	

	2.2 Si los datos introducidos son correctos, el sistema adiciona el tipo de lesión a la Base de Datos y termina el CUS.
<b>Curso alterno de los eventos</b>	
<b>Acción 2.2</b>	Si los datos introducidos por el administrador de atleta son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato erróneo e indica al usuario retornar a la acción 2.
<b>Escenario 2: Modificar tipo de lesión.</b>	
11. El administrador de atleta selecciona la opción de Modificar tipo de lesión.	1.5 El sistema muestra un listado con los tipos de lesión existentes en la Base de Datos.
12.El administrador de atleta selecciona el tipo de lesión que será modificado y selecciona la opción Modificar.	El sistema localiza los datos del tipo de lesión seleccionado y los muestra, listos para ser modificados.
13. El administrador de atleta realiza los cambios necesarios a los datos y selecciona la opción Actualizar.	3.7 El sistema verifica los datos modificados por el administrador de atleta.
	3.8 Si los datos están correctos el sistema actualiza los datos del tipo de lesión en la Base de Datos y termina el CUS.
<b>Curso alterno de los eventos</b>	
<b>Acción 3.2</b>	Si los datos introducidos por el administrador de atleta son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato erróneo e indica al usuario retornar a la acción 3.
<b>Escenario 3: Eliminar tipo de lesión.</b>	
11 El administrador de atleta selecciona la opción de Eliminar tipo de lesión.	1.3 El sistema muestra un listado con los tipos de lesión existentes en la Base de Datos.
12 El administrador de atleta selecciona el	El sistema muestra un mensaje de advertencia para

o los tipos de lesión que serán eliminados y selecciona la opción Eliminar.	la acción a realizar.
13 El administrador de atleta confirma si desea o no eliminar el o los tipos de lesión seleccionados.	3.4 Si el administrador de atleta acepta, el sistema elimina los datos correspondientes al tipo de lesión, o los tipos de lesión seleccionados en la Base de Datos y termina el CUS.
<b>Curso alterno de los eventos</b>	
<b>Acción 3.1</b>	Si el administrador de atleta cancela la acción, se culmina el CUS sin ejecutar ninguna acción.
<b>Prioridad</b>	Crítico

Tabla A3.19 Descripción del caso de uso del sistema “Gestionar tipo de lesión”

<b>Caso de Uso</b>	<b>Gestionar utensilio deportivo.</b>
<b>Actores</b>	Administrador de gimnasio (inicia).
<b>Propósito</b>	Permite al administrador de gimnasio adicionar, modificar y eliminar un utensilio deportivo.
<b>Resumen</b>	El CUS se inicia cuando el administrador de gimnasio selecciona la opción Gestionar utensilio deportivo. El sistema muestra las opciones Adicionar utensilio deportivo, Modificar utensilio deportivo y Eliminar utensilio deportivo. El administrador de gimnasio sigue las instrucciones del sistema e introduce los datos necesarios para la realización de estas acciones. Estos datos se registran en la Base de Datos si son correctos y termina el CUS.
<b>Referencias</b>	RF 13
<b>Precondiciones</b>	-
<b>Poscondiciones</b>	13 Utensilio deportivo adicionado a la Base de Datos. 14 Información del utensilio deportivo, modificada en la Base de Datos. 15 Utensilio deportivo eliminado de la Base de Datos.

<b>Curso normal de los eventos</b>	
<b>Acción del actor</b>	<b>Respuesta del sistema</b>
5 El administrador de gimnasio selecciona la opción Gestionar utensilio deportivo.	1.5 El sistema muestra las opciones: Adicionar utensilio deportivo, Modificar utensilio deportivo y Eliminar utensilio deportivo.
<b>Escenario 1: Adicionar tipo de lesión.</b>	
10. El administrador de gimnasio selecciona la opción de Adicionar utensilio deportivo.	1.1 El sistema muestra el formulario a completar para la adición de un nuevo utensilio deportivo.
11. El administrador de gimnasio introduce los datos solicitados por el sistema y selecciona la opción Adicionar.	2.1 El sistema verifica los datos introducidos por el administrador de gimnasio.
	2.2 Si los datos introducidos son correctos, el sistema adiciona dicho utensilio deportivo a la Base de Datos y termina el CUS.
<b>Curso alterno de los eventos</b>	
<b>Acción 2.2</b>	Si los datos introducidos por el administrador de gimnasio son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato erróneo e indica al usuario retornar a la acción 2.
<b>Escenario 2: Modificar utensilio deportivo.</b>	
14. El administrador de gimnasio selecciona la opción de Modificar utensilio deportivo.	1.6 El sistema muestra un listado con los utensilios deportivos existentes en la Base de Datos.
15.El administrador de gimnasio selecciona el utensilio deportivo que será modificado y selecciona la opción Modificar.	El sistema localiza los datos del utensilio deportivo seleccionado y los muestra, listos para ser modificados.
16. El administrador de gimnasio realiza los	3.9 El sistema verifica los datos modificados por el

cambios necesarios a los datos y selecciona la opción Actualizar.	administrador de gimnasio.
	3.10 Si los datos están correctos el sistema actualiza los datos del utensilio deportivo en la Base de Datos y termina el CUS.
<b>Curso alterno de los eventos</b>	
<b>Acción 3.2</b>	Si los datos introducidos por el administrador de gimnasio son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato erróneo e indica al usuario retornar a la acción 3.
<b>Escenario 3: Eliminar utensilio deportivo.</b>	
14 El administrador de gimnasio selecciona la opción de Eliminar utensilio deportivo.	1.4 El sistema muestra un listado con los utensilios deportivos existentes en la Base de Datos.
15 El administrador de gimnasio selecciona el o los utensilios deportivos que serán eliminados y selecciona la opción Eliminar.	El sistema muestra un mensaje de advertencia para la acción a realizar.
16 El administrador de gimnasio confirma si desea o no eliminar el o los utensilios deportivos seleccionados.	3.5 Si el administrador de gimnasio acepta, el sistema elimina los datos correspondientes al utensilio deportivo, o los utensilios deportivos seleccionados en la Base de Datos y termina el CUS.
<b>Curso alterno de los eventos</b>	
<b>Acción 3.1</b>	Si el administrador de gimnasio cancela la acción, se culmina el CUS sin ejecutar ninguna acción.
<b>Prioridad</b>	Crítico

Tabla A3.20 Descripción del caso de uso del sistema “Gestionar utensilio deportivo”

<b>Caso de Uso</b>	<b>Gestionar rol.</b>	
<b>Actores</b>	Administrador general (inicia).	
<b>Propósito</b>	Permite al administrador asignar, modificar y eliminar rol a un usuario y adicionar, modificar y eliminar un rol.	
<b>Resumen</b>	El CUS se inicia cuando el administrador general selecciona la opción Gestionar rol. El sistema muestra las opciones Gestionar rol de usuario y Gestionar rol. El administrador general sigue las instrucciones del sistema e introduce los datos necesarios para la realización de estas acciones. Estos datos se registran en la Base de Datos si son correctos y termina el CUS.	
<b>Referencias</b>	RF 13	
<b>Precondiciones</b>	-	
<b>Poscondiciones</b>	16 Rol adicionado a la Base de Datos. 17 Información de rol, modificada en la Base de Datos. 18 Rol eliminado de la Base de Datos. 19 Rol asignado a un usuario en la Base de Datos. 20 Información de rol de usuario, modificada en la Base de Datos. 21 Rol de usuario eliminado de la Base de Datos.	
<b>Curso normal de los eventos</b>		
<b>Acción del actor</b>	<b>Respuesta del sistema</b>	
6 El administrador general selecciona la opción Gestionar rol.	1.6 El sistema muestra las opciones: Gestionar rol y Gestionar rol de usuario.	
<b>Escenario 1: Gestionar rol</b>		
1. El administrador general selecciona la opción de Gestionar rol.	El sistema muestra las opciones: Adicionar rol, Modificar rol y Eliminar rol.	
<b>Sección 1: Adicionar rol</b>		

12. El administrador general selecciona la opción de Adicionar rol.	1.1 El sistema muestra el formulario para adicionar un nuevo rol.
13. El administrador general introduce los datos solicitados por el sistema.	2.3 El sistema verifica los datos introducidos por el administrador general.
	2.4 Si los datos introducidos son correctos, el sistema adiciona el rol a la Base de Datos y termina el CUS.
<b>Curso alterno de los eventos</b>	
<b>Acción 2.2</b>	Si los datos introducidos por el administrador general son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato erróneo e indica al usuario retornar a la acción 2.
<b>Sección 2: Modificar rol.</b>	
17. El administrador general selecciona la opción de Modificar rol	1.7 El sistema muestra un listado con los roles existentes en la Base de Datos.
18.El administrador general selecciona el rol que será modificado y selecciona la opción Modificar.	2.3 El sistema localiza los datos del rol seleccionado y los muestra, listos para ser modificados.
19. El administrador general realiza los cambios necesarios a los datos y selecciona la opción Actualizar.	3.11 El sistema verifica los datos modificados por el administrador general.
	3.12 Si los datos están correctos el sistema actualiza los datos del rol en la Base de Datos y termina el CUS.
<b>Curso alterno de los eventos</b>	

<b>Acción 3.2</b>	Si los datos introducidos por el administrador general son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato erróneo e indica al usuario retornar a la acción 3.
<b>Sección 3: Eliminar rol.</b>	
17 El administrador general selecciona la opción de Eliminar rol.	1.5 El sistema muestra un listado con los roles existentes en la Base de Datos.
18 El administrador general selecciona el o los roles que serán eliminados y selecciona la opción Eliminar.	2.3 El sistema muestra un mensaje de advertencia para la acción a realizar.
19 El administrador general confirma si desea o no eliminar el o los roles seleccionados.	3.6 Si el administrador general acepta, el sistema elimina los datos correspondientes al rol, o los roles seleccionados en la Base de Datos y termina el CUS.
<b>Curso alterno de los eventos</b>	
<b>Acción 3.1</b>	Si el administrador general cancela la acción, se culmina el CUS sin ejecutar ninguna acción.
<b>Escenario 2: Gestionar rol de usuario</b>	
1. El administrador general selecciona la opción de Gestionar rol.	El sistema muestra las opciones: Asignar rol a usuario, Modificar rol de usuario y Eliminar rol de usuario.
<b>Sección 1: Asignar rol a usuario.</b>	
1 El administrador general selecciona la opción Asignar rol usuario.	1.3 El sistema muestra un formulario para insertar el número de solapín del usuario al que se le desea asignarle el rol.
2 El administrador general introduce los datos solicitados por el sistema.	2.1 El sistema verifica los datos introducidos por el administrador general.
	2.2 Si los datos introducidos son correctos, el sistema muestra un listado con los roles existentes

	en la Base de Datos.
3 El administrador general selecciona el rol que se le va a asignar al usuario y selecciona la opción Asignar.	3.1 El sistema le asigna el rol seleccionado al usuario en la Base de Datos y culmina el CUS.
<b>Curso alternativo de los eventos</b>	
<b>Acción 2.2</b>	Si los datos introducidos por el administrador general son incorrectos, el sistema muestra un mensaje de error indicando que el usuario no existe en la Base de Datos e indica al usuario retornar a la acción 2.
<b>Sección 1: Modificar rol de usuario.</b>	
1 El administrador general selecciona la opción Modificar rol de usuario.	1.1 El sistema muestra un formulario para insertar el número de solapín del usuario al que se le desea modificar su rol.
2 El administrador general introduce los datos solicitados por el sistema.	2.1 El sistema verifica los datos introducidos por el administrador general.
	2.2 Si los datos introducidos son correctos, el sistema muestra un listado con los roles de usuario existentes en la Base de Datos.
3. El administrador general selecciona el rol a modificar y selecciona la opción Modificar.	3.1 El sistema muestra el rol seleccionado listo para ser modificado.
4. El administrador general realiza los cambios necesarios a los datos y selecciona la opción Actualizar.	4.1 El sistema verifica los datos modificados por el administrador general.
	4.2 Si los datos están correctos el sistema actualiza el rol del usuario en la Base de Datos y termina el CUS.

<b>Curso alterno de los eventos</b>	
<b>Acción 2.2</b>	Si los datos introducidos por el administrador general son incorrectos, el sistema muestra un mensaje de error indicando que el usuario no existe en la Base de Datos e indica al usuario retornar a la acción 2.
<b>Acción 4.2</b>	Si los datos introducidos por el administrador general son incorrectos, el sistema muestra un mensaje de error indicando donde está el dato erróneo e indica al usuario retornar a la acción 4.
<b>Sección 3: Eliminar rol de usuario</b>	
1 El administrador general selecciona la opción Eliminar rol de usuario.	1.1 El sistema muestra un formulario para insertar el número de solapín del usuario al que se le desea eliminar su rol.
2 El administrador general introduce los datos solicitados por el sistema.	2.1 El sistema verifica los datos introducidos por el administrador general.
	2.2 Si los datos introducidos son correctos, el sistema muestra un listado con los roles del usuario, existentes en la Base de Datos.
3 El administrador general selecciona el o los roles del usuario que serán eliminados y selecciona la opción Eliminar.	3.1 El sistema muestra un mensaje de advertencia para la acción a realizar.
4 El administrador general confirma si desea o no eliminar el o los roles seleccionados.	4.1 Si el administrador general acepta, el sistema elimina el o los roles del usuario seleccionados de la Base de Datos y termina el CUS.
<b>Curso alterno de los eventos</b>	
<b>Acción 2.2</b>	Si los datos introducidos por el administrador general son incorrectos, el sistema muestra un mensaje de error indicando que el usuario no existe

---

	en la Base de Datos e indica al usuario retornar a la acción 2.
<b>Acción 4.1</b>	Si el administrador general cancela la acción, se culmina el CUS sin ejecutar ninguna acción.
<b>Prioridad</b>	Crítico

**Tabla A3.21 Descripción del caso de uso del sistema “Gestionar rol”**

Anexo # 4: Diagramas de clases del análisis.

Diagrama de clases del análisis del CUS “Generar tabla posiciones competencia”

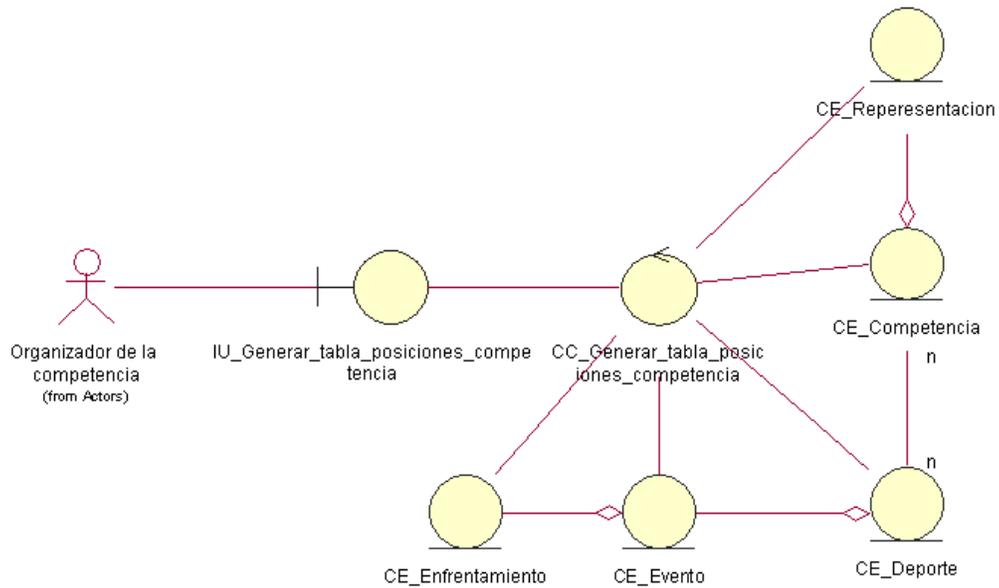


Fig. A4.1 Diagrama de clases del análisis del CUS “Generar tabla posiciones competencia”

Diagrama de clases de análisis del caso de uso “Gestiona prueba eficiencia física”.

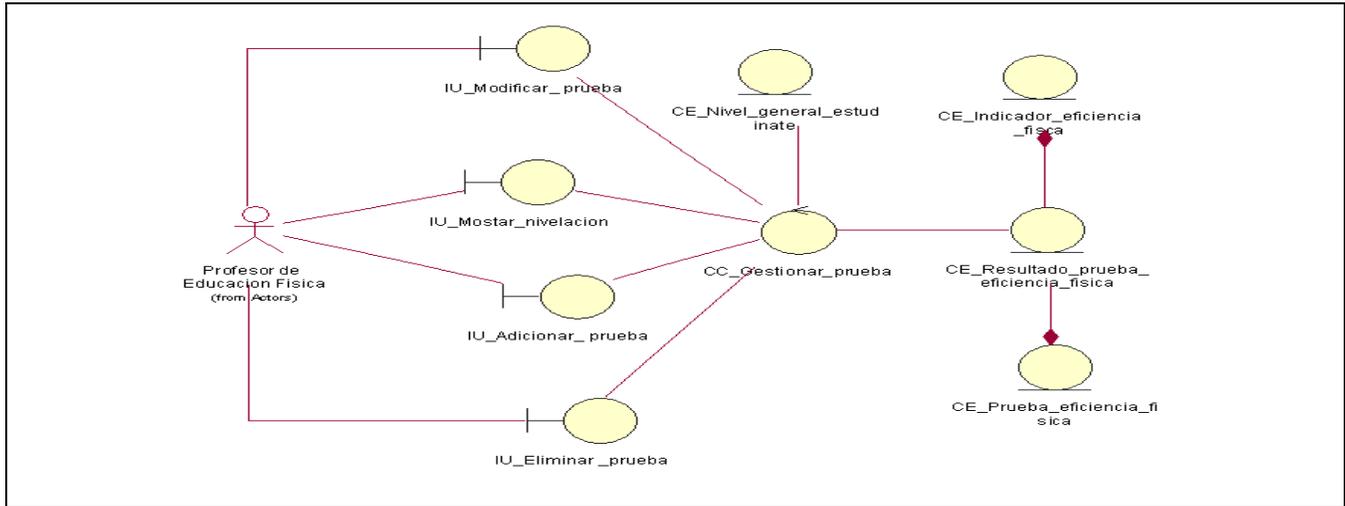


Fig. A4.2 Diagrama de clases de análisis del caso de uso “Gestiona prueba eficiencia física”.

**Diagrama de clases de análisis del caso de uso “Gestiona tipo enfrentamiento”.**

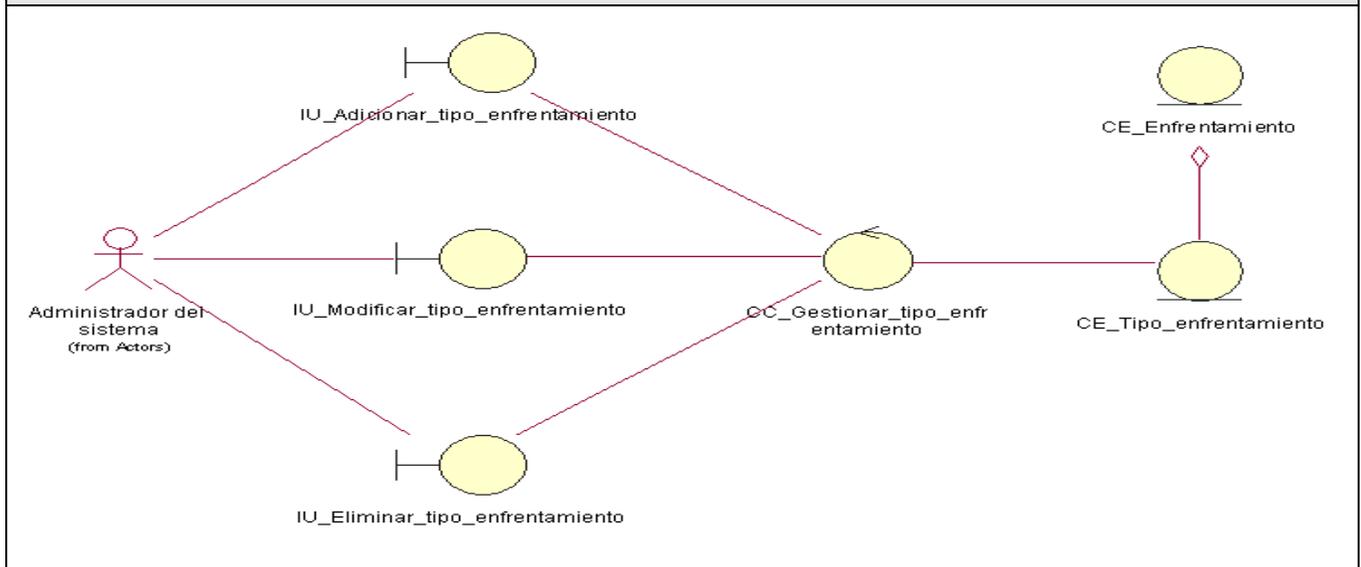


Fig. A4.3 Diagrama de clases de análisis del caso de uso “Gestiona tipo enfrentamiento”.

**Diagrama de clases de análisis del caso de uso “Gestionar atleta”.**

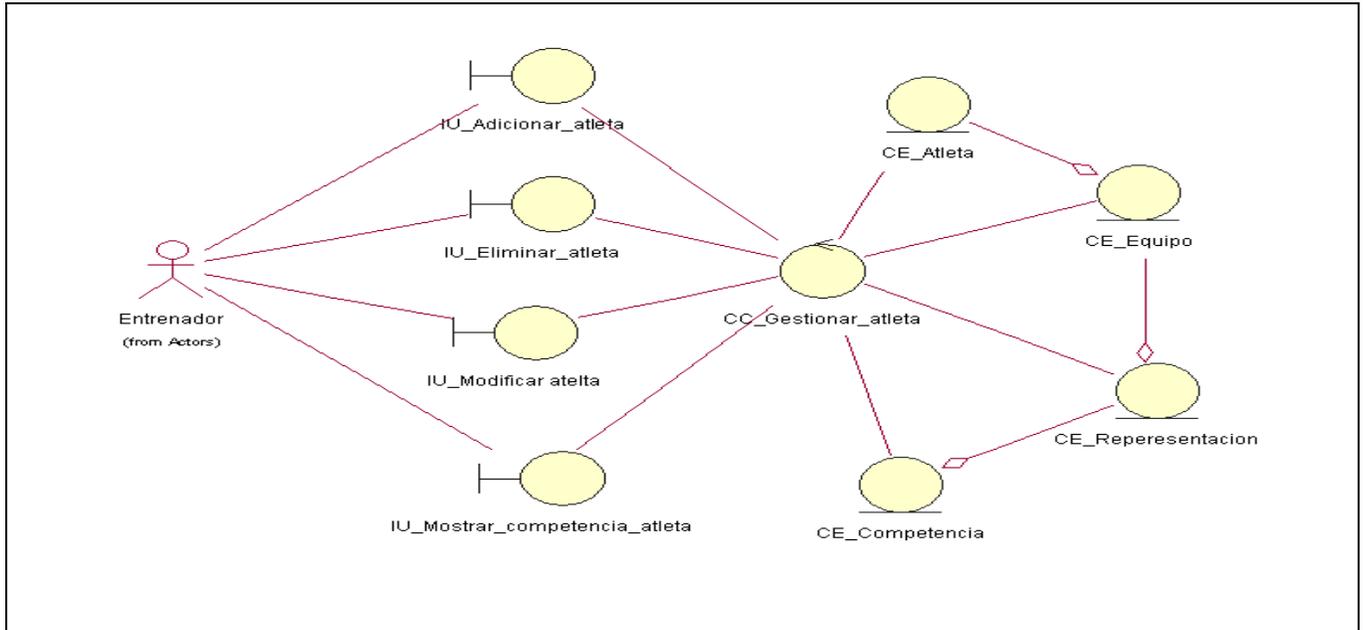


Fig. A4.4 Diagrama de clases de análisis del caso de uso “Gestionar atleta”.

**Diagrama de clases de análisis del caso de uso “Gestionar categoría competencia”.**

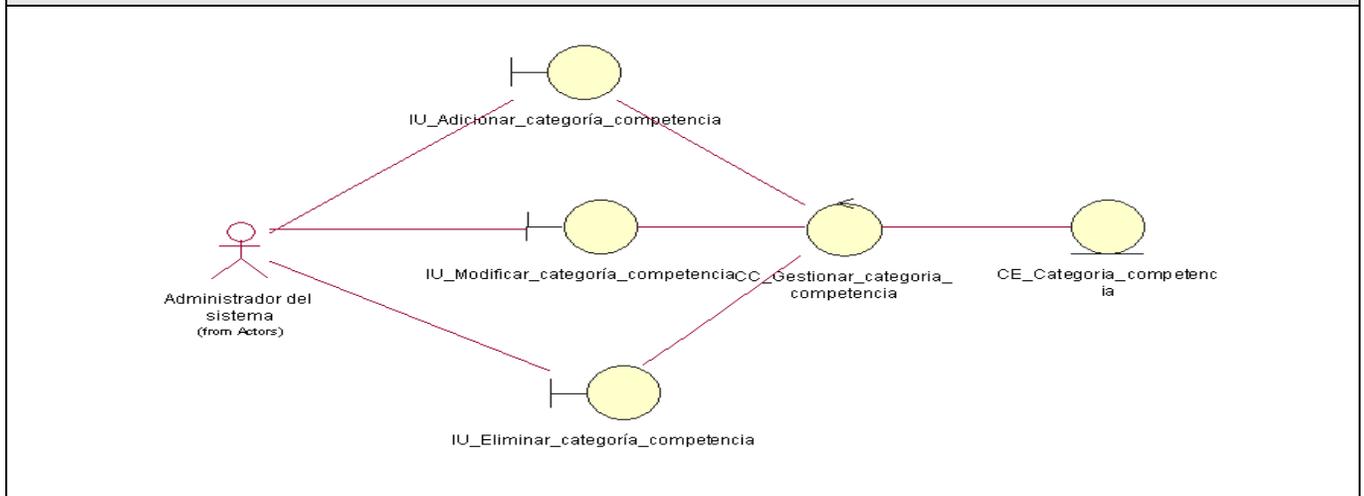


Fig. A4.5 Diagrama de clases de análisis del caso de uso “Gestionar categoría competencia”.

**Diagrama de clases de análisis del caso de uso “Gestionar color piel”.**



**Diagrama de clases de análisis del caso de uso "Gestionar deporte".**

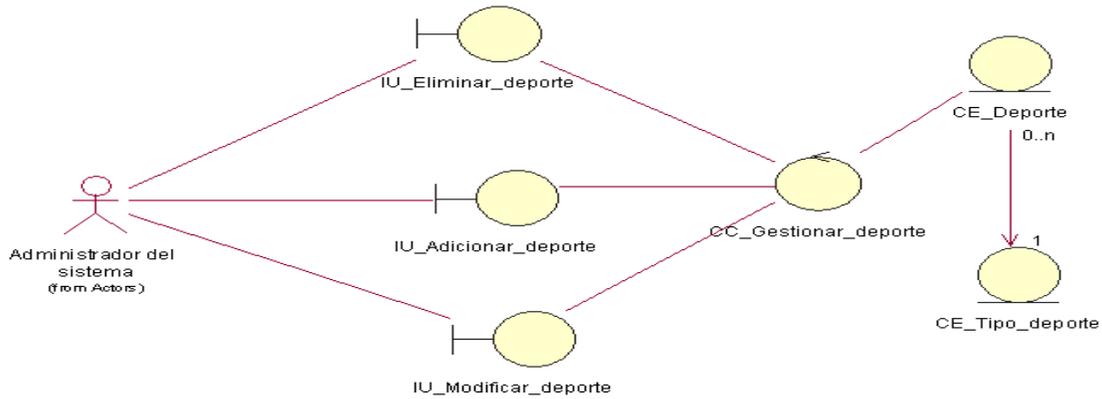


Fig. A4.8 Diagrama de clases de análisis del caso de uso "Gestionar deporte".

**Diagrama de clases de análisis del caso de uso "Gestionar enfrentamiento competencia".**

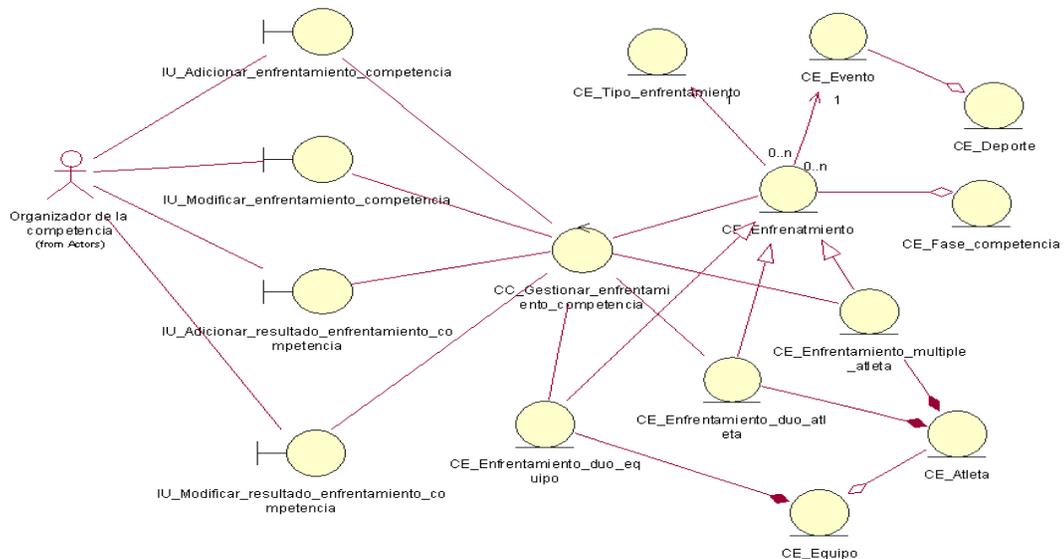
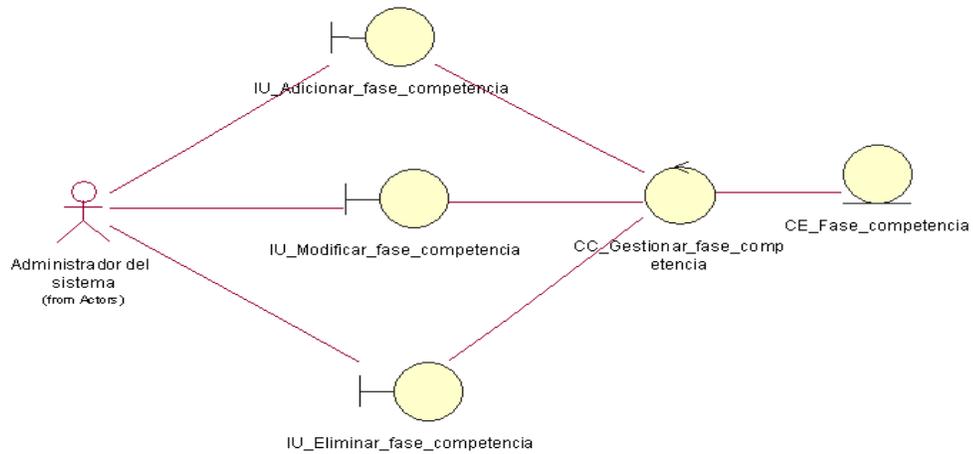


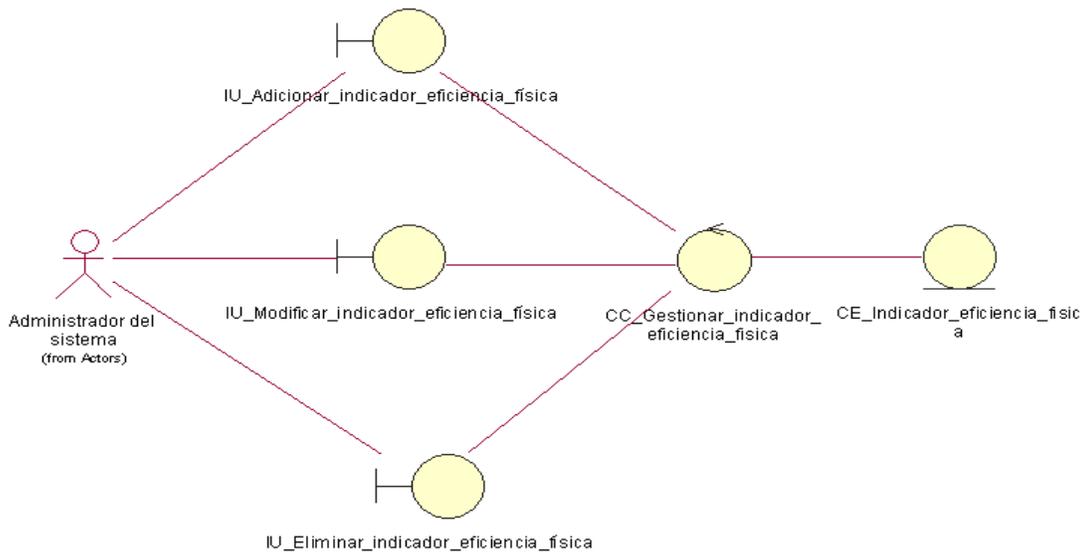
Fig. A4.9 Diagrama de clases de análisis del caso de uso "Gestionar enfrentamiento competencia".

**Diagrama de clases de análisis del caso de uso “Gestionar fase competencia”.**



**Fig. A4.10 Diagrama de clases de análisis del caso de uso “Gestionar fase competencia”.**

**Diagrama de clases de análisis del caso de uso “Gestionar indicador eficiencia física”.**



**Fig. A4.11 Diagrama de clases de análisis del caso de uso “Gestionar indicador eficiencia física”.**

**Diagrama de clases de análisis del caso de uso “Gestionar lesión atleta”.**

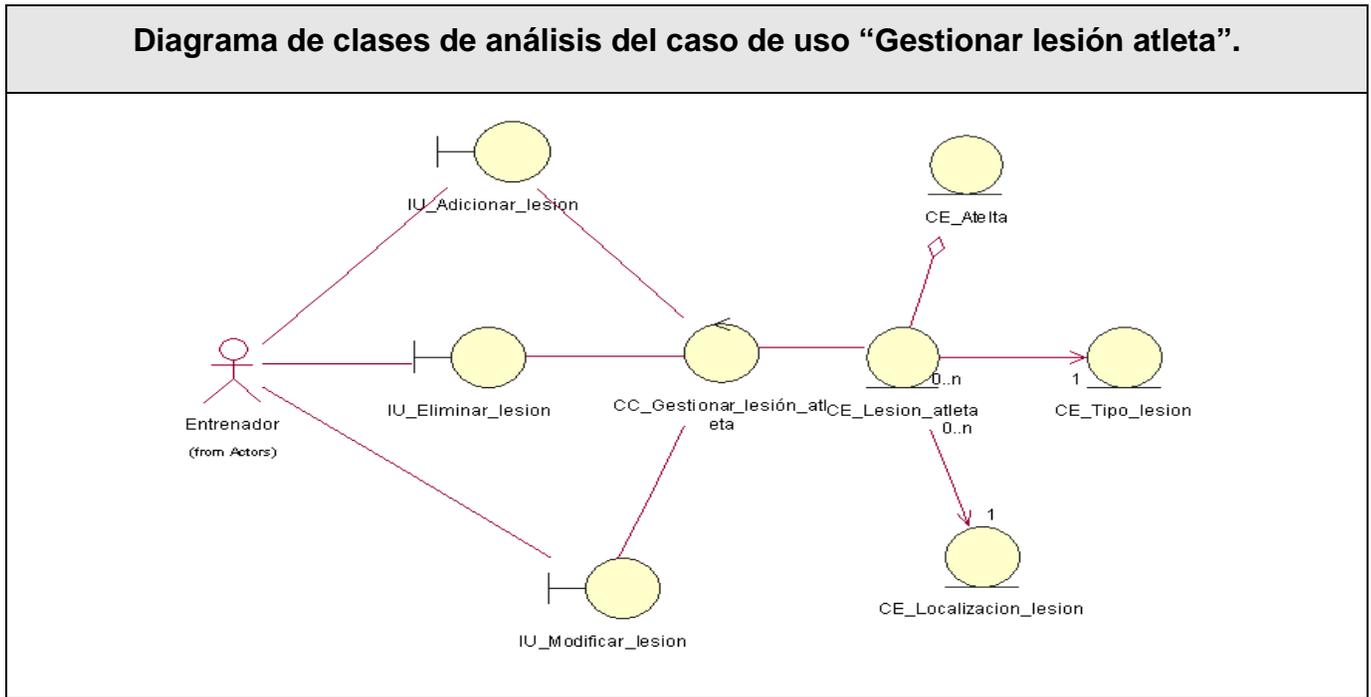


Fig. A4.12 Diagrama de clases de análisis del caso de uso “Gestionar lesión atleta”.

**Diagrama de clases de análisis del caso de uso “Gestionar localización lesión”.**

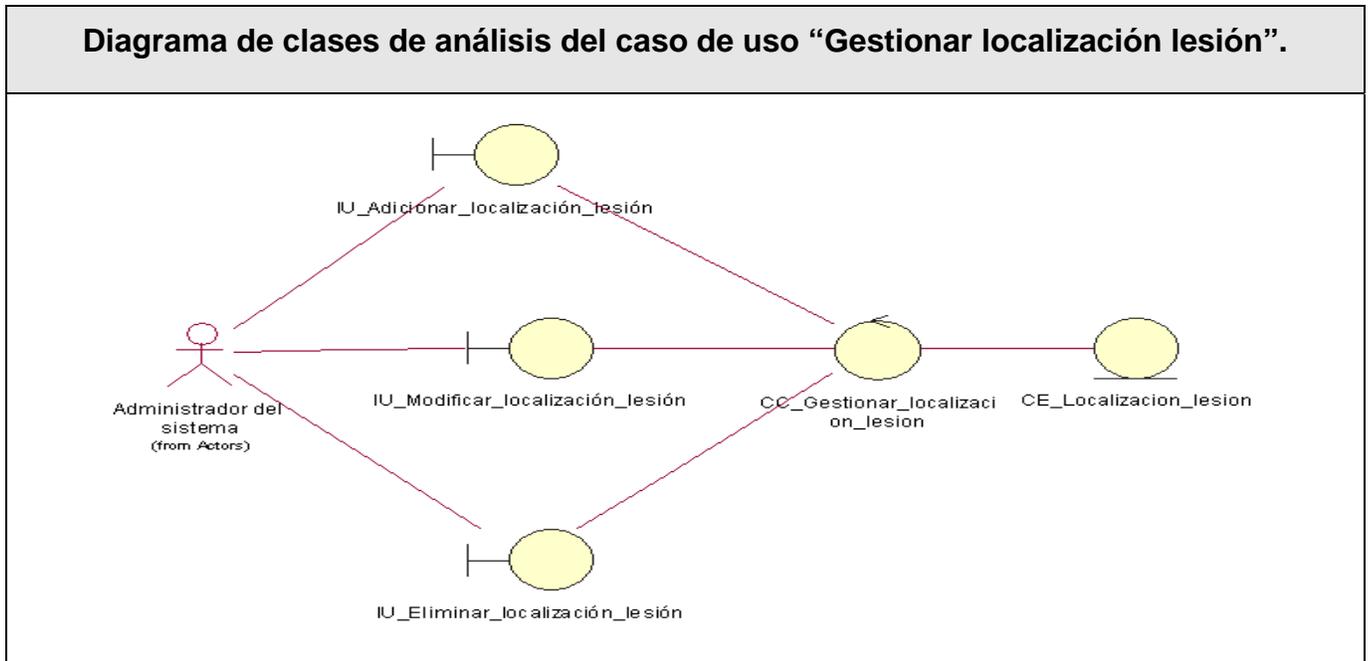


Fig. A4.13 Diagrama de clases de análisis del caso de uso “Gestionar localización lesión”.

**Diagrama de clases de análisis del caso de uso “Gestionar participantes competencia”.**

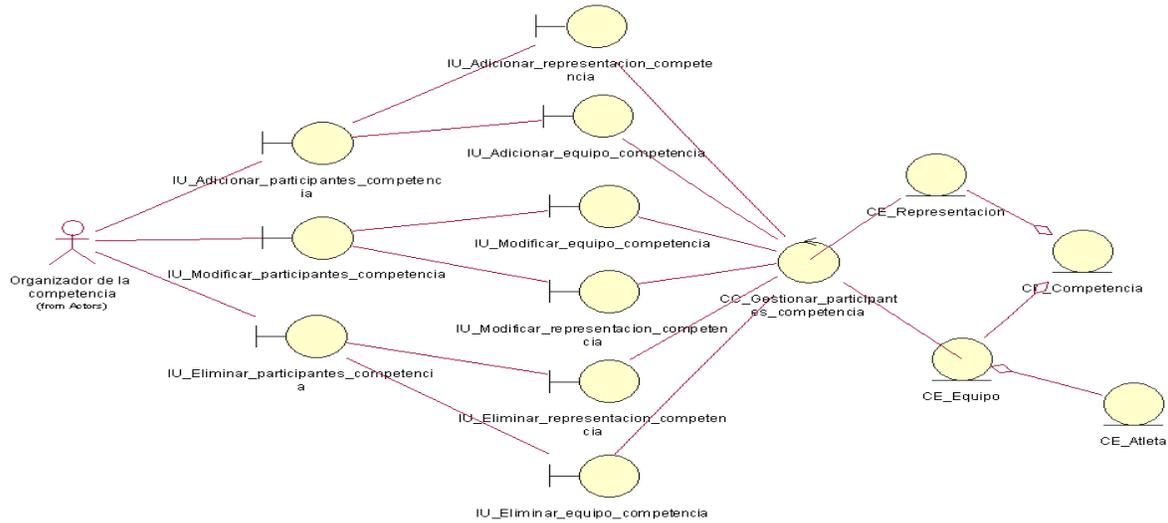


Fig. A4.14 Diagrama de clases de análisis del caso de uso “Gestionar participantes competencia”.

**Diagrama de clases de análisis del caso de uso “Gestionar préstamo utensilio”.**

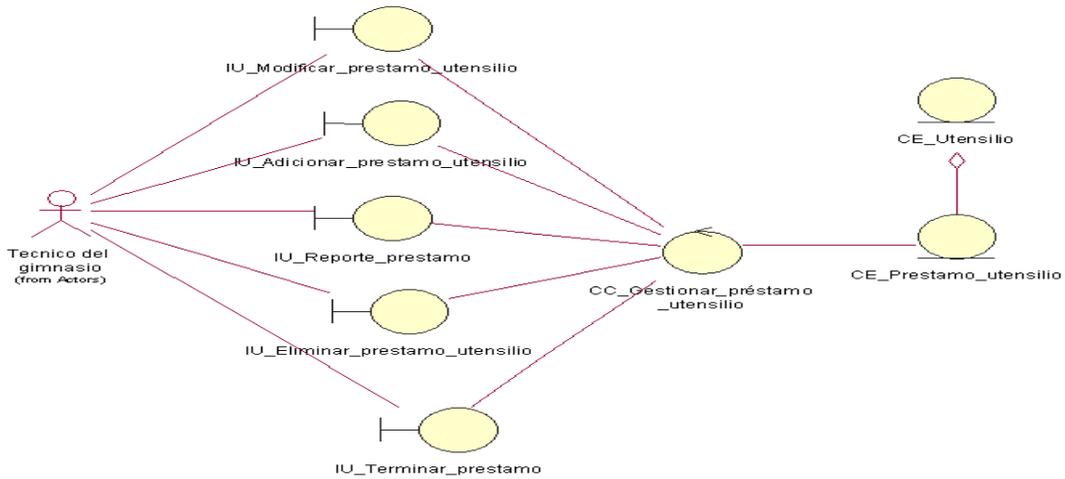


Fig. A4.15 Diagrama de clases de análisis del caso de uso “Gestionar préstamo utensilio”.

**Diagrama de clases de análisis del caso de uso “Gestionar rol”.**

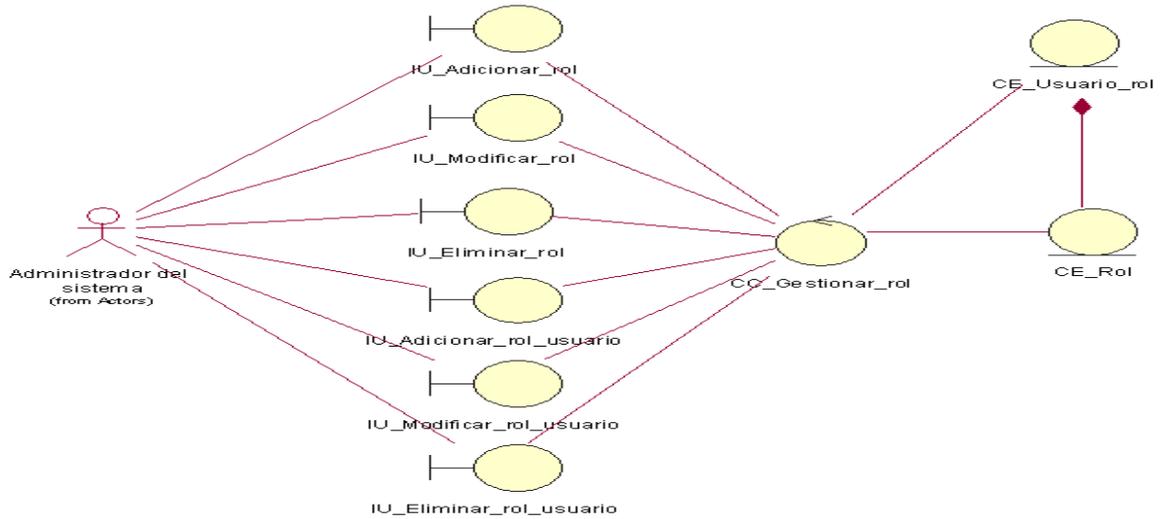


Fig. A4.16 Diagrama de clases de análisis del caso de uso “Gestionar rol”.

**Diagrama de clases de análisis del caso de uso “Gestionar sistema competencia”.**

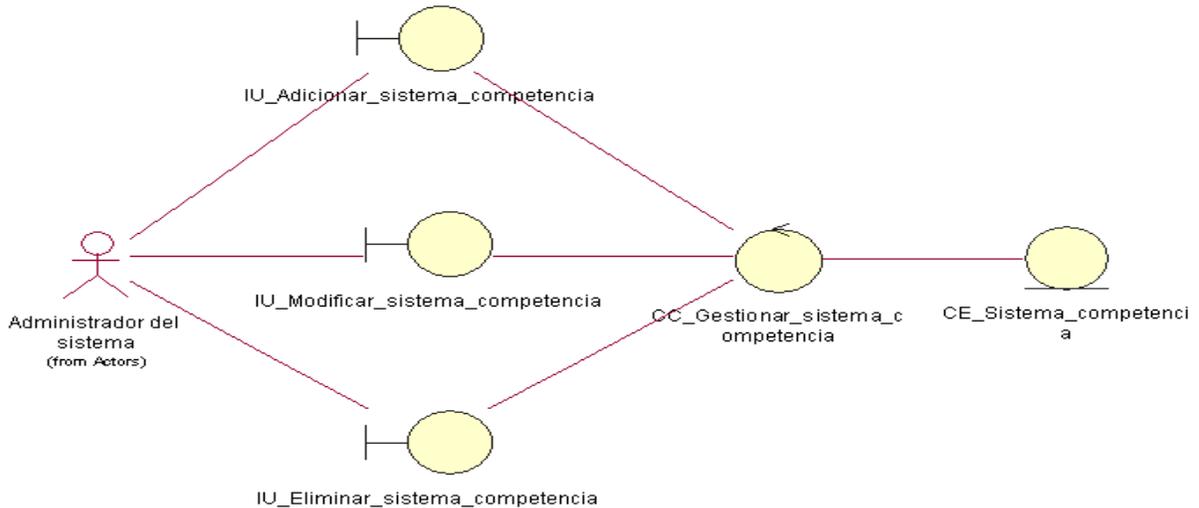


Fig. A4.17 Diagrama de clases de análisis del caso de uso “Gestionar sistema competencia”.

Diagrama de clases de análisis del caso de uso "Gestionar tipo deporte".

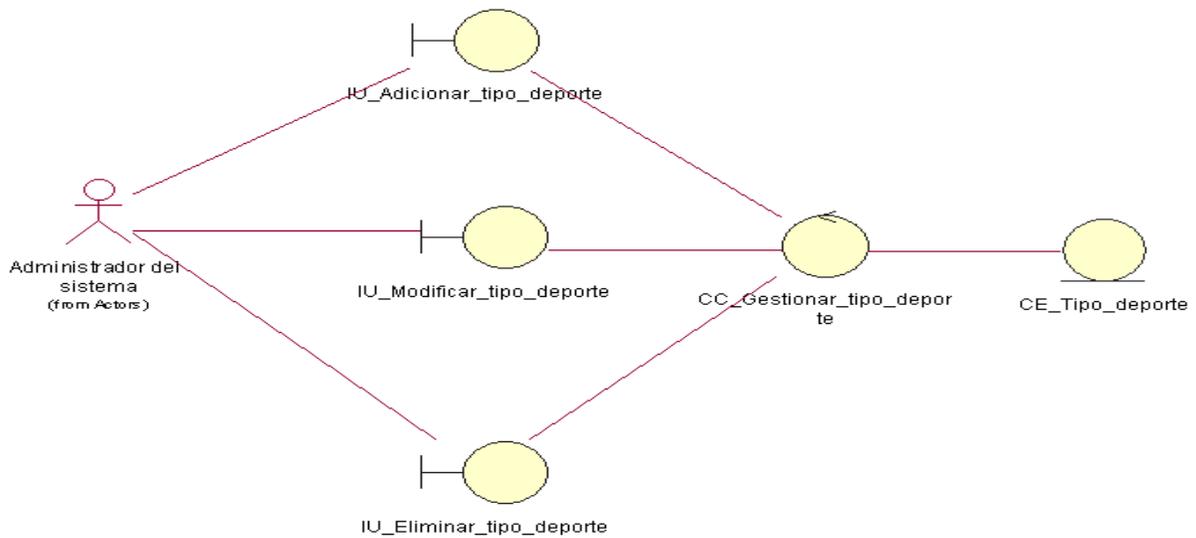


Fig. A4.18 Diagrama de clases de análisis del caso de uso "Gestionar tipo deporte".

Diagrama de clases de análisis del caso de uso "Gestionar tipo lesión".

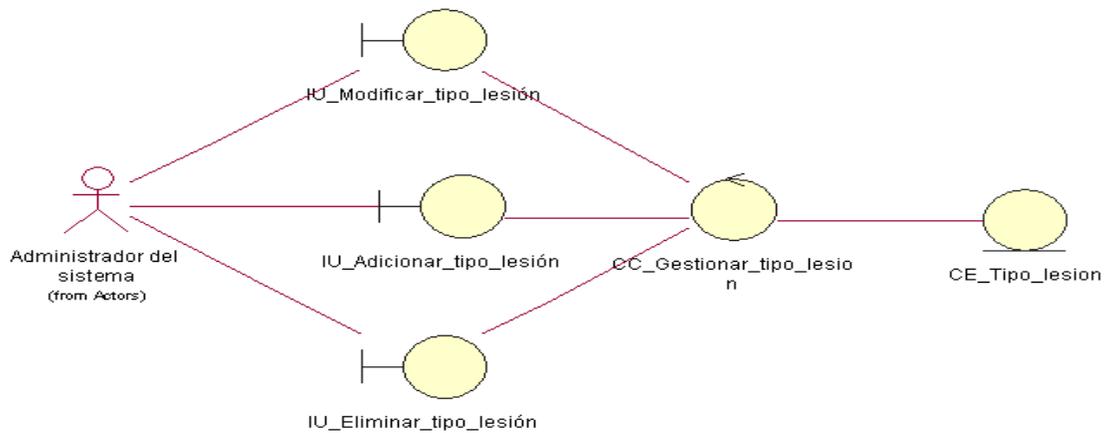


Fig. A4.19 Diagrama de clases de análisis del caso de uso "Gestionar tipo lesión".

**Diagrama de clases de análisis del caso de uso “Gestionar unidad resultado”.**

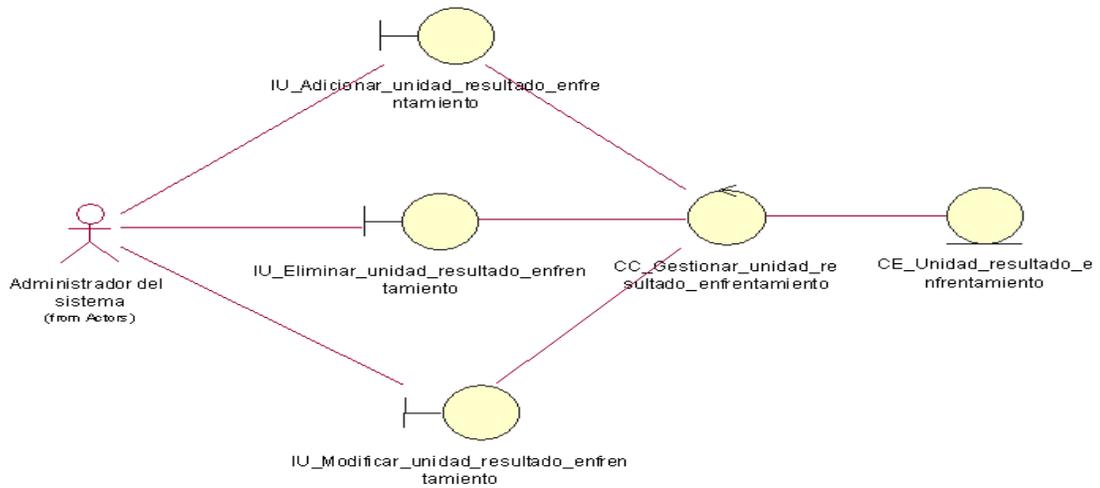


Fig. A4.20 Diagrama de clases de análisis del caso de uso “Gestionar unidad resultado”.

**Diagrama de clases de análisis del caso de uso “Gestionar utensilio”.**

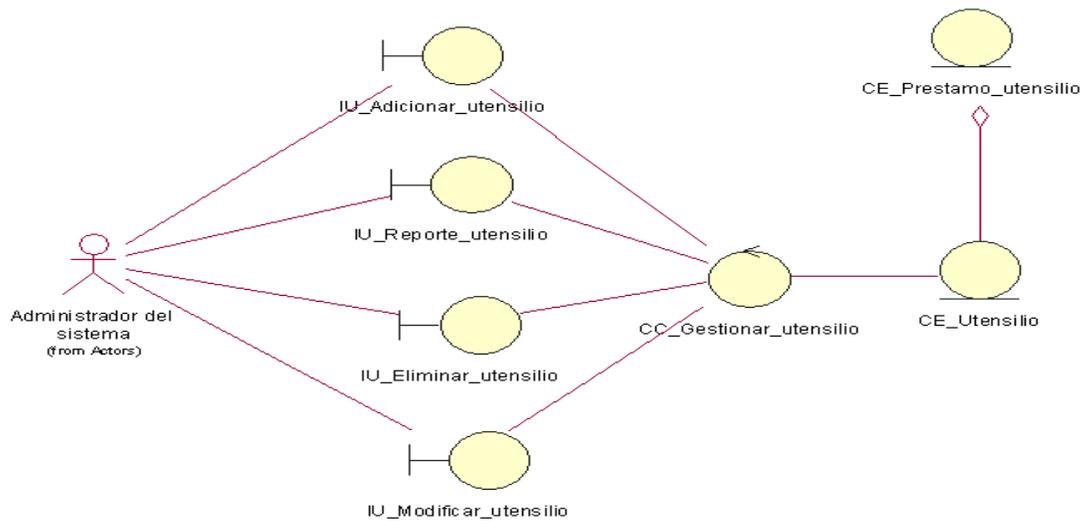


Fig. A4.21 Diagrama de clases de análisis del caso de uso “Gestionar utensilio”.

Diagrama de clases de análisis del caso de uso "Gestionar visita gimnasio".

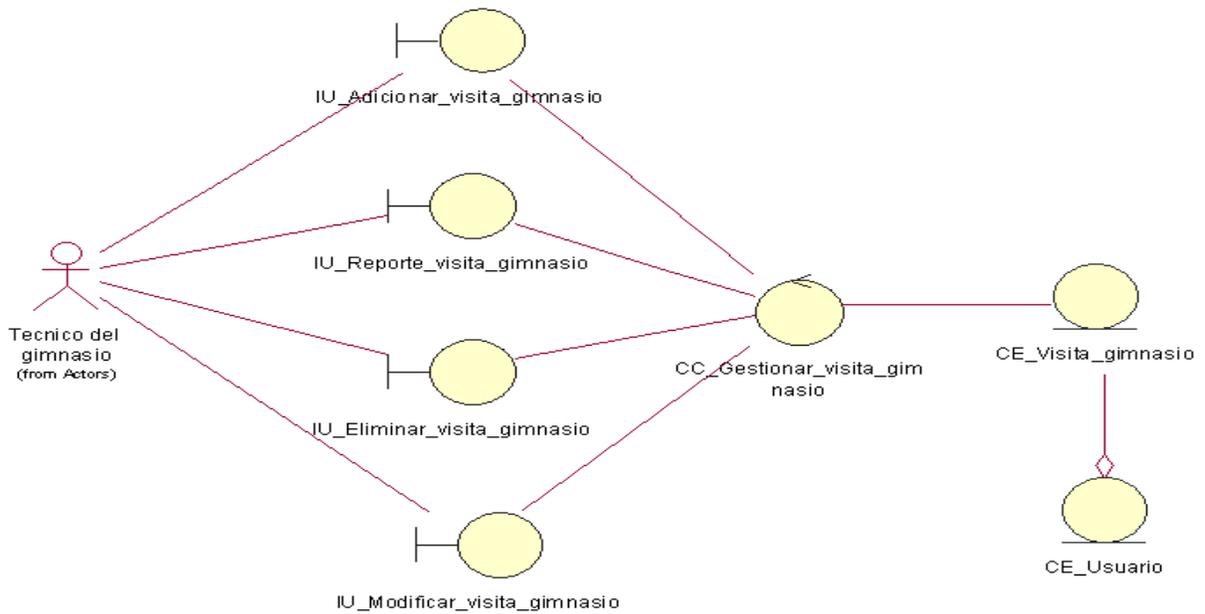


Fig. A4.22 Diagrama de clases de análisis del caso de uso "Gestionar visita gimnasio".

Anexo # 5: Diagramas de clases Web del diseño.

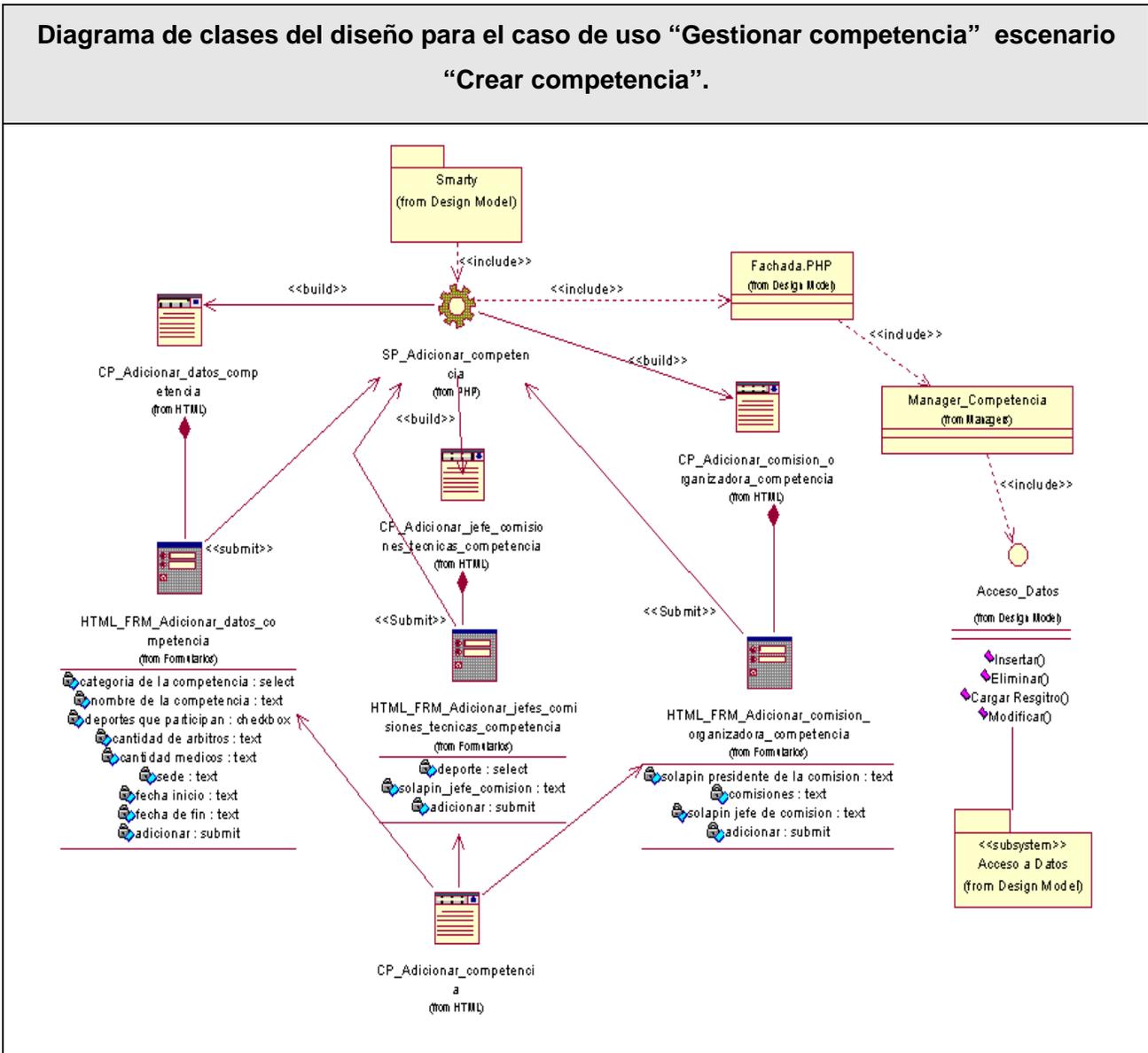


Fig. A5.1 Diagrama de clases del diseño para el caso de uso “Gestionar competencia” escenario “Crear competencia”.

a”.

Diagrama de clases del diseño para el caso de uso “Gestionar competencia” escenario “Modificar competencia”.

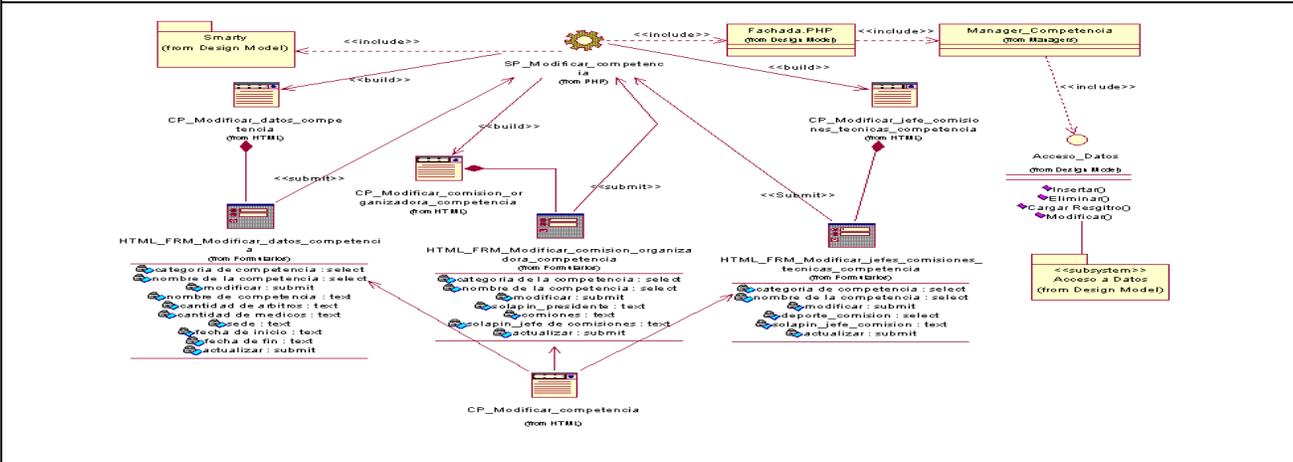


Fig. A5.2 Diagrama de clases del diseño para el caso de uso de

uso “Gestionar competencia” escenario “Modificar competencia”.

Diagrama de clases del diseño para el caso de uso “Gestionar enfrentamiento competencia” escenario “Gestionar enfrentamiento”.

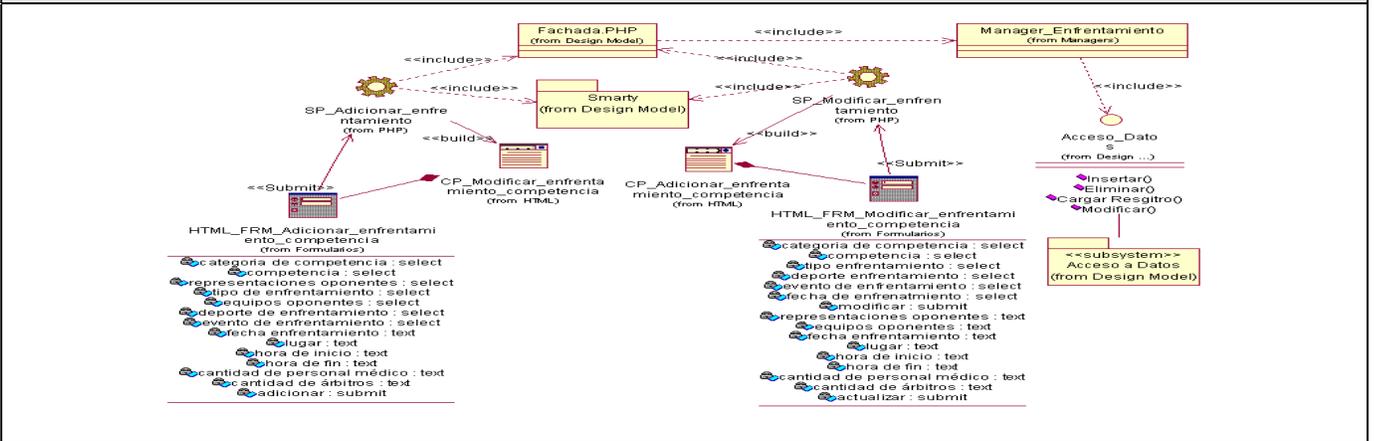


Fig. A5.3 Diagrama de clases del diseño para el caso de uso “Gestionar enfrentamiento competencia” escenario “Gestionar enfrentamiento”.

Diagrama de clases del diseño para el caso de uso “Gestionar enfrentamiento competencia” escenario “Gestionar resultado enfrentamiento”.

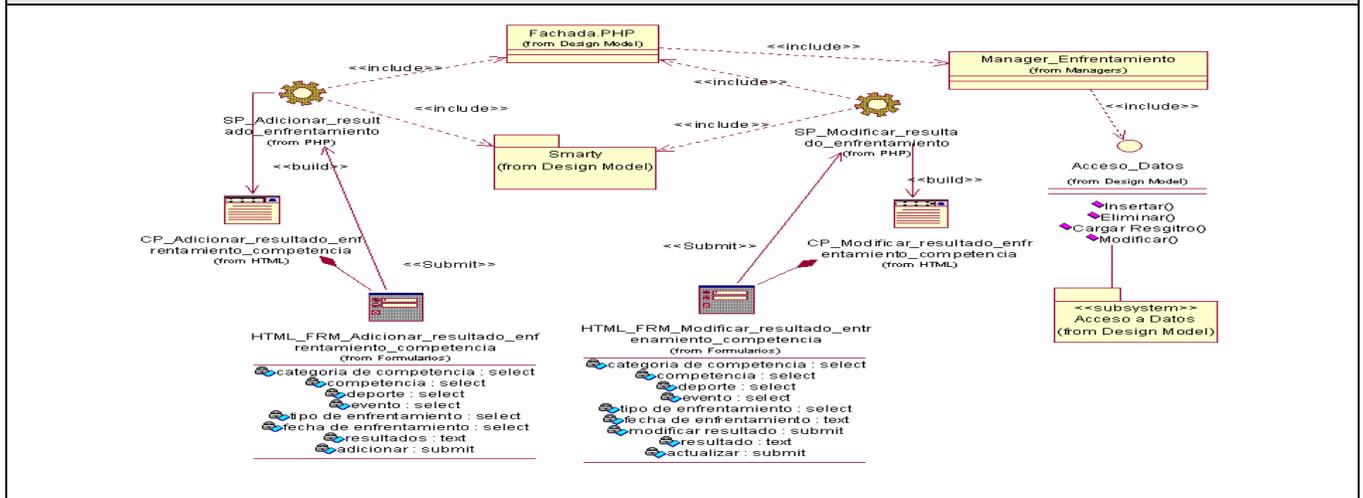


Fig. A5.4 Diagrama de clases del diseño para el caso de uso “Gestionar enfrentamiento competencia” escenario “Gestionar resultado enfrentamiento”.

Diagrama de clases del diseño para el caso de uso “Generar tabla posiciones competencia”.

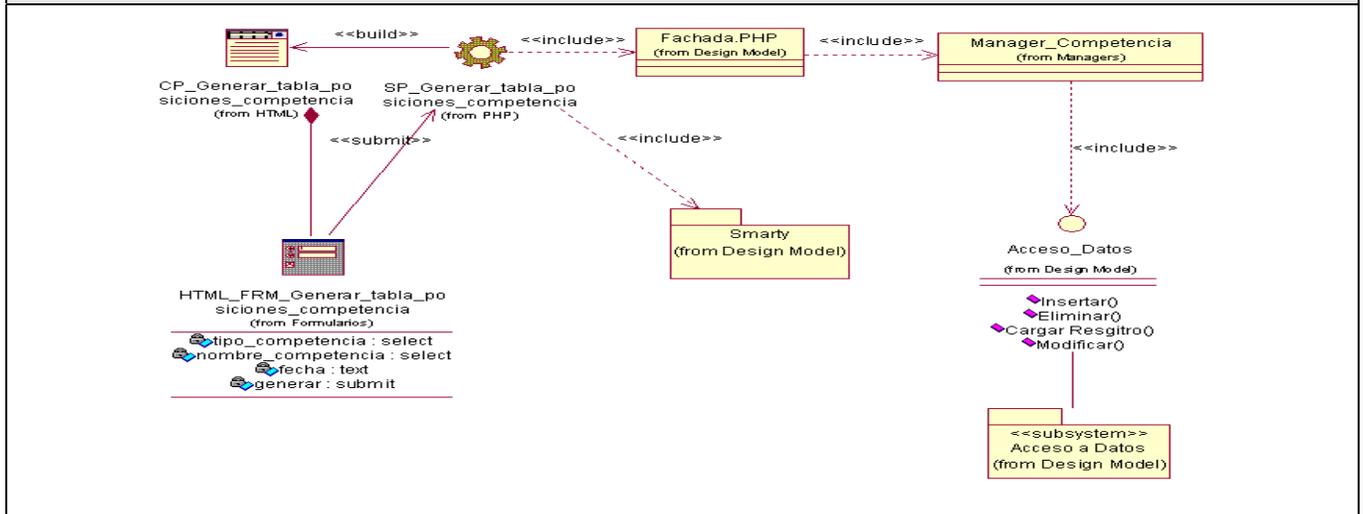


Fig. A5.5 Diagrama de clases del diseño para el caso de uso “Generar tabla posiciones competencia”.



Diagrama de clases del diseño para el caso de uso "Gestionar color piel".

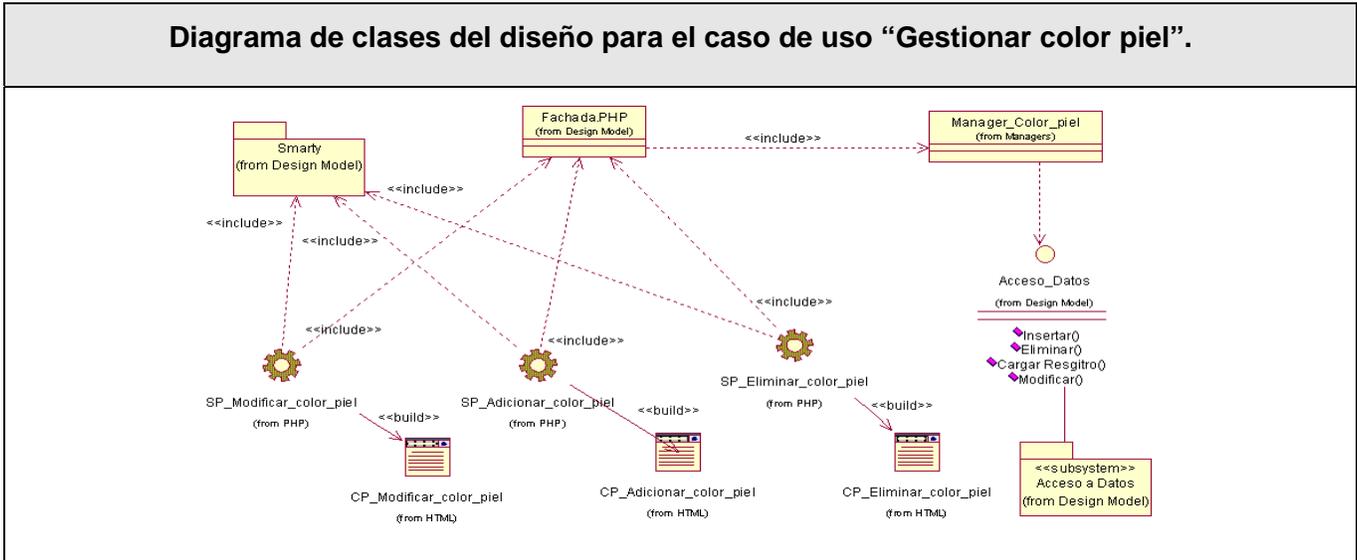


Fig. A5.8 Diagrama de clases del diseño para el caso de uso "Gestionar color piel".

Diagrama de clases del diseño para el caso de uso "Gestionar deporte".

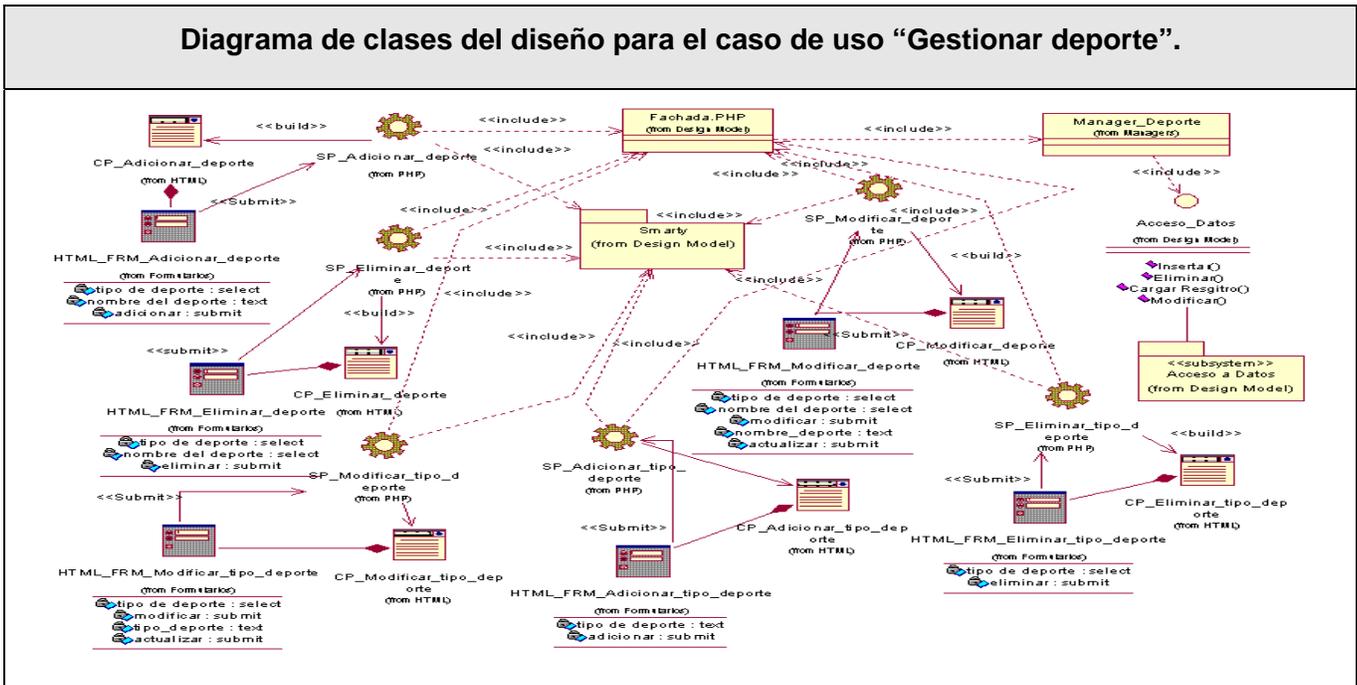


Fig. A5.9 Diagrama de clases del diseño para el caso de uso "Gestionar deporte".

Diagrama de clases del diseño para el caso de uso "Gestionar fase competencia".

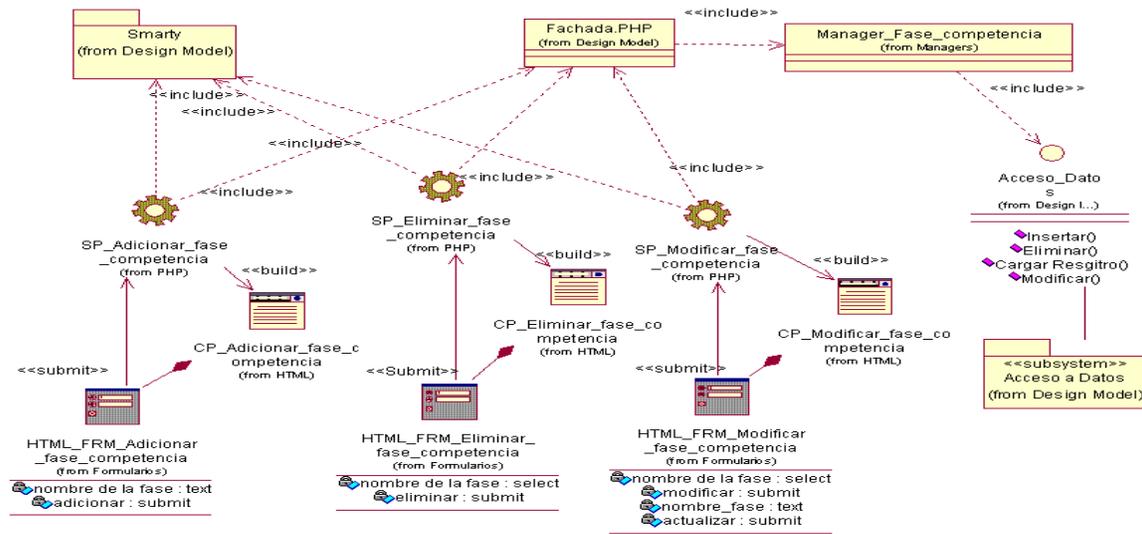


Fig. A5.10 Diagrama de clases del diseño para el caso de uso "Gestionar fase competencia".

Diagrama de clases del diseño para el caso de uso "Gestionar lesión atleta".

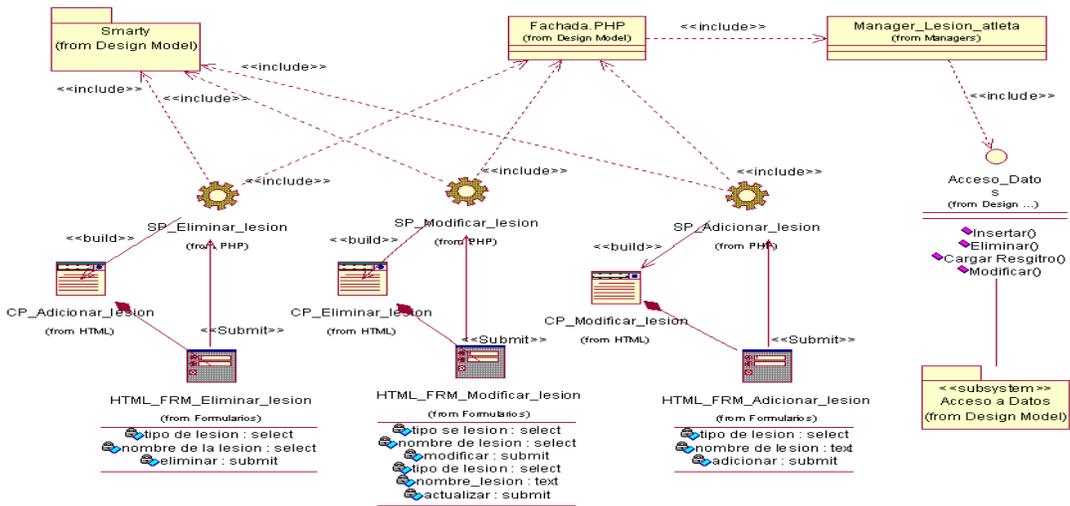


Fig. A5.11 Diagrama de clases del diseño para el caso de uso "Gestionar lesión atleta".

Diagrama de clases del diseño para el caso de uso “Gestionar localización lesión”.

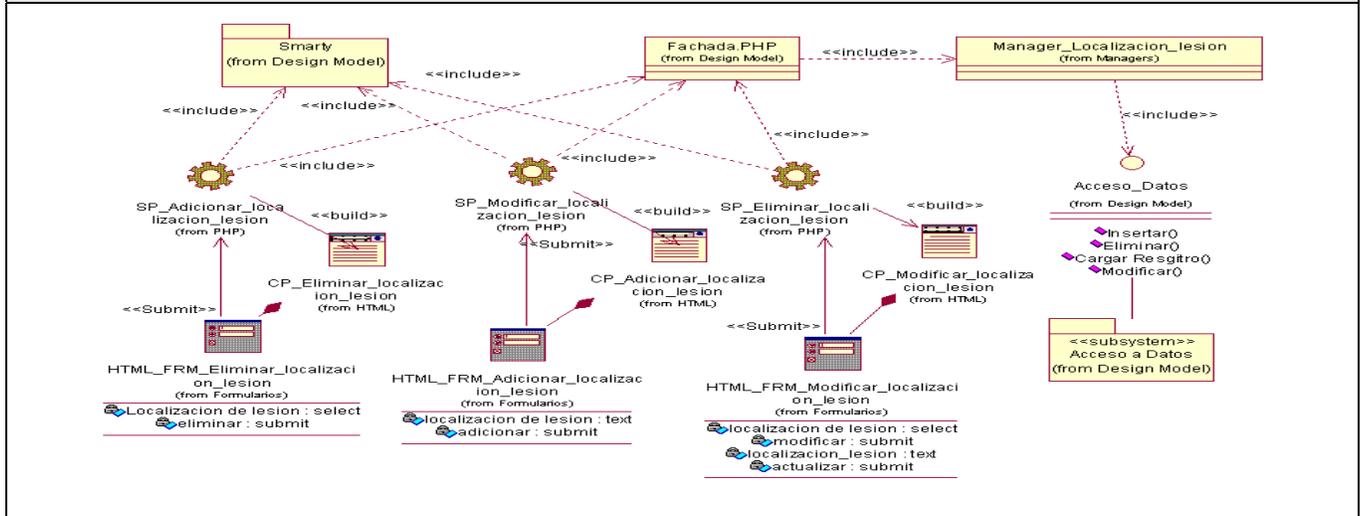


Fig. A5.12 Diagrama de clases del diseño para el caso de uso “Gestionar localización lesión”.

Diagrama de clases del diseño para el caso de uso “Gestionar participantes competencia”.

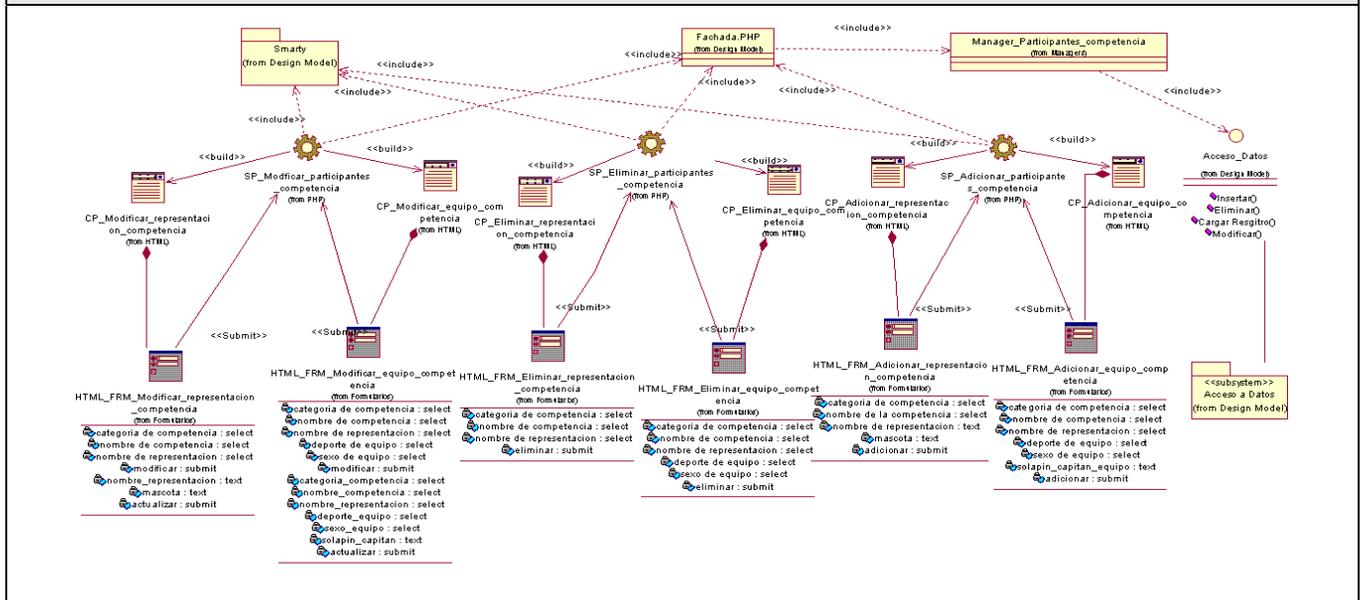


Fig. A5.13 Diagrama de clases del diseño para el caso de uso “Gestionar participantes competencia”.

Diagrama de clases del diseño para el caso de uso "Gestionar préstamo utensilio".

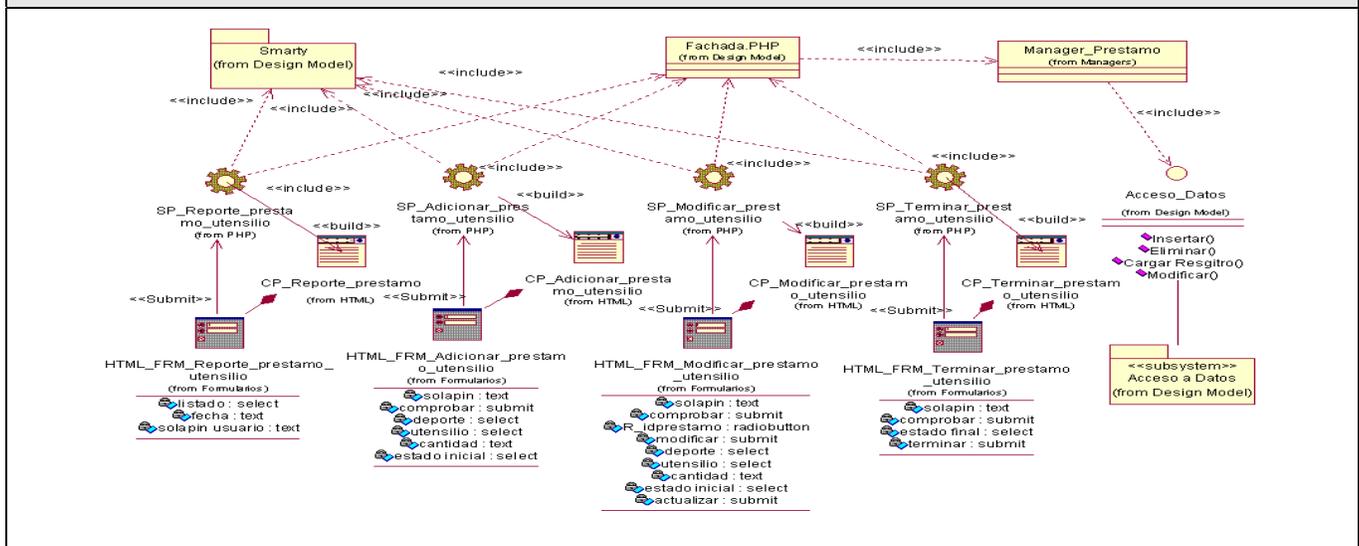


Fig. A5.14 Diagrama de clases del diseño para el caso de uso "Gestionar préstamo utensilio".

Diagrama de clases del diseño para el caso de uso "Gestionar prueba eficiencia".

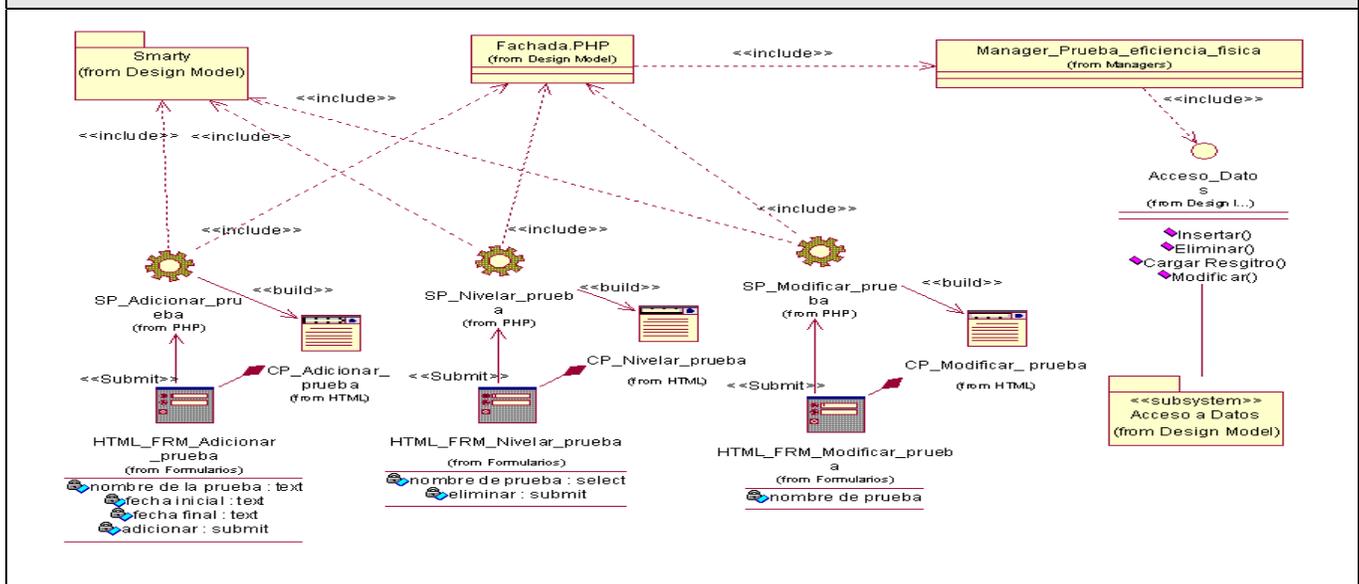


Fig. A5.15 Diagrama de clases del diseño para el caso de uso "Gestionar prueba eficiencia".

Diagrama de clases del diseño para el caso de uso "Gestionar rol".

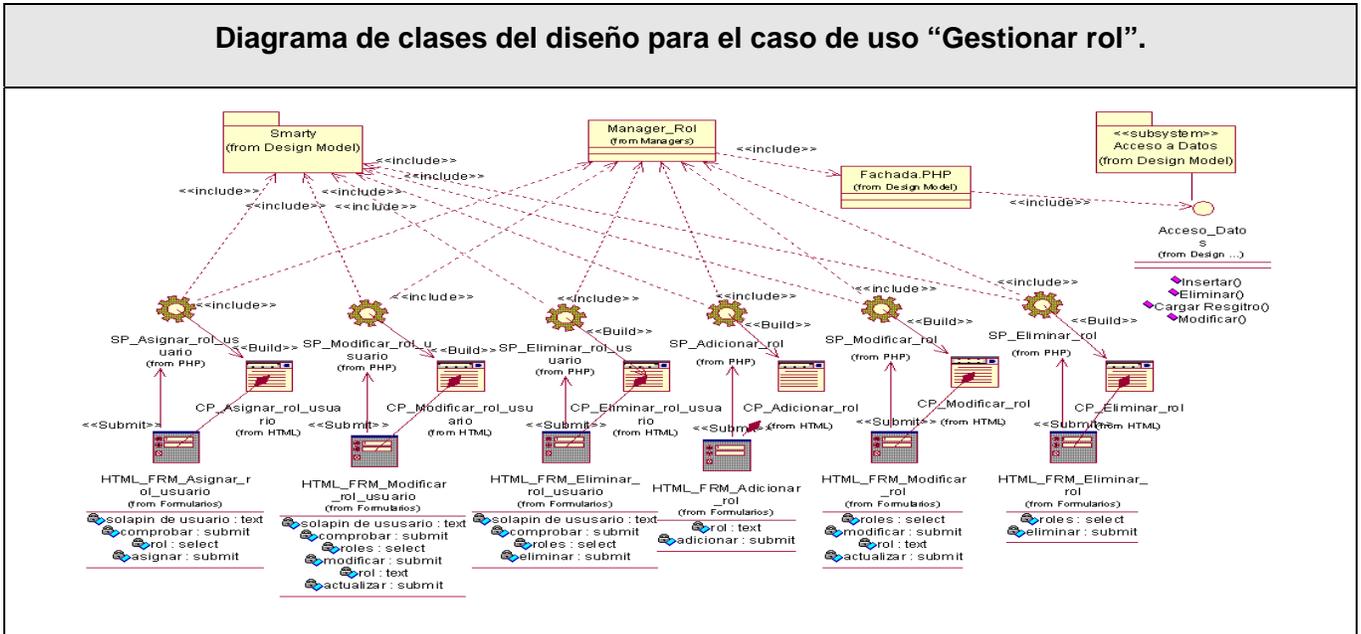


Fig. A5.16 Diagrama de clases del diseño para el caso de uso "Gestionar rol".

Diagrama de clases del diseño para el caso de uso "Gestionar sistema competencia".

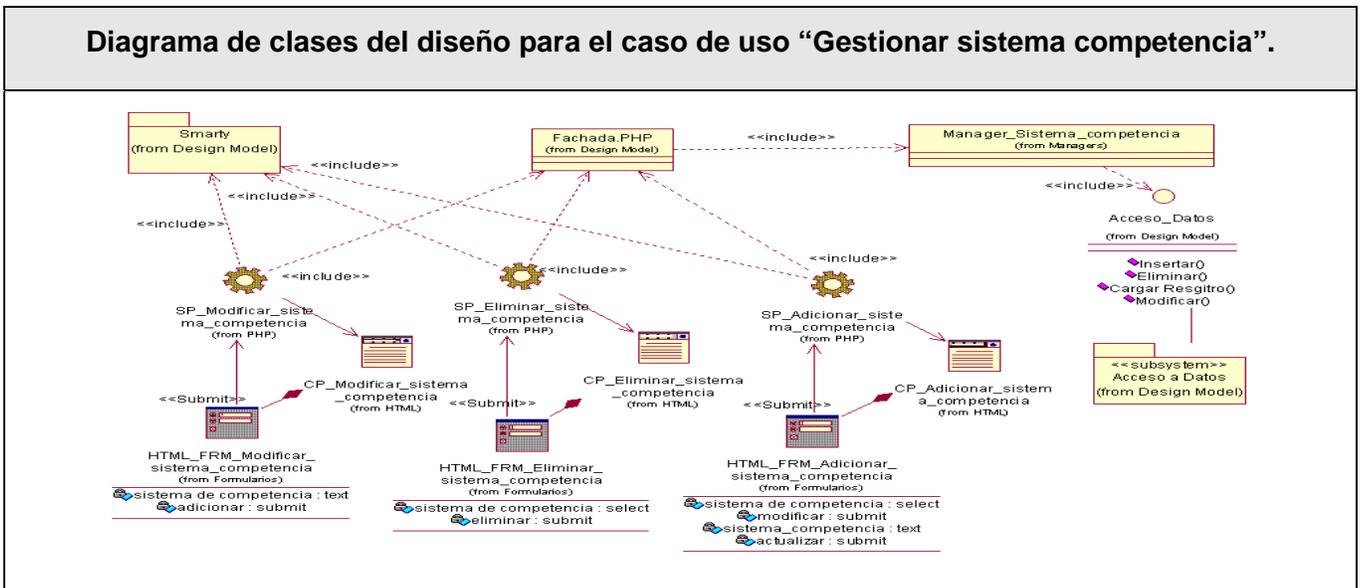


Fig. A5.17 Diagrama de clases del diseño para el caso de uso "Gestionar sistema competencia".

Diagrama de clases del diseño para el caso de uso "Gestionar tipo enfrentamiento".

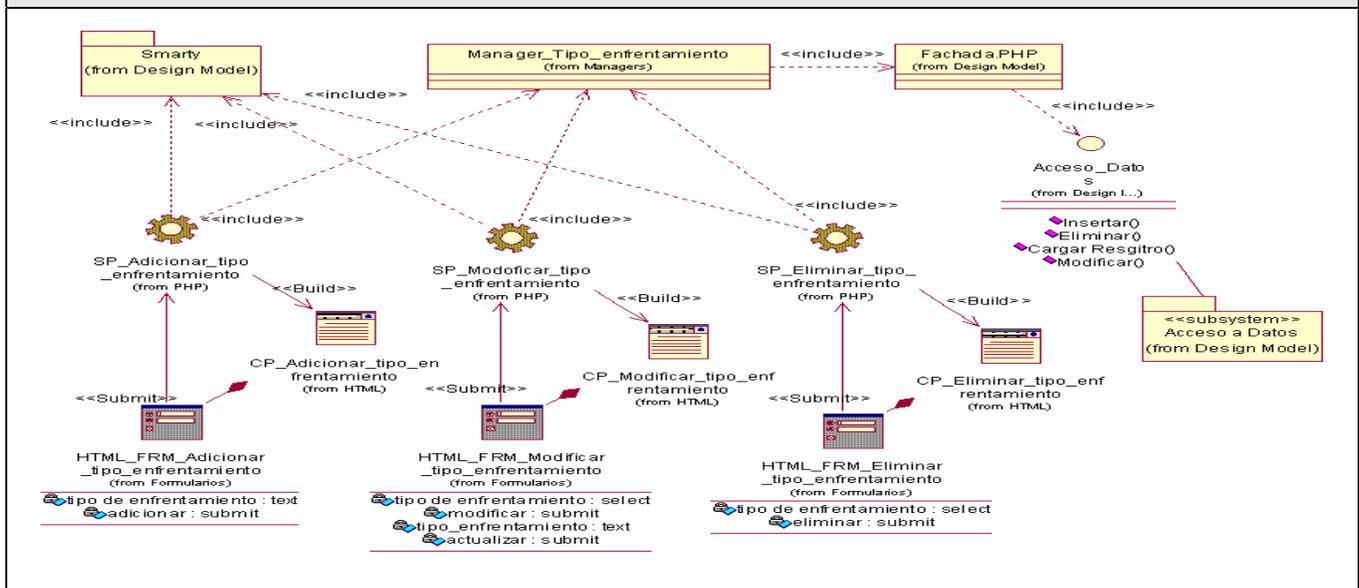


Fig. A5.18 Diagrama de clases del diseño para el caso de uso "Gestionar tipo enfrentamiento".

Diagrama de clases del diseño para el caso de uso "Gestionar tipo lesión".

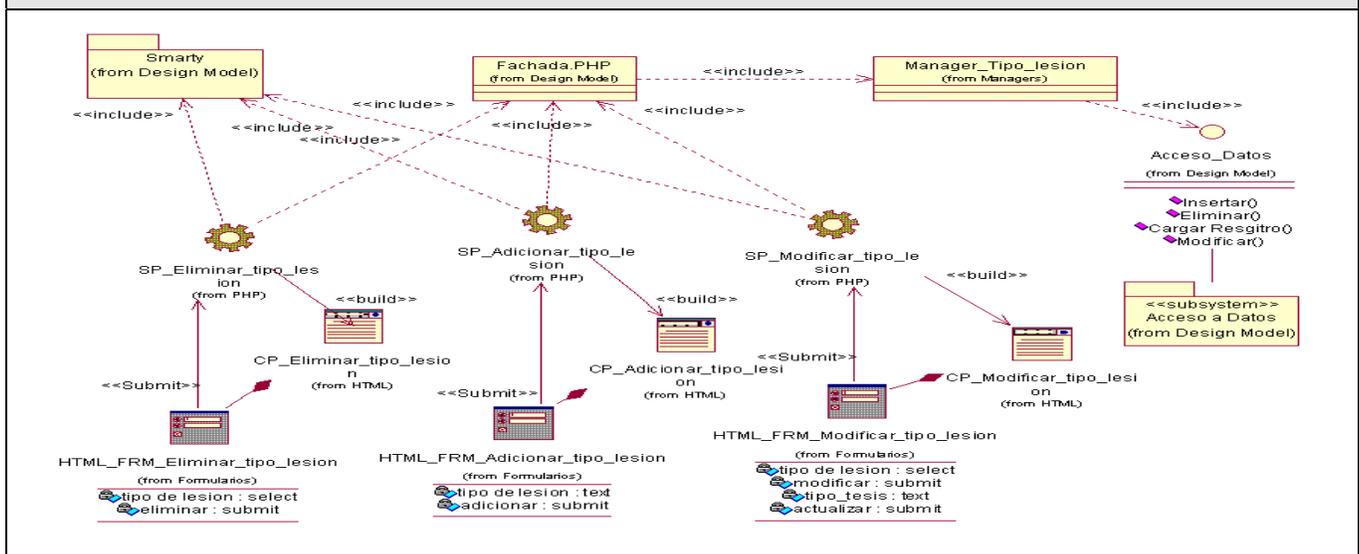


Fig. A5.19 Diagrama de clases del diseño para el caso de uso "Gestionar tipo lesión".

Diagrama de clases del diseño para el caso de uso "Gestionar unidad resultado".

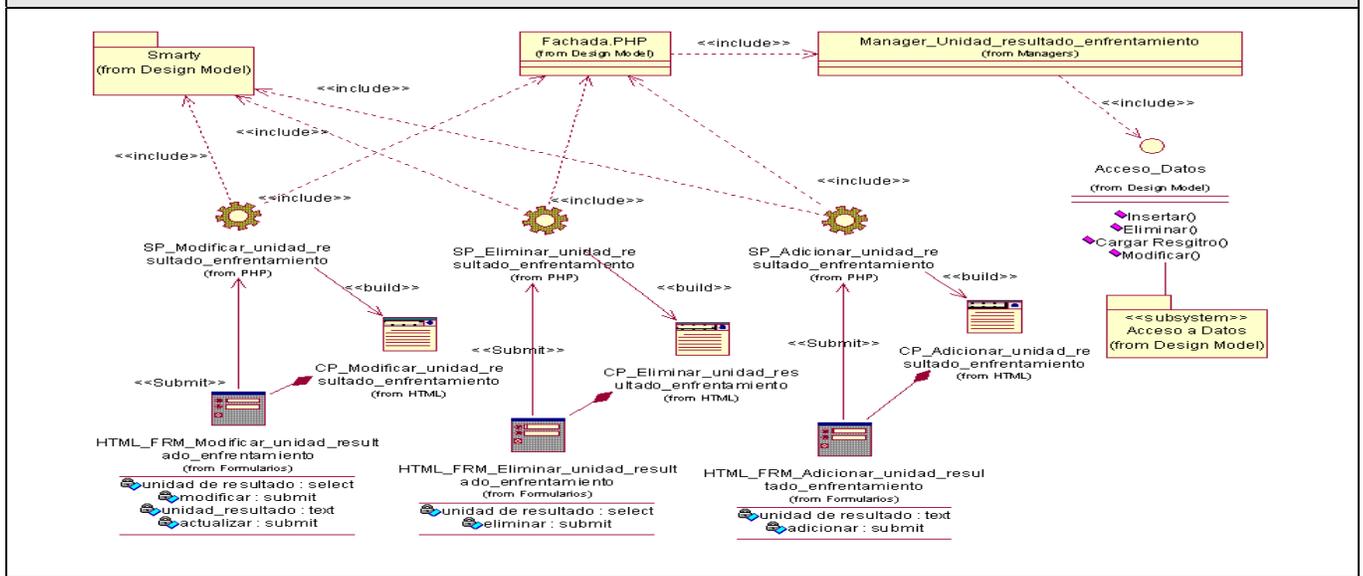


Fig. A5.20 Diagrama de clases del diseño para el caso de uso "Gestionar unidad resultado".

Diagrama de clases del diseño para el caso de uso "Gestionar utensilio".

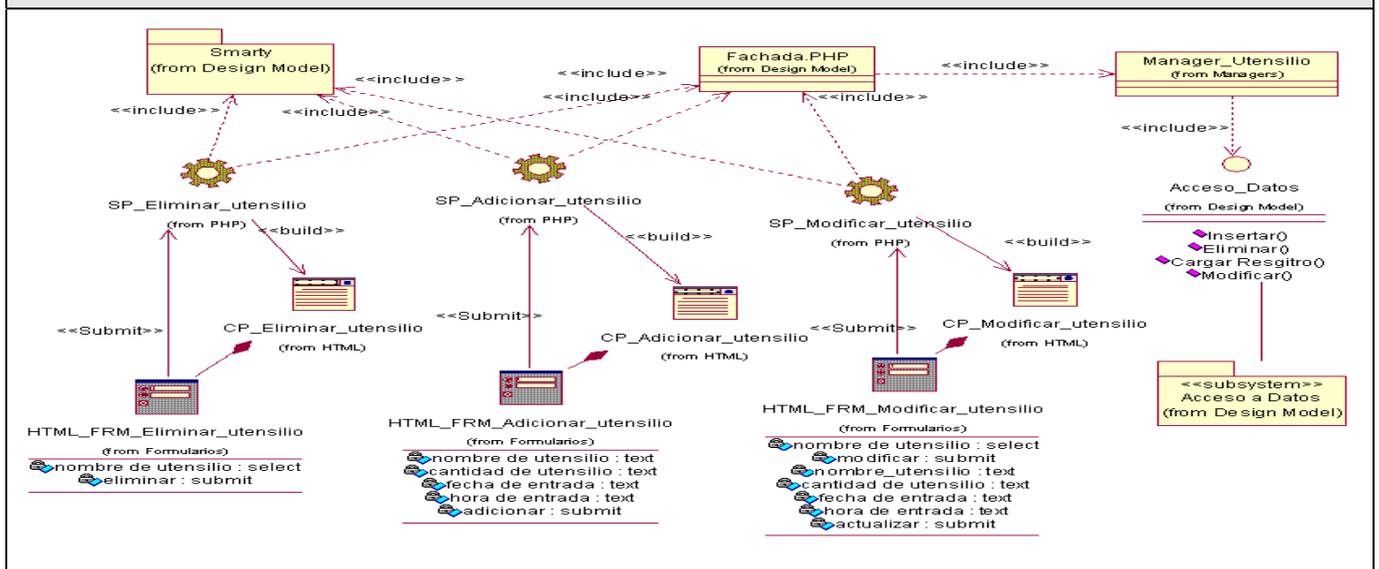


Fig. A5.21 Diagrama de clases del diseño para el caso de uso "Gestionar utensilio".



**Anexo # 6: Descripción textual de las clases Web.**

**6.1 Clases entidades.**

<b>Nombre:</b>	Catleta	
<b>Tipo de clase:</b>	Entidad	
<b>Atributo</b>	<b>Tipo</b>	
solapin	varchar	
id_equipo	int	
calzado	float	
estatura	float	
peso	float	
id_colo_piel	Int	
<b>Responsabilidad:</b>		
<b>Nombre:</b>	__construct(\$asolapin=NULL,\$aid_equipo=NULL,\$acalzado=NULL,\$aestatura=NULL,\$apeso=NULL,\$aid_colo_piel=NULL)	
	Set()	
	Get()	
<b>Descripción:</b>	Clase entidad que representa a la tabla (tb_atleta) de la base de datos.	

**Tabla A6.1 Clase entidad "Catleta".**

<b>Nombre:</b>	Ccategoria_competencia	
<b>Tipo de clase:</b>	Entidad	
<b>Atributo</b>	<b>Tipo</b>	
id_categoria	int	
nombre	varchar	
<b>Responsabilidad:</b>		
<b>Nombre:</b>	__construct(\$aid_categoria=NULL,\$anombre=NULL)	
	Set()	
	Get()	
<b>Descripción:</b>	Clase entidad que representa a la tabla (cl_categoria_competencia) de la base de datos.	

Tabla A6.2 Clase entidad “Ccategoria\_competencia”.

<b>Nombre:</b>	Ccolor_piel	
<b>Tipo de clase:</b>	Entidad	
<b>Atributo</b>	<b>Tipo</b>	
id_color	int	
color	varchar	
<b>Responsabilidad:</b>		
<b>Nombre:</b>	__construct(\$aid_color=NULL,\$acolor=NULL)	
	Set()	
	Get()	
<b>Descripción:</b>	Clase entidad que representa a la tabla (no_color_piel) de la base de datos.	

Tabla A6.3 Clase entidad “Ccolor\_piel”.

<b>Nombre:</b>	Ccomision	
<b>Tipo de clase:</b>	Entidad	
<b>Atributo</b>	<b>Tipo</b>	
id_comision	int	
nombre	varchar	
solapin	varchar	
id_comision_organizadora	int	
<b>Responsabilidad:</b>		
<b>Nombre:</b>	__construct(\$aid_comision=NULL,\$nombre=NULL, \$asolapin=NULL, \$aid_comision_organizadora=NULL)	
	Set()	
	Get()	
<b>Descripción:</b>	Clase entidad que representa a la tabla (tb_comision) de la base de datos.	

Tabla A6.4 Clase entidad “Ccomision”.

<b>Nombre:</b>	Ccomision_organizadora	
<b>Tipo de clase:</b>	Entidad	
<b>Atributo</b>	<b>Tipo</b>	
id_comision_organizadora	int	
solapin	varchar	
id_competencia	int	
<b>Responsabilidad:</b>		
<b>Nombre:</b>	__construct(\$aid_comision_organizadora=NULL,\$asolapin=NULL,\$aid_competencia=NULL)	
	Set()	
	Get()	
<b>Descripción:</b>	Clase entidad que representa a la tabla (tb_comision_organizadora) de la base de datos.	

Tabla A6.5 Clase entidad “Ccomision\_organizadora”.

<b>Nombre:</b>	Ccompetencia	
<b>Tipo de clase:</b>	Entidad	
<b>Atributo</b>	<b>Tipo</b>	
id_competencia	int	
nombre	varchar	
sede	varchar	
cant_arbitros	int	
cant_medicos	int	
fecha_inicio	time	
fecha_fin	time	
id_categoria	int	
id_curso	int	
deportes	array	
<b>Responsabilidad:</b>		
<b>Nombre:</b>	__construct(\$aid_competencia, \$nombre, \$sede, \$acant_arbitros, \$acant_medicos,\$afecha_inicio, \$afecha_fin, \$aid_categoria, \$aid_curso, \$adeportes)	

	Set()
	Get()
<b>Descripción:</b>	Clase entidad que representa a la tabla (tb_competencia) de la base de datos.

Tabla A6.6 Clase entidad “Ccompetencia”.

<b>Nombre:</b>	Ccurso
<b>Tipo de clase:</b>	Entidad
<b>Atributo</b>	<b>Tipo</b>
id_curso	int
curso	varchar
<b>Responsabilidad:</b>	
<b>Nombre:</b>	__construct(\$aid_curso=NULL,\$acurso=NULL)
	Set()
	Get()
<b>Descripción:</b>	Clase entidad que representa a la tabla (no_curso) de la base de datos.

Tabla A6.7 Clase entidad “Ccurso”.

<b>Nombre:</b>	Cdeporte
<b>Tipo de clase:</b>	Entidad
<b>Atributo</b>	<b>Tipo</b>
SGL_deporte	Varchar(3)
nombre	varchar
t_deporte	int
sexo	int
<b>Responsabilidad:</b>	
<b>Nombre:</b>	__construct(\$aSGL_deporte=NULL,\$anombre=NULL, \$at_deporte=NULL, \$asexo=NULL)
	Set()
	Get()
<b>Descripción:</b>	Clase entidad que representa a la tabla (tb_deporte) de la base de datos.

Tabla A6.8 Clase entidad “Cdeporte”.

<b>Nombre:</b>	Cenfrentamiento
<b>Tipo de clase:</b>	Entidad
Atributo	Tipo
id_enfrentamiento	int
fecha	date
cant_arbitros	int
cant_medicos	int
hora_inicio	time
hora_fin	time
lugar	varchar
id_fase	int
id_evento	int
id_sexo	int
id_tipo_enfrentamiento	int
<b>Responsabilidad:</b>	
<b>Nombre:</b>	__construct(\$aid_enfrentamiento=NULL, \$afecha=NULL, \$acant_arbitros=NULL, \$acant_medicos=NULL, \$ahora_inicio=NULL, \$ahora_fin=NULL, \$alugar=NULL, \$aid_fase=NULL, \$aid_evento=NULL, \$aid_sexo=NULL, \$aid_tipo_enfrentamiento=NULL)
	Set()
	Get()
<b>Descripción:</b>	Clase entidad que representa a la tabla (tb_enfrentamiento) de la base de datos.

Tabla A6.9 Clase entidad “Cenfrentamiento”.

<b>Nombre:</b>	Cenfrentamiento_duo_atleta
<b>Tipo de clase:</b>	Entidad
Atributo	Tipo
id_enfrentamiento	int
solapin	varchar
id_equipo	int
gano	bool
resultado	varchar
<b>Responsabilidad:</b>	
<b>Nombre:</b>	__construct(\$aid_enfrentamiento=NULL, \$asolapin=NULL, \$aid_equipo=NULL, \$agano=NULL, \$aresultado=NULL)

	Set()
	Get()
<b>Descripción:</b>	Clase entidad que representa a la tabla (tb_enfrentamiento_duo_atleta) de la base de datos.

Tabla A6.10 Clase entidad “Cenfrentamiento\_duo\_atleta”.

<b>Nombre:</b>	Cenfrentamiento_duo_equipo	
<b>Tipo de clase:</b>	Entidad	
<b>Atributo</b>	<b>Tipo</b>	
id_enfrentamiento	int	
id_equipo	int	
gano	bool	
resultado	varchar	
<b>Responsabilidad:</b>		
<b>Nombre:</b>	__construct(\$aid_enfrentamiento=NULL,\$aid_equipo=NULL,\$agano=NULL,\$aresultado=NULL)	
	Set()	
	Get()	
<b>Descripción:</b>	Clase entidad que representa a la tabla (tb_enfrentamiento_duo_equipo) de la base de datos.	

Tabla A6.11 Clase entidad “Cenfrentamiento\_duo\_equipo”.

<b>Nombre:</b>	Cenfrentamiento_multiple_atleta	
<b>Tipo de clase:</b>	Entidad	
<b>Atributo</b>	<b>Tipo</b>	
id_enfrentamiento	int	
id_equipo	int	
solapin	varchar	
resultado	varchar	
<b>Responsabilidad:</b>		
<b>Nombre:</b>	__construct(\$aid_enfrentamiento=NULL,\$asolapin=NULL,\$aid_equipo=NULL,\$aresultado=NULL)	
	Set()	

	Get()
<b>Descripción:</b>	Clase entidad que representa a la tabla (tb_enfrentamiento_multiple_atleta) de la base de datos.

Tabla A6.12 Clase entidad “Cenfrentamiento\_multiple\_atleta”.

<b>Nombre:</b>	Cequipo
<b>Tipo de clase:</b>	Entidad
<b>Atributo</b>	<b>Tipo</b>
id_equipo	int
SGLdeporte	varchar
id_representacion	int
id_sexo	int
solapin	varchar
<b>Responsabilidad:</b>	
<b>Nombre:</b>	__construct(\$aid_equipo=NULL,\$aSGLdeporte=NULL,\$aid_representacion=NULL,\$aid_sexo=NULL,\$asolapin=NULL)
	Set()
	Get()
<b>Descripción:</b>	Clase entidad que representa a la tabla (tb_equipo) de la base de datos.

Tabla A6.13 Clase entidad “Cequipo”.

<b>Nombre:</b>	Cestudiante
<b>Tipo de clase:</b>	Entidad
<b>Atributo</b>	<b>Tipo</b>
solapin	varchar
id_grupo	int
<b>Responsabilidad:</b>	
<b>Nombre:</b>	__construct(\$asolapin=NULL,\$aid_grupo=NULL)
	Set()
	Get()
<b>Descripción:</b>	Clase entidad que representa a la tabla (tb_estudiante) de la base de datos.

Tabla A6.14 Clase entidad “Cestudiante”.

<b>Nombre:</b>	Cevento
<b>Tipo de clase:</b>	Entidad
Atributo	Tipo
id_evento	int
nombre	varchar
colectivo	bool
id_sexo	int
id_sistema	int
id_unidad_resultado	int
<b>Responsabilidad:</b>	
<b>Nombre:</b>	__construct(\$aid_evento=NULL,\$anombre=NULL, \$acolectivo=NULL, \$aSGLdeporte=NULL, \$aid_sexo=NULL, \$aid_sistema=NULL, \$aid_unidad_resultado=NULL)
	Set()
	Get()
<b>Descripción:</b>	Clase entidad que representa a la tabla (tb_evento) de la base de datos.

Tabla A6.15 Clase entidad "Cevento".

<b>Nombre:</b>	Cfase_competencia
<b>Tipo de clase:</b>	Entidad
Atributo	Tipo
id_fase	int
nombre	varchar
<b>Responsabilidad:</b>	
<b>Nombre:</b>	__construct(\$aid_fase=NULL,\$anombre=NULL)
	Set()
	Get()
<b>Descripción:</b>	Clase entidad que representa a la tabla (no_fase) de la base de datos.

Tabla A6.16 Clase entidad "Cfase\_competencia".

<b>Nombre:</b>	Cgrupo
<b>Tipo de clase:</b>	Entidad

Atributo	Tipo
id_grupo	int
numero_grupo	varchar
anno	int
facultad	int
<b>Responsabilidad:</b>	
<b>Nombre:</b>	__construct(\$aid_grupo=NULL,\$anumero_grupo=NULL,\$aanno=NULL,\$afacultad=NU LL) Set() Get()
<b>Descripción:</b>	Clase entidad que representa a la tabla (tb_grupo) de la base de datos.

Tabla A6.17 Clase entidad "Cgrupo".

<b>Nombre:</b>	Cindicador
<b>Tipo de clase:</b>	Entidad
Atributo	Tipo
id_indicador	int
nombre	varchar
<b>Responsabilidad:</b>	
<b>Nombre:</b>	__construct(\$aid_indicador=NULL,\$anombre=NULL) Set() Get()
<b>Descripción:</b>	Clase entidad que representa a la tabla (no_indicador_ef) de la base de datos.

Tabla A6.18 Clase entidad "Cindicador".

<b>Nombre:</b>	Clesion
<b>Tipo de clase:</b>	Entidad
Atributo	Tipo
id_lesion	int
id_tipo_lesion	int
solapin	varchar
id_equipo	int
id_localizacion_lesion	int
<b>Responsabilidad:</b>	

<b>Nombre:</b>	__construct(\$aid_lesion=NULL,\$aid_tipo_lesion=NULL,\$asolapin=NULL,\$aid_equipo=NULL,\$aid_localizacion_lesion=NULL) Set() Get()
<b>Descripción:</b>	Clase entidad que representa a la tabla (tb_lesion) de la base de datos.

Tabla A6.19 Clase entidad “Clesion”.

<b>Nombre:</b>	Clocalizacion_lesion	
<b>Tipo de clase:</b>	Entidad	
<b>Atributo</b>	<b>Tipo</b>	
id_localizacion	int	
lugar	int	
<b>Responsabilidad:</b>		
<b>Nombre:</b>	__construct(\$aid_localizacion=NULL,\$alugar=NULL) Set() Get()	
<b>Descripción:</b>	Clase entidad que representa a la tabla (no_localizacion_lesion) de la base de datos.	

Tabla A6.20 Clase entidad “Clocalizacion\_lesion”.

<b>Nombre:</b>	Cnivel_general_est	
<b>Tipo de clase:</b>	Entidad	
<b>Atributo</b>	<b>Tipo</b>	
fecha	date	
id_prueba	int	
solapin	varchar	
nivel	int	
id_curso	int	
<b>Responsabilidad:</b>		
<b>Nombre:</b>	__construct(\$afecha=NULL,\$aid_prueba=NULL,\$asolapin=NULL,\$anivel=NULL,\$aid_curso=NULL) Set() Get()	
<b>Descripción:</b>	Clase entidad que representa a la tabla (tb_nivel_general_est) de la base de datos.	

Tabla A6.21 Clase entidad “Cnivel\_general\_est”.

<b>Nombre:</b>	Cnivel_general_grupo
<b>Tipo de clase:</b>	Entidad
<b>Atributo</b>	<b>Tipo</b>
fecha	date
id_prueba	int
id_grupo	int
nivel	int
id_curso	int
<b>Responsabilidad:</b>	
<b>Nombre:</b>	__construct(\$afecha=NULL,\$aid_prueba=NULL,\$aid_grupo=NULL,\$anivel=NULL,\$aid_curso=NULL)
	Set()
	Get()
<b>Descripción:</b>	Clase entidad que representa a la tabla (tb_nivel_grupo_general) de la base de datos.

Tabla A6.22 Clase entidad “Cnivel\_general\_grupo”.

<b>Nombre:</b>	Cnivel_grupo_indicador
<b>Tipo de clase:</b>	Entidad
<b>Atributo</b>	<b>Tipo</b>
fecha	date
id_indicador	int
id_grupo	int
nivel	int
id_curso	int
<b>Responsabilidad:</b>	
<b>Nombre:</b>	__construct(\$afecha=NULL,\$aid_indicador=NULL,\$aid_grupo=NULL,\$anivel=NULL,\$aid_curso=NULL)
	Set()
	Get()
<b>Descripción:</b>	Clase entidad que representa a la tabla (tb_nivel_grupo_indicador) de la base de datos.

Tabla A6.23 Clase entidad “Cnivel\_grupo\_indicador”.

<b>Nombre:</b>	Cpersona
<b>Tipo de clase:</b>	Entidad
<b>Atributo</b>	<b>Tipo</b>
Solapin	varchar
direccion_particular	varchar
id_ocupacion	int
nombre	varchar
papellido	varchar
sapellido	varchar
edad	int
usuario	varchar
id_municipio	int
id_provincia	int
id_sexo	int
<b>Responsabilidad:</b>	
<b>Nombre:</b>	__construct(\$aSolapin=NULL, \$adireccion_particular=NULL, \$aid_ocupacion=NULL, \$anombre=NULL, \$apapellido=NULL, \$asapellido=NULL, \$aedad=NULL, \$ausuario=NULL, \$aid_municipio=NULL, \$aid_provincia=NULL, \$aid_sexo=NULL)
	Set()
	Get()
<b>Descripción:</b>	Clase entidad que representa a la tabla (tb_persona) de la base de datos.

Tabla A6.24 Clase entidad "Cpersona".

<b>Nombre:</b>	Cprestamo
<b>Tipo de clase:</b>	Entidad
<b>Atributo</b>	<b>Tipo</b>
id_prestamo	int
id_utensilio	int
solapin	varchar
fecha_inicio	date
fecha_fin	date
hora_inicio	time
hora_fin	time
estado_inicial	int
estado_final	int
cantidad	int

entregado	bool
<b>Responsabilidad:</b>	
<b>Nombre:</b>	__construct(\$aid_prestamo=NULL,\$aid_utensilio=NULL,\$asolapin=NULL,\$afecha_inicio=NULL,\$afecha_fin=NULL,\$ahora_inicio=NULL,\$ahora_fin=NULL,\$aestado_inicial=NULL,\$aestado_final=NULL,\$acantidad=NULL,\$aentregado=NULL)
	Set()
	Get()
<b>Descripción:</b>	Clase entidad que representa a la tabla (tb_prestamo) de la base de datos.

Tabla A6.25 Clase entidad "Cprestamo".

<b>Nombre:</b>	Cprofesor_ef
<b>Tipo de clase:</b>	Entidad
<b>Atributo</b>	<b>Tipo</b>
solapin	varchar
id_grupo	int
<b>Responsabilidad:</b>	
<b>Nombre:</b>	__construct(\$asolapin=NULL,\$aid_grupo=NULL)
	Set()
	Get()
<b>Descripción:</b>	Clase entidad que representa a la tabla (tb_profesor_ef) de la base de datos.

Tabla A6.26 Clase entidad "Cprofesor\_ef".

<b>Nombre:</b>	Crepresentacion
<b>Tipo de clase:</b>	Entidad
<b>Atributo</b>	<b>Tipo</b>
id_representacion	Int
puntos_acumulados	Int
nombre	varchar
mascota	varchar
id_competencia	Int
<b>Responsabilidad:</b>	
<b>Nombre:</b>	__construct(\$aid_representacion=NULL,\$apuntos_acumulados=NULL,\$anombre=NULL,\$amascota=NULL,\$aid_competencia=NULL)

	Set()
	Get()
<b>Descripción:</b>	Clase entidad que representa a la tabla (tb_representacion) de la base de datos.

Tabla A6.27 Clase entidad "Crepresentacion".

<b>Nombre:</b>	Cresultado
<b>Tipo de clase:</b>	Entidad
Atributo	Tipo
fecha	date
id_indicador	Int
id_prueba	Int
solapin	varchar
nivel_indicador	Int
resultado	Int
<b>Responsabilidad:</b>	
<b>Nombre:</b>	__construct(\$afecha=NULL,\$aid_indicador=NULL,\$aid_prueba=NULL,\$asolapin=NULL,\$anivel_indicador=NULL,\$aresultado=NULL)
	Set()
	Get()
<b>Descripción:</b>	Clase entidad que representa a la tabla (tb_resultado) de la base de datos.

Tabla A6.28 Clase entidad "Cresultado".

<b>Nombre:</b>	Croll
<b>Tipo de clase:</b>	Entidad
Atributo	Tipo
id_roll	Int
roll	varchar
<b>Responsabilidad:</b>	
<b>Nombre:</b>	__construct(\$aid_roll=NULL,\$aroll=NULL)
	Set()
	Get()
<b>Descripción:</b>	Clase entidad que representa a la tabla (no_rol) de la base de datos.

Tabla A6.29 Clase entidad "Croll".

<b>Nombre:</b>	Csistema_competencia	
<b>Tipo de clase:</b>	Entidad	
<b>Atributo</b>	<b>Tipo</b>	
id_sistema	Int	
nombre	varchar	
<b>Responsabilidad:</b>		
<b>Nombre:</b>	__construct(\$aid_sistema=NULL,\$nombre=NULL)	
	Set()	
	Get()	
<b>Descripción:</b>	Clase entidad que representa a la tabla (no_sistema_competencia) de la base de datos.	

Tabla A6.30 Clase entidad "Csistema\_competencia".

<b>Nombre:</b>	Ctipo_deporte	
<b>Tipo de clase:</b>	Entidad	
<b>Atributo</b>	<b>Tipo</b>	
id_tipo	Int	
nombre	varchar	
<b>Responsabilidad:</b>		
<b>Nombre:</b>	__construct(\$aid_tipo=NULL,\$nombre=NULL)	
	Set()	
	Get()	
<b>Descripción:</b>	Clase entidad que representa a la tabla (cl_tipo_deporte) de la base de datos.	

Tabla A6.31 Clase entidad "Ctipo\_deporte".

<b>Nombre:</b>	Ctipo_enfrentamiento	
<b>Tipo de clase:</b>	Entidad	
<b>Atributo</b>	<b>Tipo</b>	

id_enfrentamiento	Int
nombre	varchar
<b>Responsabilidad:</b>	
<b>Nombre:</b>	__construct(\$aid_enfrentamiento=NULL,\$nombre=NULL)
	Set()
	Get()
<b>Descripción:</b>	Clase entidad que representa a la tabla (cl_tipo_enfrentamiento) de la base de datos.

Tabla A6.32 Clase entidad "Ctipo\_enfrentamiento".

<b>Nombre:</b>	Ctipo_lesion	
<b>Tipo de clase:</b>	Entidad	
<b>Atributo</b>	<b>Tipo</b>	
id_tipo_lesion	Int	
lesion	varchar	
<b>Responsabilidad:</b>		
<b>Nombre:</b>	__construct(\$aid_tipo_lesion=NULL,\$lesion=NULL)	
	Set()	
	Get()	
<b>Descripción:</b>	Clase entidad que representa a la tabla (cl_tipo_lesion) de la base de datos.	

Tabla A6.33 Clase entidad "Ctipo\_lesion".

<b>Nombre:</b>	Cunidad_resultado_enf	
<b>Tipo de clase:</b>	Entidad	
<b>Atributo</b>	<b>Tipo</b>	
id_unidad	Int	
nombre	varchar	
<b>Responsabilidad:</b>		
<b>Nombre:</b>	__construct(\$aid_unidad=NULL,\$nombre=NULL)	
	Set()	
	Get()	
<b>Descripción:</b>	Clase entidad que representa a la tabla (no_unidad_resultado) de la base de datos.	

Tabla A6.34 Clase entidad “Cunidad\_resultado\_enf”.

<b>Nombre:</b>	Cusuario
<b>Tipo de clase:</b>	Entidad
<b>Atributo</b>	<b>Tipo</b>
solapin	varchar
pass	varchar
roles	array
<b>Responsabilidad:</b>	
<b>Nombre:</b>	__construct(\$asolapin=NULL,\$apass=NULL,\$aroles=NULL)
	Set()
	Get()
<b>Descripción:</b>	Clase entidad que representa a la tabla (tb_usuario) de la base de datos.

Tabla A6.35 Clase entidad “Cusuario”.

<b>Nombre:</b>	Cutensilio
<b>Tipo de clase:</b>	Entidad
<b>Atributo</b>	<b>Tipo</b>
id_utensilio	int
nombre	varchar
cantidad	Int
fecha_hora_ent	Date/time
SGL_deporte	varchar
<b>Responsabilidad:</b>	
<b>Nombre:</b>	__construct(\$aid_utensilio=NULL, \$anombre=NULL,\$acantidad=NULL, \$afecha_hora_ent=NULL, \$aSGL_deporte=NULL)
	Set()
	Get()
<b>Descripción:</b>	Clase entidad que representa a la tabla (tb_utensilio) de la base de datos.

Tabla A6.36 Clase entidad “Cutensilio”.

<b>Nombre:</b>	Cvisita
<b>Tipo de clase:</b>	Entidad

Atributo		Tipo
id_visita		int
Solapin		varchar
hora_ent		time
hora_sal		time
fecha		date
terminada		bool
<b>Responsabilidad:</b>		
<b>Nombre:</b>	__construct(\$aid_visita=NULL,\$aSolapin=NULL, \$ahora_ent=NULL, \$ahora_sal=NULL, \$afecha=NULL,\$aterminada=NULL)	
	Set()	
	Get()	
<b>Descripción:</b>	Clase entidad que representa a la tabla (tb_visita) de la base de datos.	

Tabla A6.37 Clase entidad "Cvisita".

### 6.2 Clases de acceso a datos.

<b>Nombre:</b>	CAcceso_Datos	
<b>Tipo de clase:</b>	DAO	
Atributo		Tipo
servidor		string
usuario		string
password		string
basedato		string
conexión		string
error		string
Ado_DB		ADONewConnection
<b>Responsabilidad:</b>		
<b>Nombre:</b>	Base_Datos(\$s=SERVIDOR, \$u=USUARIO, \$p=PASSWORD, \$bd = BASEDATOS)	
	Abrir_Conexion()	
	Cerrar_Conexion()	
	Ejecutar_Consulta(sql)	

	Cargar_Registro(sql)
	Empezar_Transaccion()
	Terminar_Transaccion()
	Ocurrio_Error_Transaccion()
	Forzar_Transaccion()
<b>Descripción:</b>	Clase encargada de utilizar la herramienta adodb para realizar las operaciones de acceso a la base de datos.

Tabla A6.38 Clase de acceso a datos "CAcceso\_Datos".

<b>Nombre:</b>	Catleta_DAO	
<b>Tipo de clase:</b>	DAO	
<b>Atributo</b>	<b>Tipo</b>	
<b>Responsabilidad:</b>		
<b>Nombre:</b>	Insertar_atleta(Catleta \$atleta)	
	Modificar_atleta(Catleta \$atleta)	
	Eliminar_atleta(Catleta \$atleta)	
	Listar_atletas()	
	Listar_atleta_deporte()	
	Listar_atleta_equipo()	
	Listar_atleta_bajo_peso()	
	Listar_atleta_competencia()	
<b>Descripción:</b>	Controla las operaciones de acceso a la tabla (tb_atleta).	

Tabla A6.39 Clase de acceso a datos "Catleta\_DAO".

<b>Nombre:</b>	Ccategoria_competencia_DAO	
<b>Tipo de clase:</b>	DAO	
<b>Atributo</b>	<b>Tipo</b>	
<b>Responsabilidad:</b>		
<b>Nombre:</b>	Insertar_categoria(Ccategoria_competencia \$categoria)	
	Modificar_categoria(Ccategoria_competencia \$categoria)	

	Eliminar_categoria(Ccategoria_competencia \$categoria)
	Listar_categoria()
<b>Descripción:</b>	Controla las operaciones de acceso a la tabla (cl_categoria_competencia).

Tabla A6.40 Clase de acceso a datos "Ccategoria\_competencia\_DAO".

<b>Nombre:</b>	Ccomision_DAO	
<b>Tipo de clase:</b>	DAO	
<b>Atributo</b>	<b>Tipo</b>	
<b>Responsabilidad:</b>		
<b>Nombre:</b>	Insertar_comision(Ccomision \$comision)	
	Modificar_comision(Ccomision \$comision)	
	Eliminar_comision(Ccomision \$comision)	
	Listar_comisiones()	
<b>Descripción:</b>	Controla las operaciones de acceso a la tabla (tb_comision).	

Tabla A6.41 Clase de acceso a datos "Ccomision\_DAO".

<b>Nombre:</b>	Ccomision_organizadora_DAO	
<b>Tipo de clase:</b>	DAO	
<b>Atributo</b>	<b>Tipo</b>	
<b>Responsabilidad:</b>		
<b>Nombre:</b>	Insertar_comision_organizadora(Ccomision_organizadora \$comision_organizadora)	
	Modificar_comision_organizadora(Ccomision_organizadora \$comision_organizadora)	
	Eliminar_comision_organizadora(Ccomision_organizadora \$comision_organizadora)	
	Listar_comisiones_organizadoras()	
<b>Descripción:</b>	Controla las operaciones de acceso a la tabla (tb_comision_organizadora).	

Tabla A6.42 Clase de acceso a datos "Ccomision\_organizadora\_DAO".

<b>Nombre:</b>	Ccompetencia_DAO	
<b>Tipo de clase:</b>	DAO	
<b>Atributo</b>	<b>Tipo</b>	

<b>Responsabilidad:</b>	
<b>Nombre:</b>	Insertar_competencia(Ccompetencia \$competencia)
	Modificar_competencia(Ccompetencia \$competencia)
	Eliminar_competencia(Ccompetencia \$competencia)
	Listar_competencias()
<b>Descripción:</b>	Controla las operaciones de acceso a la tabla (tb_competencia).

Tabla A6.43 Clase de acceso a datos "Ccompetencia\_DAO".

<b>Nombre:</b>	Cdeporte_DAO	
<b>Tipo de clase:</b>	DAO	
<b>Atributo</b>	<b>Tipo</b>	
<b>Responsabilidad:</b>		
<b>Nombre:</b>	Insertar_Deporte(Cdeporte \$deporte)	
	Existe_Deporte(Cdeporte \$deporte)	
	Listado_Deportes()	
	Eliminar_Deporte(CDeporte \$deporte)	
	Detalle_Deporte(CDeporte \$deporte)	
	Modificar_Deporte(CDeporte \$deporte)	
	Listar_Tipos_Deportes()	
	Listar_Sexos_Deportes()	
<b>Descripción:</b>	Controla las operaciones de acceso a la tabla (tb_deporte).	

Tabla A6.44 Clase de acceso a datos "Cdeporte\_DAO".

<b>Nombre:</b>	Cenfrentamiento_DAO	
<b>Tipo de clase:</b>	DAO	
<b>Atributo</b>	<b>Tipo</b>	
<b>Responsabilidad:</b>		
<b>Nombre:</b>	Insertar_enfrentamiento(Cenfrentamiento \$enfrentamiento)	
	Modificar_enfrentamiento(Cenfrentamiento \$enfrentamiento)	
	Eliminar_enfrentamiento(Cenfrentamiento \$enfrentamiento)	

<b>Descripción:</b>	Controla las operaciones de acceso a la tabla (tb_enfrentamiento).
---------------------	--

**Tabla A6.45 Clase de acceso a datos “Cenfrentamiento\_DAO”.**

<b>Nombre:</b>	Cenfrentamiento_duo_atleta_DAO	
<b>Tipo de clase:</b>	DAO	
<b>Atributo</b>	<b>Tipo</b>	
<b>Responsabilidad:</b>		
<b>Nombre:</b>	Insertar_Cenfrentamiento_duo_atleta(Cenfrentamiento_duo_atleta \$Cenfrentamiento_duo_atleta)	
	Modificar_Cenfrentamiento_duo_atleta(Cenfrentamiento_duo_atleta \$Cenfrentamiento_duo_atleta)	
	Eliminar_Cenfrentamiento_duo_atleta(Cenfrentamiento_duo_atleta \$Cenfrentamiento_duo_atleta)	
<b>Descripción:</b>	Controla las operaciones de acceso a la tabla (tb_enfrentamiento_duo_atleta).	

**Tabla A6.46 Clase de acceso a datos “Cenfrentamiento\_duo\_atleta\_DAO”.**

<b>Nombre:</b>	Cenfrentamiento_duo_equipo_DAO	
<b>Tipo de clase:</b>	DAO	
<b>Atributo</b>	<b>Tipo</b>	
<b>Responsabilidad:</b>		
<b>Nombre:</b>	Insertar_Cenfrentamiento_duo_equipo(Cenfrentamiento_duo_equipo \$Cenfrentamiento_duo_equipo)	
	Modificar_Cenfrentamiento_duo_equipo(Cenfrentamiento_duo_equipo \$Cenfrentamiento_duo_equipo)	
	Eliminar_Cenfrentamiento_duo_equipo(Cenfrentamiento_duo_equipo \$Cenfrentamiento_duo_equipo)	
<b>Descripción:</b>	Controla las operaciones de acceso a la tabla (tb_enfrentamiento_duo_equipo).	

**Tabla A6.47 Clase de acceso a datos “Cenfrentamiento\_duo\_equipo\_DAO”.**

<b>Nombre:</b>	Cenfrentamiento_multiple_atleta_DAO	
<b>Tipo de clase:</b>	DAO	

Atributo	Tipo
<b>Responsabilidad:</b>	
<b>Nombre:</b>	Insertar_Cenfrentamiento_multiple_atleta(Cenfrentamiento_multiple_atleta \$Cenfrentamiento_multiple_atleta)
	Modificar_Cenfrentamiento_multiple_atleta(Cenfrentamiento_multiple_atleta \$Cenfrentamiento_multiple_atleta)
	Eliminar_Cenfrentamiento_multiple_atleta(Cenfrentamiento_multiple_atleta \$Cenfrentamiento_multiple_atleta)
<b>Descripción:</b>	Controla las operaciones de acceso a la tabla (tb_enfrentamiento_multiple_atleta).

Tabla A6.48 Clase de acceso a datos “Cenfrentamiento\_multiple\_atleta\_DAO”.

<b>Nombre:</b>	Cequipo_DAO
<b>Tipo de clase:</b>	DAO
Atributo	Tipo
<b>Responsabilidad:</b>	
<b>Nombre:</b>	Insertar_equipo(Cequipo \$equipo)
	Modificar_equipo(Cequipo \$equipo)
	Eliminar_equipo(Cequipo \$equipo)
	Listar_equipos()
	Listar_equipo_deporte()
	Listar_equipo_competencia()
<b>Descripción:</b>	Controla las operaciones de acceso a la tabla (tb_equipo).

Tabla A6.49 Clase de acceso a datos “Cequipo\_DAO”.

<b>Nombre:</b>	Cestudiante_DAO
<b>Tipo de clase:</b>	DAO
Atributo	Tipo
<b>Responsabilidad:</b>	
<b>Nombre:</b>	Insertar_estudiante(Cestudiante \$estudiante)
	Modificar_estudiante(Cestudiante \$estudiante)
	Eliminar_estudiante(Cestudiante \$estudiante)

	Listar_estudiantes() Listar_estudiante_grupo(Cestudiante \$estudiante)
<b>Descripción:</b>	Controla las operaciones de acceso a la tabla (tb_estudiante).

Tabla A6.50 Clase de acceso a datos "Cestudiante\_DAO".

<b>Nombre:</b>	Cevento_DAO
<b>Tipo de clase:</b>	DAO
<b>Atributo</b>	<b>Tipo</b>
<b>Responsabilidad:</b>	
<b>Nombre:</b>	Insertar_Evento(Cevento \$evento) Existe_Evento(Cevento \$evento) Listado_Eventos() Eliminar_Evento(Cevento \$evento) Detalle_Evento(Cevento \$evento) Modificar_Evento(Cevento \$evento)
<b>Descripción:</b>	Controla las operaciones de acceso a la tabla (tb_evento).

Tabla A6.51 Clase de acceso a datos "Cevento\_DAO".

<b>Nombre:</b>	Cfase_competencia_DAO
<b>Tipo de clase:</b>	DAO
<b>Atributo</b>	<b>Tipo</b>
<b>Responsabilidad:</b>	
<b>Nombre:</b>	Insertar_fase(Cfase_competencia \$fase) Modificar_fase(Cfase_competencia \$fase) Eliminar_fase(Cfase_competencia \$fase)
<b>Descripción:</b>	Controla las operaciones de acceso a la tabla (tb_fase).

Tabla A6.52 Clase de acceso a datos "Cfase\_competencia\_DAO".

<b>Nombre:</b>	Cgrupo_DAO
<b>Tipo de clase:</b>	DAO

Atributo	Tipo
<b>Responsabilidad:</b>	
<b>Nombre:</b>	Insertar_grupo(Cgrupo \$grupo)
	Modificar_grupo(Cgrupo \$grupo)
	Eliminar_grupo(Cgrupo \$grupo)
	Listar_grupos()
	Listar_grupo_profesor_ef(Cgrupo \$grupo)
<b>Descripción:</b>	Controla las operaciones de acceso a la tabla (tb_grupo).

Tabla A6.53 Clase de acceso a datos "Cgrupo\_DAO".

<b>Nombre:</b>	Cindicador_DAO	
<b>Tipo de clase:</b>	DAO	
Atributo	Tipo	
<b>Responsabilidad:</b>		
<b>Nombre:</b>	Insertar_indicador(Cindicador \$indicador)	
	Modificar_indicador(Cindicador \$indicador)	
	Eliminar_indicador(Cindicador \$indicador)	
	Listar_indicador()	
<b>Descripción:</b>	Controla las operaciones de acceso a la tabla (tb_indicador).	

Tabla A6.54 Clase de acceso a datos "Cindicador\_DAO".

<b>Nombre:</b>	Clesion_DAO	
<b>Tipo de clase:</b>	DAO	
Atributo	Tipo	
<b>Responsabilidad:</b>		
<b>Nombre:</b>	Insertar_lesion(Clesion \$lesion)	
	Modificar_lesion(Clesion \$lesion)	
	Eliminar_lesion(Clesion \$lesion)	
	Listar_lesiones()	
<b>Descripción:</b>	Controla las operaciones de acceso a la tabla (tb_lesion).	

Tabla A6.55 Clase de acceso a datos “Clesion\_DAO”.

<b>Nombre:</b>	Clocalizacion_lesion_DAO	
<b>Tipo de clase:</b>	DAO	
<b>Atributo</b>	<b>Tipo</b>	
<b>Responsabilidad:</b>		
<b>Nombre:</b>	Insertar_lugar_lesion(Clocalizacion_lesion \$lugar)	
	Modificar_lugar_lesion(Clocalizacion_lesion \$lugar)	
	Eliminar_lugar_lesion(Clocalizacion_lesion \$lugar)	
	Listar_localizaciones_lesion()	
<b>Descripción:</b>	Controla las operaciones de acceso a la tabla (no_localizacion_lesion).	

Tabla A6.56 Clase de acceso a datos “Clocalizacion\_lesion\_DAO”.

<b>Nombre:</b>	Cnivel_general_est_DAO	
<b>Tipo de clase:</b>	DAO	
<b>Atributo</b>	<b>Tipo</b>	
<b>Responsabilidad:</b>		
<b>Nombre:</b>	Insertar_nivel_general_est(Cnivel_general_est \$nivel_general_est)	
	Modificar_nivel_general_est(Cnivel_general_est \$nivel_general_est)	
	Listar_nivel_general_est()	
<b>Descripción:</b>	Controla las operaciones de acceso a la tabla (tb_nivel_general_est).	

Tabla A6.57 Clase de acceso a datos “Cnivel\_general\_est\_DAO”.

<b>Nombre:</b>	Cnivel_general_grupo_DAO	
<b>Tipo de clase:</b>	DAO	
<b>Atributo</b>	<b>Tipo</b>	
<b>Responsabilidad:</b>		
<b>Nombre:</b>	Insertar_nivel_general_grupo(Cnivel_general_grupo \$nivel_general_grupo)	

	Modificar_nivel_general_grupo(Cnivel_general_grupo \$nivel_general_grupo)
	Listar_nivel_general_grupo()
<b>Descripción:</b>	Controla las operaciones de acceso a la tabla (tb_nivel_grupo_general).

Tabla A6.58 Clase de acceso a datos “Cnivel\_general\_grupo\_DAO”.

<b>Nombre:</b>	Cnivel_grupo_indicador_DAO
<b>Tipo de clase:</b>	DAO
<b>Atributo</b>	<b>Tipo</b>
<b>Responsabilidad:</b>	
<b>Nombre:</b>	Insertar_nivel_grupo_indicador(Cnivel_grupo_indicador \$nivel_grupo_indicador)
	Modificar_nivel_grupo_indicador(Cnivel_grupo_indicador \$nivel_grupo_indicador)
	Listar_nivel_grupo_indicador()
<b>Descripción:</b>	Controla las operaciones de acceso a la tabla (tb_nivel_grupo_indicador).

Tabla A6.59 Clase de acceso a datos “Cnivel\_grupo\_indicador\_DAO”.

<b>Nombre:</b>	Cnormativa_DAO
<b>Tipo de clase:</b>	DAO
<b>Atributo</b>	<b>Tipo</b>
<b>Responsabilidad:</b>	
<b>Nombre:</b>	Insertar_normativa(Cnormativa \$normativa)
	Modificar_normativa(Cnormativa \$normativa)
	Eliminar_normativa(Cnormativa \$normativa)
	Listar_normativas()
	Buscar_Nivel_de_normativa(Cnormativa \$normativa)
	Buscar_Nivel_general_estudiante(Cnormativa \$normativa)
<b>Descripción:</b>	Controla las operaciones de acceso a la tabla (tb_normativa).

Tabla A6.60 Clase de acceso a datos “Cnormativa\_DAO”.

<b>Nombre:</b>	Cpersona_DAO
<b>Tipo de clase:</b>	DAO

Atributo	Tipo
<b>Responsabilidad:</b>	
<b>Nombre:</b>	Insertar_Persona(Cpersona \$Persona)
	Existe_Persona(CPersona \$Persona)
	Listado_Persona()
	Eliminar_Persona(CPersona \$Persona)
	Datos_Persona(CPersona \$Persona)
	Persona_NAF(CPersona \$Persona)
	Modificar_Persona(CPersona \$Persona)
<b>Descripción:</b>	Controla las operaciones de acceso a la tabla (tb_persona).

Tabla A6.61 Clase de acceso a datos "Cpersona\_DAO".

<b>Nombre:</b>	Cprestamo_DAO	
<b>Tipo de clase:</b>	DAO	
Atributo	Tipo	
<b>Responsabilidad:</b>		
<b>Nombre:</b>	Insertar_Prestamo(Cprestamo \$prestamo)	
	Listado_Prestamos_Pendientes()	
	Listado_Prestamos_Pendientes_Usuario(Cprestamo \$prestamo )	
	Listado_Prestamos_Fecha(Cprestamo \$prestamo)	
	Listado_Prestamos_Usuario(Cprestamo \$prestamo)	
	Listado_Prestamos_Completados()	
	Detalle_Prestamo(Cprestamo \$prestamo)	
	Modificar_Prestamo(Cprestamo \$prestamo)	
	Prestamo_Cant_UTE(Cprestamo \$prestamo)	
	Terminar_prestamo(Cprestamo \$prestamo)	
<b>Descripción:</b>	Controla las operaciones de acceso a la tabla (tb_prestamo).	

Tabla A6.62 Clase de acceso a datos "Cprestamo\_DAO".

<b>Nombre:</b>	Cprofesor_ef_DAO	
<b>Tipo de clase:</b>	DAO	
Atributo	Tipo	

<b>Responsabilidad:</b>	
<b>Nombre:</b>	Insertar_profesor_ef(Cprofesor_ef \$profesor_ef)
	Modificar_profesor_ef(Cprofesor_ef \$profesor_ef)
	Eliminar_profesor_ef(Cprofesor_ef \$profesor_ef)
	Listar_profesor_ef()
<b>Descripción:</b>	Controla las operaciones de acceso a la tabla (tb_profesor_ef).

Tabla A6.63 Clase de acceso a datos "Cprofesor\_ef\_DAO".

<b>Nombre:</b>	Cprueba_DAO	
<b>Tipo de clase:</b>	DAO	
<b>Atributo</b>	<b>Tipo</b>	
<b>Responsabilidad:</b>		
<b>Nombre:</b>	Insertar_prueba(Cprueba \$prueba)	
	Modificar_prueba(Cprueba \$prueba)	
	Eliminar_prueba(Cprueba \$prueba)	
	Listar_pruebas()	
	Listar_prueba_grupo()	
<b>Descripción:</b>	Controla las operaciones de acceso a la tabla (tb_prueba_ef).	

Tabla A6.64 Clase de acceso a datos "Cprueba\_DAO".

<b>Nombre:</b>	Crepresentacion_DAO	
<b>Tipo de clase:</b>	DAO	
<b>Atributo</b>	<b>Tipo</b>	
<b>Responsabilidad:</b>		
<b>Nombre:</b>	Insertar_representacion(Crepresentacion \$representacion)	
	Modificar_representacion(Crepresentacion \$representacion)	
	Eliminar_representacion(Crepresentacion \$representacion)	
	Listar_representaciones()	
	Listar_representacion_competencia()	
<b>Descripción:</b>	Controla las operaciones de acceso a la tabla (tb_representacion).	

Tabla A6.65 Clase de acceso a datos "Crepresentacion\_DAO".

<b>Nombre:</b>	Cresultado_DAO	
<b>Tipo de clase:</b>	DAO	
<b>Atributo</b>	<b>Tipo</b>	
<b>Responsabilidad:</b>		
<b>Nombre:</b>	Insertar_resultado(Cresultado \$resultado)	
	Modificar_resultado(Cresultado \$resultado)	
	Eliminar_resultado(Cresultado \$resultado)	
	Listar_resultados()	
	Buscar_nivel()	
	Mostrar_Nivel_Est_Ind(Cresultado \$resultado)	
<b>Descripción:</b>	Controla las operaciones de acceso a la tabla (tb_resultado).	

Tabla A6.66 Clase de acceso a datos "Cresultado\_DAO".

<b>Nombre:</b>	Croll_DAO	
<b>Tipo de clase:</b>	DAO	
<b>Atributo</b>	<b>Tipo</b>	
<b>Responsabilidad:</b>		
<b>Nombre:</b>	Insertar_roll(Croll \$roll)	
	Modificar_roll(Croll \$roll)	
	Eliminar_roll(Croll \$roll)	
	Listar_roll()	
	Existe_roll(Croll \$roll)	
<b>Descripción:</b>	Controla las operaciones de acceso a la tabla (tb_resultado).	

Tabla A6.67 Clase de acceso a datos "Croll\_DAO".

<b>Nombre:</b>	Csistema_competencia_DAO	
<b>Tipo de clase:</b>	DAO	
<b>Atributo</b>	<b>Tipo</b>	

<b>Responsabilidad:</b>	
<b>Nombre:</b>	Insertar_sistema(Csistema_competencia \$sistema)
	Modificar_sistema(Csistema_competencia \$sistema)
	Eliminar_sistema(Csistema_competencia \$sistema)
	Listar_sistema()
<b>Descripción:</b>	Controla las operaciones de acceso a la tabla (no_sistema_competencia).

Tabla A6.68 Clase de acceso a datos "Csistema\_competencia\_DAO".

<b>Nombre:</b>	Ctipo_deporte_DAO	
<b>Tipo de clase:</b>	DAO	
<b>Atributo</b>	<b>Tipo</b>	
<b>Responsabilidad:</b>		
<b>Nombre:</b>	Insertar_tipo_deporte(Ctipo_deporte \$tipo_deporte)	
	Modificar_tipo_deporte(Ctipo_deporte \$tipo_deporte)	
	Eliminar_tipo_deporte(Ctipo_deporte \$tipo_deporte)	
	Listar_tipo_deportes()	
<b>Descripción:</b>	Controla las operaciones de acceso a la tabla (cl_tipo_deporte).	

Tabla A6.69 Clase de acceso a datos "Ctipo\_deporte\_DAO".

<b>Nombre:</b>	Ctipo_enfrentamiento_DAO	
<b>Tipo de clase:</b>	DAO	
<b>Atributo</b>	<b>Tipo</b>	
<b>Responsabilidad:</b>		
<b>Nombre:</b>	Insertar_tipo_enfrentamiento(Cenfrentamiento \$enfrentamiento)	
	Modificar_tipo_enfrentamiento(Cenfrentamiento \$enfrentamiento)	
	Eliminar_tipo_enfrentamiento(Cenfrentamiento \$enfrentamiento)	
	Listar_tipos_enfrentamientos()	
<b>Descripción:</b>	Controla las operaciones de acceso a la tabla (cl_tipo_enfrentamiento).	

Tabla A6.70 Clase de acceso a datos "Ctipo\_enfrentamiento\_DAO".

<b>Nombre:</b>	Ctipo_lesion_DAO
----------------	------------------

<b>Tipo de clase:</b>	DAO
<b>Atributo</b>	<b>Tipo</b>
<b>Responsabilidad:</b>	
<b>Nombre:</b>	Insertar_tipo_lesion(Ctipo_lesion \$lesion)
	Modificar_tipo_lesion(Ctipo_lesion \$lesion)
	Eliminar_tipo_lesion(Ctipo_lesion \$lesion)
	Listar_tipos_lesiones()
<b>Descripción:</b>	Controla las operaciones de acceso a la tabla (cl_tipo_lesion).

Tabla A6.71 Clase de acceso a datos "Ctipo\_lesion\_DAO".

<b>Nombre:</b>	Cunidad_resultado_enf_DAO
<b>Tipo de clase:</b>	DAO
<b>Atributo</b>	<b>Tipo</b>
<b>Responsabilidad:</b>	
<b>Nombre:</b>	Insertar_unidad(Cunidad_resultado_enf \$unidad)
	Modificar_unidad(Cunidad_resultado_enf \$unidad)
	Eliminar_unidad(Cunidad_resultado_enf \$unidad)
	Listar_unidad()
<b>Descripción:</b>	Controla las operaciones de acceso a la tabla (no_unidad_resultado).

Tabla A6.72 Clase de acceso a datos "Cunidad\_resultado\_enf\_DAO".

<b>Nombre:</b>	Cusuario_DAO
<b>Tipo de clase:</b>	DAO
<b>Atributo</b>	<b>Tipo</b>
<b>Responsabilidad:</b>	
<b>Nombre:</b>	Insertar_rol_usuario(Cusuario \$usuario)
	Existe_rol_usuario(Cusuario \$usuario)
	Eliminar_rol_usuario(Cusuario \$usuario)
	Listar_rol_usuario(Cusuario \$usuario)
	Listar_usuarios()

	Existe_usuario(Cusuario \$usuario)
<b>Descripción:</b>	Controla las operaciones de acceso a la tabla (tb_usuario).

Tabla A6.73 Clase de acceso a datos "Cusuario\_DAO".

<b>Nombre:</b>	Cutensilio_DAO	
<b>Tipo de clase:</b>	DAO	
<b>Atributo</b>	<b>Tipo</b>	
<b>Responsabilidad:</b>		
<b>Nombre:</b>	Insertar_Utensilio(Cutensilio \$utensilio)	
	Existe_Utensilio(Cutensilio \$utensilio)	
	Modificar_Utensilio(Cutensilio \$utensilio)	
	Eliminar_Utensilio(Cutensilio \$utensilio)	
	Listado_Utensilios(Cdeporte \$deporte)	
	Listado_UTE()	
	Detalle_Utensilio(Cutensilio \$utensilio)	
<b>Descripción:</b>	Controla las operaciones de acceso a la tabla (tb_utensilio).	

Tabla A6.74 Clase de acceso a datos "Cutensilio\_DAO".

<b>Nombre:</b>	Cvisita_DAO	
<b>Tipo de clase:</b>	DAO	
<b>Atributo</b>	<b>Tipo</b>	
<b>Responsabilidad:</b>		
<b>Nombre:</b>	Registrar_Visita(Cvisita \$visita)	
	Listado_Visitas()	
	Eliminar_Visita(Cvisita \$visita)	
	Detalle_Visita(Cvisita \$visita)	
	Registrar_Salida(Cvisita \$visita)	
	Listar_Visita_Fecha(Cvisita \$visita)	
	Listar_Visita_Usuarios_frecuentes()	
	Verificar_Visita(Cvisita \$visita)	
<b>Descripción:</b>	Controla las operaciones de acceso a la tabla (tb_visita).	

Tabla A6.75 Clase de acceso a datos "Cvisita\_DAO".

### 6.3 Clase Fachada.

<b>Nombre:</b>	CFachada	
<b>Tipo de clase:</b>	Interfaz (PHP, servidor)	
<b>Atributo</b>	<b>Tipo</b>	
gestion_deporte	Cdeporte_Manager	
gestion_evento	Cevento_Manager	
gestion_persona	Cpersona_Manager	
gestion_prestamo	Cprestamo_Manager	
gestion_utensilio	Cutensilio_Manager	
gestion_categoria_competencia	Ccategoria_competencia_Manager	
gestion_fase_competencia	Cfase_competencia_Manager	
gestion_indicador	Cindicador_Manager	
gestion_sistema_competencia	Csistema_competencia_Manager	
gestion_tipo_enfrentamiento	Ctipo_enfrentamiento_Manager	
gestion_unidad_resultado	Cunidad_resultado_enf_Manager	
gestion_color_piel	Ccolor_piel_Manager	
gestion_localizacion_lesion	Clocalizacion_lesion_Manager	
gestion_tipo_lesion	Ctipo_lesion_Manager	
gestion_atleta	Catleta_Manager	
gestion_curso	Ccurso_Manager	
gestion_comision	Ccomision_Manager	
gestion_comision_organizadora	Ccomision_organizadora_Manager	
gestion_competencia	Ccompetencia_Manager	
gestion_enfrentamiento	Cenfrentamiento_Manager	
gestion_enfrentamiento_duo_atleta	Cenfrentamiento_duo_atleta_Manager	
gestion_enfrentamiento_duo_equipo	Cenfrentamiento_duo_equipo_Manager	
gestion_enfrentamiento_multiple_atleta	Cenfrentamiento_multiple_atleta_Manager	
gestion_equipo	Cequipo_Manager	
gestion_estudiante	Cestudiante_Manager	

gestion_grupo	Cgrupo_Manager
gestion_lesion	Clesion_Manager
gestion_nivel_general_est	Cnivel_general_est_Manager
gestion_nivel_general_grupo	Cnivel_general_grupo_Manager
gestion_nivel_grupo_indicador	Cnivel_grupo_indicador_Manager
gestion_profesor_ef	Cprofesor_ef_Manager
gestion_representacion	Crepresentacion_Manager
gestion_resultado	Cresultado_Manager
gestion_usuario	Cusuario_Manager
gestion_roll	Croll_Manager
gestion_visita	Cvisita_Manager

**Responsabilidad:**

<b>Nombre:</b>	Insertar_Deporte
	Existe_Deporte
	Listado_Deportes
	Eliminar_Deporte
	Detalle_Deporte
	Modificar_Deporte
	Insertar_Evento
	Existe_Evento
	Listado_Eventos
	Eliminar_Evento
	Detalle_Evento
	Modificar_Evento
	Insertar_Persona
	Existe_Persona
	Listado_Persona
	Eliminar_Personal
Datos_Persona	

Persona_NAF
Modificar_Persona
Insertar_Prestamo
Listado_Prestamos_Pendientes
Listado_Prestamos_Fecha
Listado_Prestamos_Usuario
Listado_Prestamos_Pendientes_Usuario
Listado_Prestamos_Completados
Detalle_Prestamo
Modificar_Prestamo
Prestamo_Cant_UTE
Terminar_prestamo
Listado_Utensilios
Listado_UTE
Insertar_categoria
Modificar_categoria
Eliminar_categoria
Listar_categoria
Insertar_fase
Modificar_fase
Eliminar_fase
Insertar_indicador
Modificar_indicador
Eliminar_indicador
Listar_indicador
Insertar_sistema
Modificar_sistema
Eliminar_sistema
Listar_sistema

	Insertar_tipo_enfrentamiento
	Modificar_tipo_enfrentamiento
	Eliminar_tipo_enfrentamiento
	Listar_tipos_enfrentamientos
	Insertar_unidad
	Modificar_unidad
	Eliminar_unidad
	Listar_unidad
	Insertar_color
	Modificar_color
	Eliminar_color
	Listar_color
	Insertar_lugar_lesion
	Modificar_lugar_lesion
	Eliminar_lugar_lesion
	Listar_localizaciones_lesion
	Insertar_tipo_lesion
	Modificar_tipo_lesion
	Eliminar_tipo_lesion
	Listar_tipos_lesiones
	Insertar_atleta
	Modificar_atleta
	Eliminar_atleta
	Listar_atletas
	Listar_atleta_deporte
	Listar_atleta_equipo
	Listar_atleta_bajo_peso
	Listar_atleta_competencia
	Insertar_tipo_deporte

Modificar_tipo_deporte
Eliminar_tipo_deporte
Listar_tipo_deporte
Insertar_curso
Modificar_curso
Eliminar_curso
Listar_cursos
Insertar_comision
Modificar_comision
Eliminar_comision
Listar_comisiones
Insertar_comision_organizadora
Modificar_comision_organizadora
Eliminar_comision_organizadora
Listar_comisiones_organizadoras
Insertar_competencia
Modificar_competencia
Eliminar_competencia
Listar_competencias
Insertar_Deporte_Competencia
Insertar_enfrentamiento
Modificar_enfrentamiento
Eliminar_enfrentamiento
Insertar_enfrentamiento_duo_atleta
Modificar_enfrentamiento_duo_atleta
Insertar_enfrentamiento_duo_equipo
Modificar_enfrentamiento_duo_equipo
Eliminar_enfrentamiento_duo_equipo
Insertar_enfrentamiento_multiple_atleta

	Modificar_enfrentamiento_multiple_atleta
	Eliminar_enfrentamiento_multiple_atleta
	Insertar_equipo
	Modificar_equipo
	Eliminar_equipo
	Listar_equipos
	Listar_equipo_deporte
	Listar_equipo_competencia
	Insertar_estudiante
	Modificar_estudiante
	Eliminar_estudiante
	Listar_estudiantes
	Listar_estudiante_grupo
	Insertar_grupo
	Modificar_grupo
	Eliminar_grupo
	Listar_grupos
	Insertar_lesion
	Modificar_lesion
	Eliminar_lesion
	Listar_lesiones
	Insertar_nivel_general_est
	Modificar_nivel_general_est
	Listar_nivel_general_est
	Insertar_nivel_general_grupo
	Modificar_nivel_general_grupo
	Listar_nivel_general_grupo
	Insertar_nivel_grupo_indicador
	Modificar_nivel_grupo_indicador

	Listar_nivel_grupo_indicador
	Insertar_profesor_ef
	Modificar_profesor_ef
	Eliminar_profesor_ef
	Listar_profesor_ef
	Listar_profesor_ef_grupo
	Insertar_representacion
	Modificar_representacion
	Eliminar_representacion
	Listar_representaciones
	Listar_representacion_competencia
	Insertar_resultado
	Modificar_resultado
	Eliminar_resultado
	Listar_resultados
	Insertar_usuario
	Modificar_usuario
	Eliminar_usuario
	Listar_usuarios
	Insertar_roll
	Modificar_roll
	Eliminar_roll
	Registrar_Visita
	Registrar_Salida
	Eliminar_visita
	Listado_Visitas
	Detalle_Visita
<b>Descripción:</b>	Clase encargada de establecer la comunicación entre la interfaz del sistema y las clase controladoras.

Tabla A6.76 Clase de acceso a datos “CFachada”.

**6.4 Clases controladoras.**

<b>Nombre:</b>	Catleta_Manager
<b>Tipo de clase:</b>	Controladora
<b>Responsabilidad:</b>	
<b>Nombre:</b>	Insertar_atleta(\$asolapin)
	Modificar_atleta(\$asolapin)
	Eliminar_atleta(\$asolapin)
	Listar_atletas()
	Listar_atleta_equipo()
	Listar_atleta_bajo_peso()
	Listar_atleta_competencia()
<b>Descripción:</b>	Controla la lógica de todo el proceso de las actividades de los atletas del sistema.

Tabla A6.77 Clase controladora “Catleta\_Manager”

<b>Nombre:</b>	Ccategoria_competencia_Manager
<b>Tipo de clase:</b>	Controladora
<b>Responsabilidad:</b>	
<b>Nombre:</b>	Insertar_categoria(\$anombre)
	Modificar_categoria(\$anombre)
	Eliminar_categoria(\$anombre)
	Listar_categoria()
<b>Descripción:</b>	Controla la lógica de todo el proceso de gestión de las categorías de las competencias del sistema.

Tabla A6.78 Clase controladora “Ccategoria\_competencia\_Manager”

<b>Nombre:</b>	Ccompetencia_Manager
----------------	----------------------

<b>Tipo de clase:</b>	Controladora
<b>Responsabilidad:</b>	
<b>Nombre:</b>	Insertar_competencia(\$nombre, \$sede, \$acant_arbitros, \$acant_medicos, \$afecha_inicio, \$afecha_fin, \$aid_categoria, \$aid_curso, \$adeportes)
	Modificar_competencia(\$nombre, \$sede, \$acant_arbitros, \$acant_medicos, \$afecha_inicio, \$afecha_fin, \$aid_categoria, \$aid_curso, \$adeportes)
	Eliminar_competencia(\$nombre)
	Listar_competencias()
<b>Descripción:</b>	Controla la lógica de todo el proceso de gestión de las competencias del sistema.

Tabla A6.79 Clase controladora “Ccompetencia\_Manager”

<b>Nombre:</b>	Cdeporte_Manager
<b>Tipo de clase:</b>	Controladora
<b>Responsabilidad:</b>	
<b>Nombre:</b>	Insertar_Deporte(\$aSGL_deporte, \$nombre_deporte, \$at_deporte, \$asexo)
	Existe_Deporte(\$nombre_deporte)
	Listado_Deportes()
	Eliminar_Deporte(\$nombre_deporte)
	Detalle_Deporte(\$nombre_deporte)
	Modificar_Deporte(\$aSGL_deporte, \$nombre_deporte, \$at_deporte, \$asexo)
<b>Descripción:</b>	Controla la lógica de todo el proceso de gestión de los deportes del sistema.

Tabla A6.80 Clase controladora “Cdeporte\_Manager”

<b>Nombre:</b>	Ccomision_Manager
<b>Tipo de clase:</b>	Controladora
<b>Responsabilidad:</b>	
<b>Nombre:</b>	Insertar_comision(\$nombre, \$asolapin, \$aid_comision_organizadora)
	Modificar_comision(\$nombre)
	Eliminar_comision(\$nombre)

	Listar_comisiones()
<b>Descripción:</b>	Controla la lógica de todo el proceso de gestión de las comisiones de las competencias del sistema.

**Tabla A6.81 Clase controladora “Ccomision\_Manager”**

<b>Nombre:</b>	Ccomision_organizadora_Manager
<b>Tipo de clase:</b>	Controladora
<b>Responsabilidad:</b>	
<b>Nombre:</b>	Insertar_comision_organizadora(\$asolapin,\$aid_competencia)
	Modificar_comision_organizadora(\$asolapin,\$aid_competencia)
	Eliminar_comision_organizadora(\$aid_comision_organizadora)
	Listar_comisiones_organizadoras()
<b>Descripción:</b>	Controla la lógica de todo el proceso de gestión de las comisiones organizadoras de las competencias del sistema.

**Tabla A6.82 Clase controladora “Ccomision\_organizadora\_Manager”**

<b>Nombre:</b>	Cenfrentamiento_duo_atleta_Manager
<b>Tipo de clase:</b>	Controladora
<b>Responsabilidad:</b>	
<b>Nombre:</b>	Insertar_enfrentamiento_duo_atleta(\$aid_enfrentamiento,\$asolapin,\$aid_equipo,\$agano,\$aresultado)
	Modificar_enfrentamiento_duo_atleta(\$aid_enfrentamiento,\$asolapin,\$aid_equipo,\$agano,\$aresultado)
	Eliminar_enfrentamiento_duo_atleta(\$aid_enfrentamiento)
<b>Descripción:</b>	Controla la lógica de todo el proceso de gestión de los enfrentamientos individuales de los atletas.

**Tabla A6.83 Clase controladora “Cenfrentamiento\_duo\_atleta\_Manager”**

<b>Nombre:</b>	Cenfrentamiento_duo_equipo_Manager
<b>Tipo de clase:</b>	Controladora

<b>Responsabilidad:</b>	
<b>Nombre:</b>	Insertar_enfrentamiento_duo_equipo(\$aid_enfrentamiento,\$aid_equipo,\$agano,\$aresultado)
	Modificar_enfrentamiento_duo_equipo(\$aid_enfrentamiento,\$aid_equipo,\$agano,\$aresultado)
	Eliminar_enfrentamiento_duo_equipo(\$aid_enfrentamiento)
<b>Descripción:</b>	Controla la lógica de todo el proceso de gestión de los enfrentamientos de los equipos.

**Tabla A6.84 Clase controladora “Cenfrentamiento\_duo\_equipo\_Manager”**

<b>Nombre:</b>	Cenfrentamiento_Manager
<b>Tipo de clase:</b>	Controladora
<b>Responsabilidad:</b>	
<b>Nombre:</b>	Insertar_enfrentamiento(\$afecha, \$acant_arbitros, \$acant_medicos, \$ahora_inicio, \$ahora_fin, \$alugar, \$aid_fase, \$aid_evento, \$aidsexo, \$aid_tipo_enfrentamiento)
	Modificar_enfrentamiento(\$afecha, \$acant_arbitros, \$acant_medicos, \$ahora_inicio, \$ahora_fin, \$alugar, \$aid_fase, \$aid_evento, \$aidsexo, \$aid_tipo_enfrentamiento)
	Eliminar_enfrentamiento(\$aid_enfrentamiento)
<b>Descripción:</b>	Controla la lógica de todo el proceso de gestión de los enfrentamientos.

**Tabla A6.85 Clase controladora “Cenfrentamiento\_Manager”**

<b>Nombre:</b>	Cenfrentamiento_multiple_atleta_Manager
<b>Tipo de clase:</b>	Controladora
<b>Responsabilidad:</b>	
<b>Nombre:</b>	Insertar_enfrentamiento_multiple_atleta(\$aid_enfrentamiento,\$asolapin,\$aid_equipo,\$aresultado)
	Modificar_enfrentamiento_multiple_atleta(\$aid_enfrentamiento,\$asolapin,\$aid_equipo,\$aresultado)
	Eliminar_enfrentamiento_multiple_atleta(\$aid_enfrentamiento)

<b>Descripción:</b>	Controla la lógica de todo el proceso de gestión de los enfrentamientos individuales donde participan más de 2 personas.
---------------------	--

**Tabla A6.86 Clase controladora “Cenfrentamiento\_multiple\_atleta\_Manager”**

<b>Nombre:</b>	Cequipo_Manager
<b>Tipo de clase:</b>	Controladora
<b>Responsabilidad:</b>	
<b>Nombre:</b>	Insertar_equipo(\$aSGLdeporte,\$aid_representacion,\$aid_sexo,\$asolapin)
	Modificar_equipo(\$aSGLdeporte,\$aid_representacion,\$aid_sexo,\$asolapin)
	Eliminar_equipo(\$aid_equipo)
	Listar_equipos()
	Listar_equipo_deporte()
	Listar_equipo_competencia()
<b>Descripción:</b>	Controla la lógica de todo el proceso de gestión de los equipos del sistema.

**Tabla A6.87 Clase controladora “Cequipo\_Manager”**

<b>Nombre:</b>	Cestudiante_Manager
<b>Tipo de clase:</b>	Controladora
<b>Responsabilidad:</b>	
<b>Nombre:</b>	Insertar_estudiante(\$asolapin,\$aid_grupo)
	Modificar_estudiante(\$asolapin,\$aid_grupo)
	Eliminar_estudiante(\$asolapin,\$aid_grupo)
	Listar_estudiantes()
	Listar_estudiante_grupo(\$aid_grupo)
<b>Descripción:</b>	Controla la lógica de todo el proceso de gestión de los estudiantes del sistema.

**Tabla A6.88 Clase controladora “Cestudiante\_Manager”**

<b>Nombre:</b>	Cevento_Manager
<b>Tipo de clase:</b>	Controladora

<b>Responsabilidad:</b>	
<b>Nombre:</b>	Insertar_Evento(\$nombre, \$colectivo, \$SGLdeporte, \$aidsexo, \$aid_sistema, \$aid_unidad_resultado)
	Existe_Evento(\$nombre)
	Listado_Eventos()
	Eliminar_Evento(\$nombre)
	Detalle_Evento(\$nombre)
	Modificar_Evento(\$nombre, \$colectivo, \$SGLdeporte, \$aidsexo, \$aid_sistema, \$aid_unidad_resultado)
<b>Descripción:</b>	Controla la lógica de todo el proceso de gestión de los eventos del sistema.

Tabla A6.89 Clase controladora “Cevento\_Manager”

<b>Nombre:</b>	Cfase_competencia_Manager
<b>Tipo de clase:</b>	Controladora
<b>Responsabilidad:</b>	
<b>Nombre:</b>	Insertar_fase(\$nombre)
	Modificar_fase(\$nombre)
	Eliminar_fase(\$nombre)
<b>Descripción:</b>	Controla la lógica de todo el proceso de gestión de las fases del sistema.

Tabla A6.90 Clase controladora “Cfase\_competencia\_Manager”

<b>Nombre:</b>	Cgrupo_Manager
<b>Tipo de clase:</b>	Controladora
<b>Responsabilidad:</b>	
<b>Nombre:</b>	Insertar_grupo(\$numero_grupo,\$aanno,\$afacultad)
	Modificar_grupo(\$numero_grupo,\$aanno,\$afacultad)
	Eliminar_grupo(\$id_grupo)
	Listar_grupos()
	Listar_grupo_profesor_ef(\$asolapin)

<b>Descripción:</b>	Controla la lógica de todo el proceso de gestión de los grupos del sistema.
---------------------	---

**Tabla A6.91 Clase controladora “Cgrupo\_Manager”**

<b>Nombre:</b>	Cindicador_Manager
<b>Tipo de clase:</b>	Controladora
<b>Responsabilidad:</b>	
<b>Nombre:</b>	Insertar_indicador(\$nombre)
	Modificar_indicador(\$nombre)
	Eliminar_indicador(\$nombre)
	Listar_indicador()
<b>Descripción:</b>	Controla la lógica de todo el proceso de gestión de los grupos del sistema.

**Tabla A6.92 Clase controladora “Cindicador\_Manager”**

<b>Nombre:</b>	Clesion_Manager
<b>Tipo de clase:</b>	Controladora
<b>Responsabilidad:</b>	
<b>Nombre:</b>	Insertar_lesion(\$aid_tipo_lesion,\$asolapin,\$aid_equipo,\$aid_localizacion_lesion)
	Modificar_lesion(\$aid_tipo_lesion,\$asolapin,\$aid_equipo,\$aid_localizacion_lesion)
	Eliminar_lesion(\$id_lesion)
	Listar_lesiones()
<b>Descripción:</b>	Controla la lógica de todo el proceso de gestión de las lesiones de los atletas del sistema.

**Tabla A6.93 Clase controladora “Clesion\_Manager”**

<b>Nombre:</b>	Clocalizacion_lesion_Manager
<b>Tipo de clase:</b>	Controladora
<b>Responsabilidad:</b>	
<b>Nombre:</b>	Insertar_lugar_lesion(\$alugar)
	Modificar_lugar_lesion(\$alugar)

	Eliminar_lugar_lesion(\$alugar)
	Listar_localizaciones_lesion()
<b>Descripción:</b>	Controla la lógica de todo el proceso de gestión de la localización de las lesiones de los atletas del sistema.

**Tabla A6.94 Clase controladora “Clocalizacion\_lesion\_Manager”**

<b>Nombre:</b>	Cnivel_general_est_Manager
<b>Tipo de clase:</b>	Controladora
<b>Responsabilidad:</b>	
<b>Nombre:</b>	Insertar_nivel_general_est(\$afecha,\$aid_prueba,\$asolapin,\$anivel,\$aid_curso)
	Modificar_nivel_general_est(\$afecha,\$aid_prueba,\$asolapin,\$anivel,\$aid_curso)
	Listar_nivel_general_est()
<b>Descripción:</b>	Controla la lógica de todo el proceso de gestión de la nivelación general de los estudiantes en la pruebas de eficiencia física.

**Tabla A6.95 Clase controladora “Cnivel\_general\_est\_Manager”**

<b>Nombre:</b>	Cnormativa_Manager
<b>Tipo de clase:</b>	Controladora
<b>Responsabilidad:</b>	
<b>Nombre:</b>	Insertar_normativa (\$aid_indicador, \$aedad_I, \$aedad_F, \$aidsexo,\$arango_I, \$arango_F, \$anivel)
	Modificar_normativa(\$aid_indicador,\$aedad_I,\$aedad_F,\$aidsexo,\$arango_I, \$arango_F,\$anivel)
	Eliminar_normativa(\$aid_indicador,\$aedad_I,\$aedad_F,\$aidsexo,\$arango_I, \$arango_F,\$anivel)
	Listar_normativas()
	Buscar_Nivel_de_normativa(\$id_indicador,\$sexo_atleta,\$edad_atleta,\$are resultado)
	Buscar_Nivel_general_estudiante(\$are resultado)
<b>Descripción:</b>	Controla la lógica de todo el proceso de gestión de los resultados y las normativas de

	los estudiantes.
--	------------------

**Tabla A6.96 Clase controladora “Cnormativa\_Manager”**

<b>Nombre:</b>	Cpersona_Manager
<b>Tipo de clase:</b>	Controladora
<b>Responsabilidad:</b>	
<b>Nombre:</b>	Insertar_Persona(\$asolapin, \$adireccion_particular, \$aid_ocupacion, \$anombre, \$apapellido, \$asapellido, \$aedad, \$ausuario, \$aid_municipio, \$aid_provincia, \$aid_sexo)
	Existe_Persona(\$asolapin)
	Eliminar_Personal(\$asolapin, \$anombre)
	Datos_Persona(\$anombre)
	Persona_NAF(\$asolapin)
	Modificar_Persona(\$asolapin, \$adireccion_particular, \$aid_ocupacion, \$anombre, \$apapellido, \$asapellido, \$aedad, \$ausuario, \$aid_municipio, \$aid_provincia, \$aid_sexo)
	Listado_Persona()
<b>Descripción:</b>	Controla la lógica de todo el proceso de gestión de las personas que van a interactuar con el sistema.

**Tabla A6.97 Clase controladora “Cprestamo\_Manager”**

<b>Nombre:</b>	Cprestamo_Manager
<b>Tipo de clase:</b>	Controladora
<b>Responsabilidad:</b>	
<b>Nombre:</b>	Insertar_Prestamo(\$aid_utensilio, \$asolapin, \$afecha_inicio, \$ahora_inicio, \$aestado_inicial, \$acantidad)
	Listado_Prestamos_Pendientes()
	Listado_Prestamos_Fecha(\$afecha)
	Listado_Prestamos_Usuario(\$asolapin)

	Listado_Prestamos_Pendientes_Usuario(\$asolapin)
	Listado_Prestamos_Completados()
	Detalle_Prestamo(\$afecha, \$asolapin )
	Modificar_Prestamo(\$aid_prestamo, \$aid_utensilio, \$aestado_inicial, \$acantidad)
	Prestamo_Cant_UTE(\$aid_utensilio)
	Terminar_prestamo(\$aid_prestamo,\$afecha_fin,\$ahora_fin,\$aestado_final,\$aentregado)
<b>Descripción:</b>	Controla la lógica de todo el proceso de gestión de los préstamos del gimnasio.

**Tabla A6.98 Clase controladora “Cprestamo\_Manager”**

<b>Nombre:</b>	Cprofesor_ef_Manager
<b>Tipo de clase:</b>	Controladora
<b>Responsabilidad:</b>	
<b>Nombre:</b>	Insertar_profesor_ef(\$asolapin,\$aid_grupo)
	Modificar_profesor_ef(\$asolapin,\$aid_grupo)
	Eliminar_profesor_ef(\$asolapin,\$aid_grupo)
	Listar_profesor_ef()
	Listar_profesor_ef_grupo(\$aid_grupo)
<b>Descripción:</b>	Controla la lógica de todo el proceso de gestión de los profesores de educación física del gimnasio.

**Tabla A6.99 Clase controladora “Cprofesor \_ef\_Manager”**

<b>Nombre:</b>	Cusuario_Manager
<b>Tipo de clase:</b>	Controladora
<b>Responsabilidad:</b>	
<b>Nombre:</b>	Insertar_rol_usuario(\$asolapin,\$aroles)
	Existe_rol_usuario(\$asolapin,\$aroles)
	Listar_rol_usuario(\$asolapin)
	Eliminar_rol_usuario(\$asolapin,\$aroles)

	Listar_usuarios()
	Existe_usuario(\$ausuario,\$apass)
<b>Descripción:</b>	Controla la lógica de todo el proceso de gestión de los usuarios del sistema.

**Tabla A6.100 Clase controladora "Cusuario\_Manager"**

<b>Nombre:</b>	Cutensilio_Manager
<b>Tipo de clase:</b>	Controladora
<b>Responsabilidad:</b>	
<b>Nombre:</b>	Listado_Utensilios(\$anombre_deporte)
	Listado_UTE()
	Insertar_Utensilio()
	Eliminar_Utensilio()
	Modificar_Utensilio()
	Detalle_Utensilio(\$aid_utensilio)
<b>Descripción:</b>	Controla la lógica de todo el proceso de gestión de los utensilios del sistema.

**Tabla A6.101 Clase controladora "Cutensilio\_Manager"**



**Anexo # 8: Descripción textual de las tablas de la base de datos.**

<b>Nombre:</b>	cl_categoria_competencia	
<b>Descripción:</b>	Esta tabla almacena las categorías de las competencias.	
<b>Atributo</b>	<b>Tipo</b>	<b>Descripción</b>
id_categoria	Int	La llave primaria de la tabla
nombre	Varchar	Es un atributo que tiene el nombre de la categoría.

**Tabla A8.1 Descripción textual de la tabla “cl\_categoria\_competencia”**

<b>Nombre:</b>	cl_tipo_deporte	
<b>Descripción:</b>	Esta tabla almacena los tipos de deportes.	
<b>Atributo</b>	<b>Tipo</b>	<b>Descripción</b>
id_tipo	Int	La llave primaria de la tabla
nombre	Varchar	Es un atributo que tiene el nombre del tipo de deporte.

**Tabla A8.2 Descripción textual de la tabla “cl\_tipo\_deporte”**

<b>Nombre:</b>	cl_tipo_enfrentamiento	
<b>Descripción:</b>	Esta tabla almacena los tipos de enfrentamientos que puede haber en una competencia.	
<b>Atributo</b>	<b>Tipo</b>	<b>Descripción</b>
id_tipo	Int	La llave primaria de la tabla
nombre	Varchar	Es un atributo que tiene el nombre del tipo de enfrentamiento.

**Tabla A8.3 Descripción textual de la tabla “cl\_tipo\_enfrentamiento”**

<b>Nombre:</b>	cl_tipo_lesion	
<b>Descripción:</b>	Esta tabla almacena los tipos de lesiones que puede tener un atleta.	
<b>Atributo</b>	<b>Tipo</b>	<b>Descripción</b>

id_tipo_lesion	Int	La llave primaria de la tabla
lesion	Varchar	Es un atributo que tiene el nombre del tipo de lesión.

Tabla A8.4 Descripción textual de la tabla “cl\_tipo\_lesion”

<b>Nombre:</b>	no_fase	
<b>Descripción:</b>	Esta tabla almacena las fases de los sistemas de competencias.	
<b>Atributo</b>	<b>Tipo</b>	<b>Descripción</b>
id_fase	Int	La llave primaria de la tabla
nombre	Varchar	Es un atributo que tiene el nombre de la fase.

Tabla A8.5 Descripción textual de la tabla “no\_fase”

<b>Nombre:</b>	tb_atleta	
<b>Descripción:</b>	Esta tabla almacena a los atletas y algunos datos adicionales de ellos	
<b>Atributo</b>	<b>Tipo</b>	<b>Descripción</b>
solapin	Varchar	Es un atributo que hace referencia a la tabla (tb_persona)
id_equipo	Int	Es un atributo que hace referencia a la tabla (tb_equipo).
calzado	Varchar	Es un atributo que tiene el tamaño del pie.
estatura	Float	Es un atributo que tiene la estatura.
peso	Float	Es un atributo que tiene el peso.
id_color_piel	Int	Es un atributo que hace referencia a la tabla (no_color_piel).

Tabla A8.6 Descripción textual de la tabla “tb\_atleta”

<b>Nombre:</b>	no_indicador_ef	
<b>Descripción:</b>	Esta tabla almacena los indicadores que se tienen en cuenta para hacer las pruebas de eficiencia física.	
<b>Atributo</b>	<b>Tipo</b>	<b>Descripción</b>
id_indicador	Int	La llave primaria de la tabla
nombreindicador	Varchar	Es un atributo que tiene el nombre del indicador.

Tabla A8.7 Descripción textual de la tabla “no\_indicador\_ef”

<b>Nombre:</b>	no_localizacion_lesion	
<b>Descripción:</b>	Esta tabla almacena las posibles localizaciones de las lesiones que puede sufrir un atleta.	
<b>Atributo</b>	<b>Tipo</b>	<b>Descripción</b>
id_localizacion_lesion	Int	La llave primaria de la tabla
localizacion	Varchar	Es un atributo que tiene el nombre de la localización de la lesión.

Tabla A8.8 Descripción textual de la tabla “no\_localizacion\_lesion”

<b>Nombre:</b>	no_rol	
<b>Descripción:</b>	Esta tabla almacena los posibles roles que puede tener un usuario.	
<b>Atributo</b>	<b>Tipo</b>	<b>Descripción</b>
id_rol	Int	La llave primaria de la tabla
rol	Varchar	Es un atributo que tiene el nombre del rol.

Tabla A8.9 Descripción textual de la tabla “no\_rol”

<b>Nombre:</b>	no_sexo	
<b>Descripción:</b>	Esta tabla almacena los sexos de los eventos, incluyendo mixto.	
<b>Atributo</b>	<b>Tipo</b>	<b>Descripción</b>
id_sexo	Int	La llave primaria de la tabla
sexo	Varchar	Es un atributo que tiene el sexo.

Tabla A8.10 Descripción textual de la tabla “no\_sexo”

<b>Nombre:</b>	no_sistema_competencia	
<b>Descripción:</b>	Esta tabla almacena el nombre de los sistemas de competencias que se pueden tener en cuenta para un evento.	
<b>Atributo</b>	<b>Tipo</b>	<b>Descripción</b>
id_sistema	Int	La llave primaria de la tabla

nombre	Varchar	Es un atributo que tiene el nombre.
--------	---------	-------------------------------------

Tabla A8.11 Descripción textual de la tabla “no\_sistema\_competencia”

<b>Nombre:</b>	no_unidad_resultado	
<b>Descripción:</b>	Esta tabla almacena las unidades en que se pueden dar los resultados de cada deporte.	
<b>Atributo</b>	<b>Tipo</b>	<b>Descripción</b>
id_unidad_resultado	Int	La llave primaria de la tabla
unidad_resultado	Varchar	Es un atributo que tiene el nombre de la unidad.

Tabla A8.12 Descripción textual de la tabla “no\_unidad\_resultado”

<b>Nombre:</b>	tb_comision	
<b>Descripción:</b>	Esta tabla almacena algunos de los datos de las comisiones que se crean para cada competencia.	
<b>Atributo</b>	<b>Tipo</b>	<b>Descripción</b>
id_comision	Int	La llave primaria de la tabla
nombre	Varchar	Es un atributo que tiene el nombre de la comisión.
solapin	Varchar	Es un atributo que hace referencia a la tabla (tb_persona), este solapin seria el del jefe de la comisión.
id_comision_organizadora	Int	Es un atributo que hace referencia a la tabla (tb_comision_organizadora).

Tabla A8.13 Descripción textual de la tabla “tb\_comision”

<b>Nombre:</b>	tb_comision_organizadora	
<b>Descripción:</b>	Esta tabla almacena los datos de las comisiones organizadoras que se crean por cada comisión.	
<b>Atributo</b>	<b>Tipo</b>	<b>Descripción</b>
id_comision_organizadora	Int	La llave primaria de la tabla
solapin	Varchar	Es un atributo que hace referencia a la tabla (tb_persona).
id_competencia	Int	Es un atributo que hace referencia a la tabla

		(tb_competencia).
--	--	-------------------

Tabla A8.14 Descripción textual de la tabla “tb\_comision\_organizadora”

<b>Nombre:</b>	tb_comision_tecnica	
<b>Descripción:</b>	Esta tabla almacena los datos de las comisiones técnicas que se crean por cada comisión.	
<b>Atributo</b>	<b>Tipo</b>	<b>Descripción</b>
solapin	Int	Es un atributo que hace referencia a la tabla (tb_persona).
id_competencia	Int	Es un atributo que hace referencia a la tabla (tb_competencia).
SGLdeporte	Varchar	Es un atributo que hace referencia a la tabla (tb_deporte).

Tabla A8.15 Descripción textual de la tabla “tb\_comision\_tecnica”

<b>Nombre:</b>	tb_competencia	
<b>Descripción:</b>	Esta tabla almacena algunos de los datos de las competencias.	
<b>Atributo</b>	<b>Tipo</b>	<b>Descripción</b>
id_competencia	Int	La llave primaria de la tabla.
cantidad_arbitros	Int	Es un atributo que tiene cantidad de árbitros que van a participar en la competencia.
cantidad_medicos	Int	Es un atributo que tiene cantidad de médicos que van a participar en la competencia.
nombre	Varchar	Es un atributo que tiene el nombre de la competencia.
sede	Varchar	Es un atributo que tiene la sede.
fechainicio	Date	Es un atributo que tiene la fecha de inicio de la competencia.
fechafin	Date	Es un atributo que tiene la fecha de clausura de la competencia.
id_categoria	Int	Es un atributo que hace referencia a la tabla (cl_categoria_competencia).

Tabla A8.16 Descripción textual de la tabla “tb\_competencia”

<b>Nombre:</b>	tb_deporte	
<b>Descripción:</b>	Esta tabla almacena algunos de los datos de las competencias.	
<b>Atributo</b>	<b>Tipo</b>	<b>Descripción</b>
SGLdeporte	Varchar	La llave primaria de la tabla.
nombre	Varchar	Es un atributo que tiene el nombre del deporte.
id_sexo	Int	Es un atributo que hace referencia a la tabla (no_sexo).
id_tipo	Int	Es un atributo que hace referencia a la tabla (cl_tipo_deporte).

Tabla A8.17 Descripción textual de la tabla “tb\_deporte”

<b>Nombre:</b>	tb_deporte_competencia	
<b>Descripción:</b>	Esta tabla almacena los deportes que van a tenerse en cuenta en cada competencia.	
<b>Atributo</b>	<b>Tipo</b>	<b>Descripción</b>
SGLdeporte	Varchar	Es un atributo que hace referencia a la tabla (tb_deporte).
id_competencia	Int	Es un atributo que hace referencia a la tabla (tb_competencia).

Tabla A8.18 Descripción textual de la tabla “tb\_deporte\_competencia”

<b>Nombre:</b>	tb_determinar_niveles	
<b>Descripción:</b>	Esta tabla almacena los datos que se van a consultar para determinar los niveles generales de cada estudiante en las pruebas de eficiencia física.	
<b>Atributo</b>	<b>Tipo</b>	<b>Descripción</b>
nivel	Varchar	Es un atributo que tiene el nivel.
variante	Int	Es un atributo que tiene el numero que se que se compara con los resultados del estudiante para obtener su nivel.

Tabla A8.19 Descripción textual de la tabla “tb\_determinar\_niveles”

<b>Nombre:</b>	tb_normativa	
<b>Descripción:</b>	Esta tabla almacena los datos que se van a consultar para determinar los niveles por indicadores de cada estudiante en las pruebas de eficiencia física.	
<b>Atributo</b>	<b>Tipo</b>	<b>Descripción</b>
id_indicador	Int	Es un atributo que hace referencia a la tabla (no_indicador_ef).
edad_I	Int	Es un atributo que tiene la edad inicial dentro del rango a comparar.
edad_F	Int	Es un atributo que tiene la edad final dentro del rango a comparar.
id_sexo	Int	Es un atributo que hace referencia a la tabla (no_sexo).
rango_I	Float	Es un atributo que tiene el valor inicial dentro del rango a comparar.
rango_F	Float	Es un atributo que tiene el valor final dentro del rango a comparar.
nivel	Int	Es un atributo que tiene el nivel que se obtiene después de comprar todos los parámetros.

Tabla A8.20 Descripción textual de la tabla “tb\_normativa”

<b>Nombre:</b>	tb_enfrentamiento	
<b>Descripción:</b>	Esta tabla almacena los datos de los enfrentamientos que se realizan en una competencia.	
<b>Atributo</b>	<b>Tipo</b>	<b>Descripción</b>
id_enf	Int	La llave primaria de la tabla.
fecha	Date	Es un atributo que tiene la fecha.
cantidad_arbitros	Int	Es un atributo que tiene la cantidad de árbitros que estuvieron presentes en el enfrentamiento.

cantidad_medicos	Int	Es un atributo que tiene la cantidad de médicos que estuvieron presentes en el enfrentamiento.
Horainicio	Time	Es un atributo que tiene la hora inicial.
HoraFin	Time	Es un atributo que tiene la hora final.
lugar	Varchar	Es un atributo que tiene el lugar donde se realizo el enfrentamiento.
id_fase	Int	Es un atributo que hace referencia a la tabla (no_fase).
id_evento	Int	Es un atributo que hace referencia a la tabla (tb_evento).
idsexo	Int	Es un atributo que hace referencia a la tabla (nosexo).
id_tipo_enfrentamiento	Int	Es un atributo que hace referencia a la tabla (no_tipo_enfrentamiento).

Tabla A8.21 Descripción textual de la tabla “tb\_enfrentamiento”

<b>Nombre:</b>	tb_enfrentamiento_duo_atleta	
<b>Descripción:</b>	Esta tabla almacena los datos de los enfrentamientos donde participan dos atletas solamente.	
<b>Atributo</b>	<b>Tipo</b>	<b>Descripción</b>
id_enf	Int	Es un atributo que hace referencia a la tabla (tb_enfrentamiento).
solapin	Varchar	Es un atributo que hace referencia a la tabla (tb_persona).
id_equipo	Int	Es un atributo que hace referencia a la tabla (tb_equipo).
gano	Boolean	Es un atributo que tiene el valor de 1 cuando gana y 0 cuando perdió.
resultado	Float	Es un atributo que tiene el resultado del atleta en el enfrentamiento.

Tabla A8.22 Descripción textual de la tabla “tb\_enfrentamiento\_duo\_atleta”

<b>Nombre:</b>	tb_enfrentamiento_duo_equipo
<b>Descripción:</b>	Esta tabla almacena los datos de los enfrentamientos donde participan

	dos equipos.	
Atributo	Tipo	Descripción
id_enf	Int	Es un atributo que hace referencia a la tabla (tb_enfrentamiento).
id_equipo	Int	Es un atributo que hace referencia a la tabla (tb_equipo).
gano	Boolean	Es un atributo que tiene el valor de 1 cuando gano y 0 cuando perdió.
resultado	Varchar	Es un atributo que tiene el resultado del equipo en el enfrentamiento.

Tabla A8.23 Descripción textual de la tabla “tb\_enfrentamiento\_duo\_equipo”

<b>Nombre:</b>	tb_enfrentamiento_multiple_atleta	
<b>Descripción:</b>	Esta tabla almacena los datos de los enfrentamientos donde participan varios atletas solamente.	
Atributo	Tipo	Descripción
id_enf	Int	Es un atributo que hace referencia a la tabla (tb_enfrentamiento).
solapin	Varchar	Es un atributo que hace referencia a la tabla (tb_persona).
id_equipo	Int	Es un atributo que hace referencia a la tabla (tb_equipo).
resultado	Varchar	Es un atributo que tiene el resultado del atleta en el enfrentamiento.

Tabla A8.24 Descripción textual de la tabla “tb\_enfrentamiento\_multiple\_atleta”

<b>Nombre:</b>	tb_equipo	
<b>Descripción:</b>	Esta tabla almacena los datos de los equipos que participan en las competencias.	
Atributo	Tipo	Descripción
id_equipo	Int	La llave primaria de la tabla.

SGLdeporte	Varchar	Es un atributo que hace referencia a la tabla (tb_deporte).
id_representacion	Int	Es un atributo que hace referencia a la tabla (tb_representacion).
id_sexo	Int	Es un atributo que hace referencia a la tabla (no_sexo).
solapin	Varchar	Es un atributo que hace referencia a la tabla (tb_persona).

Tabla A8.25 Descripción textual de la tabla “tb\_equipo”

<b>Nombre:</b>	tb_estudiante	
<b>Descripción:</b>	Esta tabla almacena los grupos con sus estudiantes.	
<b>Atributo</b>	<b>Tipo</b>	<b>Descripción</b>
solapin	Varchar	Es un atributo que hace referencia a la tabla (tb_persona).
id_grupo	Int	Es un atributo que hace referencia a la tabla (tb_grupo).

Tabla A8.26 Descripción textual de la tabla “tb\_estudinate”

<b>Nombre:</b>	tb_evento	
<b>Descripción:</b>	Esta tabla almacena los datos de los eventos.	
<b>Atributo</b>	<b>Tipo</b>	<b>Descripción</b>
id_evento	Int	La llave primaria de la tabla.
nombre	Varchar	Es un atributo que tiene el nombre del evento.
colectivo	Boolean	Es un atributo que tiene valor 1 si es colectivo el evento sino 0.
SGLdeporte	Varchar	Es un atributo que hace referencia a la tabla (tb_deporte).
id_sexo	Int	Es un atributo que hace referencia a la tabla (no_sexo).
id_sistema	Int	Es un atributo que hace referencia a la tabla (no_sistema_competencia).
id_unidad_resultado	Int	Es un atributo que hace referencia a la tabla (no_unidad_resultado).

Tabla A8.27 Descripción textual de la tabla “tb\_evento”

<b>Nombre:</b>	tb_grupo	
<b>Descripción:</b>	Esta tabla almacena los datos de los grupos.	
<b>Atributo</b>	<b>Tipo</b>	<b>Descripción</b>
id_grupo	Int	La llave primaria de la tabla.
numerogrupo	Varchar	Es un atributo que tiene el número del grupo.
anno	Int	Es un atributo que tiene el año que cursa el grupo.
facultad	Varchar	Es un atributo que tiene la facultad a la que pertenece el grupo.
solapin	Varchar	Es un atributo que hace referencia a la tabla (tb_persona).

Tabla A8.28 Descripción textual de la tabla “tb\_grupo”

<b>Nombre:</b>	tb_lesion	
<b>Descripción:</b>	Esta tabla almacena las lesiones de los atletas.	
<b>Atributo</b>	<b>Tipo</b>	<b>Descripción</b>
id_lesion	Int	La llave primaria de la tabla.
id_tipo_lesion	Int	Es un atributo que hace referencia a la tabla (no_tipo_lesion).
solapin	Varchar	Es un atributo que hace referencia a la tabla (tb_persona).
id_equipo	Int	Es un atributo que hace referencia a la tabla (tb_equipo).
id_localizacion_lesion	Int	Es un atributo que hace referencia a la tabla (no_localizacion_lesion).

Tabla A8.29 Descripción textual de la tabla “tb\_lesion”

<b>Nombre:</b>	tb_nivel_general_est	
<b>Descripción:</b>	Esta tabla almacena los niveles generales de los estudiantes que hacen pruebas de eficiencia física.	

Atributo	Tipo	Descripción
fecha	Date	Es un atributo que hace referencia a la tabla (tb_resultado).
id_prueba	Int	Es un atributo que hace referencia a la tabla (tb_prueba_ef).
solapin	Varchar	Es un atributo que hace referencia a la tabla (tb_persona).
nivel	Int	Es un atributo que tiene el nivel del estudiante.

Tabla A8.30 Descripción textual de la tabla “tb\_nivel\_general\_est”

<b>Nombre:</b>	tb_persona	
<b>Descripción:</b>	Esta tabla almacena la mayoría de los datos de las personas que interactúan con el sistema.	
Atributo	Tipo	Descripción
solapin	Varchar	La llave primaria de la tabla.
direccion_particular	Varchar	Es un atributo que tiene la dirección particular de la persona.
id_ocupacion	Int	Es un atributo que hace referencia a la tabla (no_ocupacion).
nombre	Varchar	Es un atributo que tiene el nombre de la persona.
papellido	Varchar	Es un atributo que tiene el primer apellido de la persona.
sapellido	Varchar	Es un atributo que tiene el segundo apellido de la persona.
edad	Int	Es un atributo que tiene la edad de la persona.
usuario	Varchar	Es un atributo que tiene el usuario de la persona.
id_municipio	Int	Es un atributo que hace referencia a la tabla (no_municipio).
id_provincia	Int	Es un atributo que hace referencia a la tabla (no_provincia).

id_sexo	Int	Es un atributo que hace referencia a la tabla (no_sexo).
foto	Varchar	Es un atributo que tiene la dirección de donde se encuentra la foto.

Tabla A8.31 Descripción textual de la tabla “tb\_persona”

<b>Nombre:</b>	tb_prestamo	
<b>Descripción:</b>	Esta tabla almacena los datos de los préstamos que se efectúan en el gimnasio.	
<b>Atributo</b>	<b>Tipo</b>	<b>Descripción</b>
id_prestamo	Int	La llave primaria de la tabla.
id_utensilio	Int	Es un atributo que hace referencia a la tabla (tb_utensilio).
solapin	Varchar	Es un atributo que hace referencia a la tabla (tb_persona).
fecha_inicio	Date	Es un atributo que tiene la fecha que se inicio el préstamo.
fecha_fin	Date	Es un atributo que tiene la fecha que se finalizo el préstamo.
hora_inicio	Time	Es un atributo que tiene la hora en que se inicio el préstamo.
hora_fin	Time	Es un atributo que tiene la hora en que se finalizo el préstamo.
estado_inicial	Boolean	Es un atributo que tiene el valor 1 cuando esta en buen estado el utensilio que es prestado y valor 0 en caso contrario.
estado_final	Boolean	Es un atributo que tiene el valor 1 cuando se entrega en buen estado el utensilios prestado y 0 en caso contrario.
cant_utensilio	Int	Es un atributo que tiene la cantidad de utensilios prestados, por ahora siempre es 1.

entregado	Boolean	Es un atributo que tiene valor 0 cuando el utensilio prestado no se ha entregado todavía, y 1 en caso contrario.
-----------	---------	--

Tabla A8.32 Descripción textual de la tabla “tb\_prestamo”

<b>Nombre:</b>	tb_profesor_ef	
<b>Descripción:</b>	Esta tabla almacena el solapin de las personas que son profesores de educación física.	
<b>Atributo</b>	<b>Tipo</b>	<b>Descripción</b>
solapin	Varchar	Es un atributo que hace referencia a la tabla (tb_persona).

Tabla A8.33 Descripción textual de la tabla “tb\_profesor\_ef”

<b>Nombre:</b>	tb_prueba_ef	
<b>Descripción:</b>	Esta tabla almacena los datos de las pruebas de eficiencia física.	
<b>Atributo</b>	<b>Tipo</b>	<b>Descripción</b>
id_prueba	Int	La llave primaria de la tabla.
nombre	Varchar	Es un atributo que tiene el nombre de la prueba de eficiencia física.
fecha_inicial	Date	Es un atributo que tiene la fecha inicial del plazo en que se hace la prueba de eficiencia física.
fecha_final	Date	Es un atributo que tiene la fecha final del plazo en que se hace la prueba de eficiencia física.

Tabla A8.34 Descripción textual de la tabla “tb\_prueba\_ef”

<b>Nombre:</b>	tb_representacion	
<b>Descripción:</b>	Esta tabla almacena las representaciones que participan en las competencias.	

Atributo	Tipo	Descripción
id_representacion	Int	La llave primaria de la tabla.
puntos_acumulados	Int	Es un atributo que tiene la cantidad de puntos que va acumulando la representación durante la competencia.
nombre	Varchar	Es un atributo que tiene el nombre de la representación.
mascota	Varchar	Es un atributo que tiene la dirección de la foto de la mascota.
id_competencia	Int	Es un atributo que hace referencia a la tabla (tb_competencia).

Tabla A8.35 Descripción textual de la tabla “tb\_representacion”

<b>Nombre:</b>	tb_resultado	
<b>Descripción:</b>	Esta tabla almacena los resultados y los niveles obtenidos de los estudiantes que hacen las pruebas de eficiencia física.	
Atributo	Tipo	Descripción
fecha	Date	La llave primaria de la tabla.
id_indicador	Int	Es un atributo que hace referencia a la tabla (no_indicador).
id_prueba	Int	Es un atributo que hace referencia a la tabla (tb_prueba_ef).
solapin	Varchar	Es un atributo que hace referencia a la tabla (tb_persona).
nivel_indicador	Int	Es un atributo que tiene el valor del nivel obtenido.
resultado	Varchar	Es un atributo que tiene el resultado del estudiante en la prueba de eficiencia física.

Tabla A8.36 Descripción textual de la tabla “tb\_resultado”

<b>Nombre:</b>	tb_usuario
<b>Descripción:</b>	Esta tabla almacena algunos de los datos del usuario.

Atributo	Tipo	Descripción
solapin	Varchar	Es un atributo que hace referencia a la tabla (tb_persona).
contrasena	Varchar	Es un atributo que tiene la contraseña del usuario.

Tabla A8.37 Descripción textual de la tabla “tb\_usuario”

<b>Nombre:</b>	tb_usuario_roll	
<b>Descripción:</b>	Esta tabla almacena los roles de los usuarios del sistema.	
Atributo	Tipo	Descripción
solapin	Varchar	Es un atributo que hace referencia a la tabla (tb_persona).
id_roll	Int	Es un atributo que hace referencia a la tabla (no_rol).

Tabla A8.38 Descripción textual de la tabla “tb\_usuario\_roll”

<b>Nombre:</b>	tb_utensilio	
<b>Descripción:</b>	Esta tabla almacena los utensilios que se tienen en el gimnasio.	
Atributo	Tipo	Descripción
id_utensilio	Int	La llave primaria de la tabla.
nombre	Varchar	Es un atributo que tiene el nombre del utensilio.
cantidad	Int	Es un atributo que tiene la cantidad de utensilios de cada tipo.
fecha_hora_ent	DateTime	Es un atributo que tiene hecha y hora de entrada a la bd.
SGLdeporte	Varchar	Es un atributo que hace referencia a la tabla (tb_deporte).

Tabla A8.39 Descripción textual de la tabla “tb\_utensilio”

<b>Nombre:</b>	tb_visita	
<b>Descripción:</b>	Esta tabla almacena las visitas al gimnasio de las personas.	
Atributo	Tipo	Descripción

id_visita	Int	La llave primaria de la tabla.
solapin	Varchar	Es un atributo que hace referencia a la tabla (tb_persona).
hora_ent	Time	Es un atributo que tiene la hora de entrada al gimnasio.
hora_sal	Time	Es un atributo que tiene la hora de salida al gimnasio.
fecha	Date	Es un atributo que tiene la fecha de entrada al gimnasio.
terminada	Boolean	Es un atributo que tiene valor 1 cuando a sido terminada la visita en caso contrario tiene valor 0.

Tabla A8.40 Descripción textual de la tabla “tb\_visita”

**Anexo # 9: Diagrama de componentes de los casos de uso del sistema.**

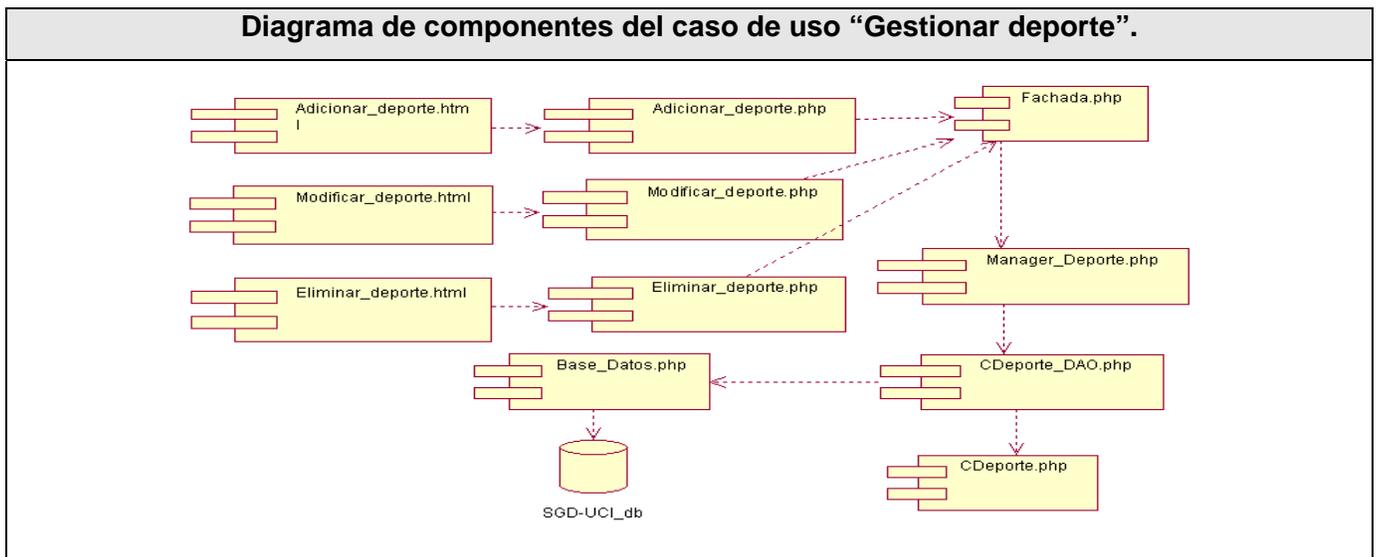


Fig. A9.1 Diagrama de componente del caso de uso “Gestionar deporte”

**Diagrama de componentes del caso de uso “Gestionar préstamo utensilio”.**

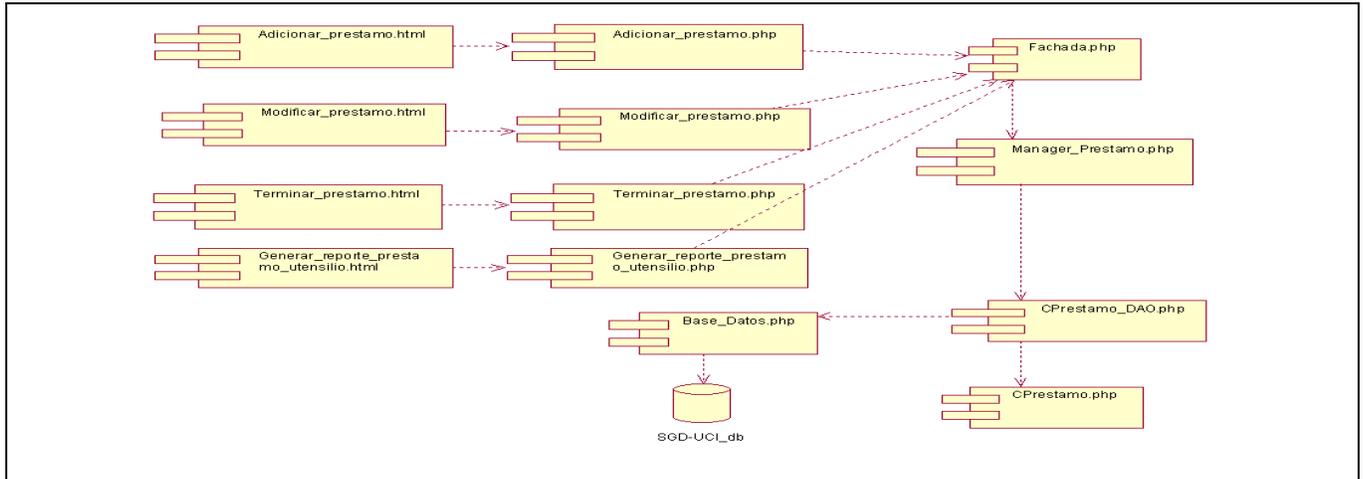


Fig. A9.2 Diagrama de componentes del caso de uso "Gestionar préstamo utensilio".

**Diagrama de componentes del caso de uso "Gestionar prueba".**

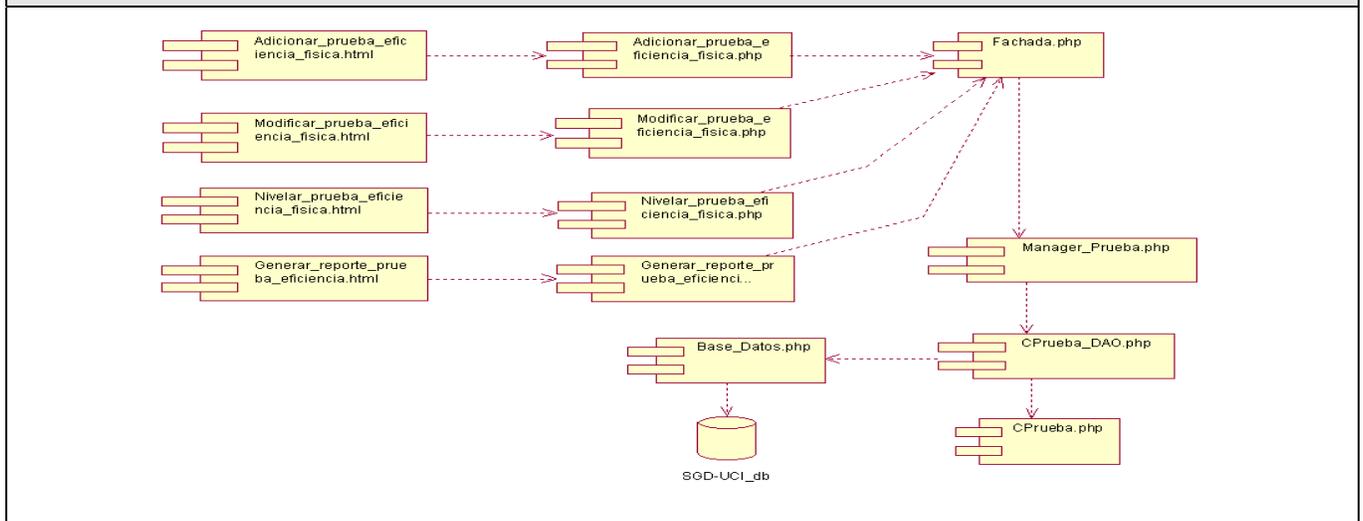


Fig. A9.3 Diagrama de componentes del caso de uso "Gestionar prueba".

**Diagrama de componentes del caso de uso "Gestionar rol".**

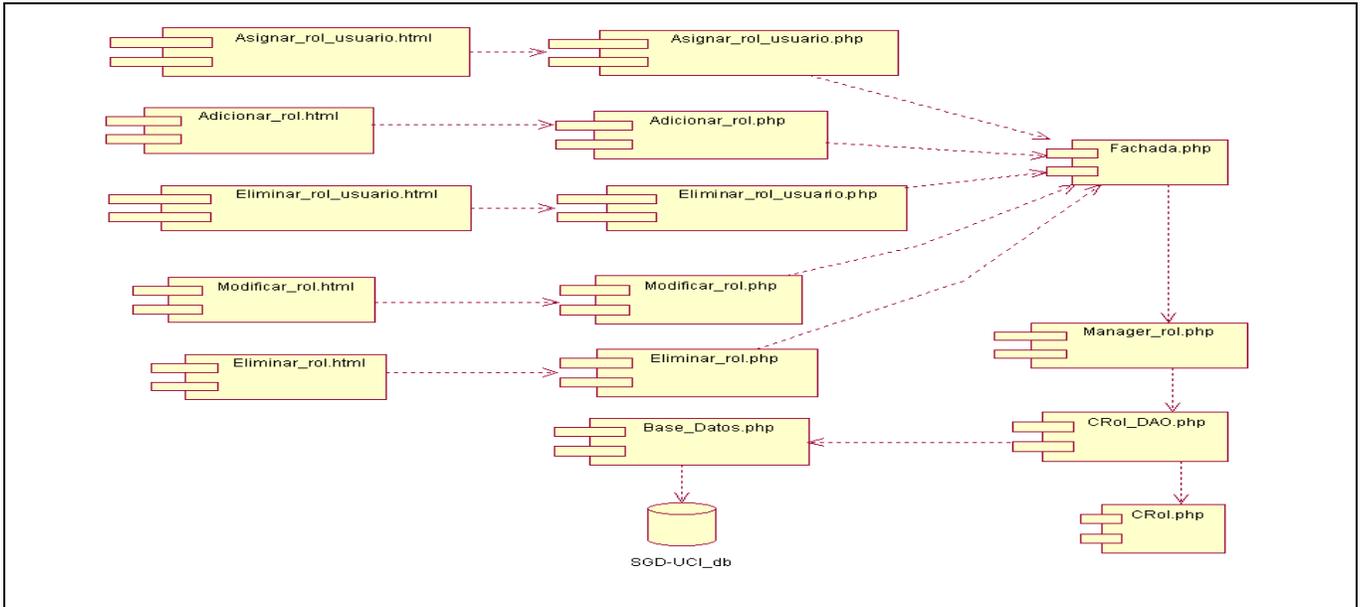


Fig. A9.4 Diagrama de componentes del caso de uso "Gestionar rol".

**Diagrama de componentes del caso de uso "Gestionar utensilio".**

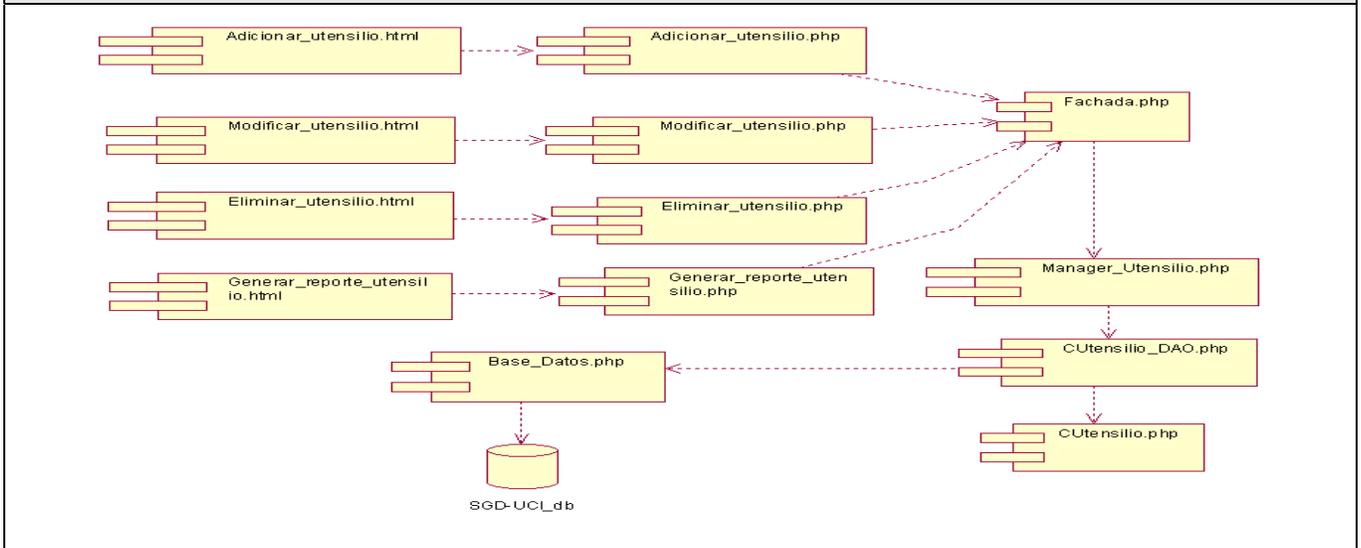


Fig. A9.5 Diagrama de componentes del caso de uso "Gestionar utensilio".

**Diagrama de componentes del caso de uso "Gestionar visita".**

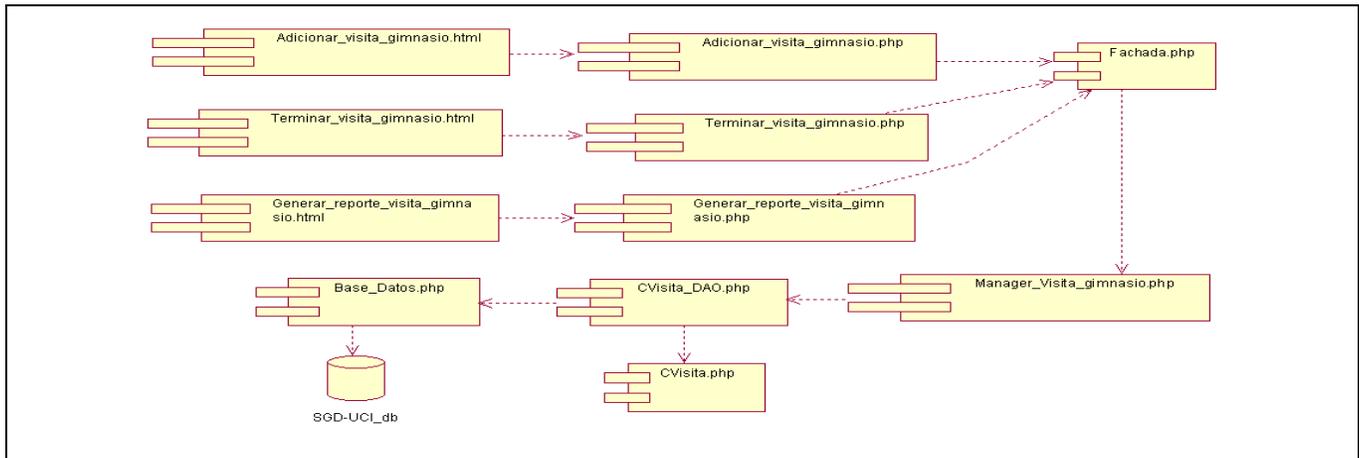


Fig. A9.6 Diagrama de componentes del caso de uso “Gestionar visita”.

**Anexo # 10: Prueba de caja negra de los casos de uso del sistema.**

Condición de Entrada	Casos Válidos	Casos no Válidos
Nombre	Letras	Introducir números o caracteres extraños.
Tipo de deporte	Seleccionar tipo de deporte	No seleccionar el tipo de deporte
Siglas	3 Letras	Siglas mayor o menor que 3 letras. Introducir números o caracteres extraños.
Sexo que participan	Seleccionar sexo	No seleccionar sexo

*Caso válido*

<b>Caso de uso:</b>	Gestionar deporte
<b>Caso de prueba:</b>	Entrar un deporte introduciendo correctamente todos los datos.
<b>Entrada:</b>	El administrador decide entrar un deporte, para ello debe seleccionar la opción de Administrar en el módulo de administración y después la

	opción Gestionar deporte. Nombre: Fútbol Tipo de deporte: Colectivo Siglas: Ftb Nombre: Masculino
<b>Resultado:</b>	El sistema introduce el deporte en la base de datos del sistema.
<b>Condiciones:</b>	Los datos seleccionados e introducidos deben de estar en el formato correcto. El deporte a ingresar no debe aparecer anteriormente en la base de datos del sistema.

Caso inválido

<b>Caso de uso:</b>	Gestionar deporte
<b>Caso de prueba:</b>	Entrar un deporte introduciendo algún dato erróneo.
<b>Entrada:</b>	El administrador decide entrar un deporte, para ello debe seleccionar la opción de Administrar en el módulo de administración y después la opción Gestionar deporte. Nombre: Fútbol12 Tipo de deporte: Colectivo Siglas: Ftb Nombre: Masculino
<b>Resultado:</b>	El sistema muestra un mensaje de error notificando que el nombre introducido es incorrecto.
<b>Condiciones:</b>	Los datos seleccionados e introducidos deben de estar en el formato correcto. El deporte a ingresar no debe aparecer anteriormente en la base de datos del sistema.

Tabla A10.1 Pruebas de Caja Negra para el Caso de uso “Gestionar deporte” escenario “Entrar deporte”.

Condición de Entrada	Casos Válidos	Casos no Válidos
Deporte a modificar	Seleccionar deporte	No seleccionar deporte
Nombre de deporte	Letras	Introducir números o caracteres extraños.

Tipo de deporte	Seleccionar tipo de deporte	No seleccionar tipo de deporte
Siglas	3 Letras	Siglas mayor o menor que 3 letras. Introducir números o caracteres extraños.
Sexo que participan	Seleccionar sexo	No seleccionar sexo

Caso válido

<b>Caso de uso:</b>	Gestionar deporte
<b>Caso de prueba:</b>	Modificar un deporte introduciendo correctamente todos los datos.
<b>Entrada:</b>	<p>El administrador decide modificar un deporte, para ello debe seleccionar la opción de Administrar en el módulo de administración y después la opción Gestionar deporte.</p> <p>Deporte a modificar: Fútbol</p> <p>Nombre de deporte: Fútbol</p> <p>Tipo de deporte: Colectivo</p> <p>Siglas: Ftb</p> <p>Nombre: Masculino</p>
<b>Resultado:</b>	El sistema modifica el deporte en la base de datos del sistema.
<b>Condiciones:</b>	Los datos seleccionados e introducidos deben de estar en el formato correcto. El deporte a ingresar no debe aparecer anteriormente en la base de datos del sistema.

Caso inválido

<b>Caso de uso:</b>	Gestionar deporte
<b>Caso de prueba:</b>	Modificar un deporte introduciendo algún dato erróneo.
<b>Entrada:</b>	El administrador decide modificar un deporte, para ello debe seleccionar la opción de Administrar en el módulo de administración y después la opción Gestionar deporte.

	Deporte a modificar: Fútbol Nombre de deporte: Fútbol Tipo de deporte: Colectivo Siglas: Ftb1 Nombre: Masculino
<b>Resultado:</b>	El sistema muestra un mensaje de error notificando que las siglas introducidas son incorrectas.
<b>Condiciones:</b>	Los datos seleccionados e introducidos deben de estar en el formato correcto. El deporte a ingresar no debe aparecer anteriormente en la base de datos del sistema.

Tabla A10.2 Pruebas de Caja Negra para el Caso de uso “Gestionar deporte” escenario “Eliminar deporte”.

Condición de Entrada	Casos Válidos	Casos no Válidos
Deporte	Seleccionar deporte	No seleccionar deporte

Caso válido

<b>Caso de uso:</b>	Gestionar deporte
<b>Caso de prueba:</b>	Eliminar un deporte introduciendo correctamente todos los datos.
<b>Entrada:</b>	El administrador decide eliminar un deporte, para ello debe seleccionar la opción de Administrar en el módulo de administración y después la opción Gestionar deporte.  Deporte: Fútbol
<b>Resultado:</b>	El sistema elimina el deporte en la base de datos del sistema.
<b>Condiciones:</b>	El dato seleccionado debe de estar en el formato correcto.

Caso inválido

<b>Caso de uso:</b>	Gestionar deporte
<b>Caso de prueba:</b>	Eliminar un deporte introduciendo algún dato erróneo.

<b>Entrada:</b>	El administrador decide eliminar un deporte, para ello debe seleccionar la opción de Administrar en el módulo de administración y después la opción Gestionar deporte. Deporte: Seleccione
<b>Resultado:</b>	El sistema muestra un mensaje de error notificando que no seleccionado ningún deporte.
<b>Condiciones:</b>	El dato seleccionado debe de estar en el formato correcto.

Tabla A10.3 Pruebas de Caja Negra para el Caso de uso “Gestionar deporte” escenario “Entrar tipo de deporte”.

Condición de Entrada	Casos Válidos	Casos no Válidos
Tipo	Letras	Introducir números o caracteres extraños.

Caso válido

<b>Caso de uso:</b>	Gestionar deporte
<b>Caso de prueba:</b>	Entrar tipo de deporte introduciendo correctamente todos los datos.
<b>Entrada:</b>	El administrador decide entrar un tipo de deporte, para ello debe seleccionar la opción de Administrar en el módulo de administración y después la opción Gestionar deporte. Tipo: Colectivo
<b>Resultado:</b>	El sistema entra el tipo de deporte en la base de datos del sistema.
<b>Condiciones:</b>	El dato introducido debe de estar en el formato correcto. El tipo de deporte a ingresar no debe aparecer anteriormente en la base de datos del sistema.

Caso inválido

<b>Caso de uso:</b>	Gestionar deporte
<b>Caso de prueba:</b>	Entrar tipo de deporte introduciendo algún dato erróneo.
<b>Entrada:</b>	El administrador decide entrar un tipo de deporte, para ello debe

	seleccionar la opción de Administrar en el módulo de administración y después la opción Gestionar deporte. Tipo: Colectivo12
<b>Resultado:</b>	El sistema muestra un mensaje de error notificando que el tipo de deporte introducido es incorrecto.
<b>Condiciones:</b>	El dato introducido debe de estar en el formato correcto. El tipo de deporte a ingresar no debe aparecer anteriormente en la base de datos del sistema.

Tabla A10.4 Pruebas de Caja Negra para el Caso de uso “Gestionar deporte” escenario “Modificar tipo de deporte”.

Condición de Entrada	Casos Válidos	Casos no Válidos
Tipo	Seleccionar tipo de deporte	No seleccionar tipo de deporte
Nuevo tipo	Letras	Introducir números o caracteres extraños.

*Caso válido*

<b>Caso de uso:</b>	Gestionar deporte
<b>Caso de prueba:</b>	Modificar tipo de deporte introduciendo correctamente todos los datos.
<b>Entrada:</b>	El administrador decide modificar un tipo de deporte, para ello debe seleccionar la opción de Administrar en el módulo de administración y después la opción Gestionar deporte. Tipo: Colectivos Nuevo tipo: Colectivo
<b>Resultado:</b>	El sistema modifica el tipo de deporte en la base de datos del sistema.
<b>Condiciones:</b>	El dato introducido debe de estar en el formato correcto. El tipo de deporte a ingresar no debe aparecer anteriormente en la base de datos del sistema.

*Caso inválido*

<b>Caso de uso:</b>	Gestionar deporte
<b>Caso de prueba:</b>	Modificar tipo de deporte introduciendo algún dato erróneo.
<b>Entrada:</b>	El administrador decide modificar un tipo de deporte, para ello debe seleccionar la opción de Administrar en el módulo de administración y después la opción Gestionar deporte. Tipo: Colectivo Nuevo tipo: Colectivo12
<b>Resultado:</b>	El sistema muestra un mensaje de error notificando que el tipo de deporte introducido es incorrecto.
<b>Condiciones:</b>	El dato introducido debe de estar en el formato correcto. El tipo de deporte a ingresar no debe aparecer anteriormente en la base de datos del sistema.

Tabla A10.5 Pruebas de Caja Negra para el Caso de uso “Gestionar deporte” escenario “Eliminar tipo de deporte”.

Condición de Entrada	Casos Válidos	Casos no Válidos
Tipo	Seleccionar tipo de deporte	No seleccionar tipo de deporte

*Caso válido*

<b>Caso de uso:</b>	Gestionar deporte
<b>Caso de prueba:</b>	Eliminar tipo de deporte introduciendo correctamente todos los datos.
<b>Entrada:</b>	El administrador decide eliminar un tipo de deporte, para ello debe seleccionar la opción de Administrar en el módulo de administración y después la opción Gestionar deporte. Tipo: Colectivos
<b>Resultado:</b>	El sistema elimina el tipo de deporte en la base de datos del sistema.
<b>Condiciones:</b>	El dato introducido debe de estar en el formato correcto.

*Caso inválido*

<b>Caso de uso:</b>	Gestionar deporte
<b>Caso de prueba:</b>	Eliminar tipo de deporte introduciendo algún dato erróneo.
<b>Entrada:</b>	El administrador decide eliminar un tipo de deporte, para ello debe seleccionar la opción de Administrar en el módulo de administración y después la opción Gestionar deporte. Tipo: Seleccionar
<b>Resultado:</b>	El sistema muestra un mensaje de error notificando que no ha seleccionado el tipo de deporte.
<b>Condiciones:</b>	El dato introducido debe de estar en el formato correcto.

Tabla A10.6 Pruebas de Caja Negra para el Caso de uso “Gestionar utensilio” escenario “Entrar utensilio”.

Condición de Entrada	Casos Válidos	Casos no Válidos
Nombre	Letras	Introducir números o caracteres extraños.
Deporte	Seleccionar deporte	No seleccionar deporte
Cantidad	Números	Introducir letras o caracteres extraños.

*Caso válido*

<b>Caso de uso:</b>	Gestionar utensilio
<b>Caso de prueba:</b>	Entrar utensilio introduciendo correctamente todos los datos.
<b>Entrada:</b>	El administrador decide entrar un utensilio, para ello debe seleccionar la opción de Administrar gimnasio en el módulo de administración. Nombre: Bate Deporte: Beisbol Cantidad: 50
<b>Resultado:</b>	El sistema introduce el utensilio en la base de datos del sistema.
<b>Condiciones:</b>	Los datos introducidos deben de estar en el formato correcto. El utensilio

	a ingresar no debe aparecer anteriormente en la base de datos del sistema.
--	--

*Caso inválido*

<b>Caso de uso:</b>	Gestionar utensilio
<b>Caso de prueba:</b>	Entrar utensilio introduciendo algún dato erróneo.
<b>Entrada:</b>	El administrador decide entrar un utensilio, para ello debe seleccionar la opción de Administrar gimnasio en el módulo de administración.  Nombre: Bate Deporte: Seleccionar Cantidad: 50
<b>Resultado:</b>	El sistema muestra un mensaje de error notificando que no ha seleccionado el deporte.
<b>Condiciones:</b>	Los datos introducidos deben de estar en el formato correcto. El utensilio a ingresar no debe aparecer anteriormente en la base de datos del sistema.

Tabla A10.7 Pruebas de Caja Negra para el Caso de uso “Gestionar utensilio” escenario “Modificar utensilio”.

Condición de Entrada	Casos Válidos	Casos no Válidos
Utensilio	Seleccionar utensilio	No seleccionar utensilio
Nombre	Letras	Introducir números o caracteres extraños.
Deporte	Seleccionar deporte	No seleccionar deporte
Cantidad	Números	Introducir letras o caracteres extraños.

*Caso válido*

<b>Caso de uso:</b>	Gestionar utensilio
<b>Caso de prueba:</b>	Modificar utensilio introduciendo correctamente todos los datos.

<b>Entrada:</b>	<p>El administrador decide modificar un utensilio, para ello debe seleccionar la opción de Administrar gimnasio en el módulo de administración.</p> <p>Utensilio: Bates                  Nombre: Bate                  Deporte: Beisbol                  Cantidad: 50</p>
<b>Resultado:</b>	El sistema modifica el utensilio en la base de datos del sistema.
<b>Condiciones:</b>	Los datos introducidos deben de estar en el formato correcto. El utensilio a ingresar no debe aparecer anteriormente en la base de datos del sistema.

*Caso inválido*

<b>Caso de uso:</b>	Gestionar utensilio
<b>Caso de prueba:</b>	Modificar utensilio introduciendo algún dato erróneo.
<b>Entrada:</b>	<p>El administrador decide modificar un utensilio, para ello debe seleccionar la opción de Administrar gimnasio en el módulo de administración.</p> <p>Utensilio: Seleccionar                  Nombre: Bate                  Deporte: Beisbol                  Cantidad: 50</p>
<b>Resultado:</b>	El sistema muestra un mensaje de error notificando que no ha seleccionado el utensilio.
<b>Condiciones:</b>	Los datos introducidos deben de estar en el formato correcto. El utensilio a ingresar no debe aparecer anteriormente en la base de datos del sistema.

Tabla A10.8 Pruebas de Caja Negra para el Caso de uso “Gestionar utensilio” escenario “Eliminar utensilio”.

Condición de Entrada	Casos Válidos	Casos no Válidos
Utensilio	Seleccionar utensilio	No seleccionar utensilio

*Caso válido*

<b>Caso de uso:</b>	Gestionar utensilio
<b>Caso de prueba:</b>	Eliminar utensilio introduciendo correctamente todos los datos.
<b>Entrada:</b>	El administrador decide eliminar un utensilio, para ello debe seleccionar la opción de Administrar gimnasio en el módulo de administración. Utensilio: Bate
<b>Resultado:</b>	El sistema elimina el utensilio en la base de datos del sistema.
<b>Condiciones:</b>	Los datos introducidos deben de estar en el formato correcto.

*Caso inválido*

<b>Caso de uso:</b>	Gestionar utensilio
<b>Caso de prueba:</b>	Eliminar utensilio introduciendo algún dato erróneo.
<b>Entrada:</b>	El administrador decide eliminar un utensilio, para ello debe seleccionar la opción de Administrar gimnasio en el módulo de administración. Utensilio: Seleccionar
<b>Resultado:</b>	El sistema muestra un mensaje de error notificando que no ha seleccionado el utensilio.
<b>Condiciones:</b>	Los datos introducidos deben de estar en el formato correcto.

Tabla A10.9 Pruebas de Caja Negra para el Caso de uso “Gestionar préstamo de utensilio” escenario “Crear préstamo”.

Condición de Entrada	Casos Válidos	Casos no Válidos
Solapin	Números	Introducir letras o caracteres extraños.
Deporte	Seleccionar deporte	No seleccionar deporte

Utensilio	Seleccionar utensilio	No seleccionar utensilio
Estado inicial	Selecciona estado	

*Caso válido*

<b>Caso de uso:</b>	Gestionar préstamo de utensilio
<b>Caso de prueba:</b>	Crear préstamo introduciendo correctamente todos los datos.
<b>Entrada:</b>	El técnico del gimnasio decide crear un nuevo préstamo, para ello debe seleccionar la opción de Gimnasio, y luego seleccionar Gestión de préstamo dentro del modulo del técnico.  Solapin: 54849 Deporte: Beisbol Utensilio: Bate Estado inicial: Bueno
<b>Resultado:</b>	El sistema crea el préstamo en la base de datos del sistema.
<b>Condiciones:</b>	Los datos introducidos deben de estar en el formato correcto.

*Caso inválido*

<b>Caso de uso:</b>	Gestionar préstamo de utensilio
<b>Caso de prueba:</b>	Crear préstamo introduciendo algún dato erróneo.
<b>Entrada:</b>	El técnico del gimnasio decide crear un nuevo préstamo, para ello debe seleccionar la opción de Gimnasio, y luego seleccionar Gestión de préstamo dentro del modulo del técnico.  Solapin: 54849 Deporte: Seleccionar Utensilio: Seleccionar Estado inicial: Bueno
<b>Resultado:</b>	El sistema muestra un mensaje de error notificando que no ha seleccionado el deporte.
<b>Condiciones:</b>	Los datos introducidos deben de estar en el formato correcto.

Tabla A10.10 Pruebas de Caja Negra para el Caso de uso “Gestionar préstamo de utensilio” escenario “Terminar préstamo”.

Condición de Entrada	Casos Válidos	Casos no Válidos
Solapín	Números	Introducir letras o caracteres extraños.
Estado final	Seleccionar estado	
Préstamo	Seleccionar préstamo	No seleccionar préstamo

*Caso válido*

<b>Caso de uso:</b>	Gestionar préstamo de utensilio
<b>Caso de prueba:</b>	Terminar préstamo introduciendo correctamente todos los datos.
<b>Entrada:</b>	El técnico decide terminar un préstamo, para ello debe seleccionar la opción de Gimnasio, y luego seleccionar Gestión de préstamo dentro del modulo del técnico.  Solapín: 54849 Estado final: Bueno Préstamo: Selección
<b>Resultado:</b>	El sistema termina el préstamo en la base de datos del sistema.
<b>Condiciones:</b>	Los datos introducidos deben de estar en el formato correcto. Se debe seleccionar al menos un préstamo.

*Caso inválido*

<b>Caso de uso:</b>	Gestionar préstamo de utensilio
<b>Caso de prueba:</b>	Terminar préstamo introduciendo algún dato erróneo.
<b>Entrada:</b>	El técnico decide terminar un préstamo, para ello debe seleccionar la opción de Gimnasio, y luego seleccionar Gestión de préstamo dentro del modulo del técnico.  Solapín: 54849

	Estado final: Bueno Préstamo: No selección
<b>Resultado:</b>	El sistema muestra un mensaje de error notificando que no ha seleccionado un préstamo.
<b>Condiciones:</b>	Los datos introducidos deben de estar en el formato correcto. Se debe seleccionar al menos un préstamo.

Tabla A10.11 Pruebas de Caja Negra para el Caso de uso “Gestionar préstamo de utensilio” escenario “Modificar préstamo”.

Condición de Entrada	Casos Válidos	Casos no Válidos
Solapin	Números	Introducir letras o caracteres extraños.
Préstamo	Seleccionar préstamo	No seleccionar préstamo
Deporte	Seleccionar deporte	No seleccionar deporte
Utensilio	Seleccionar utensilio	No seleccionar utensilio
Estado inicial	Seleccionar estado	
Cantidad	Números.	Introducir letras o caracteres extraños.

Caso válido

<b>Caso de uso:</b>	Gestionar préstamo de utensilio
<b>Caso de prueba:</b>	Modificar préstamo introduciendo correctamente todos los datos.
<b>Entrada:</b>	El técnico decide modificar un préstamo, para ello debe seleccionar la opción de Gimnasio, y luego seleccionar Gestión de préstamo dentro del modulo del técnico.  Solapin: 54849 Préstamo: Selección Deporte: Beisbol Utensilio: Bate Estado final: Bueno Cantidad: 1

<b>Resultado:</b>	El sistema modifica el préstamo en la base de datos del sistema.
<b>Condiciones:</b>	Los datos introducidos deben de estar en el formato correcto. Se debe seleccionar un préstamo solamente.

Caso inválido

<b>Caso de uso:</b>	Gestionar préstamo de utensilio
<b>Caso de prueba:</b>	Modificar préstamo introduciendo algún dato erróneo.
<b>Entrada:</b>	El técnico decide modificar un préstamo, para ello debe seleccionar la opción de Gimnasio, y luego seleccionar Gestión de préstamo dentro del modulo del técnico. Solapin: 54849 Préstamo: Selección Deporte: Beisbol Utensilio: Selecciona Estado final: Bueno Cantidad: 1
<b>Resultado:</b>	El sistema muestra un mensaje de error notificando que no ha seleccionado un utensilio.
<b>Condiciones:</b>	Los datos introducidos deben de estar en el formato correcto. Se debe seleccionar al menos un préstamo.

Tabla A10.12 Pruebas de Caja Negra para el Caso de uso “Gestionar préstamo de utensilio” escenario “Reporte de préstamo por fecha”.

Condición de Entrada	Casos Válidos	Casos no Válidos
Reporte	Seleccionar fecha	No seleccionar fecha
Fecha	Seleccionar fecha	No seleccionar fecha

Caso válido

<b>Caso de uso:</b>	Gestionar préstamo de utensilio
<b>Caso de prueba:</b>	Reporte de préstamo por fecha introduciendo correctamente todos los datos.
<b>Entrada:</b>	El usuario decide realizar un Reporte de préstamo por fecha, para ello debe seleccionar la opción de Gimnasio, y luego seleccionar Reporte de préstamo.  Reporte: Fecha Fecha: 2007/06/06
<b>Resultado:</b>	El sistema muestra los datos de los préstamos realizados en esa fecha.
<b>Condiciones:</b>	Los datos introducidos deben de estar en el formato correcto.

Caso inválido

<b>Caso de uso:</b>	Gestionar préstamo de utensilio
<b>Caso de prueba:</b>	Reporte de préstamo por fecha introduciendo algún dato erróneo.
<b>Entrada:</b>	El usuario decide realizar un Reporte de préstamo por fecha, para ello debe seleccionar la opción de Gimnasio, y luego seleccionar Reporte de préstamo.  Reporte: Fecha Fecha: 0000/00/00
<b>Resultado:</b>	El sistema muestra un mensaje de error notificando que la fecha es incorrecta.
<b>Condiciones:</b>	Los datos introducidos deben de estar en el formato correcto.

Tabla A10.13 Pruebas de Caja Negra para el Caso de uso “Gestionar préstamo de utensilio” escenario “Reporte de préstamo por usuario”.

Condición de Entrada	Casos Válidos	Casos no Válidos
----------------------	---------------	------------------

Reporte	Seleccionar usuario	No seleccionar usuario
Solapin	Números	Introducir letras o caracteres extraños.

Caso válido

<b>Caso de uso:</b>	Gestionar préstamo de utensilio
<b>Caso de prueba:</b>	Reporte de préstamo por usuario introduciendo correctamente todos los datos.
<b>Entrada:</b>	El usuario decide realizar un Reporte de préstamo por usuario, para ello debe seleccionar la opción de Gimnasio, y luego seleccionar Reporte de préstamo. Reporte: Usuario Solapin: 54849
<b>Resultado:</b>	El sistema muestra los datos de los préstamos realizados por ese usuario.
<b>Condiciones:</b>	Los datos introducidos deben de estar en el formato correcto.

Caso inválido

<b>Caso de uso:</b>	Gestionar préstamo de utensilio
<b>Caso de prueba:</b>	Reporte de préstamo por usuario introduciendo algún dato erróneo.
<b>Entrada:</b>	El usuario decide realizar un Reporte de préstamo por fecha, para ello debe seleccionar la opción de Gimnasio, y luego seleccionar Reporte de préstamo. Reporte: Usuario Solapin: 54637L
<b>Resultado:</b>	El sistema muestra un mensaje de error notificando que el solapin es incorrecto.
<b>Condiciones:</b>	Los datos introducidos deben de estar en el formato correcto.

Tabla A10.14 Pruebas de Caja Negra para el Caso de uso “Gestionar préstamo de utensilio” escenario “Reporte de préstamo pendientes”.

Condición de Entrada	Casos Válidos	Casos no Válidos
Reporte	Seleccionar pendientes	No seleccionar pendientes

Caso válido

<b>Caso de uso:</b>	Gestionar préstamo de utensilio
<b>Caso de prueba:</b>	Reporte de préstamos pendientes introduciendo correctamente todos los datos.
<b>Entrada:</b>	El usuario decide realizar un Reporte de préstamo pendientes, para ello debe seleccionar la opción de Gimnasio, y luego seleccionar Reporte de préstamo. Reporte: Pendientes
<b>Resultado:</b>	El sistema muestra los datos de los préstamos pendientes.
<b>Condiciones:</b>	Los datos introducidos deben de estar en el formato correcto.

Caso inválido

<b>Caso de uso:</b>	Gestionar préstamo de utensilio
<b>Caso de prueba:</b>	Reporte de préstamos pendientes introduciendo algún dato erróneo.
<b>Entrada:</b>	El usuario decide realizar un Reporte de préstamo por fecha, para ello debe seleccionar la opción de Gimnasio, y luego seleccionar Reporte de préstamo. Reporte: Seleccione
<b>Resultado:</b>	El sistema muestra un mensaje de error notificando que no se ha seleccionado ninguno parámetro.

<b>Condiciones:</b>	Los datos introducidos deben de estar en el formato correcto.
---------------------	---

Tabla A10.15 Pruebas de Caja Negra para el Caso de uso “Gestionar préstamo de utensilio” escenario “Reporte de utensilio por deporte”.

Condición de Entrada	Casos Válidos	Casos no Válidos
Reporte	Seleccionar deporte	No seleccionar deporte
Deporte	Seleccionar deporte	No seleccionar deporte

Caso válido

<b>Caso de uso:</b>	Gestionar préstamo de utensilio
<b>Caso de prueba:</b>	Reporte de utensilio por deporte introduciendo correctamente todos los datos.
<b>Entrada:</b>	El usuario decide realizar un Reporte de utensilios por deporte, para ello debe seleccionar la opción de Gimnasio, y luego seleccionar Reporte de utensilio. Reporte: Deporte Deporte: Beisbol
<b>Resultado:</b>	El sistema muestra los datos de los utensilios del deporte seleccionado.
<b>Condiciones:</b>	Los datos introducidos deben de estar en el formato correcto.

Caso inválido

<b>Caso de uso:</b>	Gestionar préstamo de utensilio
<b>Caso de prueba:</b>	Reporte de utensilio por deporte introduciendo algún dato erróneo.
<b>Entrada:</b>	El usuario decide realizar un Reporte de utensilios por deporte, para ello debe seleccionar la opción de Gimnasio, y luego seleccionar Reporte de utensilio. Reporte: Deporte

	Deporte: Seleccione
<b>Resultado:</b>	El sistema muestra un mensaje de error notificando que no se ha seleccionado ninguno deporte.
<b>Condiciones:</b>	Los datos introducidos deben de estar en el formato correcto.

Tabla A10.16 Pruebas de Caja Negra para el Caso de uso “Gestionar préstamo de utensilio” escenario “Reporte de utensilio más prestados”.

Condición de Entrada	Casos Válidos	Casos no Válidos
Reporte	Seleccionar más prestados	No seleccionar más prestados

Caso válido

<b>Caso de uso:</b>	Gestionar préstamo de utensilio
<b>Caso de prueba:</b>	Reporte de utensilios más prestados introduciendo correctamente todos los datos.
<b>Entrada:</b>	El usuario decide realizar un Reporte de utensilios por deporte, para ello debe seleccionar la opción de Gimnasio, y luego seleccionar Reporte de utensilio. Reporte: Más prestados
<b>Resultado:</b>	El sistema muestra los datos de los utensilios más prestados.
<b>Condiciones:</b>	Los datos introducidos deben de estar en el formato correcto.

Caso inválido

<b>Caso de uso:</b>	Gestionar préstamo de utensilio
<b>Caso de prueba:</b>	Reporte de utensilios más prestados introduciendo algún dato erróneo.
<b>Entrada:</b>	El usuario decide realizar un Reporte de utensilios por deporte, para ello debe seleccionar la opción de Gimnasio, y luego seleccionar Reporte de utensilio.

	Reporte: Seleccione
<b>Resultado:</b>	El sistema muestra un mensaje de error notificando que no se ha seleccionado ninguno parámetro.
<b>Condiciones:</b>	Los datos introducidos deben de estar en el formato correcto.

Tabla A10.17 Pruebas de Caja Negra para el Caso de uso “Gestionar visita al gimnasio” escenario “Registrar visita”.

Condición de Entrada	Casos Válidos	Casos no Válidos
Solapin	Números	Introducir letras o caracteres extraños.

Caso válido

<b>Caso de uso:</b>	Gestionar visita al gimnasio.
<b>Caso de prueba:</b>	Registrar visita introduciendo correctamente todos los datos.
<b>Entrada:</b>	El técnico decide registrar una nueva visita, para ello debe seleccionar la opción de Gimnasio, y luego seleccionar Gestión de visita dentro del modulo del técnico.  Solapin: 54849
<b>Resultado:</b>	El sistema registra la visita en la base de datos del sistema.
<b>Condiciones:</b>	Los datos introducidos deben de estar en el formato correcto. El usuario no debe de estar dentro del gimnasio.

Caso inválido

<b>Caso de uso:</b>	Gestionar visita al gimnasio.
<b>Caso de prueba:</b>	Registrar visita introduciendo algún dato erróneo.
<b>Entrada:</b>	El técnico decide registrar una nueva visita, para ello debe seleccionar la opción de Gimnasio, y luego seleccionar Gestión de visita dentro del modulo del técnico.  Solapin: 548493
<b>Resultado:</b>	El sistema muestra un mensaje de error notificando que ese solapin no

	existe.
<b>Condiciones:</b>	Los datos introducidos deben de estar en el formato correcto. El usuario no debe de estar dentro del gimnasio.

Tabla A10.18 Pruebas de Caja Negra para el Caso de uso “Gestionar visita al gimnasio” escenario “Registrar salida”.

Condición de Entrada	Casos Válidos	Casos no Válidos
Solapin	Números	Introducir letras o caracteres extraños.

Caso válido

<b>Caso de uso:</b>	Gestionar visita al gimnasio.
<b>Caso de prueba:</b>	Registrar salida introduciendo correctamente todos los datos.
<b>Entrada:</b>	El técnico decide registrar una salida, para ello debe seleccionar la opción de Gimnasio, y luego seleccionar Gestión de visita dentro del modulo del técnico. Solapin: 54849
<b>Resultado:</b>	El sistema registra la salida en la base de datos del sistema.
<b>Condiciones:</b>	Los datos introducidos deben de estar en el formato correcto. El usuario no debe de estar fuera del gimnasio.

Caso inválido

<b>Caso de uso:</b>	Gestionar visita al gimnasio.
<b>Caso de prueba:</b>	Registrar salida introduciendo algún dato erróneo.
<b>Entrada:</b>	El técnico decide registrar una salida, para ello debe seleccionar la opción de Gimnasio, y luego seleccionar Gestión de visita dentro del modulo del técnico. Solapin: 548493
<b>Resultado:</b>	El sistema muestra un mensaje de error notificando que ese solapin no existe.

<b>Condiciones:</b>	Los datos introducidos deben de estar en el formato correcto. El usuario no debe de estar fuera del gimnasio.
---------------------	---

Tabla A10.19 Pruebas de Caja Negra para el Caso de uso “Gestionar visita al gimnasio” escenario “Reporte de visita por fecha”.

Condición de Entrada	Casos Válidos	Casos no Válidos
Reporte	Seleccionar fecha	No seleccionar fecha
Fecha	Seleccionar fecha	No seleccionar fecha

Caso válido

<b>Caso de uso:</b>	Gestionar visita al gimnasio.
<b>Caso de prueba:</b>	Reporte de visita por fecha introduciendo correctamente todos los datos.
<b>Entrada:</b>	El usuario decide realizar un Reporte de visitas por fecha, para ello debe seleccionar la opción de Gimnasio, y luego seleccionar Reporte de visita.  Reporte: Fecha Fecha: 2007/06/06
<b>Resultado:</b>	El sistema muestra los datos de las visitas que se efectuaron en esa fecha.
<b>Condiciones:</b>	Los datos introducidos deben de estar en el formato correcto.

Caso inválido

<b>Caso de uso:</b>	Gestionar visita al gimnasio.
<b>Caso de prueba:</b>	Reporte de visita por fecha introduciendo algún dato erróneo.
<b>Entrada:</b>	El usuario decide realizar un Reporte de visitas por fecha, para ello debe seleccionar la opción de Gimnasio, y luego seleccionar Reporte

	de visita. Reporte: Fecha Fecha: 0000/00/00
<b>Resultado:</b>	El sistema muestra un mensaje de error notificando que la fecha es incorrecta.
<b>Condiciones:</b>	Los datos introducidos deben de estar en el formato correcto.

Tabla A10.20 Pruebas de Caja Negra para el Caso de uso “Gestionar visita al gimnasio” escenario “Reporte de visita de los usuarios más visitantes”.

Condición de Entrada	Casos Válidos	Casos no Válidos
Reporte	Seleccionar usuarios más visitantes	No seleccionar usuarios más visitantes

Caso válido

<b>Caso de uso:</b>	Gestionar visita al gimnasio.
<b>Caso de prueba:</b>	Reporte de visitas de los usuarios más visitantes introduciendo correctamente todos los datos.
<b>Entrada:</b>	El usuario decide realizar un Reporte de visitas de los usuarios más visitantes, para ello debe seleccionar la opción de Gimnasio, y luego seleccionar Reporte de visita. Reporte: Usuarios más visitantes
<b>Resultado:</b>	El sistema muestra los 10 usuarios que mas frecuentan al gimnasio.
<b>Condiciones:</b>	Los datos introducidos deben de estar en el formato correcto.

Caso inválido

<b>Caso de uso:</b>	Gestionar visita al gimnasio.
<b>Caso de prueba:</b>	Reporte de visitas de los usuarios más visitantes introduciendo algún

	dato erróneo.
<b>Entrada:</b>	El usuario decide realizar un Reporte de visitas por fecha, para ello debe seleccionar la opción de Gimnasio, y luego seleccionar Reporte de visita.  Reporte: Seleccione
<b>Resultado:</b>	El sistema muestra un mensaje de error notificando que no se ha seleccionado ninguno parámetro.
<b>Condiciones:</b>	Los datos introducidos deben de estar en el formato correcto.

Condición de Entrada	Casos Válidos	Casos no Válidos
Nombre	Seleccionar nombre	No seleccionar nombre
Indicadores	Seleccionar indicadores	No seleccionar indicadores
Fecha	Seleccionar fecha	No seleccionar fecha
Grupo	Seleccionar grupo	No seleccionar grupo
Resultados	Números	Introducir letras o caracteres extraños.

Caso válido

<b>Caso de uso:</b>	Gestionar prueba de eficiencia física.
<b>Caso de prueba:</b>	Crear prueba introduciendo correctamente todos los datos.
<b>Entrada:</b>	El profesor de educación física decide introducir los datos de una prueba de eficiencia física, para ello debe seleccionar la opción de Eficiencia física, y luego seleccionar Crear prueba de eficiencia física dentro del modulo de profesor.  Nombre: Primera prueba de eficiencia física Indicadores: Selección de los indicadores que se van a tener en cuenta.  Fecha: 2008/06/06 Grupo: 8502

	Resultados: Resultados en los indicadores.
<b>Resultado:</b>	El sistema crea la prueba de eficiencia física en la base de datos del sistema y muestra la nivelación de los estudiantes en dicha prueba.
<b>Condiciones:</b>	Los datos introducidos deben de estar en el formato correcto.

Caso inválido

<b>Caso de uso:</b>	Gestionar prueba de eficiencia física.
<b>Caso de prueba:</b>	Crear prueba introduciendo algún dato erróneo.
<b>Entrada:</b>	<p>El profesor de educación física decide introducir los datos de una prueba de eficiencia física, para ello debe seleccionar la opción de Eficiencia física, y luego seleccionar Crear prueba de eficiencia física dentro del modulo de profesor.</p> <p>Nombre: Primera prueba de eficiencia física</p> <p>Indicadores: Selección de los indicadores que se van a tener en cuenta.</p> <p>Fecha: 2008/06/06</p> <p>Grupo: Seleccione</p> <p>Resultados: Resultados en los indicadores.</p>
<b>Resultado:</b>	El sistema muestra un mensaje de error notificando que no se ha seleccionado grupo.
<b>Condiciones:</b>	Los datos introducidos deben de estar en el formato correcto.

Tabla A10.21 Pruebas de Caja Negra para el Caso de uso “Gestionar prueba de eficiencia física” escenario “Crear prueba de eficiencia física”.

Condición de Entrada	Casos Válidos	Casos no Válidos
Nombre	Seleccionar nombre	No seleccionar nombre
Fecha	Seleccionar fecha	No seleccionar fecha
Grupo	Seleccionar grupo	No seleccionar grupo

Resultados	Números	Introducir letras o caracteres extraños.
------------	---------	--

Caso válido

<b>Caso de uso:</b>	Gestionar prueba de eficiencia física.
<b>Caso de prueba:</b>	Modificar prueba introduciendo correctamente todos los datos.
<b>Entrada:</b>	<p>El profesor de educación física decide modificar algún dato de una prueba de eficiencia física, para ello debe seleccionar la opción de Eficiencia física, y luego seleccionar Modificar prueba de eficiencia física dentro del modulo de profesor.</p> <p>Nombre: Primera prueba de eficiencia física                  Fecha: 2008/06/06                  Grupo: 8502                  Resultados: Resultados en los indicadores.</p>
<b>Resultado:</b>	El sistema modifica la prueba de eficiencia física en la base de datos del sistema y muestra la nueva nivelación de los estudiantes en dicha prueba.
<b>Condiciones:</b>	Los datos introducidos deben de estar en el formato correcto.

Caso inválido

<b>Caso de uso:</b>	Gestionar prueba de eficiencia física.
<b>Caso de prueba:</b>	Modificar prueba introduciendo algún dato incorrecto.
<b>Entrada:</b>	<p>El profesor de educación física decide modificar algún dato de una prueba de eficiencia física, para ello debe seleccionar la opción de Eficiencia física, y luego seleccionar Modificar prueba de eficiencia física dentro del modulo de profesor.</p> <p>Nombre: Primera prueba de eficiencia física                  Fecha: 2008/06/06                  Grupo: 8502                  Resultados: Un resultado (2,3L)</p>

<b>Resultado:</b>	El sistema muestra un mensaje de error notificando que el valor del indicador es incorrecto.
<b>Condiciones:</b>	Los datos introducidos deben de estar en el formato correcto.

Tabla A10.22 Pruebas de Caja Negra para el Caso de uso “Gestionar prueba de eficiencia física” escenario “Modificar prueba de eficiencia física”.

Condición de Entrada	Casos Válidos	Casos no Válidos
Nombre	Seleccionar nombre	No seleccionar nombre
Fecha	Seleccionar fecha	No seleccionar fecha
Grupo	Seleccionar grupo	No seleccionar grupo

Caso válido

<b>Caso de uso:</b>	Gestionar prueba de eficiencia física.
<b>Caso de prueba:</b>	Reportes de nivelación introduciendo correctamente todos los datos.
<b>Entrada:</b>	El usuario decide realizar un reporte de la nivelación de una prueba de eficiencia física, para ello debe seleccionar la opción de Eficiencia física, y luego seleccionar Reportes de nivelación. Nombre: Primera prueba de eficiencia física Fecha: 2008/06/06 Grupo: 8502
<b>Resultado:</b>	El sistema muestra la nueva nivelación de los estudiantes en dicha prueba.
<b>Condiciones:</b>	Los datos introducidos deben de estar en el formato correcto.

Caso inválido

<b>Caso de uso:</b>	Gestionar prueba de eficiencia física.
<b>Caso de prueba:</b>	Reportes de nivelación introduciendo algún dato incorrecto.

<b>Entrada:</b>	El usuario decide realizar un reporte de la nivelación de una prueba de eficiencia física, para ello debe seleccionar la opción de Eficiencia física, y luego seleccionar Reportes de nivelación. Nombre: Primera prueba de eficiencia física Fecha: 2008/06/06 Grupo: Seleccione
<b>Resultado:</b>	El sistema muestra un mensaje de error notificando que el grupo no se a seleccionado.
<b>Condiciones:</b>	Los datos introducidos deben de estar en el formato correcto.

Tabla A10.23 Pruebas de Caja Negra para el Caso de uso “Gestionar prueba de eficiencia física” escenario “Reportes de nivelación”.