

UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS

Facultad 8

“Análisis, diseño e implementación de un sistema de registro y procesamiento de información estadística para los Campeonatos Zonales Nacionales de voleibol en las categorías menores.”

**TRABAJO DE DIPLOMA PARA OPTAR POR EL TÍTULO DE INGENIERO EN
CIENCIAS INFORMÁTICAS.**

Autores: Adrián Suárez Rodríguez

Yaneisis Vargas Labañino.

Tutor: MsC. Jeanne Cadet Ochoa.

Ciudad de La Habana
junio de 2010



“El deporte es fuente de voluntad, constancia, vigor físico y agilidad mental...”

Fidel Castro Ruz.

Declaración de autoría

Declaramos que somos los únicos autores del trabajo “Análisis, diseño e implementación de un sistema de registro y procesamiento de información estadística para los campeonatos zonales nacionales del voleibol en las categorías menores” y autorizamos a la Facultad 8 de la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

Para que así conste firmamos la presente a los 10 días del mes de junio del año 2010.

Autores:

Yaneisis Vargas Labañino

Nombre y apellidos del autor

Adrián Suárez Rodríguez

Nombre y apellidos del autor

Tutor:

MsC. Jeanne Cadet Ochoa

Nombre y apellidos del tutor

OPINIÓN DEL USUARIO DEL TRABAJO DE DIPLOMA

El Trabajo de Diploma, titulado “Análisis, diseño e implementación de un sistema de registro y procesamiento de información estadística para los Campeonatos Zonales Nacionales de voleibol en las categorías menores”, fue realizado en la Universidad de las Ciencias Informáticas. Esta entidad considera que, en correspondencia con los objetivos trazados, el trabajo realizado le satisface

Totalmente

Parcialmente en un ____ %

Los resultados de este Trabajo de Diploma le reportan a esta entidad los beneficios siguientes:

Trabajo muy importante para el voleibol cubano, puede ser usado en eventos internacionales.

Como resultado de la implantación de este trabajo se reportará un efecto económico que asciende a una suma importante en moneda nacional.

Y para que así conste, se firma la presente a los 24 días del mes de mayo del año 2010.

Argelio Hernández Rodríguez

Secretario General FCV

Representante de la entidad

Cargo

Firma

Cuño

AGRADECIMIENTOS

A mi papá por tanta ayuda y comprensión en estos 5 años.

A mi hermano por ser mi orgullo, camino a seguir y por tantos consejos.

A mi padrastro por ser más que un padre para mí.

A mis abuelos por ser lo más grande que tengo en la vida.

A mis tíos Chavi y Elena por tanta ayuda y confianza.

A mis hermanos Adalberto, Yamilka y Yamilet por tanto apoyo y creer en mí.

A mi cuñada Dunia por tanto cariño y amor, por esas comidas que me quitaron el sueño.

A mi familia en general por tanto cariño y apoyo.

A mi novia Yaneisis por robar mi corazón, nada de esto estuviera pasando hoy si no fuera por ella.

A la familia de mi novia por tratarme como un hijo y por tanto amor, especialmente a mi suegra Ramona, mi suegro Ernesto, mi cuñado Angelito y a Melquiades.

A Alejandro Medina por apoyarme tanto y ser mi guía en el desarrollo de esta tesis, por siempre estar presente cuando lo necesité, gracias a él comprendí el significado de la palabra incondicional.

A mi hermana Odalis, a mi tía Caridad y a mis primos Heber y Daymel por su apoyo.

A Danay por ser siempre mí mejor amiga.

A Luis Carlos, Luis Ernesto, Kiki, Raciél, Francisco, Andy, Rubén, Ramón y Pilo por ser como mis hermanos.

A Raciél Yera por tanta ayuda y consejos.

A mi tutor Jeanne Cadet por su apoyo, confianza y por aguantarnos tanto.

A los miembros del tribunal especialmente a María de los Ángeles por guiarnos en el transcurso de la tesis.

A mi prima Jana por su cariño y apoyo, por esas piñas que tanto me ayudaron a aliviar las penas.

A nuestro oponente Marcel Puentes por su ayuda y consejos.

A José y Elaine por hacerme sentir parte de su familia, por tanta ayuda y consejos.

Al equipo de béisbol de Las Tunas por ayudarme a cumplir uno de mis grandes sueños, especialmente a Yosvani y Yordanis Alarcón, Alexander Guerrero, Henry, Osmani y Ermidelio Urrutia, Joan Carlos Pedroso, Billy Arias y Jesús Guerra.

A la UCI y a la Revolución por permitir que lograra mis objetivos.

AGRADECIMIENTOS

Les agradezco de corazón a todos mis amigos por ayudarme a aguantar estos 5 años de estudio y sacrificio, por ayudarme siempre cuando lo necesité, no por ser los últimos dejan de ser menos importantes especialmente a Leo, Carlos Navarro, Karina, Luis Orlando, Alejandro Rey, Susel, Jorge Tabel, Alejandro Carreras, Elizabeth, Abelardo, Morgado, Jacobo, Yordanis, Maryenis, Yeni, Ana Delia, Yadira, Yusleymis, Neima, Osmani Montes de Oca, Ana María, Franklin, Hector Zúñiga, Viquillón, Laura, Jailen, Marianela, Ramiro y a todos los que de una forma u otra me ayudaron a llegar aquí.

Adrián

A mi tía Madeleine por todo el amor y la confianza que siempre me ha tenido, por ser como una madre para mí.

A mi padrastro por todo su cariño y el apoyo en todo momento.

A mis tías Mique y Fela por su apoyo y toda la ayuda que me han brindado.

A mi prima Marilín por todo su apoyo y consejos.

A toda mi familia que de una forma u otra estuvieron a mi lado y me apoyaron.

A mi novio por compartir estos cinco cursos a mi lado, por apoyarme en los momentos más difíciles, por su amor y su cariño, por darme fuerza para seguir, por soportar todas mis malacrianzas y confiar en mí, porque sin él todo hubiera sido diferente.

A mi suegra por todo el apoyo y la ayuda que me ha dado, por su confianza y amor.

A mi suegro por toda su ayuda y cariño depositado en mí.

A la familia de mi novio por acogerme como un miembro más, en especial a mi cuñado Ángel, a esos abuelos maravillosos, a Dalvis y Chavi.

A mi amiga Dagmara, porque sin su ayuda no hubiese podido entrar a esta Universidad.

A mis amigas Yenisleydis, Ana Delia, Maryenis y Ana María por estar siempre en los momentos que las necesitaba.

A mis amigos Alejandro Medina y Raciél Yera por estar siempre disponible cuando lo necesitaba, por toda su ayuda.

A mi tutor Jeanne por estar siempre a nuestro lado, por apoyarnos y confiar en nosotros.

A todos mis compañeros del grupo 8101 y 8503 por soportarme estos 5 cursos, por todas las ayudas que de una forma u otra me han brindado.

A los profesores miembros del tribunal y especialmente a la profesora María de los Ángeles, por toda su ayuda en los momentos que la necesité.

Yaneisis

DEDICATORIA

A mi mamá, porque todo lo que hoy soy se lo debo a ella. Por llorar a mi lado y apoyarme incondicionalmente cuando lo necesité. Si tuviera que ponerle nombre a mi vida sería el suyo porque no existen palabras para describirla.

Adrián

A los tres grandes amores de mi vida: mi mamá, mi papá y mi hermano, por todo su amor y cariño, por guiarme durante toda mi vida estudiantil, por todo su sacrificio incondicional y darme lo mejor para seguir adelante y hacer de mí lo que hoy soy.

Yaneisis

RESUMEN

En la actualidad no existe un sistema que gestione la información estadística generada en los Campeonatos Zonales Naciones del voleibol cubano. Este trabajo tiene como objetivo desarrollar un sistema que permita llevar a cabo el control estadístico de los objetivos pedagógicos trazados por la Federación Cubana de Voleibol (FCV) y los resultados deportivos obtenidos en estos campeonatos, además que tenga utilidad para posteriores análisis de la información con el fin de lograr que el voleibol en Cuba crezca desde la base. Todo el proceso de desarrollo fue guiado por la metodología tradicional RUP. Se realizó un estudio de algunas de las tecnologías, herramientas y lenguajes de programación más usados en la actualidad para el desarrollo de aplicaciones de escritorio. Se realizó el levantamiento de requisitos para definir los requerimientos funcionales y no funcionales del sistema propuesto utilizando técnicas de recopilación de información, así como los diagramas correspondientes al análisis y diseño del sistema. La creación del sistema le permitirá a la FCV conocer cuáles son las condiciones en que se encuentra el voleibol en las categorías menores y podrá comparar los resultados obtenidos en un año con los anteriores verificando que las estrategias trazadas sean las correctas.

Palabras Claves: Federación, estadística, categorías menores.

INTRODUCCIÓN.....	1
CAPÍTULO 1	5
FUNDAMENTACIÓN TEÓRICA.	5
1.1 INTRODUCCIÓN.....	5
1.2 ESTUDIO DE SISTEMAS ESTADÍSTICOS.	5
1.2.1 Data Volley.....	5
1.2.2 Sistema de Información de Voleibol (VIS).....	5
1.2.3 Stat Trak.....	6
1.2.4 VolleyStats.	6
1.3 METODOLOGÍAS DE DESARROLLO.....	7
1.3.1 Proceso Unificado de Desarrollo (RUP):.....	7
1.3.2 Microsoft Solutions Framework (MSF).	9
1.3.3 Extreme Programming (XP).....	10
1.3.5 Adaptive Software Development (ASD)	14
1.3.6 ¿Por qué RUP?	15
1.4 LENGUAJES DE MODELADO.	15
1.4.1 Lenguaje Unificado de Modelado (UML).....	16
1.4.2 Goal-oriented Requirement Language (GRL)	17
1.4.3 El lenguaje i*	17
1.5 HERRAMIENTAS CASE.....	18
1.5.1 ARGO-UML.....	18
1.5.2 Poseidón para UML.....	19
1.5.3 Visual Paradigm.	19
1.5.4 Rational Rose.....	21
1.5.5 ¿Por qué Visual Paradigm?	21
1.6 SISTEMAS GESTORES DE BASES DE DATOS (SGBD).....	22
1.6.1 MySQL.....	22
1.6.2 Oracle.....	23
1.6.3. PostgreSQL.....	24
1.6.4 ¿Por qué PostgreSQL?	25
1.7 LENGUAJES DE PROGRAMACIÓN.	25
1.7.1 Java.....	26
1.7.2 C#.....	26
1.7.3 C++	28
1.7.4 ¿Por qué Java?	29
1.8 ENTORNO DE DESARROLLO INTEGRADO.	29
1.8.1 Eclipse.....	29
1.8.2 NetBeans.....	30
1.8.3 Jbuilder.....	31
1.8.4 JCreator.....	31
1.8.3 ¿Por qué NetBeans?	32
1.9 FRAMEWORKS.....	32
1.9.1 Hibernate.....	32

1.9.2 JasperReports	33
1.10 PLATAFORMA DE DESARROLLO	34
1.10.1 Java 2 Enterprise Edition (J2EE)	34
1.11 ARQUITECTURA DE SOFTWARE	35
1.11.1 Arquitectura en tres capas	35
1.11.2 Arquitectura Modelo Vista Controlador (MVC)	36
1.11.3 ¿Por qué arquitectura en tres capas?	38
1.12 CONCLUSIONES	39
CAPÍTULO 2	40
ANÁLISIS Y DISEÑO	40
2.1 INTRODUCCIÓN	40
2.2 MODELO DEL NEGOCIO	40
2.2.1 Descripción del Negocio	40
2.2.2 Reglas del Negocio	41
2.2.5 Entidades del Negocio	42
2.2.6 Caso de Uso del Negocio	42
2.2.7 Diagrama de Casos de Uso del Negocio	42
2.2.9 Diagramas de Actividades	44
2.3 REQUISITOS DEL SISTEMA	44
2.3.1 Requerimientos Funcionales	44
2.3.2 Requerimientos No Funcionales	46
2.5 ACTORES DEL SISTEMA	47
2.6 CASOS DE USO DEL SISTEMA	48
2.8 MODELO DE ANÁLISIS	55
2.8.1 Diagramas de Clases de Análisis	56
2.9 MODELO DE DISEÑO	57
2.9.1 Patrones de Diseño	57
2.9.1.1 DAO	57
2.9.2 Diagramas de Interacción (Colaboración o Secuencia)	58
2.9.2.1 Diagramas de Secuencia	58
2.9.3 Diagramas de Clases del Diseño	60
2.9.4 Diseño de la Base de Datos	66
2.9.5 Modelo de Datos	67
2.9.5.1 Descripción de las Tablas de la Base de Datos	67
2.10 CONCLUSIONES	68
CAPÍTULO 3	69
IMPLEMENTACIÓN Y PRUEBA	69
3.1 INTRODUCCIÓN	69
3.2 MODELO DE IMPLEMENTACIÓN	69
3.2.1 Diagrama de Despliegue	69
3.2.2 Diagrama de Componentes	70
3.3 MODELO DE PRUEBA	70

ÍNDICE DE CONTENIDO

3.3.1 Pruebas de Caja Negra.....	71
3.3.2 Casos de Prueba.....	71
3.4 CONCLUSIONES.....	72
CONCLUSIONES GENERALES.....	73
RECOMENDACIONES.....	74
BIBLIOGRAFÍA.....	75
REFERENCIAS BIBLIOGRÁFICAS.....	77
GLOSARIO DE TÉRMINOS.....	80
ANEXOS.....	81

ÍNDICE DE TABLAS

Tabla 1: Diagramas de UML.....	16
Tabla 2: Actores del Negocio.....	41
Tabla 3: Trabajador del Negocio.....	42
Tabla 4: Entidades del Negocio.....	42
Tabla 5: Descripción del Caso de Uso del Negocio.....	44
Tabla 6: Actores del Sistema.....	48
Tabla 7: Descripción del Caso de Uso Gestionar Juego.....	53
Tabla 8: Descripción del Caso de Uso Consultar Boletín Diario.....	55
Tabla 9: Clases del Análisis.....	56
Tabla 10: Descripción de la Tabla de la Base de Datos "evento".....	68
Tabla 11: Caso de Prueba del Caso de Uso Consultar Reporte Boletín Diario.....	72
Tabla 12: Descripción del Caso de Uso Gestionar Evento.....	85
Tabla 13: Descripción del Caso de Uso Gestionar Jugador.....	89
Tabla 14: Descripción del Caso de Uso Consultar Evento.....	92
Tabla 15: Descripción del Caso de Uso Consultar Jugador.....	96
Tabla 16: Descripción del Caso de Uso Consultar Juego.....	98
Tabla 17: Descripción del Caso de Uso Consultar Reporte Resumen Estadístico de cada Equipo.....	102
Tabla 18: Descripción del Caso de Uso Consultar Reporte Resultados de los Objetivos Pedagógicos.....	103
Tabla 19: Descripción del Caso de Uso Consultar Reporte Resultados por Provincia.....	104
Tabla 20: Descripción del Caso de Uso Consultar Reporte Resultados Alcanzados por las Provincias.....	106
Tabla 21: Descripción del Caso de Uso Consultar Reporte Equipo Ideal.....	107
Tabla 22: Descripción del Caso de Uso Consultar Reporte Control Competitivo.....	108
Tabla 23: Descripción del Caso de Uso Consultar Reporte Historial de Usuario.....	109
Tabla 24: Descripción del Caso de Uso Autenticar Usuario.....	110
Tabla 25: Descripción del Caso de Uso Gestionar Usuario.....	114
Tabla 26: Caso de Prueba del Caso de Uso Gestionar Juego.....	124
Tabla 27: Caso de Prueba del Caso de Uso Gestionar Evento.....	127
Tabla 28: Caso de Prueba del Caso de Uso Gestionar Jugador.....	132
Tabla 29: Caso de Prueba del Caso de Uso Gestionar Usuario.....	135
Tabla 30: Caso de Prueba del Caso de Uso Autenticar Usuario.....	135
Tabla 31: Caso de Prueba del Caso de Uso Consultar Reporte Resumen Estadístico de cada Equipo.....	136
Tabla 32: Caso de Prueba del Caso de Uso Consultar Reporte Resultados por Provincia.....	137
Tabla 33: Caso de Prueba del Caso de Uso Consultar Reporte Resultados Alcanzados por las Provincias.....	138
Tabla 34: Caso de Prueba del Caso de Uso Consultar Reporte Resultados de los Objetivos Pedagógicos.....	138
Tabla 35: Caso de Prueba del Caso de Uso Consultar Reporte Control Competitivo.....	139
Tabla 36: Caso de Prueba del Caso de Uso Consultar Reporte Equipo Ideal.....	140
Tabla 37: Caso de Prueba del Caso de Uso Consultar Reporte Historial de Usuario.....	140
Tabla 38: Caso de Prueba del Caso de Uso Consultar Evento.....	142
Tabla 39: Caso de Prueba del Caso de Uso Consultar Juego.....	144
Tabla 40: Caso de Prueba del Caso de Uso Consultar Jugador.....	145
Tabla 41: Caso de Prueba del Caso de Uso Consultar Usuario.....	146
Tabla 42: Descripción de la Tabla de la Base de Datos "equipo".....	146

ÍNDICE DE TABLAS

Tabla 43: Descripción de la Tabla de la Base de Datos "evento_ equipo"	147
Tabla 44: Descripción de la Tabla de la Base de Datos "jugador"	148
Tabla 45: Descripción de la Tabla de la Base de Datos "juego"	149
Tabla 46: Descripción de la Tabla de la Base de Datos "usuario"	150
Tabla 47: Descripción de la Tabla de la Base de Datos "datosVariables"	151

ÍNDICE DE FIGURAS

Figura 1: Diagrama de Caso de Uso del Negocio.....	42
Figura 2: Diagrama de Actividades del Caso de Uso del Negocio Solicitar acciones de los juegos.....	44
Figura 3: Diagrama de Casos de Uso del Sistema	55
Figura 4: Diagrama de Clases del Análisis del Caso de Uso Gestionar Juego	56
Figura 5: Diagrama de Clases del Análisis del Caso de Uso Consultar Reporte Boletín Diario.	57
Figura 6: Diagrama de Secuencia del Caso de Uso Gestionar Juego Sección Insertar Juego .	59
Figura 7: Diagrama de Secuencia del Caso de Uso Gestionar Juego Sección Modificar Juego	59
Figura 8: Diagrama de Secuencia del Caso de Uso Gestionar Juego Sección Eliminar Juego.	60
Figura 9: Diagrama de Secuencia del Caso de Uso Gestionar Juego Sección Ver Juego	60
Figura 10: Diagrama de Clases del Diseño del Caso de Uso Gestionar Juego.....	61
Figura 11: Diagrama de Clases Persistentes.....	66
Figura 12: Modelo de Datos	67
Figura 13: Diagrama de Despliegue	69
Figura 14: Diagrama de Componentes	70
Figura 15: Diagrama de Clases del Análisis del Caso de Uso Autenticar Usuario	151
Figura 16: Diagrama de Clases del Análisis del Caso de Uso Gestionar Evento	151
Figura 17: Diagrama de Clases del Análisis del Caso de Uso Gestionar Jugador	152
Figura 18: Diagrama de Clases del Análisis del Caso de Uso Gestionar Usuario	152
Figura 19: Diagrama de Clases del Análisis del Caso de Uso Consultar Reporte Resultados de los Objetivos Pedagógicos	153
Figura 20: Diagrama de Clases del Análisis del Caso de Uso Consultar Reporte Resumen Estadístico de cada Equipo.	153
Figura 21: Diagrama de Clases del Análisis del Caso de Uso Consultar Reporte Equipo Ideal.	154
Figura 22: Diagrama de Clases del Análisis del Caso de Uso Consultar Reporte Control Competitivo	154
Figura 23: Diagrama de Clases del Análisis del Caso de Uso Consultar Reporte Resultados Alcanzados por las Provincias.....	154
Figura 24: Diagrama de Clases del Análisis del Caso de Uso Consultar Reporte Resultados por Provincias.....	155
Figura 25: Diagrama de Clases del Análisis del Caso de Uso Consultar Reporte Historial de Usuario.....	155
Figura 26: Diagrama de Clase del Análisis del Caso de Uso Consultar Evento	155
Figura 27: Diagrama de Clase del Análisis del Caso de Uso Consultar Juego	156
Figura 28: Diagrama de Clase del Análisis del Caso de Uso Consultar Jugador	156
Figura 29: Diagrama de Clase de Análisis del Caso de Uso Consultar Usuario.....	156
Figura 30: Diagrama de Secuencia del Caso de Uso Autenticar Usuario	157
Figura 31: Diagrama de Secuencia del Caso de Uso Gestionar Usuario Sección Insertar Usuario.....	157
Figura 32: Diagrama de Secuencia del Caso de Uso Gestionar Usuario Sección Eliminar Usuario	158
Figura 33: Diagrama de Secuencia del Caso de Uso Gestionar Usuario Sección Modificar Usuario.....	158
Figura 34: Diagrama de Secuencia del Caso de Uso Gestionar Usuario Sección Ver Usuario	159
Figura 35: Diagrama de Secuencia del Caso de Uso Gestionar Evento Sección Abrir/Cerrar Evento.	159
Figura 36: Diagrama de Secuencia del Caso de Uso Gestionar Evento Sección Insertar Evento.	160

ÍNDICE DE FIGURAS

Figura 37: Diagrama de Secuencia del Caso de Uso Gestionar Evento Sección Modificar Evento	160
Figura 38: Diagrama de Secuencia del Caso de Uso Gestionar Evento Sección Ver Evento .	161
Figura 39: Diagrama de Secuencia del Caso de Uso Gestionar Jugador Sección Insertar Jugador	161
Figura 40: Diagrama de Secuencia del Caso de Uso Gestionar Jugador Sección Modificar Jugador	162
Figura 41: Diagrama de Secuencia del Caso de Uso Gestionar Jugador Sección Eliminar Jugador	162
Figura 42: Diagrama de Secuencia del Caso de Uso Gestionar Jugador Sección Ver Jugador	163
Figura 43: Diagrama de Secuencia del Caso de Uso Consultar Reporte Boletín Diario	163
Figura 44: Diagrama de Secuencia del Caso de Uso Consultar Reporte Resultados de los Objetivos Pedagógicos	164
Figura 45: Diagrama de Secuencia del Caso de Uso Consultar Reporte Control Competitivo	164
Figura 46: Diagrama de Secuencia del Caso de Uso Consultar Reporte Historial de Usuario	165
Figura 47: Diagrama de Secuencia del Caso de Uso Consultar Reporte Resumen Estadístico de cada Equipo	165
Figura 48: Diagrama de Secuencia del Caso de Uso Consultar Reporte Equipo Ideal	166
Figura 49: Diagrama de Secuencia del Caso de Uso Consultar Reporte Resultados Alcanzados por las Provincias	166
Figura 50: Diagrama de Secuencia del Caso de Uso Consultar Reporte Resultados por Provincias	167
Figura 51: Diagrama de Secuencia del Caso de Uso Consultar Evento	167
Figura 52: Diagrama de Secuencia del Caso de Uso Consultar Juego	168
Figura 53: Diagrama de Secuencia del Caso de Uso Consultar Jugador	168
Figura 54: Diagrama de Secuencia del Caso de Uso Consultar Usuario	169
Figura 55: Diagrama de Clases del Diseño del Caso de Uso Gestionar Evento	170
Figura 56: Diagrama de Clases del Diseño del Caso de Uso Gestionar Jugador	171
Figura 57: Diagrama de Clases del Diseño del Caso de Uso Gestionar Usuario y Autenticar Usuario	172
Figura 58: Diagrama de Clases del Diseño del Caso de Uso Consultar Reporte Boletín Diario, Consultar Reporte Resumen Estadístico de cada Equipo, Consultar Reporte Resultados por Provincia, Consultar Reporte Resultados Alcanzados por las Provincias, Consultar Reporte Control Competitivo, Consultar Reporte Resultados de los Objetivos Pedagógicos.	173
Figura 59: Diagrama de Clases del Diseño del Caso de Uso Consultar Reporte Equipo Ideal	174
Figura 60: Diagrama de Clases del Diseño del Caso de Uso Consultar Reporte Historial de Usuario	175
Figura 61: Diagrama de Clases del Diseño del Caso de Uso Consultar Jugador	176
Figura 62: Diagrama de Clases del Diseño del Caso de Uso Consultar Juego	177
Figura 63: Diagrama de Clases del Diseño del Caso de Uso Consultar Evento	178
Figura 64: Diagrama de Clases del Diseño del Caso de Uso Consultar Usuario	179

INTRODUCCIÓN

El Voleibol es uno de los deportes más populares y practicados del mundo. En la actualidad la Federación Internacional de Voleibol consta de 220 países adscriptos. (universal, 2008). Si bien en sus inicios era más básico, con el paso del tiempo la tecnología invadió cada uno de sus componentes, a tal punto que cada renglón de juego puede ser estudiado, investigado y analizado por personas que se han integrado a equipos de trabajo como una de sus partes esenciales.

De esta manera, se logra que el nivel de jugadores y deporte crezca de una forma acelerada, pues la aplicación de programas informáticos posibilita conocer no solo al contrario, sino a su propio equipo a modo de radiografía, que muestra fortalezas y debilidades. A pesar de que es común en competencias de nivel internacional observar la influencia de los adelantos tecnológicos, ese trabajo se ha incorporado a las categorías menores, como herramienta para perfeccionar el trabajo con la base. Se logra así que los imberbes practicantes arriben a la etapa juvenil con un nivel superior al que alcanzarían si se prescindiera de la ciencia.

El Voleibol es uno de los deportes que se tienen en cuenta sus estadísticas con el fin de conocer el rendimiento propio sobre la base de datos relevados de cada jugador en particular y de los equipos. Con estas se pueden registrar todas las acciones que posibiliten marcar puntos, llevándose así los rendimientos deportivos de cada equipo y jugador, teniendo en cuenta la importancia que tiene su análisis para programar estrategias en función de mejoras en los mismos.

El uso de sistemas informáticos se ha extendido a nivel mundial en la esfera del deporte, donde son liderados por los países desarrollados, contando estos con sofisticados sistemas para evaluar patrones de rendimiento y conducta y tener para un futuro el comportamiento tanto de los atletas como de los equipos. En el mundo existe una gran variedad de sistemas destinados a llevar el control de los juegos de voleibol, tales como el Data Volley, el Sistema de Información de Voleibol y el Stat Trak. Con estos sistemas se obtienen datos de las distintas acciones de juego obteniendo una clara visión de cómo se desempeña cada jugador y la evolución de cada equipo.

El deporte cubano está enmarcado en un desarrollo de estas tecnologías que proporcionan grandes ventajas a los sistemas informáticos para el voleibol, por la rapidez y eficacia con que procesan la información. La FCV hoy en día tiende a desarrollarse en cuanto a estos sistemas

INTRODUCCIÓN

informáticos, desplegándose así software tales como el de Estudio de Contrarios para el Voleibol y Control Estadístico del Rendimiento Técnico – Táctico. La FCV contando con estos sistemas no le permite llevar todo un registro de estadísticas en las categorías menores por los datos específicos que se manejan en las mismas, actualmente este proceso es registrado en papeles o documentos digitales, que no les resultan muy eficiente por la rapidez y eficiencia de las estadísticas que deben llevar de cada juego, ya que se debe entregar a cada entrenador los reportes una vez finalizado el partido. Este trabajo llevado como es hasta el momento provoca errores a la hora de sacar las estadísticas, lo que trae como consecuencia falsas expectativas del rendimiento de los atletas y equipos, provocando a la vez que el entrenador no conozca la realidad de los mismos.

La FCV no cuenta con los recursos suficientes que les permita comprar un software destinado a estos eventos por los altos costos que poseen, es por eso que la Universidad de las Ciencias Informáticas desempeña un papel fundamental buscando soluciones informáticas a diversas problemáticas como la que se presenta a continuación: aún cuando hay suficiente información de los sistemas de registro y procesamiento estadístico en el voleibol, Cuba no cuenta con un sistema de registro y procesamiento de información estadística que le permita llevar todo este proceso en los Campeonatos Zonales Nacionales en las categorías menores del voleibol cubano. Surge así el **problema**: ¿Cómo facilitar el registro y procesamiento de forma rápida y eficiente de toda la información estadística que se genera en los Campeonatos Zonales Nacionales de Voleibol en las categorías menores?

Como **objeto de estudio** proceso de gestión y procesamiento de información estadística en el voleibol; y como **campo de acción** los sistemas de gestión y procesamiento de información estadística para el voleibol. Para dar respuesta al problema se realizará este trabajo que tiene como **objetivo** elaborar un sistema de registro y procesamiento de información estadística para los Campeonatos Zonales Nacionales de Voleibol en las categorías menores y como **preguntas científicas**:

- ¿Cuáles son los antecedentes históricos de los sistemas de registro y procesamiento estadístico en el voleibol?
- ¿Cuál es la caracterización del estado actual de la utilización de sistemas de gestión y procesamiento estadístico para el voleibol?
- ¿Cuáles son las metodologías, lenguajes de programación y herramientas que existen para el desarrollo de aplicaciones de escritorio?
- ¿Cómo desarrollar un sistema de gestión y procesamiento estadístico para los

INTRODUCCIÓN

Campeonatos Zonales Nacionales de Voleibol en las categorías menores?

Como **tareas de investigación** se tiene:

- Estudio de los antecedentes históricos de los sistemas de registro y procesamiento estadístico en el voleibol.
- Caracterización del estado actual en la elaboración de sistemas de gestión y procesamiento estadístico para el voleibol.
- Estudio de las diferentes metodologías, lenguajes de programación y herramientas que existen para el desarrollo de aplicaciones de escritorio.
- Documentación y justificación de la selección de las herramientas y tecnologías a utilizar para realizar el sistema.
- Desarrollo y prueba del sistema de gestión y procesamiento estadístico.

Para realizar la investigación se usaron métodos teóricos como la revisión documental, este permite realizar una investigación de toda la bibliografía relacionada con el objeto de estudio. El analítico-sintético y el inductivo-deductivo que actúan en el plano procedimental como operaciones lógicas del pensamiento que están presentes en todo el proceso de obtención y valoración de los datos requeridos. La modelación para realizar diferentes modelos abstractos, relacionados con el objeto de estudio y el campo de acción, donde finalmente se modeló en la realidad objetiva el resultado esperado. La generalización, donde se obtuvieron resultados parciales los cuales se generalizaron como aporte del conocimiento en el campo investigativo. En cuanto a los métodos empíricos, se aplicaron las entrevistas, que fueron útiles por el enriquecimiento del conocimiento adquirido con personas de gran experiencia en el voleibol.

La solución propuesta tiene gran importancia para los técnicos del voleibol en el país, la cual permitirá diferenciar y determinar el comportamiento de los rendimientos competitivos de cada equipo, ofreciendo una información objetiva cuantificable en el voleibol moderno, para el análisis y posterior evaluación de un partido o entrenamiento. Tecnológicamente el sistema automatiza el proceso de gestión de estadísticas en cada juego y económicamente ayudará al país pues no tendrá que comprar un software con estas características.

El presente trabajo de investigación está conformado por tres capítulos con los siguientes temas a tratar:

Capítulo 1. Fundamentación teórica: Descripción de sistemas estadísticos en el voleibol, un estudio y análisis de los lenguajes, metodologías, herramientas y tecnologías posibles a usar

INTRODUCCIÓN

para dar una solución óptima a la problemática planteada. Descripción de la arquitectura empleada en el sistema.

Capítulo 2. Análisis y diseño del sistema: Características y funcionalidades con que el sistema debe contar, casos de uso que guiarán todo el desarrollo de este y una propuesta de sistema acorde con el estudio realizado. Documentación generada durante el diseño de la aplicación que engloba una descripción de forma general de las clases utilizadas en los diagramas de clases del diseño, diagrama de interacción por cada caso de uso a cumplir, así como el modelo de la base de datos.

Capítulo 3. Implementación y prueba: Se presentan los diagramas de implementación y los casos de prueba necesarios para asegurar que el sistema cumpla con las expectativas del cliente.

Capítulo 1

Fundamentación Teórica.

1.1 Introducción.

Las nuevas tecnologías de la información y las comunicaciones (NTIC) han conseguido un cambio profundo en la sociedad. Estas tecnologías cuentan a la vez con un conjunto de innovaciones pero también con las herramientas que permiten una redefinición radical del funcionamiento de la sociedad. Para todo tipo de aplicaciones las NTIC son herramientas que facilitan el aprendizaje, desarrollo de habilidades y distintas formas de informarse. Para la elaboración de un sistema de gestión y procesamiento estadístico es necesario realizar un estudio de la situación actual de los mismos en el mundo, así como de las tecnologías y herramientas que deben considerarse a la hora de desarrollar este tipo de aplicaciones.

1.2 Estudio de sistemas estadísticos.

Son muchas las aplicaciones que se utilizan en la actualidad a nivel mundial para la gestión de estadísticas en el voleibol, a continuación se realiza una breve caracterización de las más usadas.

1.2.1 Data Volley.

Data Volley es un paquete de software estadístico proporcionado por Data Project. Es un software disponible comercialmente con el apoyo de un número de ligas de voleibol influyentes, como las ligas italianas y americanas. Data Volley sólo requiere una persona para tener las estadísticas de un juego, aunque una segunda persona, se recomienda durante el entrenamiento. Registra solamente los tiros de ataque. (Brown, 2009)

1.2.2 Sistema de Información de Voleibol (VIS).

VIS es el software estadístico producido por la Federación Internacional de Voleibol (FIVB). Se utiliza para generar estadísticas en todos los eventos de la FIVB, incluyendo los Campeonatos Mundiales y eventos de playa.

VIS tiene estadísticas en cada tiro en el juego y las filas de cada disparo, ya sea:

- Excelente
- Normal
- Error

Las estadísticas son tomadas por dos grupos de tres personas, situadas en los extremos de la pista de voleibol. Una persona es responsable de registrar las estadísticas en la computadora, otra para anunciar los disparos y la última para escribir las estadísticas sobre el papel, en caso de fallo de software. Otra persona es responsable de corregir los errores cometidos por ambos extremos. La principal desventaja de VIS es que necesita mucha mano de obra (siete personas por partido). Contiene módulos para cargar directamente las estadísticas de la FIVB y también genera una serie de informes estándares que deben ser firmados por los jurados de juego en cualquier campeonato de la FIVB. También elabora informes de los mejores jugadores en un torneo que pueden ser distribuidos a los equipos y medios de comunicación. (Brown, 2009)

1.2.3 Stat Trak.

Stat Trak es un programa completo de estadísticas fácil de usar. Calcula los servicios de ases, porcentaje de servicios, asistencias por juego, la clasificación del equipo, etc. Crea e imprime informes por equipos, múltiples equipos y cada jugador. Es ideal para todos los niveles, incluyendo las ligas de voleibol. (Brown, 2009)

1.2.4 VolleyStats.

VolleyStats es una aplicación dirigida a los equipos de voleibol que necesiten de una toma rápida, cómoda, útil y eficaz de las estadísticas de juego y entrenamientos, obteniendo así una completa información tanto de los partidos como de los jugadores de forma individual. El software permite registrar estadísticas en los partidos de voleibol desde un dispositivo portátil. De forma sencilla y rápida, VolleyStats está pensado y diseñado para que una sola persona pueda anotar todos los datos útiles de ambos equipos en un partido o de un entrenamiento. Permite consultar todos los datos estadísticos de un partido o entrenamiento tales como: aciertos, fallos, efectividad, direcciones de saques, ataques y zonas de bloqueo. Tanto de forma conjunta como por jugador. (Integra Tecnología y Comunicación, 2009)

Como se ha descrito anteriormente cada una de estas aplicaciones tiene sus potencialidades, todas destinadas al manejo de gestión de estadísticas, pero ninguna posee las características que necesita la FCV. Cuando se trabaja con el término categorías menores se refiere a las

categorías en que se compite en los Campeonatos Zonales Nacionales en Cuba, las cuales son: primaria, secundaria, pioneril, escolar, estudiantil y juvenil. La FCV necesita llevar un control de los resultados pedagógicos, donde manejan datos muy particulares que no se hacen en los otros sistemas ya descritos anteriormente, para gestionar esos resultados usan porcentos y parámetros definidos por ellos mismos que pueden cambiar en cualquier momento. Además, a Cuba se le dificulta en gran medida adquirir esos sistemas por las licencias de los mismos y los altos precios que poseen.

1.3 Metodologías de desarrollo.

Actualmente se cuenta con una buena cantidad de metodologías tradicionales como Rational Unified Process (RUP), Microsoft Solutions Framework (MSF), y metodologías ágiles como Extreme Programming (XP), SCRUM, Dynamic, Adaptive Software Development (ASD).

1.3.1 Proceso Unificado de Desarrollo (RUP):

Es un proceso de desarrollo de software. Es un marco de trabajo genérico que puede especializarse para una gran variedad de sistemas software, para diferentes áreas de aplicación, diferentes tipos de organizaciones, diferentes niveles de aptitud y diferentes tamaños de proyectos. Utiliza el Lenguaje Unificado de Modelado para preparar todos los esquemas de un sistema software. (Jacobson, y otros, 2000)

Características:

- Dirigido por casos de uso.
- Centrado en la arquitectura.
- Iterativo e incremental. (Jacobson, y otros, 2000)

Fases

- Conceptualización (Concepción o Inicio): Se desarrolla una descripción del producto final y se presenta el análisis del negocio para el producto.
- Elaboración: Se especifican en detalle la mayoría de los casos de uso del producto y se diseña la arquitectura del sistema.
- Construcción: Se crea el producto, la línea base de la arquitectura crece hasta convertirse en el sistema completo.

FUNDAMENTACIÓN TEÓRICA

- Transición: Cubre el periodo durante el cual el producto se convierte en versión beta. (Jacobson, y otros, 2000)

Flujos de trabajo:

- Modelamiento del negocio.
- Requerimientos.
- Implementación.
- Prueba.
- Administración del proyecto.
- Administración de configuración y cambios.
- Ambiente. (Jacobson, y otros, 2000)

Ventajas:

- Mitigación temprana de posibles riesgos altos.
- Progreso visible en las primeras etapas.
- Temprana retroalimentación que se ajuste a las necesidades reales.
- Gestión de la complejidad.
- Conocimiento adquirido en una iteración puede aplicarse de iteración a iteración.

Desventajas:

- La evaluación de riesgos es compleja.
- Excesiva flexibilidad para algunos proyectos.
- El cliente deberá ser capaz de describir y entender a un gran nivel de detalle para poder acordar el alcance del proyecto con él.

Buenas prácticas

- Abordar las cuestiones de alto riesgo y valor en las primeras iteraciones.
- Usuarios involucrados continuamente.
- Verificar continuamente la calidad desde el principio y con frecuencia.
- Aplicar casos de uso.
- Modelar el software visualmente.
- Gestión cuidadosa de requisitos.
- Control de cambios. (Díaz, y otros, 2009)

1.3.2 Microsoft Solutions Framework (MSF).

MSF es un compendio de las mejores prácticas en cuanto a administración de proyectos se refiere. Más que una metodología rígida de administración de proyectos, MSF es una serie de modelos que puede adaptarse a cualquier proyecto de tecnología de información. (GPI, 2009)

Fases

- **Visión y Alcance:**

La fase de visión y alcance trata uno de los requisitos fundamentales para el éxito del proyecto, la unificación del equipo detrás de una visión común. El equipo debe tener un enfoque claro de lo que quiere lograr y ser capaz de indicarlo en términos que motiven a todo el equipo y al cliente. Se definen los líderes y responsables del proyecto, adicionalmente se identifican las metas y objetivos a alcanzar; estas últimas se deben respetar durante la ejecución del proyecto en su totalidad, y se realiza la evaluación inicial de riesgos del proyecto.

- **Planificación:**

En esta fase es cuando la mayor parte de la planeación para el proyecto es terminada. El equipo prepara las especificaciones funcionales, realiza el proceso de diseño de la solución, y prepara los planes de trabajo, estimaciones de costos y cronogramas de los diferentes entregables del proyecto.

- **Desarrollo:**

Durante esta fase el equipo realiza la mayor parte de la construcción de los componentes (tanto documentación como código), sin embargo, se puede realizar algún trabajo de desarrollo durante la etapa de estabilización en respuesta a los resultados de las pruebas. La infraestructura también es desarrollada durante esta fase.

- **Estabilización:**

En esta fase se conducen pruebas sobre la solución, las pruebas de esta etapa enfatizan el uso y operación bajo condiciones realistas. El equipo se enfoca en priorizar y resolver errores y preparar la solución para el lanzamiento.

- **Implantación:**

Durante esta fase el equipo implanta la tecnología base y los componentes relacionados, estabiliza la instalación, traspasa el proyecto al personal de soporte y operaciones, y obtiene la aprobación final del cliente. (GPI, 2009)

Ventajas

- Sirve para grandes y pequeños proyectos.

Desventajas

- Se convierte en un trabajo bastante largo, ya que para cada fase se debe documentar profundamente todo lo que se haga, pero no deja de ser un modelo que tiene buenos resultados.
- Su desarrollo se basa en tecnología Microsoft la cual es cara y limita mucho las herramientas de desarrollo.
- Este modelo no utiliza UML.

1.3.3 Extreme Programming (XP)

XP es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en el desarrollo del software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo. XP se basa en retroalimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. XP se define como especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico. (Beck, 1999)

Trata de evitar los siguientes problemas del desarrollo de Software:

- Retrasos en la planificación.
- Sistemas deteriorados.
- Requisitos mal comprendidos.
- Cambios de negocio.
- Falsa riqueza.
- Cambios de personal.

Las principales características son:

- Metodología ligera.
- Cercano al desarrollo.
- Se basa en historias de usuarios.
- Fuerte comunicación con el cliente.
- El código fuente pertenece a todos.
- Programación por parejas.

- Pobre en cuanto a documentación. (Molpeceres, 2003)

Fases

- Exploración.
- Planificación de la entrega.
- Iteraciones.
- Producción.
- Mantenimiento.
- Muerte del proyecto. (Letelier, y otros, 2006)

Ventajas:

- Planificación más transparente para los clientes, conocen las fechas de entrega de funcionalidades vitales para su negocio.
- Permite definir en cada iteración cuáles son los objetivos de la siguiente.
- Permite tener realimentación de los usuarios.
- La presión está a lo largo de todo el proyecto y no en una entrega final.

Desventajas:

- Delimitar el alcance del proyecto con el cliente: el alcance y la funcionalidad exacta del software nunca se define formal y contractualmente.
- Evita cualquier tipo de documentación fuera del código fuente: UML desempeña un papel casi nulo, por lo que no se puede representar todo lo que se debería como las dependencias entre componentes lo que dificulta utilizar la experiencia ganada en otros desarrollos ya que no se ha archivado o anotado nada.
- La implementación se desarrolla por parejas.

Prácticas básicas de la programación extrema

- **El juego de la planificación**

Es un espacio frecuente de comunicación entre el cliente y los programadores. El equipo técnico realiza una estimación del esfuerzo requerido para la implementación de las historias de usuario y los clientes deciden sobre el ámbito y tiempo de las entregas y de cada iteración.

- **Entregas pequeñas**

FUNDAMENTACIÓN TEÓRICA

La idea es producir rápidamente versiones del sistema que sean operativas, aunque obviamente no cuenten con toda la funcionalidad pretendida para el sistema pero sí que constituyan un resultado de valor para el negocio. Una entrega no debería tardar más de tres meses.

- **Metáfora**

En XP no se enfatiza la definición temprana de una arquitectura estable para el sistema. Dicha arquitectura se asume evolutiva y los posibles inconvenientes que se generarían por no contar con ella explícitamente en el comienzo del proyecto se solventan con la existencia de una metáfora.

- **Diseño simple**

Se debe diseñar la solución más simple que pueda funcionar y ser implementada en un momento determinado del proyecto.

- **Pruebas**

La producción de código está dirigida por las pruebas unitarias. Estas son establecidas antes de escribir el código y son ejecutadas constantemente ante cada modificación del sistema.

- **Refactorización**

La refactorización es una actividad constante de reestructuración del código con el objetivo de remover duplicación de código, mejorar su legibilidad, simplificarlo y hacerlo más flexible para facilitar los posteriores cambios.

- **Programación en parejas**

Toda la producción de código debe realizarse con trabajo en parejas de programadores.

- **Propiedad colectiva del código**

Cualquier programador puede cambiar cualquier parte del código en cualquier momento.

- **Integración continua**

Cada pieza de código es integrada en el sistema una vez que esté lista. Así, el sistema puede llegar a ser integrado y construido varias veces en un mismo día.

- **Cuarenta horas por semana**

Se debe trabajar un máximo de cuarenta horas por semana. No se trabajan horas extras en dos semanas seguidas.

- **Cliente in-situ**

El cliente tiene que estar presente y disponible todo el tiempo para el equipo. Gran parte del éxito del proyecto XP se debe a que es el cliente quien conduce constantemente el trabajo

hacia lo que aportará mayor valor de negocio y los programadores pueden resolver de manera inmediata cualquier duda asociada.

- **Estándares de programación**

XP enfatiza la comunicación de los programadores a través del código, con lo cual es indispensable que se sigan ciertos estándares de programación (del equipo, de la organización u otros estándares reconocidos para los lenguajes de programación utilizados). Los estándares de programación mantienen el código legible para los miembros del equipo, facilitando los cambios. (Letelier, y otros, 2006)

1.3.4 Scrum

Desarrollada por Ken Schwaber, Jeff Sutherland y Mike Beedle. Define un marco para la gestión de proyectos, que se ha utilizado con éxito durante los últimos 10 años. Está especialmente indicada para proyectos con un rápido cambio de requisitos. Sus principales características se pueden resumir en dos. El desarrollo de software se realiza mediante iteraciones, con una duración de treinta días. El resultado de cada iteración es un incremento ejecutable que se muestra al cliente. La segunda característica importante es las reuniones a lo largo del proyecto, entre ellas destaca la reunión diaria de 15 minutos del equipo de desarrollo para coordinación e integración. (Letelier, y otros, 2006)

Scrum se focaliza en priorizar el trabajo en función del valor que tenga para el negocio, maximizando la utilidad de lo que se construye y el retorno de inversión. Está diseñado especialmente para adaptarse a los cambios en los requerimientos, por ejemplo en un mercado de alta competitividad. Los requerimientos y las prioridades se revisan y ajustan durante el proyecto en intervalos muy cortos y regulares. De esta forma, se puede adaptar en tiempo real el producto que se está construyendo a las necesidades del cliente. Se pretende entregar un software que realmente resuelva las necesidades, aumentando la satisfacción del cliente. El equipo se focaliza en una única cosa: construir software de calidad. Por el otro lado, la gestión de un proyecto Scrum se basa en definir cuáles son las características que debe tener el producto a construir (qué construir, qué no y en qué orden) y en remover cualquier obstáculo que pudiera entorpecer la tarea del equipo de desarrollo. Se busca que los equipos sean lo más efectivos y productivos posible.

Ventajas

- Posibilidad de ajustar la funcionalidad en base a la necesidad de negocio del cliente.

- Visualización del proyecto día a día.
- Alcance viable.
- Equipos integrados y comprometidos con el proyecto.

Desventajas

- No genera toda la documentación de otras metodologías.
- No es apto para todos los proyectos.

1.3.5 Adaptive Software Development (ASD)

Sus principales características son: iterativo, orientado a los componentes software más que a las tareas y tolerante a los cambios. El ciclo de vida que propone tiene tres fases esenciales: especulación, colaboración y aprendizaje. En la primera de ellas se inicia el proyecto y se planifican las características del software; en la segunda desarrollan las características y finalmente en la tercera se revisa su calidad, y se entrega al cliente. La revisión de los componentes sirve para aprender de los errores y volver a iniciar el ciclo de desarrollo. (Letelier, y otros, 2006)

Fases

- **Especular**

En esta fase se establecen los principales objetivos y metas del proyecto en su conjunto y comprende las limitaciones con las que operará el proyecto. Se realizan estimaciones para la correcta atención de los trabajadores que se mueven dentro de plazos de forma que puedan priorizar sus tareas. Se decide el número de iteraciones para consumir el proyecto, prestando atención a las características que pueden ser utilizadas por el cliente al final de la iteración.

- **Colaborar**

Esta fase se centra en la mayor parte del desarrollo manteniendo un componente cíclico, es decir, se desarrollan las características del software.

- **Aprender**

En esta fase se captura lo que se ha aprendido, tanto positivo como negativo. Es un elemento crítico para la eficacia de los equipos. Se revisa la calidad, y si no se tiene errores se entrega al cliente.

Ventajas

FUNDAMENTACIÓN TEÓRICA

- La tercera fase del ciclo de vida, sirve para aprender de los errores y volver a iniciar el ciclo de desarrollo.
- Utiliza información disponible acerca de cambios para mejorar el comportamiento del software.
- Anticipa cambios y trata automáticamente con ellos dentro de un programa en ejecución, sin la necesidad de un programador.

Desventaja

- El ciclo de desarrollo es bueno permitiendo entregar productos con alta calidad, pero hay errores que no son detectados o cambiados durante las reuniones lo que afecta tanto a la calidad del producto como a su costo total.

1.3.6 ¿Por qué RUP?

Después de analizar varias de las metodologías tanto ligeras como tradicionales se decidió utilizar a RUP pues *junto con el Lenguaje Unificado de Modelado constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos* (Mosqueda, 2008). XP plantea en sus prácticas básicas que el cliente forma parte del equipo de desarrollo, aspecto que no se puede cumplir en este caso debido a que el mismo no dispone del tiempo suficiente para estar inmerso en el desarrollo del sistema, sin embargo, en RUP no es necesario que el cliente esté tan comprometido en el proceso de desarrollo, en caso que haga falta se hará a través de reuniones. Por otra parte, en la Programación Extrema los requisitos no funcionales tienen un papel secundario y en la práctica estos son tan importantes como los funcionales. MSF se basa en el desarrollo de tecnología Microsoft la cual es cara y limita mucho las herramientas de desarrollo. Otro aspecto positivo del Proceso Unificado de Desarrollo es que propone todos los artefactos que tienen utilidad para la ingeniería de software, sin embargo, se puede utilizar de ellos sólo los que se necesitan para crear un proceso de desarrollo de software propio. Scrum no genera toda la documentación necesaria para un proyecto, además no es adaptable a todo tipo de proyecto. ASD permite obtener productos de muy buena calidad pero no son detectados todos los errores, por lo que afecta la calidad del producto y su costo total.

1.4 Lenguajes de modelado.

1.4.1 Lenguaje Unificado de Modelado (UML)

El UML es un lenguaje para construir modelos; no guía al desarrollador en la forma de realizar el análisis y diseño orientados a objetos ni le indica cual proceso de desarrollo adoptar. (Jacobson, y otros, 2000)

Su propósito es visualizar, especificar, construir y documentar proyectos de software. (Giraldo, y otros, 2005)

Tiene una notación gráfica muy expresiva que permite representar en mayor o menor medida todas las fases de un proyecto informático: desde el análisis con los casos de uso, el diseño con los diagramas de clases, objetos, hasta la implementación y configuración con los diagramas de despliegue. UML sirve para el modelado completo de sistemas complejos, tanto en el diseño de los sistemas software como para la arquitectura hardware donde se ejecuten. Otro objetivo de este modelado visual es que sea independiente del lenguaje de implementación, de tal forma que los diseños realizados usando UML se puedan implementar en cualquier lenguaje que soporte las posibilidades del mismo (principalmente lenguajes orientados a objetos).

Diagramas	Tipos
Diagramas de estructuras	Diagrama de clase – Diagrama de objeto Diagrama de componente – Diagrama de implementación
Diagramas de comportamiento	Diagrama de actividades - Diagrama de CU Diagrama de estado
Diagramas de iteración	Diagrama de secuencia – Diagrama de colaboración

Tabla 1: Diagramas de UML.

Ventajas:

- Mayor rigor en la especificación.
- Se puede usar para modelar distintos tipos de sistemas de software, de hardware y organizaciones del mundo real.
- Permite realizar una verificación y validación del modelo realizado.
- Se pueden automatizar determinados procesos y permite generar código a partir de los modelos y a la inversa (a partir del código fuente generar los modelos).
- Sirve para el modelado completo de sistemas complejos, tanto en el diseño de los

sistemas software como para la arquitectura hardware donde se ejecuten.

Desventajas:

- No es un método de desarrollo, no dice cómo pasar del análisis al diseño y de este al código.

1.4.2 Goal-oriented Requirement Language (GRL)

Es un lenguaje para apoyar el modelado orientado a objetivos con requisitos no funcionales. Distingue tres principales categorías de conceptos: elementos intencionales, relaciones intencionales y actores (que no admiten especializaciones). GRL posee elementos adicionales de justificación y/o contextualización como creencias, correlaciones, tipos de contribuciones y etiquetas de evaluación para especificar los estados de satisfacción. (Ayala, y otros, 2001)

1.4.3 El lenguaje i*

Permite expresar de forma clara y sencilla los objetivos de los actores que aparecen en los modelos y las dependencias entre ellos. Cuenta con una notación gráfica que permite tener una visión intuitiva y unificada del entorno modelado mostrando tales actores y dependencias. Es sencillo y clarifica enormemente los objetivos de los actores que aparecen en el modelo y las dependencias entre ellos. Durante la construcción de un modelo basado en i* se deben detectar los actores y sus necesidades u objetivos. Estas necesidades deben ser cubiertas por otras partes del sistema o por otros sistemas. El uso de este lenguaje entraña un riesgo que se descubre pronto, pues no existe una definición única del lenguaje. Propone el uso de dos modelos, cada uno correspondiente a un nivel de abstracción: el nivel intencional representado por el Strategic Dependency Model (SD) y el nivel racional representado por el Strategic Rationale Model (SR). El modelo SD consiste en un conjunto de nodos que representan actores y un conjunto de relaciones entre estos actores. Las relaciones representan dependencias que permiten expresar que un actor depende de otro actor para conseguir un determinado propósito. El modelo permite definir la importancia de la dependencia para cada uno de los actores que intervienen en ella. Existen tres grados de importancia: abierta, comprometida o crítica. El modelo SR permite visualizar los elementos intencionales dentro de los límites de un actor para refinar el modelo SD y añadirle capacidad de razonamiento. (Ayala, y otros, 2001)

1.4.4 ¿Por qué UML?

Se escoge UML ya que la metodología que se va a usar es RUP y esta utiliza este lenguaje de modelado para preparar todos los esquemas de un sistema software.

GRL solo permite relaciones entre actores. Las versiones de i* no están completas. UML permite expresar de una forma gráfica un sistema de forma que otro lo puede entender. Permite especificar cuáles son las características de un sistema antes de su construcción. Los propios elementos gráficos sirven como documentación del sistema desarrollado, sirviendo para su futura revisión.

1.5 Herramientas CASE.

Las Herramientas CASE (Computer Aided Software Engineering, Ingeniería de Software Asistida por Ordenador) son diversas aplicaciones destinadas a aumentar la productividad en el desarrollo de software reduciendo el coste de las mismas en términos de tiempo y de dinero. Estas herramientas pueden ayudar en todos los aspectos del ciclo de vida de desarrollo del software en tareas como el proceso de realizar un diseño del proyecto, cálculo de costes, implementación de parte del código automáticamente con el diseño dado, compilación automática, documentación o detección de errores entre otras. (Camaleón, 2010)

1.5.1 ARGO-UML.

Herramienta que contiene funciones avanzadas en las etapas de diseño y modelación de software. (Giraldo, y otros, 2005)

Es una aplicación de diagramado de UML escrita en Java y publicada bajo la Licencia BSD (Berkeley Software Distribution) código abierto. Dado que es una aplicación Java, está disponible en cualquier plataforma soportada por la misma. Sin embargo, desde la versión 0.20, ArgoUML está incompleto. No es conforme completamente a los estándares UML y carece de soporte completo para algunos tipos de diagramas incluyendo los diagramas de secuencia y los de colaboración.

Características:

- Tiene muy buenas ayudas.
- Modular y entendible.
- Soporta todas las especificaciones UML.
- Integrado con la Web. (Giraldo, y otros, 2005)

Ventajas:

- Genera código automáticamente.

- Propone soluciones a algunos errores.
- Panel de propiedades y de tareas pendientes bastante útil.

Desventajas:

- Instalación costosa.
- Los modelos a veces no pueden ser re-abiertos.
- No hay llamadas reflexivas en los diagramas de secuencia.

1.5.2 Poseidón para UML.

Es una herramienta para modelar cualquier clase de sistema que esté o no relacionado con programación. (Giraldo, y otros, 2005)

Características:

- Soporta diagramas UML.
- Opciones avanzadas de impresión.
- Soporta gráficos en la mayoría de formatos.
- Generación de código para Java y exportación como HTML.
- Fácil de instalar y actualizar.
- Compatibilidad entre ediciones. (Giraldo, y otros, 2005)

Ventajas:

Herramienta hecha completamente en Java, por lo que es independiente de la plataforma.

Interfaz de usuario muy bien diseñada, fácil de aprender a usar e intuitiva.

1.5.3 Visual Paradigm.

Visual Paradigm es una herramienta considerada muy completa y fácil de usar, con soporte multiplataforma y que proporciona excelentes facilidades de interoperabilidad con otras aplicaciones. Fue creada para el ciclo vital completo del desarrollo de software que lo automatiza y acelera, permitiendo la captura de requisitos, análisis, diseño e implementación. Tiene la capacidad de crear el esquema de clases a partir de una base de datos y crear la definición de la misma a partir del esquema de clases. Permite invertir código fuente de programas, archivos ejecutables y binarios en modelos UML al instante, creando de manera simple toda la documentación. Está diseñada para usuarios interesados en sistemas de

FUNDAMENTACIÓN TEÓRICA

software de gran escala con el uso del acercamiento orientado a objeto, además apoya los estándares más recientes de las notaciones de Java y de UML. Incorpora el soporte para trabajo en equipo, que permite que varios desarrolladores trabajen a la vez en el mismo diagrama y vean en tiempo real los cambios hechos por sus compañeros. (Rodríguez Guerra, y otros, 2009)

Visual Paradigm permite importación desde Rational Rose, exportación/importación XML, generador de informes, editor de figuras, integración con MS Visio, plug-in, integración IDE con Visual Studio, IntelliJ IDEA, Eclipse, NetBeans.

Características:

- Soporta aplicaciones Web.
- Las imágenes y reportes generados, no son de muy buena calidad.
- Varios idiomas.
- Generación de código para Java y exportación como HTML.
- Fácil de instalar y actualizar.
- Compatibilidad entre ediciones.

Ventajas:

- Tiene apoyo adicional en cuanto a generación de artefactos automáticamente.
- Genera modelos VP-UML instantáneamente a partir de código binario .NET.
- Generación de documentación en formatos HTML y PDF.
- Tiene disponibilidad en múltiples plataformas: Microsoft Windows (98, 2000, XP, o Vista), Linux, Mac OS X, Solaris o Java.
- Brinda la posibilidad de generar código a partir de los diagramas, para plataformas como .Net, Java y PHP, así como obtener diagramas a partir del código.
- Permite documentar todo el trabajo sin necesidad de utilizar herramientas externas.

Desventajas:

Las imágenes y reportes generados, no son de alta calidad.

1.5.4 Rational Rose

Es la herramienta CASE que comercializan los desarrolladores de UML (Booch, Rumbaugh y Jacobson) y que soporta de forma completa la especificación del UML. Esta herramienta propone la utilización de cuatro tipos de modelos para realizar un diseño del sistema, utilizando una vista estática, otra dinámica de los modelos del sistema, una lógica y otra física; que permite crear y refinar estas vistas creando de esta forma un modelo completo que representa el dominio del problema y el sistema de software. (Larman, 1999)

Ventajas

Proporciona un lenguaje común de modelado para el equipo que facilita la creación de software de calidad más rápidamente. Utiliza modelado UML para trabajar en diseños de base de datos, con capacidad de representar la integración de los datos y los requerimientos de aplicación a través de diseños lógicos y físicos. Es una herramienta muy completa y estable, tiene facilidad de uso para la modificación y creación de nuevos diagramas.

Desventajas

- Es un software propietario.
- Su entorno gráfico no es muy amigable para el usuario.
- Necesidad de alta capacidad de procesamiento.
- No se puede crear el entorno del sistema para los diagramas de casos de uso.

1.5.5 ¿Por qué Visual Paradigm?

Se decide utilizar Visual Paradigm porque entre sus ventajas se encuentra que tiene disponibilidad en múltiples plataformas y brinda la posibilidad de intercambiar información mediante la importación y exportación de ficheros e imágenes. Soporta la generación de código e ingeniería inversa para Java. En el sitio oficial se explican detalladamente todas las funcionalidades existiendo así una abundante documentación. La herramienta es colaborativa porque soporta múltiples usuarios trabajando sobre el mismo proyecto, permite el control de versiones y genera la documentación en varios formatos. Visual Paradigm es una herramienta CASE que se destaca además por su usabilidad y portabilidad, se integra con NetBeans. Rational Rose a pesar de ser una herramienta muy completa, se dificulta su utilización al no ser libre. ArgoUML es una herramienta muy útil para la elaboración de sistemas informáticos pero su instalación es muy costosa, además no es conforme completamente a los estándares UML y

carece de soporte completo para algunos tipos de diagramas incluyendo los diagramas de secuencia y los de colaboración.

1.6 Sistemas Gestores de Bases de Datos (SGBD).

Un sistema gestor de base de datos se define como el conjunto de programas que administran y gestionan la información contenida en una base de datos. Ayuda a realizar las siguientes acciones:

- Definición de los datos.
- Mantenimiento de la integridad de los datos dentro de la base de datos.
- Control de la seguridad y privacidad de los datos.
- Manipulación de los datos. (Álvarez, 2007)

1.6.1 MySQL.

MySQL es un sistema de administración de bases de datos relacional (RDBMS) capaz de almacenar una enorme cantidad de datos de gran variedad y de distribuirlos para cubrir las necesidades de cualquier tipo de organización, desde pequeños establecimientos comerciales a grandes empresas y organismos administrativos. Es la base de datos de código abierto más popular del mundo. (Gilfillan, 2003)

Características:

En las últimas versiones se pueden destacar las siguientes características principales:

- El principal objetivo de MySQL es velocidad y robustez.
- Soporta gran cantidad de tipos de datos para las columnas.
- Gran portabilidad entre sistemas, puede trabajar en distintas plataformas y sistemas operativos.
- Cada base de datos cuenta con tres archivos: uno de estructura, uno de datos y uno de índice y soporta hasta treinta y dos índices por tabla.
- Aprovecha la potencia de sistemas multiproceso, gracias a su implementación multihilo.
- Flexible sistema de contraseñas y gestión de usuarios, con buen nivel de seguridad en los datos. (Ruiz Valdez, 2009)
- El servidor soporta mensajes de error en distintos idiomas.

Ventajas:

- Velocidad al realizar las operaciones, lo que le hace uno de los gestores con mejor rendimiento.
- Facilidad de configuración e instalación.
- Soporta gran variedad de Sistemas Operativos.
- Conectividad y seguridad. (Buenas Tareas, 2010)

Desventajas:

- Un gran porcentaje de las utilidades de MySQL no están documentadas. (Buenas Tareas, 2010)

1.6.2 Oracle.

Oracle es una potente herramienta cliente/servidor para la gestión de Bases de Datos. (Masip, 2002)

Ayuda a analizar datos y efectuar recomendaciones concernientes a mejorar el rendimiento y la eficiencia en el manejo de las informaciones que se encuentran almacenadas. Apoya el diseño y optimización de modelos de datos, y la definición de estándares de diseño y nomenclatura de objetos. Permite documentar y mantener un registro periódico de las mantenciones, actualizaciones de hardware y software, cambios en las aplicaciones y, en general, todos aquellos eventos relacionados con cambios en el entorno de utilización de una base de datos.

Ventajas:

- Oracle es el motor de base de datos relacional más usado a nivel mundial.
- Es multiplataforma.
- Permite el uso de particiones para la mejora de la eficiencia, de replicación e incluso ciertas versiones admiten la administración de bases de datos distribuidas.
- Es la base de datos con más orientación hacia Internet.
- Aceptable soporte. (Arce Morell, 2009)

Desventajas:

- Las licencias son excesivamente caras.

- También es elevado el coste de la formación, y sólo últimamente han comenzado a aparecer buenos libros sobre asuntos técnicos distintos de la simple instalación y administración.
- Necesidad de configuración.

1.6.3. PostgreSQL.

PostgreSQL es un Sistema de Gestión de Bases de Datos Objeto-Relacionales (ORDBMS) que ha sido desarrollado de varias formas desde la década de 1980. El proyecto PostgreSQL sigue actualmente un activo proceso de desarrollo a nivel mundial gracias a un equipo de desarrolladores y contribuidores de código abierto. Es ampliamente considerado como una de las alternativas de sistema de bases de datos de código abierto. (Espinoza, 2005)

Características:

- **Soporte SQL92/SQL99.**

PostgreSQL implementa un subconjunto extendido de los estándares SQL92 y SQL99

- **Bloqueos de tabla y filas.**

Ofrece varios modos de bloqueo para controlar el acceso concurrente a los datos en tablas. Algunos de estos modos de bloqueo los adquiere PostgreSQL automáticamente antes de la ejecución de una declaración, mientras que otros son proporcionados para ser usados por las aplicaciones.

- **Múltiples tipos de datos predefinidos.**

Implementa los tipos de datos definidos para el estándar SQL3 y aumenta algunos otros.

- **Soporte de tipos y funciones de usuario.**

Soporta operadores, funciones métodos de acceso y tipos de datos definidos por el usuario. Incorpora una estructura de datos Array.

- **Conectividad TCP/IP, JDBC y ODBC.**
- **Interfaz con diversos lenguajes (C, C++, Java, Delphi, Python, Perl, PHP, Bash).**
(Espinoza, 2005)

Ventajas:

- Ha sido diseñado y creado para tener un mantenimiento y ajuste mucho menor que otros productos, conservando todas las características, estabilidad y rendimiento.
- El código fuente está disponible para todos sin costo.

- Está disponible en casi cualquier Unix (treinta y cuatro plataformas en la última versión estable).
- Existen varias herramientas gráficas de alta calidad para administrar las bases de datos y para hacer diseño de bases de datos.

Desventajas:

- Falta de un conjunto de herramientas que permitan una fácil gestión de los usuarios y de las bases de datos que contenga el sistema.
- La velocidad de respuesta que ofrece este gestor con bases de datos relativamente pequeñas puede parecer deficiente, aunque esta misma velocidad la mantiene al gestionar bases de datos realmente grandes.

1.6.4 ¿Por qué PostgreSQL?

Se decide utilizar PostgreSQL porque es un motor de bases de datos avanzado y de código abierto. Oracle presenta altos costos de soporte y desarrollo mientras PostgreSQL no tiene costos de licencias. PostgreSQL es un gestor de base de datos de tipo objeto relacional mientras que Oracle y MySQL son sólo relacionales. PostgreSQL tiene una amplia variedad de lenguajes de desarrollo, disponibilidad de herramientas de desarrollo, administración, monitoreo.

1.7 Lenguajes de Programación.

Un lenguaje de programación es un conjunto de símbolos y reglas sintácticas y semánticas que definen su estructura y el significado de sus elementos y expresiones. Es utilizado para controlar el comportamiento físico y lógico de una máquina. Un lenguaje de programación permite a uno o más programadores especificar de manera precisa sobre qué datos debe operar una computadora, cómo estos datos deben ser almacenados o transmitidos y qué acciones debe tomar bajo una variada gama de circunstancias. Una característica relevante de los lenguajes de programación es precisamente que más de un programador pueda tener un conjunto común de instrucciones que puedan ser comprendidas entre ellos para realizar la construcción del programa de forma colaborativa. (Rodríguez Guerra, y otros, 2009)

1.7.1 Java.

El lenguaje de programación Java surge a principios de los años 90 en los laboratorios de Sun Microsystems. A diferencia de los lenguajes convencionales, que generalmente están diseñados para ser compilados a código nativo, Java es compilado a un código intermedio o bytecode, el cual es interpretado por una máquina virtual de Java. La máquina virtual hace posible que una aplicación que haya sido implementada en Java se ejecute en cualquier sistema operativo con soporte para ella. Proporciona un entorno de ejecución que convierte el código neutro de Java al código nativo del ambiente en que está siendo ejecutada.

Como lenguaje de programación es multipropósito, reúne todas las características de un ambiente orientado a objetos: es sencillo, cuenta con capacidad de generación de aplicaciones distribuidas, segura, de arquitectura neutral, portable, multihilo, dinámico y de alto rendimiento.

Ventajas:

- Se adapta con facilidad a los cambios en los componentes fundamentales del lenguaje.
- Cuenta con características de seguridad que evitan la infección por virus o trampas de programación, comunes en programas hechos con otros lenguajes.
- Minimiza los errores que se escapan a la fase de prueba.
- Puede integrar componentes de Internet y bases de datos remotas.
- En Java se maneja el modelo de máquina virtual para poder desarrollar código portable.

Desventajas:

- Al poseer clases de librerías el manejo del lenguaje se hace más difícil y consume mucho tiempo.
- En ocasiones puede ocurrir un desperdicio de memoria, lo que permite que la ejecución del código sea lenta.

1.7.2 C#

C# es un lenguaje orientado a objetos, moderno y seguro. (Jeff Ferguson, 2003)

Características

FUNDAMENTACIÓN TEÓRICA

- **Sencillez:** Elimina muchos elementos que otros lenguajes incluyen y que son innecesarios en .NET. Por ejemplo:

El código escrito en C# es auto contenido, lo que significa que no necesita de ficheros adicionales de la propia fuente tales como ficheros de cabecera o ficheros IDL.

El tamaño de los tipos de datos básicos es fijo e independiente del compilador, sistema operativo o máquina para quienes se compile, lo que facilita la portabilidad del código.

No se incluyen elementos poco útiles de C++ como macros, herencia múltiple o la necesidad de un operador diferente del punto (.) para acceder a miembros de espacios de nombres (::).

- **Modernidad:** Incorpora en el propio lenguaje elementos que a lo largo de los años ha ido demostrándose, son muy útiles para el desarrollo de aplicaciones y que en otros lenguajes como Java o C++ hay que simular, como un tipo básico decimal, la inclusión de una instrucción foreach que permita recorrer colecciones con facilidad y es ampliable a tipos definidos por el usuario, la inclusión de un tipo básico String para representar cadenas o la distinción de un tipo bool específico para representar valores lógicos.
- **Orientación a objetos:** Una diferencia de este enfoque orientado a objetos respecto al de otros lenguajes como C++ es que el de C# es más puro en tanto que no admite ni funciones ni variables globales sino que todo el código y datos han de definirse dentro de definiciones de tipos de datos, lo que reduce problemas por conflictos de nombres y facilita la legibilidad del código.
- **Seguridad de tipos:** Incluye mecanismos que permiten asegurar que los accesos a tipos de datos siempre se realicen correctamente, evitando que se produzcan errores difíciles de detectar por acceso a memoria no perteneciente a ningún objeto. (Tala, 2004)

Ventajas:

- Compila a código intermedio independiente del lenguaje en que haya sido escrita la aplicación e independiente de la máquina donde vaya a ejecutarse.
- Eliminación del uso de punteros.

- Concepto formalizado de los métodos Get y Set, con lo que se consigue código mucho más legible.
- Facilita muchas tareas comunes de la programación.

Desventajas:

- No es multiplataforma.

1.7.3 C++

C++ es un potente lenguaje de programación que surgió en 1980, continuando con las ventajas, flexibilidad y eficacia del C. Es un lenguaje de programación que permite programar desde sistemas operativos, compiladores, aplicaciones de bases de datos, procesadores de texto, juegos. C++ ha ido evolucionando y ha contribuido a la aparición de Java. (Tala, 2004)

Características:

- Herencia múltiple.
- Sobrecarga de operadores y funciones.
- Derivación.
- Funciones virtuales.
- Plantillas.
- Gestión de excepciones.

Ventajas:

- Al ser un lenguaje muy utilizado en la actualidad existe una gran cantidad de libros, cursos y páginas web que se dedican al estudio del mismo.
- Es un lenguaje rápido en cuanto a ejecución.

Desventajas:

- No es multiplataforma.
- Uso de librerías dinámicas muy complejas.
- C++ puede dar como resultado aplicaciones muy potentes, pero debe asegurarse de que el código funciona bien. Un error en la escritura del programa puede hacer que toda

la aplicación falle o se comporte de forma inesperada.

1.7.4 ¿Por qué Java?

Se decide utilizar Java porque es un lenguaje multiplataforma a diferencia de C++ y C#. Java fue creado para obtener un lenguaje parecido a C++, pero reduciendo los errores más comunes de la programación, algunos de ellos son: necesidad de liberar memoria, aritmética de punteros, no existen referencias. Java realiza verificaciones en busca de problemas tanto en tiempo de ejecución como en tiempo de compilación, lo que posibilita encontrar errores lo más pronto posible, algunas de esas verificaciones son excepciones y comprobación de punteros, permitiendo que sea un lenguaje robusto a diferencia de C++ que no las realiza. Otros aspectos de interés que llevan a Java a la vanguardia es que restringe el uso de aspectos críticos del sistema para evitar la codificación de virus.

1.8 Entorno de Desarrollo Integrado.

Un Entorno de Desarrollo Integrado (IDE) es un conjunto de herramientas para el programador que incluyen un buen editor de código, administrador de proyectos y archivos, enlace transparente a compiladores e integración con sistemas controladores de versiones o repositorios. (Luciano, 2008)

1.8.1 Eclipse.

Facilita enormemente las tareas de edición, compilación y ejecución de programas durante su fase de desarrollo. Aunque pretende ser un entorno versátil soportando varios lenguajes de programación, ha ganado una gran popularidad y tiene una mejor integración con Java. (Chapela Martínez, 2007)

Eclipse da soporte a todo tipo de proyectos que abarcan todo el ciclo de vida del desarrollo de aplicaciones, incluyendo soporte para modelado. Es principalmente una plataforma de programación, usada para crear entornos integrados de desarrollo. Fue desarrollado originalmente por IBM como el sucesor de su familia de herramientas para VisualAge. Es ahora desarrollado por la Fundación Eclipse, es libre y multiplataforma.

Características:

- Dispone de un editor visual de texto con resaltado de sintaxis coloreada.
- La compilación es en tiempo real.

- Tiene pruebas unitarias con JUnit, asistentes para creación de proyectos, clases, pruebas y refactorización.
- Modifica e inspecciona valores de variables.
- Avisa cuando se cometen errores mediante una ventana secundaria.
- Depura código que resida en una máquina remota.

1.8.2 NetBeans.

NetBeans es un entorno de desarrollo escrito en Java, pero puede servir para cualquier otro lenguaje de programación. Es una aplicación de código abierto diseñada para el desarrollo de aplicaciones fácilmente portables entre las distintas plataformas, haciendo uso de la tecnología Java. Dispone de soporte para crear interfaces gráficas de forma visual, desarrollo de aplicaciones Web, control de versiones, colaboración entre varias personas y creación de aplicaciones compatibles con teléfonos móviles. (Rodríguez Guerra, y otros, 2009)

Características

- Editor de código sensible al contenido. Con soporte para autocompletar el mismo, coloreado de etiquetas y uso de abreviaturas para varios lenguajes de programación.
- Soporte para Java, C, C++, XML, lenguajes HTML, JSP, RMI, CORBA, JINI, JDBC y tecnologías Servlet.
- Incluye control de versiones y compilación avanzada.
- Posibilidad de utilizar otras versiones de compiladores y depuradores.
- Creación visual de componentes gráficos.
- Herramientas con asistentes para facilitar la escritura de código. (Luciano, 2008)
- Dispone de soporte para crear interfaces gráficas de forma visual, agregando y alineando el espacio de trabajo.
- Multiplataforma (Disponible para Windows, GNU/Linux, Mac OS X y Solaris).
- Separa el diseño de software de la implementación con modelado UML.

La plataforma de NetBeans es una base modular y extensible usada como estructura de integración para crear grandes aplicaciones de escritorio. Empresas independientes asociadas, especializadas en desarrollo de software, proporcionan extensiones adicionales que se integran fácilmente en la plataforma y que pueden también utilizarse para desarrollar sus propias herramientas y soluciones. (Rodríguez Guerra, y otros, 2009)

Entre las **características** de la plataforma están:

- Administración de las interfaces de usuario (ej. menús y barras de herramientas).
- Administración de ventanas.
- Framework basado en asistentes (diálogo paso a paso).
- Administración de las configuraciones del usuario.
- Administración del almacenamiento (guardando y cargando cualquier tipo de dato).

1.8.3 Jbuilder.

Es un IDE Java de Borland. En la versión 2007 ya es multiplataforma, está disponible para Windows, Linux y MacOS X. Soporta plenamente la última versión del kit de desarrollo de Java proporcionado por Sun, la versión 1.1.x. (Rodríguez Guerra, y otros, 2009)

Características:

- Es potente, rápido y fiable.
- Sus programas se pueden ejecutar en cualquier plataforma que soporte Java.
- El entorno visual ofrece un entorno seguro para crear aplicaciones multiplataforma y también se puede alternar entre las herramientas visuales y el código, manteniendo siempre la coherencia entre las mismas.

Desventajas:

- Es muy caro y consume muchos recursos.

1.8.4 JCreator.

Es un IDE específico para la programación en Java, que engloba en la misma aplicación: editor, depurador y compilador. Se trata de uno de los programas más potentes y eficaces que existen, incluso comparado con otros como JDeveloper, JBuilder, Eclipse o NetBeans, ya que está escrito por completo en C++, por lo que consume menos recursos. (Xinox Software, 2009)

Características

- Gestor de proyectos y plantillas.
- Editor con resaltado en color de los comandos.
- Depurador de fallos.

- Interfaz personalizable.
- Permite indicar la ruta de JDK JavaDocs, para permitir la consulta de toda la documentación oficial de Java.
- Es comercial pero también existe la versión gratuita.

1.8.3 ¿Por qué NetBeans?

Se decide utilizar el NetBeans porque es multiplataforma, gratuito y de código abierto para desarrolladores de software. Tiene al alcance todas las herramientas necesarias para crear aplicaciones profesionales para entornos de escritorio en Java. El nuevo programa de instalación permite a los desarrolladores personalizar su entorno de desarrollo, eligiendo únicamente aquellas herramientas que necesiten. Eclipse a pesar de ser muy buena su arquitectura está basado en plug-ins, haciéndolo muy dependiente de estos, por lo que tiende a retrasar el trabajo. JBuilder tiene características muy buenas pero consume muchos recursos y es muy caro.

1.9 Frameworks.

Es una estructura de soporte definida en la cual un proyecto de software puede ser organizado y desarrollado. Típicamente, un framework puede incluir soporte de programas, bibliotecas y un lenguaje interpretado entre otros software para ayudar a desarrollar y unir los diferentes componentes de un proyecto. Además provee una estructura y una metodología de trabajo la cual extiende o utiliza las aplicaciones del dominio. (Rodríguez Guerra, y otros, 2009)

1.9.1 Hibernate.

Hibernate es un entorno de trabajo que tiene como objetivo facilitar la persistencia de objetos Java en bases de datos relacionales y al mismo tiempo la consulta de estas para obtener objetos. (Rosés Albiol, 2003)

Permite diseñar objetos persistentes que podrán incluir polimorfismo, relaciones, colecciones y un gran número de tipos de datos. Facilita el mapeo de atributos entre una base de datos relacional tradicional y el modelo de objetos de una aplicación, mediante archivos declarativos XML que permiten establecer estas relaciones. De una manera muy rápida y optimizada se puede generar base de datos en cualquiera de los entornos soportados: Oracle, DB2, MySQL, PostgreSQL. Se integra en cualquier tipo de aplicación justo por encima del contenedor de datos.

Características:

- Permite diseñar objetos persistentes que podrán incluir polimorfismo, relaciones, colecciones y un gran número de tipos de datos.
- Persistencia transparente. Ofrece soporte para un amplio conjunto de las colecciones de Java, propiedades del estilo de persistencia de JavaBeans, etc.
- Mapeado flexible debido a las asociaciones bidireccionales, la persistencia transitiva y las colecciones de tipos básicos el mapeado resulta mucho más flexible.
- Facilidades en metadatos. Soporta el formato del mapeado de XML, diseñado para ser editado a mano y el mapeado basado en anotaciones, además de validación basada en anotaciones. (Hibernate, 2009)

Ventajas:

- Es posible declarar si se quiere obtener los objetos relacionados al que se está obteniendo o no.
- Se puede especificar tamaño o límites de los objetos a levantar.
- Permite hacer uso del llamado caché de segundo nivel y el caché de consultas.
- Posibilita declarar consultas y después usarlas.
- Es muy cómodo y hace más ágil el desarrollo, e incluso hay herramientas que ya generan código para Hibernate.
- Software libre. Está bajo licencia LGPL (Lesser GNU Public License).
- Facilidades en consultas. Debido en parte a que se realizan en un potente lenguaje de consultas orientado a objetos.

Desventajas:

- Se usan clases generadas en tiempo de ejecución para el levantado de los objetos de la base, esto genera más sobrecarga, que las consultas directas en SQL.
- Si la base de datos cambia, ya sea agregar un campo u otra operación, los objetos deben ser modificados por tanto también los XML y todo lo relativo a Hibernate.

1.9.2 JasperReports.

JasperReports es una poderosa herramienta para generar reportes en Java, con la habilidad de producir contenido completo para la pantalla, directo para impresora o en diferentes formatos de archivo (XLS, PDF, CSV, XML y otros). Es completamente Java, código abierto y puede

reutilizarse tanto en aplicaciones cliente y cliente/servidor como en aplicaciones web, J2EE, etc. JasperReports permite organizar la información obtenida desde una base de datos relacional en diseños de reportes predefinidos en un formato XML. (JasperReports, 2001)

El propósito principal de JasperReports es ayudar a crear documentos de tipo páginas, preparados para imprimir en una forma simple y flexible. Se usa comúnmente con iReport, herramienta visual de diseños de informes para JasperReports potente, intuitiva y fácil de usar.

1.10 Plataforma de desarrollo.

Es el entorno de software común en el cual se desenvuelve la programación de un grupo definido de aplicaciones, comúnmente se encuentra relacionada directamente a un sistema operativo; sin embargo, también es posible encontrarla ligada a una familia de lenguajes de programación o a una interfaz de programación de aplicaciones. (Caballero Franco, y otros, 2009)

1.10.1 Java 2 Enterprise Edition (J2EE).

Por la necesidad del mercado de desarrollo de software de contar con medios y herramientas que permitan construir aplicaciones corporativas se diseñó la plataforma abierta y estándar de Java para este fin, mejor conocida como J2EE. Se le denomina plataforma porque proporciona técnicas específicas que describen el lenguaje, pero, además, provee las herramientas para implementar productos de software basados en dichas especificaciones.

Es una especificación que define una plataforma para crear aplicaciones empresariales utilizando un modelo de multicapas, dividiendo la aplicación en diferentes niveles, cada uno especializándose en una tarea en particular.

La especificación de J2EE define su arquitectura basándose en los conceptos de capas, contenedores, componentes, servicios y las características de cada uno de éstos. Las aplicaciones J2EE son divididas en cuatro capas: la capa cliente, la capa web, la capa negocio y la capa datos. (Barrios Núñez, 2003)

Ventajas

Posee muchas ventajas como las de proveer soporte para múltiples sistemas operativos: al ser una plataforma Java, es posible desarrollar arquitecturas basadas en J2EE usando cualquier

sistema operativo donde pueda estarse ejecutando una máquina virtual de Java, tiene una independencia total de la arquitectura de hardware. Sobre esta plataforma es posible crear arquitecturas basadas por completo en productos de software libre. Además, pone a disposición de los arquitectos de software varias soluciones libres para cada una de las partes de su arquitectura. La experiencia y madurez de la plataforma favorecen a J2EE, ya que fue creada en el año 1997, y en dicho tiempo se han ido desarrollando multitud de productos y servicios, a la vez que se han ido corrigiendo errores y cubriendo las carencias y necesidades detectadas, por lo que hoy cuenta con una gama de productos altamente consolidados.

Se seleccionó J2EE, siempre teniendo en cuenta el tipo de aplicación que se pretende desarrollar. Para la selección de la plataforma se tuvo en cuenta, como elemento indispensable, el entorno de ejecución, ya que se requiere de un sistema multiplataforma, de alto rendimiento, escalabilidad y seguridad, por lo que la opción más acertada para estos requerimientos es la plataforma J2EE.

1.11 Arquitectura de software.

La arquitectura de software comprende los componentes del software, las propiedades de esos componentes visibles externamente y las relaciones entre ellos. (Pressman, 2002)

1.11.1 Arquitectura en tres capas.

La arquitectura tres capas proporciona una buena organización y estructuración entre los distintos niveles de abstracción, donde, un cambio en uno de estos niveles no debe proporcionar cambios en los restantes. Las capas que la componen son:

- **Capa de presentación:** es la que ve el usuario, presenta el sistema al mismo, le comunica la información y captura los datos que el usuario introduce en la aplicación. Esta capa se comunica únicamente con la del negocio enviando peticiones a la misma. También es conocida como interfaz gráfica y debe tener la característica de ser entendible y fácil de usar para el usuario.
- **Capa de negocio:** es donde residen los programas que se ejecutan durante la ejecución de la aplicación. Tratan las peticiones del usuario y envían las respuestas tras el proceso. En ella se establecen todas las reglas que deben cumplirse. Esta capa se comunica con la de presentación, para recibir las peticiones y presentar los resultados, y con la de acceso a datos, para interactuar con la base de datos, consultando y modificando sus informaciones.

- **Capa de acceso a datos:** es donde residen los datos y la encargada de acceder y modificar a los mismos. Está formada por uno o más sistemas de gestión de bases de datos que realizan todo el almacenamiento de informaciones y que reciben solicitudes de almacenamiento o recuperación de datos desde la capa de negocio.

Ventajas:

- Reduce el tráfico de información en la red por lo que mejora el rendimiento de los sistemas.
- Brinda una mayor flexibilidad de desarrollo y de elección de plataformas sobre la cual montar las aplicaciones.
- Provee escalabilidad horizontal y vertical.
- Se mantiene la independencia entre el código de la aplicación (reglas y conocimiento del negocio) y los datos, mejorando la portabilidad de las aplicaciones.
- Los lenguajes sobre los cuales se desarrollan las aplicaciones son estándares lo que hace más exportables las aplicaciones entre plataformas.
- Dado que mejora el rendimiento al optimizar el flujo de información entre componentes, permite construir sistemas críticos de alta fiabilidad.
- Disminuye el número de usuarios conectados a la base de datos.

1.11.2 Arquitectura Modelo Vista Controlador (MVC).

MVC es un patrón de diseño que considera dividir una aplicación en tres módulos claramente identificables y con funcionalidad bien definida: El modelo, las vistas y el controlador. (Bascón Pantoja, 2004)

Fue diseñada para reducir el esfuerzo de programación necesario en la implementación de sistemas múltiples y sincronizados de los mismos datos. Sus características principales son que el modelo, las vistas y los controladores se tratan como entidades separadas; esto hace que cualquier cambio producido en el modelo se refleje automáticamente en cada una de las vistas. (Catalani, 2007)

Modelo

Es el objeto que representa los datos del programa. Maneja las informaciones y controla todas sus transformaciones. No tiene conocimiento específico de los controladores o de las vistas, ni siquiera contiene referencias a ellos. Es el propio sistema el que tiene encomendada la responsabilidad de mantener enlaces entre el modelo y sus vistas, y notificar a las últimas cuando cambia el modelo.

Vista

Es el objeto que maneja la presentación visual de los datos representados por el modelo. Genera una representación visual de este y muestra las informaciones al usuario. Interactúa con el modelo a través de una referencia al mismo.

Controlador

El controlador es un objeto que se encarga de dirigir el flujo del control de la aplicación debido a mensajes externos, como datos introducidos por el usuario u opciones del menú seleccionadas por él. A partir de estos mensajes, el controlador se encarga de modificar el modelo o de abrir y cerrar vistas. El controlador tiene acceso al modelo y a las vistas, pero estos no conocen de la existencia del mismo. (Bascón Pantoja, 2004)

Ventajas

- La aplicación está implementada modularmente.
- Sus vistas muestran información actualizada siempre.
- El programador no debe preocuparse de solicitar que las vistas se actualicen, ya que este proceso es realizado automáticamente por el modelo de la aplicación.
- Si se desea hacer una modificación al modelo del dominio, como aumentar métodos o datos contenidos, sólo debe modificarse el modelo y las interfaces del mismo con las vistas.
- Las modificaciones a las vistas no afectan en absoluto a los otros módulos de la aplicación. (Bascón Pantoja, 2004)

Desventajas

- El tiempo de desarrollo de una aplicación que implementa el patrón de diseño MVC es mayor, al menos en la primera etapa, que el tiempo de desarrollo de una aplicación que no lo implementa, pues requiere que el programador implemente una mayor cantidad de clases que en un entorno de desarrollo común no son necesarias. Sin embargo, esta desventaja es muy relativa ya que posteriormente, en la etapa de mantenimiento de la aplicación, una aplicación MVC es mucho más extensible y modificable que una aplicación que no lo implementa.
- Necesita la existencia de una arquitectura inicial sobre la que se deben construir clases e interfaces para modificar y comunicar los módulos de una aplicación.
- Esta arquitectura inicial debe incluir, por lo menos: un mecanismo de eventos para poder proporcionar las notificaciones que genera el modelo de aplicación; una clase modelo, otra clase vista y una clase controlador genéricas que realicen todas las tareas de comunicación, notificación y actualización que serán luego transparentes para el desarrollo de la aplicación.
- MVC es un patrón de diseño orientado a objetos por lo que su implementación es sumamente costosa y difícil en lenguajes que no siguen este paradigma. (Bascón Pantoja, 2004)

1.11.3 ¿Por qué arquitectura en tres capas?

El patrón tres capas es el escogido para utilizar en el diseño de la aplicación ya que sus ventajas se adaptan perfectamente a los objetivos que se quieren lograr, con esta arquitectura se desarrollan aplicaciones de manera clara y transparente, con la ventaja de poder desarrollar varias capas al mismo tiempo. El no estar atado a un lenguaje de programación, garantiza una fácil adaptación, su principal fortaleza es que puede hacer convivir aplicaciones creadas en distintos lenguajes. Permite una centralización de los aspectos de seguridad y transaccionalidad, no replicación de lógica de negocio en los clientes permitiendo que las modificaciones y mejoras sean automáticamente aprovechadas por el conjunto de los usuarios, reduciendo los costes de mantenimiento. MVC posee muchas ventajas para ser considerado como arquitectura candidata, sin embargo, se utiliza principalmente en aplicaciones que manejan gran volumen de datos y transacciones complejas, que no es el caso.

1.12 Conclusiones

Después del estudio realizado a los sistemas que existen a nivel mundial relacionados con la gestión de estadísticas en el voleibol, se determinó que ninguno de estos resuelve las necesidades de la FCV.

Para desarrollar el sistema propuesto se realizó un estudio de algunas de las metodologías, lenguajes y herramientas que existen para el desarrollo de aplicaciones de escritorio determinando que las más adecuadas son: RUP como metodología de desarrollo con el lenguaje de modelado UML, la herramienta CASE Visual Paradigm (Versión 6.4), la plataforma de desarrollo J2EE, sistema gestor de base de datos PostgreSQL (Versión 8.3), el lenguaje de programación Java, el entorno de desarrollo integrado NetBeans (Versión 6.8) y los framework Hibernate y Jasper Report.

Capítulo 2

Análisis y Diseño

2.1 Introducción.

Para que un software esté documentado es importante definir una serie de artefactos siguiendo una metodología de desarrollo con la ayuda de herramientas que posibiliten la creación de los mismos. En este proceso es necesario tener definido el negocio para lograr un mejor entendimiento de los problemas que se pretenden resolver. Resulta sumamente importante seleccionar los requerimientos funcionales y no funcionales para comenzar a modelar el sistema que se pretende construir ya que de ellos se derivan los casos de uso, muy útiles pues sus descripciones y diagramas guían todo el funcionamiento del sistema.

2.2 Modelo del negocio.

Los objetivos del modelo del negocio son:

- Comprender la estructura y la dinámica de la organización en la cual se va a implantar un sistema.
- Comprender los problemas actuales de la organización e identificar las mejoras potenciales.
- Asegurar que los consumidores, usuarios finales y desarrolladores tengan un entendimiento común de la organización.
- Derivar los requerimientos del sistema que va a soportar la organización.

2.2.1 Descripción del Negocio.

La Federación Cubana de Voleibol realiza cada año una competencia por zonas en las categorías menores, donde se llevan estadísticas del rendimiento de cada equipo en cuanto al saque, K1 y K2 para realizar una valoración de los objetivos pedagógicos trazados por la Federación, también se registran los juegos ganados y perdidos, así como los sets que resultaron vencedores cada equipo para obtener una valoración de los resultados deportivos. Para llevar un mejor control de las estadísticas se plantea la necesidad de desarrollar un

software que cumpla con cada uno de los requisitos necesarios para llevar este proceso en cada competencia.

2.2.2 Reglas del Negocio.

Las reglas de negocio describen políticas que deben cumplirse o condiciones que deben satisfacerse, por lo que regulan algún aspecto del negocio. A continuación se mencionan las reglas por las que se rigen.

- Sólo se compete en las categorías:
 - Pioneril.
 - Escolar
 - Estudiantil
 - Primaria
 - Secundaria
 - Juvenil
- Para cualquier categoría o zona se competirá en ambos sexos.
- Los juegos se programarán de cinco sets a ganar tres.
- Se competirá en tres zonas (oriental, occidental y central) y además todos contra todos donde estarán presentes la totalidad de las provincias.

2.2.3 Actores del Negocio.

Un actor del negocio es cualquier individuo, grupo, entidad, organización, máquina o sistema de información externo; con los que el negocio interactúa.

A continuación se muestra el actor involucrado con el negocio.

Actores	Descripción
Federación Cubana de Voleibol	Es la institución encargada de llevar el control del evento.

Tabla 2: Actores del Negocio

2.2.4 Trabajadores del Negocio.

Representa a personas, o sistemas dentro del negocio que son las que realizan las actividades que están comprendidas dentro de un caso de uso. A continuación se muestran los involucrados con el negocio.

Trabajador	Descripción
Especialista	Es la persona encargada de llevar las acciones del juego.

Tabla 3: Trabajador del Negocio

2.2.5 Entidades del Negocio.

Las entidades de negocio representan a los objetos que los trabajadores del negocio toman, inspeccionan, manipulan, producen o utilizan durante la realización de los casos de uso de negocio. Comúnmente representan un documento o una parte esencial de un producto. A continuación se muestra la involucrada con este negocio.

Entidades	Descripción
Informe del juego	Contiene las acciones de cada equipo en un juego con el total de saques, k1 y k2, así como los que fueron efectivos, los juegos y sets ganados y perdidos además de la fecha.

Tabla 4: Entidades del Negocio

2.2.6 Caso de Uso del Negocio.

- Solicitar acciones de los juegos.

2.2.7 Diagrama de Casos de Uso del Negocio.

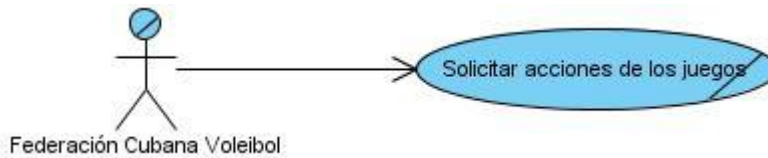


Figura 1: Diagrama de Caso de Uso del Negocio.

2.2.8 Descripción de los casos de uso del negocio.

Nombre del CU	Solicitar acciones de los juegos
Actores	Federación Cubana de Voleibol

ANÁLISIS Y DISEÑO

Propósito	Solicitar las acciones de cada uno de los juegos realizados.
Resumen	El caso de uso se inicia cuando la Federación Cubana de Voleibol solicita al especialista todas las acciones realizadas en cada uno de los juegos efectuados. Una vez obtenido los resultados el caso de uso termina.
Precondiciones	Debe haberse registrado los datos de los juegos.
Prioridad	Crítico
Sección “Solicitar acciones de los juegos”: Flujo Normal de Eventos	
Acción del Actor	Respuesta del proceso de Negocio
1. La Federación Cubana de Voleibol solicita las acciones de los juegos.	2 .El especialista elabora el informe de los juegos con cada uno de los datos del equipo ganador y el perdedor: Equipo ganador <ul style="list-style-type: none"> • Total de saques • Total de saques efectivos • Total de K1 • Total de K1 efectivos • Total de K2 • Total de K2 efectivos • Sets ganados Equipo perdedor <ul style="list-style-type: none"> • Total de saques • Total de saques efectivos • Total de K1 • Total de K1 efectivos • Total de K2 • Total de K2 efectivos • Sets ganados 4. El especialista entrega informe a la Federación Cubana de Voleibol.
7. La Federación Cubana de Voleibol recibe el informe. 8. El caso de uso termina.	

Tabla 5: Descripción del Caso de Uso del Negocio

2.2.9 Diagramas de Actividades.

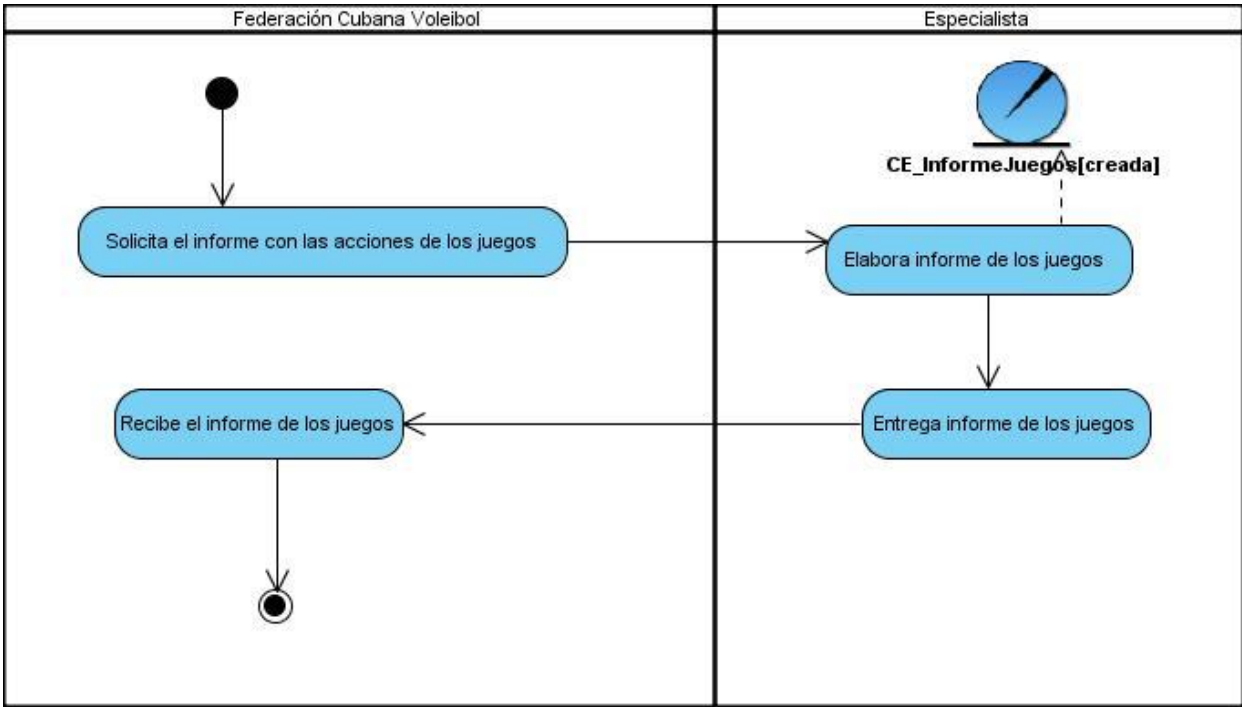


Figura 2: Diagrama de Actividades del Caso de Uso del Negocio Solicitar acciones de los juegos.

2.3 Requisitos del Sistema.

Los requisitos son la condición o capacidad que necesita un usuario para resolver un problema o lograr un objetivo. A partir de aquí se comienza a modelar el sistema que se va a construir, donde serán identificados los requerimientos funcionales y no funcionales.

2.3.1 Requerimientos Funcionales

Son capacidades o condiciones que el sistema debe cumplir. A continuación se presenta un conjunto de ellos:

RF1: Gestionar Evento.

- RF1.1 Insertar Evento.
- RF 1.2 Modificar Evento.

- RF 1.3 Abrir Evento.
- RF1.4 Cerrar Evento.
- RF 1.4 Ver Evento.

RF2: Gestionar Jugador.

- RF2.1 Insertar Jugador.
- RF 2.2 Modificar Jugador.
- RF2.3 Eliminar Jugador.
- RF 2.4 Ver Jugador.

RF 3. Gestionar Juego.

- RF3.1 Insertar Juego.
- RF3.2 Modificar Juego.
- RF 3.3 Eliminar Juego.
- RF 3.4 Ver Juego.

RF 4. Autenticar Usuario.

RF 5. Gestionar Usuario.

- RF 5.1: Insertar Usuario.
- RF 5.2: Modificar Usuario.
- RF5.3: Eliminar Usuario.
- RF 5.4 Ver Usuario.

RF 6. Consultar Evento.

RF 7. Consultar Jugador.

RF 8. Consultar Juego.

RE 9. Consultar Usuario.

RF 10. Consultar Reporte Boletín Diario.

- RF 10.1 Generar Reporte Boletín Diario.

RF11. Consultar Reporte Resumen Estadístico de Cada Equipo.

- RF 11.1 Generar Reporte Resumen Estadístico de Cada Equipo.

RF 12. Consultar Reporte Resultados de los Objetivos Pedagógicos.

- RF 12.1 Generar Reporte Resultados de los Objetivos Pedagógicos.

RF13. Consultar Reporte Resultados por Provincia.

- RF 13.1 Generar Reporte Resultados por Provincia.

RF14. Consultar Reporte Resultados Alcanzados por las Provincias.

- RF 14.1 Generar Reporte Resultados Alcanzados por las Provincias

RF15. Consultar Reporte Equipo Ideal.

- RF 15.1 Generar Reporte Equipo Ideal.

RF16. Consultar Reporte Control Competitivo.

- RF 16.1 Generar Reporte Control Competitivo.

RF17. Consultar Reporte Historial de Usuario.

2.3.2 Requerimientos No Funcionales.

Los requerimientos no funcionales son propiedades o cualidades que el producto debe tener.

- **Seguridad**

Confidencialidad: se establecerán diferentes niveles de acceso para los diferentes usuarios que interactúen con el sistema para garantizar que la información manejada esté protegida de acceso no autorizado.

- **Disponibilidad**

La información estará disponible las 24 horas del día para todos los usuarios.

- **Diseño e Implementación**

Lenguaje de programación: Java.

Herramienta para el almacenamiento de información: sistema gestor de bases de datos PostgreSQL.

- **Usabilidad**

Para utilizar el sistema es necesario poseer conocimientos elementales de computación. El sistema debe contar con menús que agrupen las funcionalidades que tienen relación.

- **Software**

Servidor de Base de datos PostgreSQL.

Máquina virtual de Java.

- **Hardware**

RAM: 256 MB o superior.

Tarjeta de video: VGA 128 Mb ATI Radeon 7000/7200/7500/9000 DDR o superior.

Disco Duro: 80 GB o superior.

- **Rendimiento**

Ante cualquier acción el sistema debe dar respuesta rápida.

- **Portabilidad**

Se requiere que el sistema sea multiplataforma.

- **Apariencia o Interfaz Externa**

Diseño sencillo, permitiendo la utilización del sistema sin mucho entrenamiento, todos los mensajes en pantalla aparecerán en idioma español.

- **Confiabilidad**

Garantía de un tratamiento adecuado de las excepciones.

- **Soporte**

Garantía de instalación y prueba del sistema.

2.4 Propuesta de solución.

Teniendo en cuenta que la información que se genera en los Campeonatos Zonales Nacionales de voleibol es registrada de forma manual en documentos digitales y la FCV no dispone de una herramienta que le permita llevar estos controles de una forma más eficiente, se propone la elaboración de un sistema informático que automatice estos procesos. Esta aplicación tendrá funcionalidades que le permitirá a la FCV gestionar toda la información que se genera en los campeonatos mencionados anteriormente y obtener reportes de forma rápida y sencilla.

La solución propuesta consiste en una aplicación de escritorio con tecnología Java, gestor de base de datos PostgreSQL 8.3, con una arquitectura en tres capas.

Esta solución aporta las siguientes ventajas:

- Navegación rápida a través de las interfaces de usuario, estos no necesitan acceder a la aplicación desde un lugar diferente de su puesto de trabajo ya que el software y las herramientas complementarias se instalarán directamente a la computadora del usuario.
- No se necesita disponer de Internet o Intranet para acceder a la aplicación.
- Fácil acceso a recursos locales: disco duro, memoria de video, sonido, ratón, teclado, etc.
- Es una aplicación multiplataforma, es decir, puede funcionar en diferentes sistemas operativos y/u ordenadores, pero el código fuente es el mismo.

Esta aplicación tendrá el nombre "SRPEV" asociado a las iniciales de las palabras Sistema, Registro, Procesamiento, Estadístico y Voleibol.

2.5 Actores del Sistema.

Actores	Descripción
----------------	--------------------

ANÁLISIS Y DISEÑO

Estadístico	Persona encargada de llevar las estadísticas y el rendimiento de cada equipo.
Administrador	Persona encargada de llevar la gestión de usuario y darle los permisos requeridos a cada uno.
Usuario	Persona que se va a autenticar para entrar al sistema.

Tabla 6: Actores del Sistema

2.6 Casos de Uso del Sistema.

Los casos de uso modelan el sistema desde el punto de vista del usuario, se basan en los requerimientos del sistema mostrando las distintas operaciones que se esperan de una aplicación y como se relacionan con su entorno.

2.6.1 Descripción de los Casos de Uso del Sistema Expandidos.

A continuación se muestra la descripción de algunos de los casos de uso del sistema más significativos, los otros se podrán encontrar en los anexos.

Nombre del CU	Gestionar Juego
Actores	Estadístico
Propósito	Insertar, modificar, ver o eliminar un juego.
Resumen	El caso de uso se inicia cuando el estadístico decide insertar un nuevo juego, modificar, eliminar o ver los datos de uno ya existente. Una vez realizada la operación deseada el caso de uso termina.
Referencias	RF 3
Precondiciones	El estadístico debe estar autenticado en el sistema.
Prioridad	Crítico
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema

ANÁLISIS Y DISEÑO

1. El caso de uso se inicia cuando el especialista selecciona la opción Gestionar Juego.	2. Brinda la posibilidad de realizar las acciones: <ul style="list-style-type: none">• Insertar Juego.• Modificar Juego• Eliminar Juego.• Ver Juego.
Sección “Insertar Juego”: Flujo Normal de Eventos	
3. Selecciona la opción Insertar juego.	4. Brinda la posibilidad de introducir: <ul style="list-style-type: none">• Total de saques• Total k1• Total k2• Cantidad Positivos de saques• Cantidad Positivos k1• Cantidad Positivos k2,• Sets Ganados• Sets Perdidos Y seleccionar: <ul style="list-style-type: none">• Fecha• Equipo• Evento Permite: <ul style="list-style-type: none">• Insertar• Cerrar
5. Inserta los datos del juego y presiona el botón Insertar.	6. Valida los datos. 7. Registra los datos. 8. Envía mensaje (“Insertado correctamente”) 9 El caso de uso termina.
Flujos Alternos	
6.a Existe algún campo vacío	

ANÁLISIS Y DISEÑO

	<p>6.a.1 El sistema envía mensaje (“Debe llenar todos los campos”)</p> <p>6. a.2 Regresa al paso 5 del Flujo Normal de Eventos.</p>
6.b Existe algún dato incorrecto	
	<p>6.b.1 El sistema envía mensaje (“Verifique los datos”)</p> <p>6. b.2 Regresa al paso 5 del Flujo Normal de Eventos.</p>
6.d El evento está cerrado	
	d.1 El sistema muestra mensaje (“ El evento está cerrado, no pude realizar ninguna acción en el sistema”)
4.a Selecciona la opción Cerrar	
	<p>4. a.1 Regresa a la vista anterior.</p> <p>4. a.2 El caso de uso termina.</p>
Sección “Modificar Juego”: Flujo Normal de Eventos	
1. Selecciona opción Modificar juego.	<p>2. Brinda la posibilidad de seleccionar:</p> <ul style="list-style-type: none"> • Evento • Sexo
3. Selecciona el evento y la fecha.	4. Muestra listado de juegos.
5. Selecciona el juego.	<p>6. Muestra los datos del juego.</p> <ul style="list-style-type: none"> • Fecha • Sexo <p>Equipo ganador</p> <ul style="list-style-type: none"> • Total de saques • Total de saques efectivos • Total de K1

ANÁLISIS Y DISEÑO

	<ul style="list-style-type: none"> • Total de K1 efectivos • Total de K2 • Total de K2 efectivos • Sets ganados <p>Equipo perdedor</p> <ul style="list-style-type: none"> • Total de saques • Total de saques efectivos • Total de K1 • Total de K1 efectivos • Total de K2 • Total de K2 efectivos • Sets ganados <p>Y permite:</p> <ul style="list-style-type: none"> • Modificar • Cerrar
<p>7. Modifica los datos y presiona el botón Modificar.</p>	<p>8. Valida los datos. 9. Actualiza los datos del juego. 10. El caso de uso termina</p>
<p>Flujos Alternos</p>	
<p>8. a. Existe algún campo vacío.</p>	
	<p>8.a.1 El sistema envía mensaje(“Debe llenar todos los campos”) 8. a.2 Regresa al paso 5 del Flujo Normal de Eventos.</p>
<p>8.b Existe algún dato incorrecto</p>	
	<p>8.b.1 El sistema envía mensaje (“Verifique los datos”) 8. b.2 Regresa al paso 5del Flujo Normal de Eventos.</p>

ANÁLISIS Y DISEÑO

8.c El evento está cerrado	
	d.1 El sistema muestra mensaje (“ El evento está cerrado, no pude realizar ninguna acción en el sistema”)
6.a Selecciona la opción de Cerrar	
	6. a.1 Regresa a la vista anterior. 6. a.2 El caso de uso termina.
Sección “Eliminar Juego”: Flujo Normal de Evento	
1. Selecciona opción Eliminar juego.	2. Brinda la posibilidad de seleccionar: <ul style="list-style-type: none"> • Evento • Fecha
3. Selecciona la fecha y el evento.	4. Muestra listado de juegos. Y permite: <ul style="list-style-type: none"> • Eliminar • Cerrar
5. Selecciona el juego y presiona el botón Eliminar.	6. Envía mensaje de confirmación y permite seleccionar: <ul style="list-style-type: none"> • Si • No
7. Selecciona Si.	8. Elimina el juego seleccionado 9. El sistema envía mensaje (“ Eliminado correctamente”) 10. El caso de uso termina.
Flujos Alternos	
6. El estadístico selecciona la opción No	
	6.1 Regresa a la vista anterior.
	6.2 El caso de uso termina.

ANÁLISIS Y DISEÑO

4. Selecciona la opción Cerrar.	
	4.1 Regresa a la vista anterior.
	4.2 El caso de uso termina.
Sección “Ver Juego”: Flujo Normal de Eventos	
1. Selecciona la opción de Ver juego.	2. Muestra los datos del juego Y permite: <ul style="list-style-type: none"> • Cerrar • Modificar los datos del juego • Eliminar a un juego
3. Selecciona la opción Cerrar	4. Regresa a la vista anterior. 5. El caso de uso termina.
Flujo Alterno	
3. a Selecciona Modificar los datos de un juego.	
	3. a.1 Brinda la posibilidad de modificar los datos de un juego. Ver Sección “Modificar juego”. 3. a.2 El caso de uso termina.
3.b El estadístico selecciona la opción de eliminar un juego de la lista mostrada	
	3. b.1 Brinda la posibilidad de eliminar un juego. Ver Sección “Eliminar Juego” del caso de uso Gestionar Juego. 3. b.2 El caso de uso termina.
Poscondiciones	Se insertó, modificó, eliminó o vio un juego.

Tabla 7: Descripción del Caso de Uso Gestionar Juego.

Nombre del CU	Consultar Reporte Boletín Diario
Actores	Estadístico

ANÁLISIS Y DISEÑO

Propósito	Consultar los datos del boletín diario.
Resumen	El caso de uso se inicia cuando el estadístico necesita consultar el reporte resumen del boletín diario. El sistema muestra los datos del reporte y termina así el caso de uso.
Referencias	RF 4
Precondiciones	Deben estar registrados los datos de los juegos diarios.
Prioridad	Crítico
Flujo Normal de Eventos	
Acciones del Actor	Respuesta del sistema
1. Selecciona la opción Consultar Reporte Boletín Diario.	2. Brinda la posibilidad de seleccionar: <ul style="list-style-type: none"> • Evento • Fecha Y permite: <ul style="list-style-type: none"> • Ver reporte. • Cerrar
3. Selecciona evento, la fecha y la opción Ver reporte.	4. Muestra reporte.
5. Consulta el reporte. 6. El caso de uso termina.	
Poscondiciones	Se consultó un reporte boletín diario.
Flujos Alternos	
4. a El sistema no encontró coincidencias.	
	4. a.1 Muestra el mensaje “No se encontraron coincidencias” y permite realizar una nueva búsqueda.
	4. a.2 Regresa al paso 3 del Flujo Básico.

2. a Selecciona la opción de Cerrar.	
	2. a.1 Regresa a la vista anterior. 2. a.2 El caso de uso termina.

Tabla 8: Descripción del Caso de Uso Consultar Boletín Diario

2.7 Diagrama de Casos de Usos del Sistema

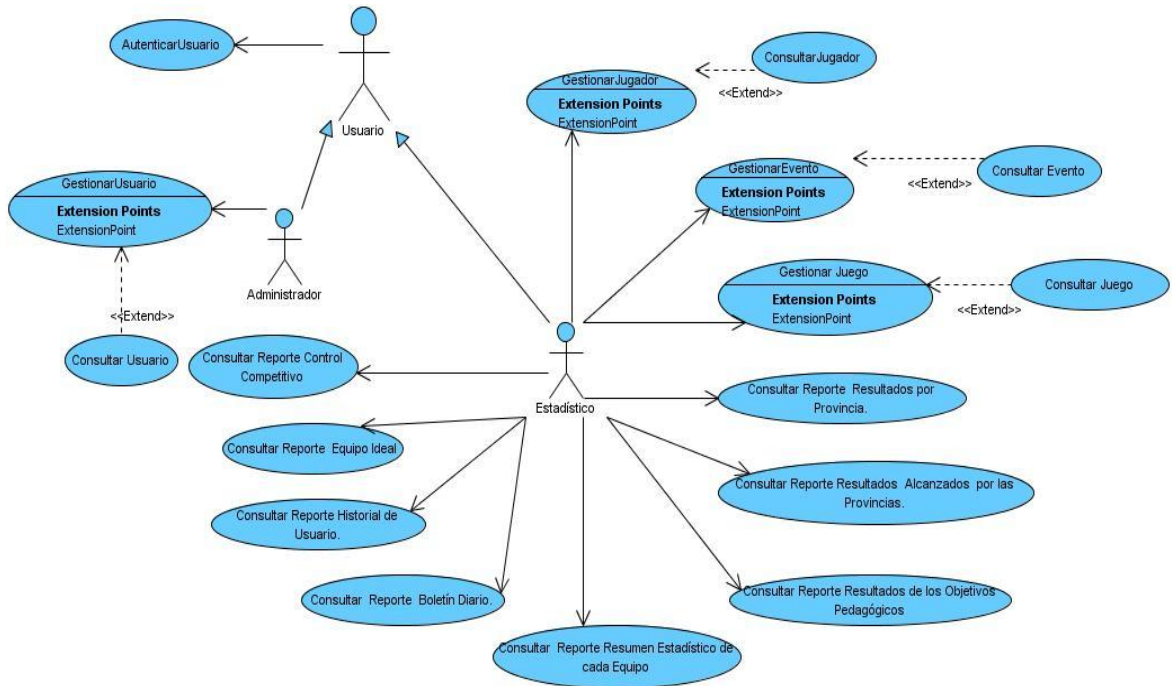


Figura 3: Diagrama de Casos de Uso del Sistema

2.8 Modelo de Análisis.

Se analizan los requerimientos que se describieron en la captura de requisitos, refinándolos y estructurándolos. El objetivo es conseguir una comprensión más precisa de los requisitos, y una descripción de los mismos que ayude a estructurar el sistema entero, incluyendo su arquitectura.

Clases de análisis: Se centran en los requisitos funcionales y son evidentes en el dominio del problema porque representan conceptos y relaciones del dominio. Tienen atributos y entre ellas se establecen relaciones de asociación, agregación / composición, generalización / especialización y tipos asociativos.

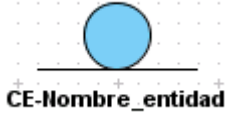
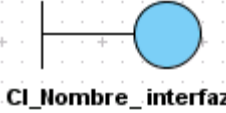

Nombre	Características	Representación
Entidad	Modelan información que poseen larga vida y que es a menudo persistente.	 CE-Nombre_entidad
Interfaz	Modelan la interacción entre el sistema.	 CI_Nombre_interfaz
Control	Coordinan la realización de uno o varios casos de uso coordinando las actividades de los objetos que implementan la funcionalidad del caso de uso.	 CC_Nombre_control

Tabla 9: Clases del Análisis

2.8.1 Diagramas de Clases de Análisis.

A continuación se muestran algunos de los diagramas de clases del análisis de los casos de usos más significativos, los otros diagramas se podrán encontrar en los anexos.

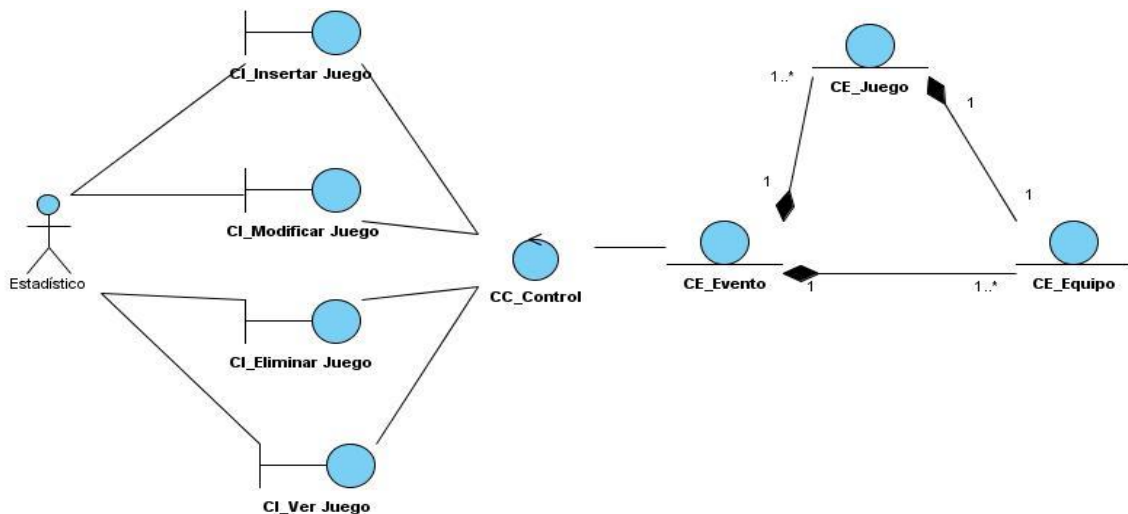


Figura 4: Diagrama de Clases del Análisis del Caso de Uso Gestionar Juego

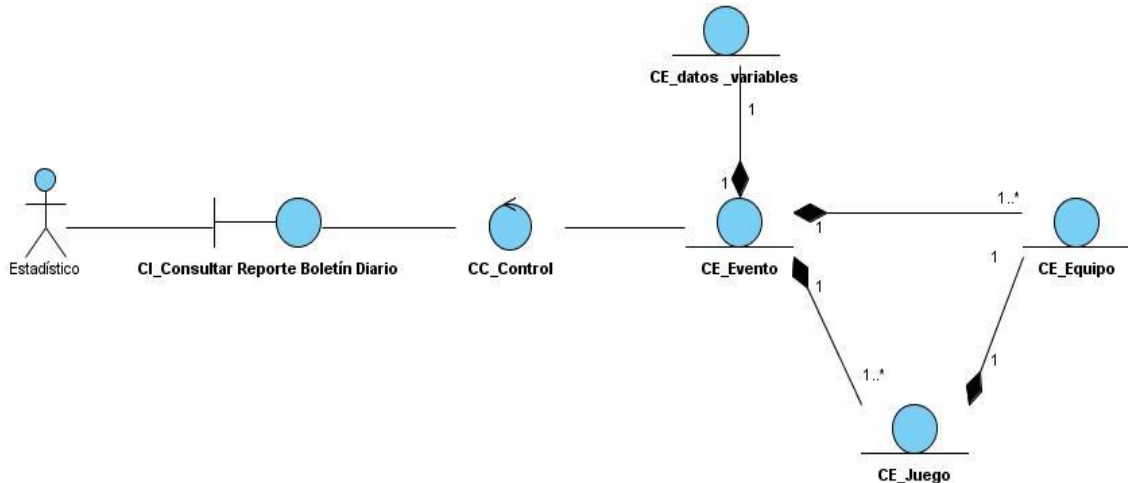


Figura 5: Diagrama de Clases del Análisis del Caso de Uso Consultar Reporte Boletín Diario

2.9 Modelo de Diseño.

Es un modelo de objetos que describe la realización física de los casos de uso, centrándose en los requisitos funcionales y no funcionales, junto con otras restricciones relacionadas con el entorno de implementación. Sirve de abstracción de la implementación del sistema y es de ese modo utilizada como una entrada fundamental de las actividades de implementación. (Jacobson, y otros, 2000)

2.9.1 Patrones de Diseño.

Los patrones de diseño expresan esquemas para definir estructuras de diseño (o sus relaciones) para elaborar software, identifican clases, instancias, roles, colaboraciones y la distribución de responsabilidades. Proponen una forma de reutilizar la experiencia de los desarrolladores, para ello clasifican y describen formas de solucionar problemas que ocurren de forma frecuente en el desarrollo. Están basados en la recopilación del conocimiento de los expertos en desarrollo de software.

Se utilizó para el diseño del sistema el patrón Data Access Object (DAO).

2.9.1.1 DAO.

El patrón de diseño DAO es usado para abstraer y encapsular todos los accesos a la fuente de datos, maneja la conexión con la fuente de datos para obtener y almacenar datos.

Ventajas

Los objetos de negocio pueden utilizar la fuente de datos sin conocer los detalles específicos de su implementación. El acceso es transparente porque los detalles de la implementación se ocultan dentro del DAO.

- Permite una migración más fácil, una capa de DAOs posibilita que una aplicación pueda migrar a una implementación de base de datos diferente. Los objetos de negocio no conocen la implementación de datos subyacente, la migración implica cambios sólo en la capa DAO.
- Reduce la complejidad del código de los objetos de negocio, como los DAOs manejan todas las complejidades del acceso a los datos, se simplifica el código de los objetos de negocio y de otros clientes que los utilizan.
- Centraliza todos los accesos a datos en una capa independiente, como todas las operaciones de acceso a los datos se ha delegado en los DAOs, esto se puede ver como una capa que aísla el resto de la aplicación de la implementación de acceso a los datos. Esta centralización hace que la aplicación sea más sencilla de mantener y de manejar.

2.9.2 Diagramas de Interacción (Colaboración o Secuencia).

Un diagrama de interacción es un conjunto de objetos y sus relaciones, incluyendo los mensajes que se pueden enviar entre ellos. Los diagramas de colaboración y secuencia se utilizan para modelar los aspectos dinámicos de un sistema.

2.9.2.1 Diagramas de Secuencia.

A continuación se encuentran los diagramas de secuencia de cada una de las secciones del caso de uso Gestionar Juego, los otros diagramas se podrán encontrar en los anexos.

ANÁLISIS Y DISEÑO

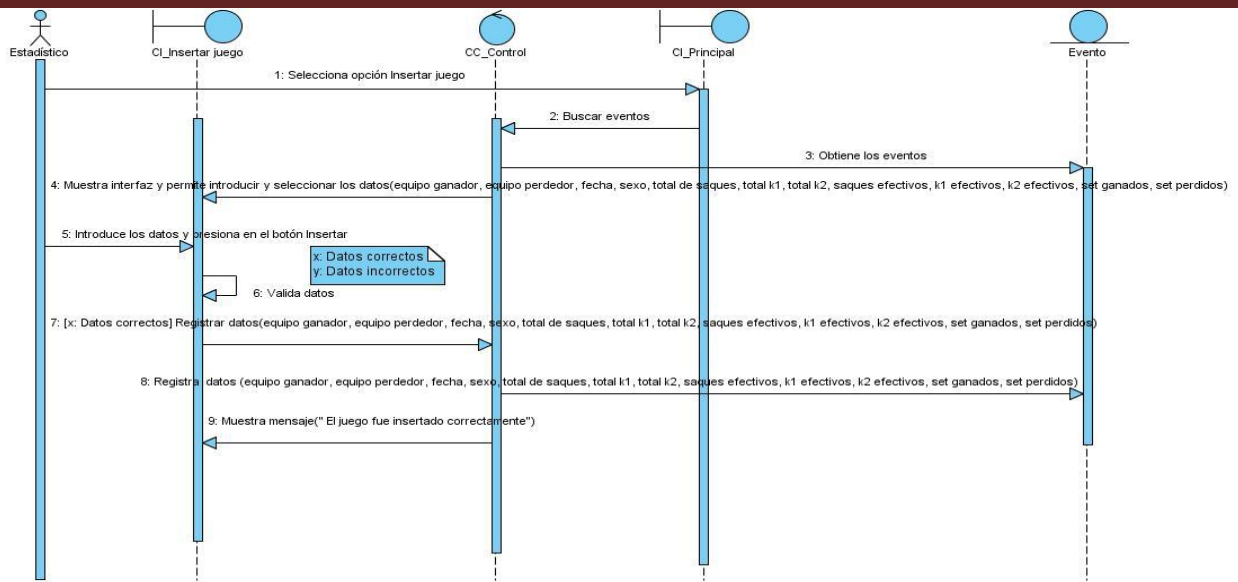


Figura 6: Diagrama de Secuencia del Caso de Uso Gestionar Juego Sección Insertar Juego

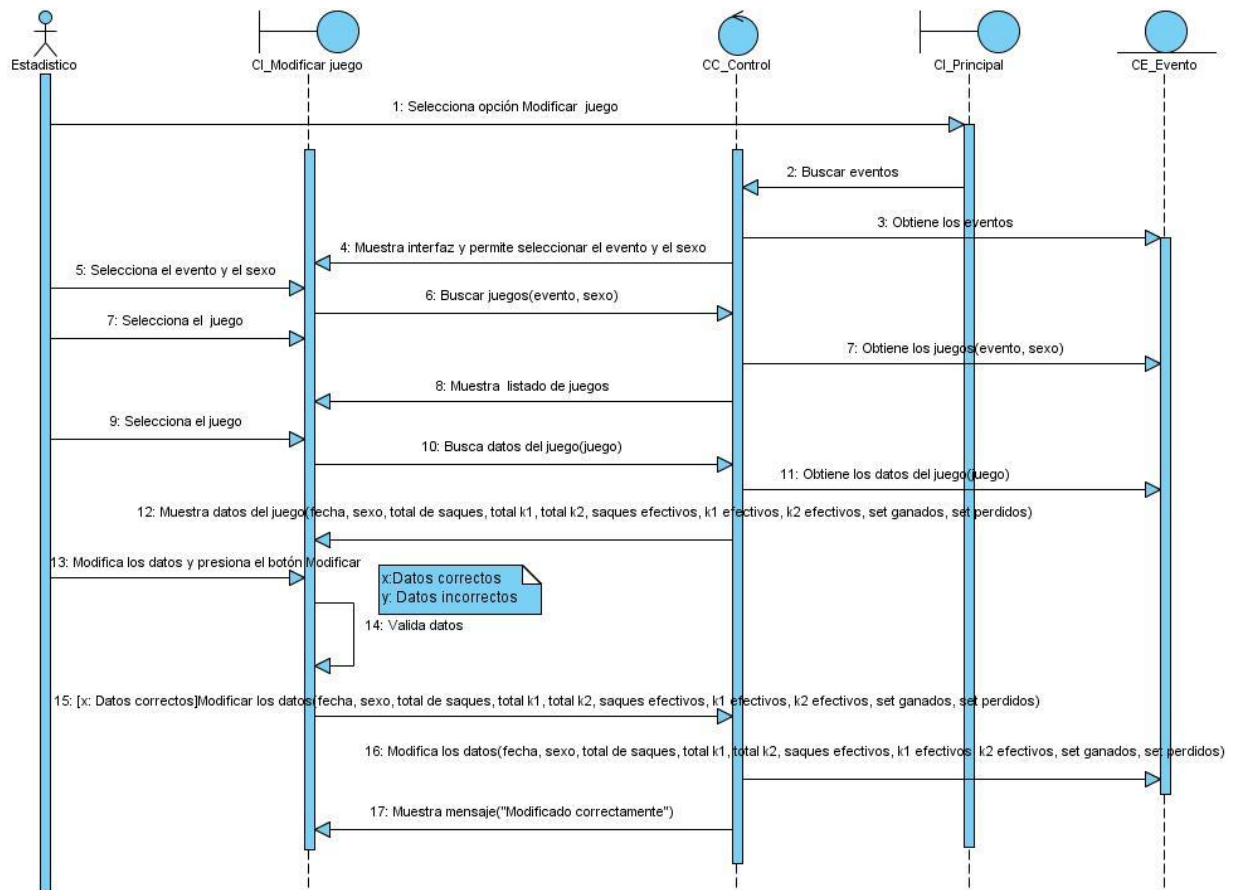


Figura 7: Diagrama de Secuencia del Caso de Uso Gestionar Juego Sección Modificar Juego

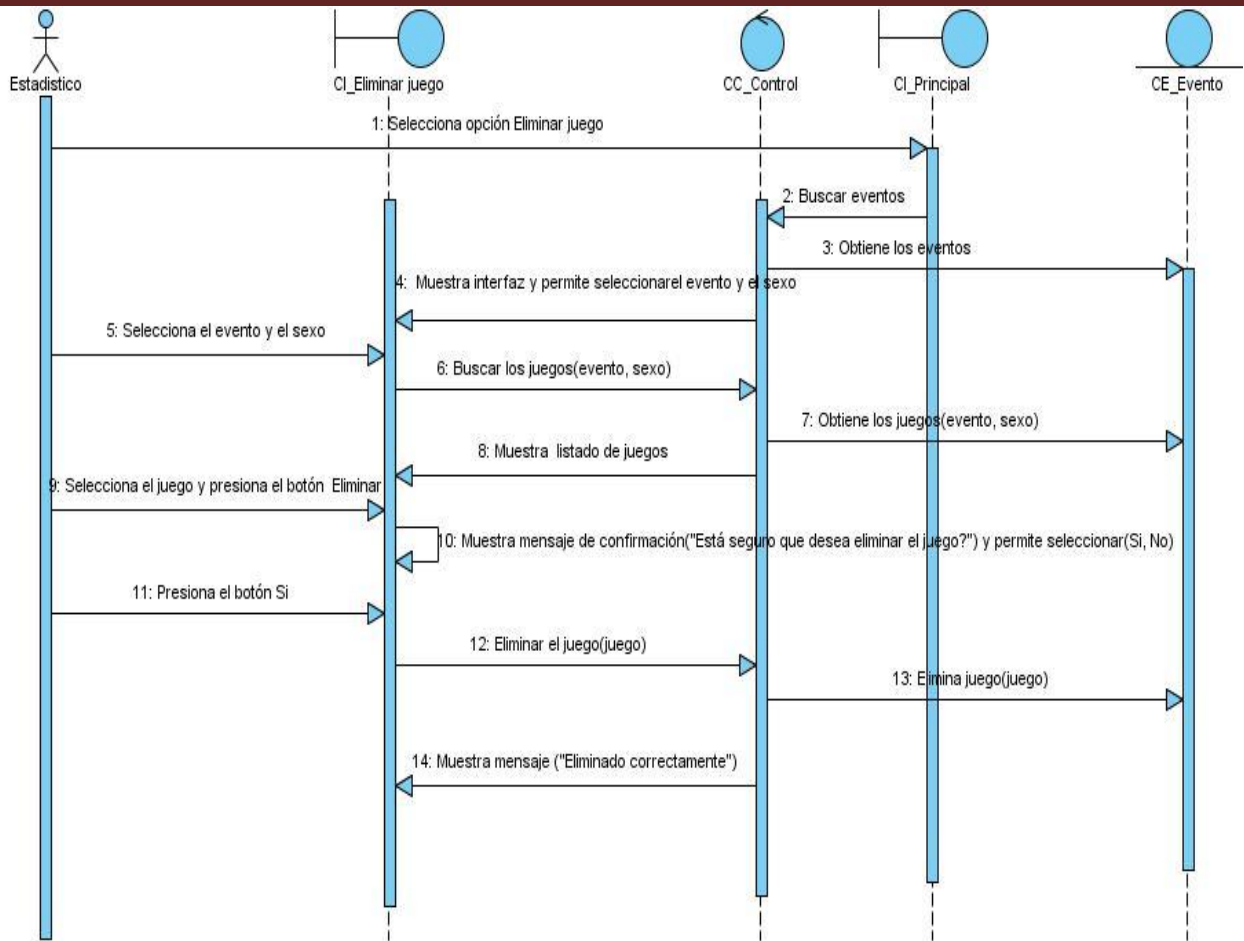


Figura 8: Diagrama de Secuencia del Caso de Uso Gestionar Juego Sección Eliminar Juego

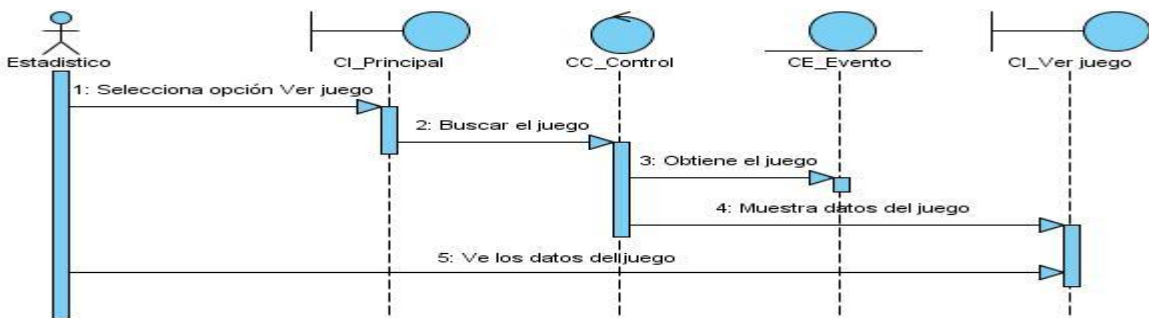


Figura 9: Diagrama de Secuencia del Caso de Uso Gestionar Juego Sección Ver Juego

2.9.3 Diagramas de Clases del Diseño.

A continuación se muestra el diagrama de clases del diseño del caso de uso Gestionar juego, los otros se podrán encontrar en los anexos.

ANÁLISIS Y DISEÑO

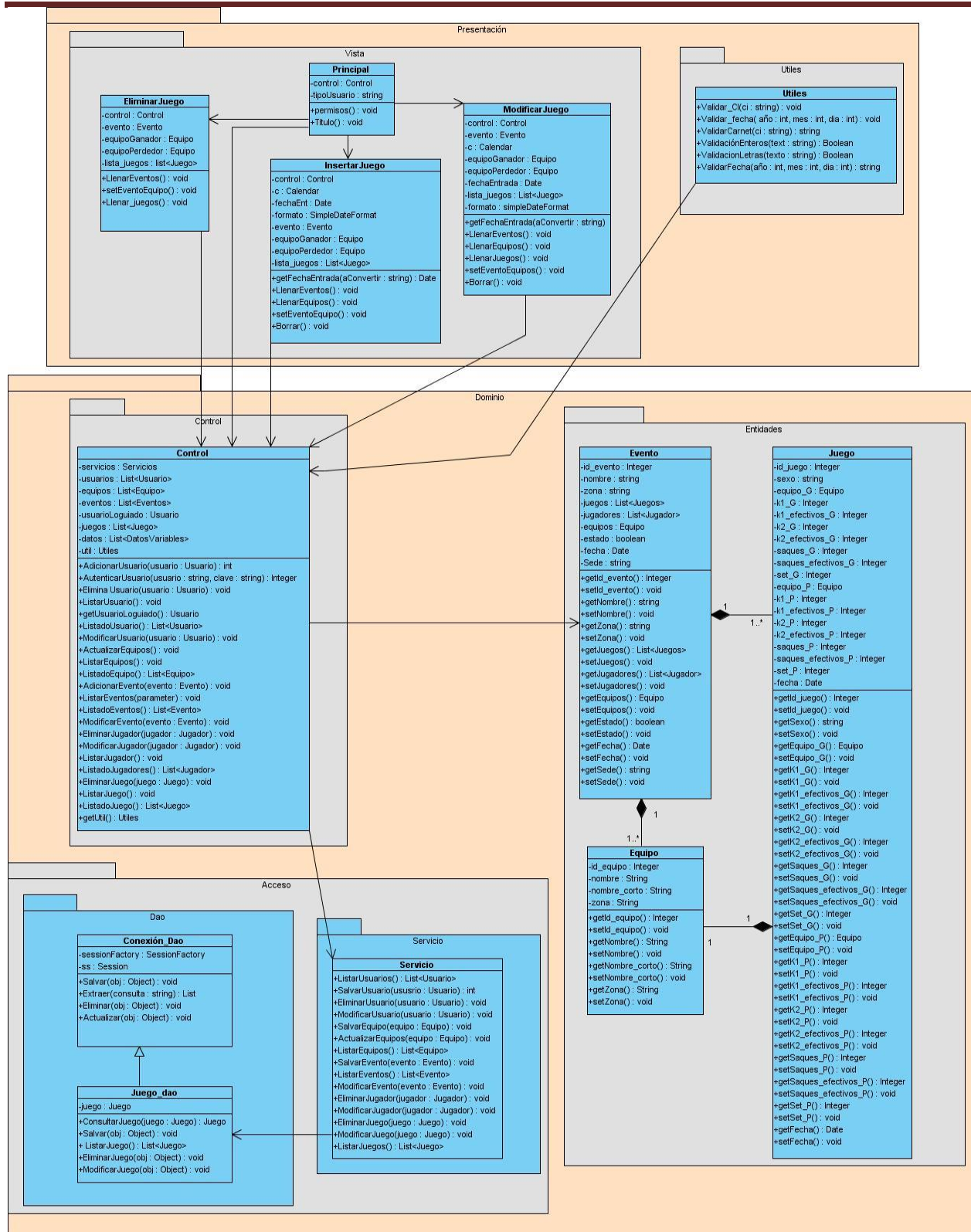


Figura 10: Diagrama de Clases del Diseño del Caso de Uso Gestionar Juego

2.9.3.1 Breve descripción de las Clases del Diseño.

Capa de Presentación:

I_AutenticaciónVisual: Esta clase es el respaldo de la vista encargada de presentar una interfaz al usuario para acceder al sistema en dependencia del rol que desempeñe en el mismo.

I_Principal: Esta clase es el respaldo de la vista encargada de presentar una interfaz al usuario con todas las funcionalidades posibles a realizar en la aplicación en dependencia al rol que desempeñe en el mismo.

I_InsertarUsuario: Esta clase es el respaldo de la vista encargada de presentar una interfaz al usuario con todos los campos necesarios para insertar un usuario nuevo, como lo son nombre y apellidos de la persona, usuario, clave, confirmar clave y categoría.

I_ModificarUsuario: Esta clase es el respaldo de la vista encargada de presentar una interfaz al usuario con todos los campos posibles a modificar, todos menos la categoría si es el único administrador que se encuentra registrado en la base de datos.

I_EliminarUsuario: Esta clase es el respaldo de la vista encargada de presentar una interfaz al usuario con todos los campos necesarios para eliminar un usuario, todos pueden ser eliminados menos el que se encuentra realizando estas operaciones.

I_InsertarEvento: Esta clase es el respaldo de la vista encargada de presentar una interfaz al usuario con todos los campos necesarios para insertar un evento nuevo, como lo son nombre, sede, categoría, zona y fecha.

I_ModificarEvento: Esta clase es el respaldo de la vista encargada de presentar una interfaz al usuario con todos los campos necesarios para modificar un evento, los mismos siempre pueden ser modificados mientras se encuentre abierto.

I_ConsultarEvento: Esta clase es el respaldo de la vista encargada de presentar una interfaz al usuario con diferentes formas para consultar eventos. Los eventos pueden ser listados por varios criterios de búsquedas, se pueden filtrar por fecha, categoría y zona. Además, permite modificar los eventos sin tener la necesidad de salir de la interfaz.

I_InsertarJugador: Esta clase es el respaldo de la vista encargada de presentar una interfaz al usuario con todos los campos necesarios para insertar un jugador nuevo, como lo son evento, equipo, nombre, apellidos, carnet de identidad, peso, altura y sexo. Además, permite modificar y eliminar los jugadores sin tener la necesidad de salir de la interfaz.

I_ModificarJugador: Esta clase es el respaldo de la vista encargada de presentar una interfaz al usuario con todos los campos necesarios para modificar un jugador.

I_EliminarJugador: Esta clase es el respaldo de la vista encargada de presentar una interfaz al usuario con todos los campos necesarios para eliminar un jugador.

I_ConsultarJugador: Esta clase es el respaldo de la vista encargada de presentar una interfaz al usuario con diferentes formas para consultar jugadores. Los jugadores pueden ser listados por varios criterios de búsquedas, se pueden filtrar por evento, equipo, sexo y carnet de identidad. Además, permite modificar y eliminar los jugadores sin tener la necesidad de salir de la interfaz.

I_InsertarJuego: Esta clase es el respaldo de la vista encargada de presentar una interfaz al usuario con todos los campos necesarios para insertar un juego nuevo, como lo son evento, fecha, del equipo ganador y equipo perdedor total de k1, k2 y saques; k1, k2 y saques efectivos y los sets que ganaron.

I_ModificarJuego: Esta clase es el respaldo de la vista encargada de presentar una interfaz al usuario con todos los campos necesarios para modificar un juego, menos los sets del equipo ganador que siempre van a ser tres.

I_EliminarJuego: Esta clase es el respaldo de la vista encargada de presentar una interfaz al usuario con todos los campos necesarios para eliminar un juego.

I_ConsultarJuego: Esta clase es el respaldo de la vista encargada de presentar una interfaz al usuario con diferentes formas para consultar juegos. Los juegos pueden ser listados por varios criterios de búsquedas, se pueden filtrar por evento, equipo ganador, equipo perdedor y fecha. Además, permite modificar y eliminar los juegos sin tener la necesidad de salir de la interfaz.

I_AbrirEvento: Esta clase es el respaldo de la vista encargada de presentar una interfaz al usuario con todos los campos necesarios que den la posibilidad de abrir un evento.

I_ConsultarReporteBoletínDiario: Esta clase es el respaldo de la vista encargada de presentar una interfaz al usuario con la posibilidad de Consultar el Reporte Boletín Diario. Además permite exportar el reporte a formato PDF y ser impreso a voluntad del usuario.

I_ConsultarReporteResumenEstadisticoCadaEquipo: Esta clase es el respaldo de la vista encargada de presentar una interfaz al usuario con la posibilidad de Consultar el Reporte Resumen Estadístico de cada Equipo. Además, permite exportar el reporte a formato PDF y ser impreso a voluntad del usuario.

I_ConsultarReporteResultadoObjetivosPedagogicos: Esta clase es el respaldo de la vista encargada de presentar una interfaz al usuario con la posibilidad de Consultar el Reporte

Resultado de los Objetivos Pedagógicos. Además, permite exportar el reporte a formato PDF y ser impreso a voluntad del usuario.

I_ConsultarReporteResultadoPorProvincias: Esta clase es el respaldo de la vista encargada de presentar una interfaz al usuario con la posibilidad de Consultar el Reporte Resultado por Provincias. Además, permite exportar el reporte a formato PDF y ser impreso a voluntad del usuario.

I_ConsultarReporteResultadoAlcanzadoPorProvincia: Esta clase es el respaldo de la vista encargada de presentar una interfaz al usuario con la posibilidad de Consultar el Reporte Resultado Alcanzado por Provincias. Además, permite exportar el reporte a formato PDF y ser impreso a voluntad del usuario.

I_ConsultarReporteEquipoIdeal: Esta clase es el respaldo de la vista encargada de presentar una interfaz al usuario con la posibilidad de Consultar el Reporte Equipo Ideal. Además, permite exportar el reporte a formato PDF y ser impreso a voluntad del usuario.

I_ConsultarReporteControlCompetitivo: Esta clase es el respaldo de la vista encargada de presentar una interfaz al usuario con la posibilidad de Consultar el Reporte Control Competitivo. Además, permite exportar el reporte a formato PDF y ser impreso a voluntad del usuario.

I_ConsultarReporteHistorialUsuario: Esta clase es el respaldo de la vista encargada de presentar una interfaz al usuario con la posibilidad de Consultar el Reporte Historial de Usuario. Además, permite exportar el reporte a formato PDF y ser impreso a voluntad del usuario.

Útiles: Esta clase contiene métodos que pueden ser usados por otras ayudando a disminuir líneas de código.

Capa lógica del negocio:

Control: Es la encargada del flujo de datos entre la capa de presentación y acceso a datos.

Usuario: Es la entidad del dominio a ser persistida por los métodos existentes en la clase Usuario_Dao.

Evento: Es la entidad del dominio a ser persistida por los métodos existentes en la clase Evento_Dao.

Jugador: Es la entidad del dominio a ser persistida por los métodos existentes en la clase Jugador_Dao.

Juego: Es la entidad del dominio a ser persistida por los métodos existentes en la clase Juego_Dao.

Equipo: Es la entidad del dominio a ser persistida por los métodos existentes en la clase Equipo_Dao.

DatosVariables: Es la entidad del dominio a ser persistida por los métodos existentes en la clase DatosVariable_Dao.

Capa acceso a datos:

Servicio: Es la clase encargada de capturar los datos de la clase Control, los valida y los deja listos para la persistencia.

Usuario_Dao: Es la clase que permite ejecutar las consultas Hibernate Query Language (HQL) para insertar, consultar, modificar o eliminar usuarios.

Evento_Dao: Es la clase que permite ejecutar las consultas HQL para insertar, consultar o modificar eventos.

Juego_Dao: Es la clase que permite ejecutar las consultas HQL para insertar, consultar, modificar o eliminar juegos.

Jugadores_Dao: Es la clase que permite ejecutar las consultas HQL para insertar, consultar, modificar o eliminar jugadores.

Equipo_Dao: Es la clase que permite ejecutar las consultas HQL para insertar o consultar equipos.

DatosVariables_Dao: Es la clase que permite ejecutar las consultas HQL para insertar, consultar, o modificar los datos.

Conexion: Las clases Usuario_Dao, Evento_Dao, Juego_Dao, Jugadores_Dao, Equipo_Dao, DatosVariables_Dao heredan de ella, es la clase encargada de crear las transacciones de los datos hacia la base de datos, para esto abre y cierra sesiones.

2.9.4 Diseño de la Base de Datos.

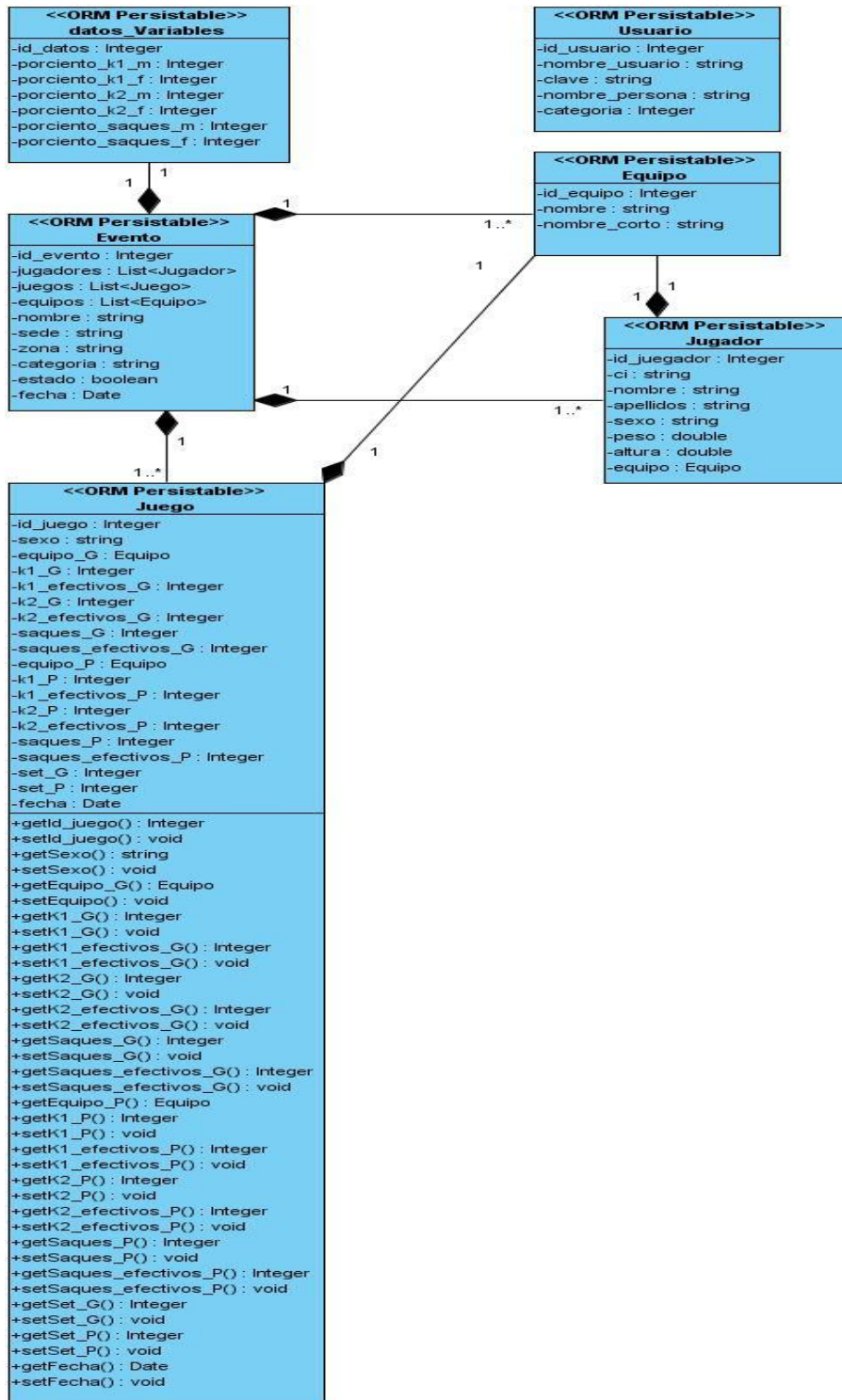


Figura 11: Diagrama de Clases Persistentes

2.9.5 Modelo de Datos.

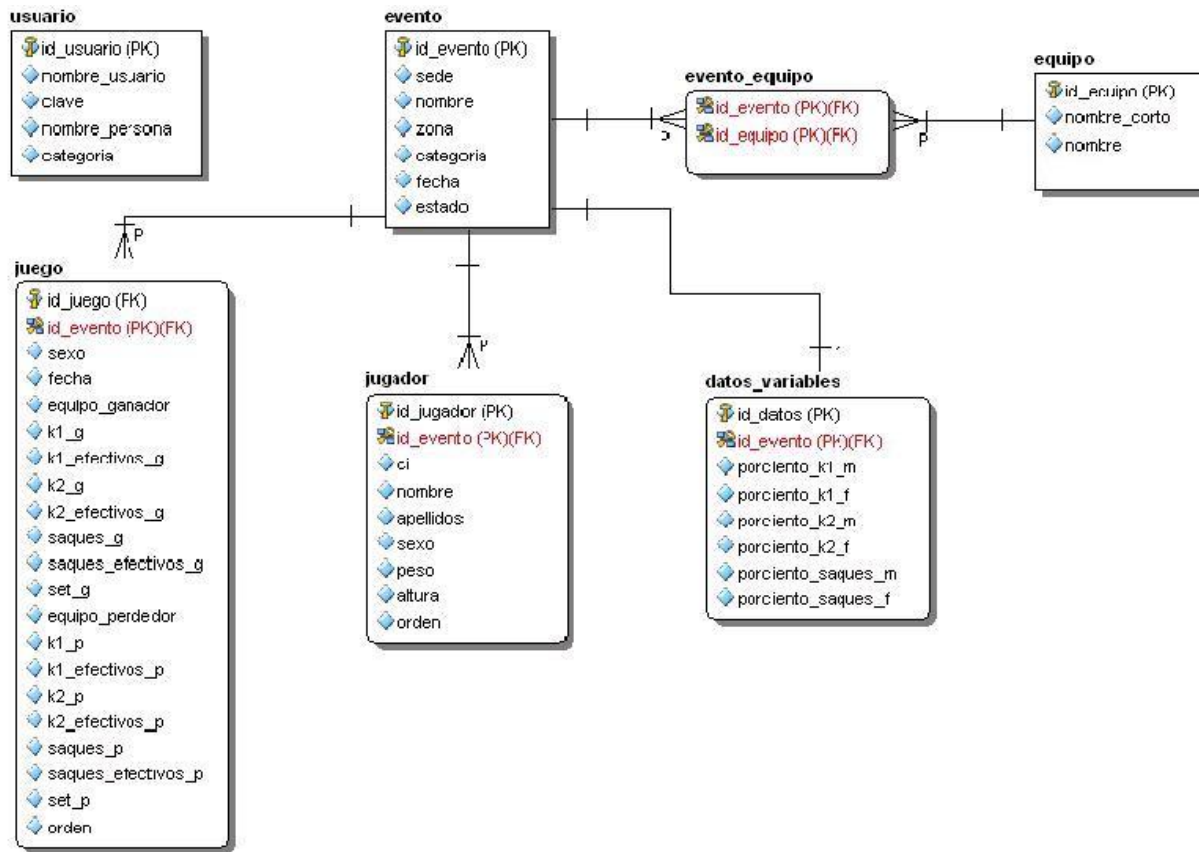


Figura 12: Modelo de Datos

2.9.5.1 Descripción de las Tablas de la Base de Datos.

A continuación se muestra la descripción de la tabla evento, las otras se podrán encontrar en los anexos.

Nombre: evento		
Descripción: Esta tabla contiene los datos de los eventos.		
Atributo	Tipo	Descripción
id_evento	Integer	Representa el identificador de la clase evento.
nombre	String	Representa el nombre del evento.
fecha	Date	Representa la fecha en que se realiza el evento.

ANÁLISIS Y DISEÑO

categoria	String	Representa la categoría que va a realizar el evento.
zona	String	Representa la zona en que se va a realizar el evento.
estado	bool	Representa si el evento está abierto o cerrado.

Tabla 10: Descripción de la Tabla de la Base de Datos "evento"

2.10 Conclusiones.

Con la realización del modelo de análisis y el modelo de diseño se obtuvieron los requerimientos funcionales y no funcionales logrando traducir los mismos a una especificación que describe cómo implementar el sistema.

Se obtuvieron diferentes artefactos como los diagramas de clases de análisis y del diseño, explicándose de una forma más detallada las responsabilidades de cada clase lográndose así una mejor visión y comprensión de lo que se quiere obtener en el sistema.

Capítulo 3

Implementación y Prueba

3.1 Introducción.

Con los artefactos obtenidos en el flujo de trabajo análisis y diseño comienza la implementación del sistema con la elaboración del diagrama de despliegue y el de componentes, los cuales conforman lo que se conoce como un modelo de implementación al describir los componentes a construir, así como su organización y dependencia entre nodos físicos en los que funcionará la aplicación. Para garantizar que la calidad del trabajo realizado sea satisfactoria resulta imprescindible la realización de las pruebas, pues con su total cumplimiento se tendrá un mayor grado de confianza en la labor realizada.

3.2 Modelo de Implementación.

Describe como los elementos del modelo de diseño se implementan en términos de componentes, ficheros de código fuente, ejecutables. Este modelo lo conforman los diagramas de despliegue y componentes.

3.2.1 Diagrama de Despliegue.

En el diagrama de despliegue se indica la situación física de los distintos nodos que componen un sistema. Es decir, se sitúa el software en el hardware que lo contiene. Cada Hardware se representa como un nodo.

Un nodo es un elemento donde se ejecutan los componentes, representan el despliegue físico de estos componentes.



Figura 13: Diagrama de Despliegue

3.2.2 Diagrama de Componentes.

Se representa como un grafo de componentes de software unidos por medio de relaciones de dependencia (compilación, ejecución), pudiendo mostrarse las interfaces que estos soporten.

Se utiliza para modelar la vista estática de un sistema. Contiene componentes, interfaces y relaciones entre ellos. Como todos los diagramas, también puede contener paquetes utilizados para agrupar elementos del modelo.

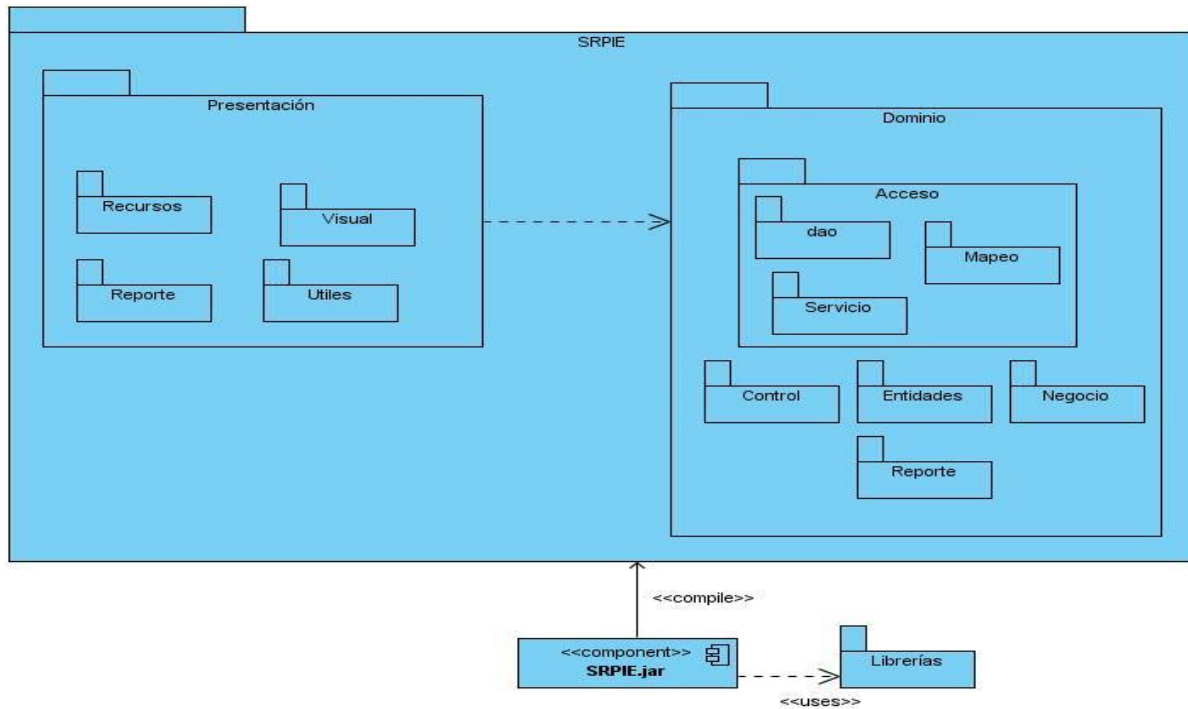


Figura 14: Diagrama de Componentes

3.3 Modelo de Prueba.

La prueba es una actividad en la cual un sistema o componente es ejecutado bajo condiciones o requerimientos especificados, los resultados son observados y registrados, y una evaluación es hecha de algún aspecto del sistema o componente. Deben centrarse en dos objetivos fundamentales: probar si el software no hace lo que debe hacer, y probar si el software hace lo que no debe hacer. Se realizan a toda la aplicación, en dependencia a lo que se quiera probar se traza una estrategia, se escoge la técnica, el método y el tipo de prueba a utilizar. Para lograr que las mismas sean satisfactorias hay que tener bien claro cuáles son sus objetivos.

IMPLEMENTACIÓN Y PRUEBA

Existen diferentes tipos de pruebas: unitarias, de integración, del sistema y de aceptación. La prueba del sistema forma parte de las pruebas de Caja Negra y permite comprobar que se cumplan los requerimientos funcionales.

3.3.1 Pruebas de Caja Negra.

Se centran principalmente en los requisitos funcionales del software. Estas pruebas permiten obtener un conjunto de condiciones de entrada que ejerciten completamente todos los requisitos funcionales de un programa. En ellas se ignora la estructura de control, centrándose en los requisitos funcionales del sistema y ejercitándolos.

Estas pruebas permiten encontrar:

- Funciones incorrectas o ausentes.
- Errores de interfaz.
- Errores en estructuras de datos.

3.3.2 Casos de Prueba.

Son un conjunto de condiciones o variables bajo las cuales se determina si el requisito de una aplicación es parcial o completamente satisfactorio. Definen un conjunto de datos de entradas, condiciones de ejecución y resultados esperados de las pruebas, identificados para hacer una evaluación de los aspectos específicos de un elemento objeto de prueba. Cada caso de prueba está asociado a un escenario de un caso de uso en particular.

A continuación se muestra el caso de prueba del caso de uso Consultar Reporte Boletín Diario, los otros se podrán encontrar en los anexos.

Entrada	Resultados	Condición
El estadístico selecciona la opción Consultar Reporte Boletín Diario.	El sistema muestra una interfaz para seleccionar el evento, sexo, fecha.	

IMPLEMENTACIÓN Y PRUEBA

<p>El estadístico selecciona Evento:" Antonio Maceo" Sexo:"F" Fecha:"17/04/10" Y presiona el botón Ver reporte.</p>	<p>El sistema comunica al usuario que no existe registrado ningún juego en esa fecha. Ejemplo:"No se han registrado juegos femeninos en esa fecha."</p>	<p>No existan juegos registrados en esa fecha.</p>
<p>El estadístico selecciona Evento:" Antonio Maceo" Sexo:"F" Fecha:"9/04/10" Y presiona el botón Ver reporte.</p>	<p>El sistema muestra el reporte boletín diario.</p>	<p>Deben existir juegos registrados en esa fecha.</p>
<p>El estadístico selecciona Evento:" Rubén Martínez Villena" Sexo:"M" Fecha:"28/04/10" Y presiona el botón Ver reporte.</p>	<p>El sistema notifica al usuario que no se han registrado juegos. Ejemplo:"No se han registrado juegos masculinos en este día"</p>	<p>No existan juegos masculinos registrados con sexo masculino.</p>

Tabla 11: Caso de Prueba del Caso de Uso Consultar Reporte Boletín Diario

3.4 Conclusiones.

Con la creación del diagrama de despliegue se alcanzó una visión más abarcadora de la distribución física del sistema. Con el diagrama de componentes quedaron descritos los elementos físicos del sistema y sus relaciones.

Se realizó la descripción de los casos de prueba para los casos de uso del sistema comprobando que la aplicación cumple satisfactoriamente con los requerimientos definidos en el flujo de trabajo análisis y diseño.

CONCLUSIONES GENERALES

- Se determinó que el principal problema de los Campeonatos Zonales Nacionales del Voleibol es la ausencia de una aplicación que automatice los procesos de información estadística en las categorías menores.
- Se demostró que los sistemas destinados a la gestión de información estadística a nivel mundial no cumplen con las necesidades de la Federación Cubana de Voleibol.
- Se determinó que las metodologías, lenguajes y herramientas más adecuadas para el desarrollo de la aplicación son: RUP como metodología de desarrollo con el lenguaje de modelado UML y la herramienta CASE Visual Paradigm 6.4.
- Se obtuvieron los requerimientos funcionales que dieron cumplimiento a las necesidades del cliente.
- Se obtuvo una aplicación que permitirá a la Federación Cubana de Voleibol controlar los procesos de información estadística en las categorías menores.

RECOMENDACIONES

- Teniendo en cuenta que la aplicación genera reportes colectivos se recomienda agregar nuevas funcionalidades que permitan obtener reportes individuales.
- Poner en práctica el sistema en todos los eventos zonales de las categorías menores de forma tal que las informaciones manejadas en los mismos queden registradas con el objetivo de que la FCV pueda comparar los resultados obtenidos en un año con los anteriores verificando así que se cumplan las estrategias trazadas por ellos.

BIBLIOGRAFÍA

- Bascón Pantoja, Ernesto. 2004.** El patrón de diseño Modelo-Vista-Controlador (MVC) y su implementación en Java Swing. *www.ucbcba.edu.bo*. [En línea] 2004. [Citado el: 12 de mayo de 2010.]
<http://www.ucbcba.edu.bo/Publicaciones/revistas/actanova/documentos/v2n4/v2.n4.bascon.pdf>.
- Becerril, Francisco. 1998.** *Java a su alcance*. s.l. : Mc Graw Hill, 1998.
- Carreón Suarez del Real, María Cristina. 2008.** Construcción de un catálogo de patrones de requisitos funcionales para ERP. Tesis de maestría. *www.essi.upc.edu*. [En línea] 25 de junio de 2008. [Citado el: 8 de febrero de 2010.]
<http://www.essi.upc.edu/~ccarreon/tesisMasterCristinaCarreon.pdf>.
- CASASOLA ROMERO, OSCAR. 2002.** Programación en castellano. Introducción a UML. *www.programacion.com*. [En línea] 25 de noviembre de 2002. [Citado el: 14 de mayo de 2010.]
http://www.programacion.com/articulo/introduccion_a_uml_181.
- Crespo, C. 2004** Adictos al trabajo.com. [En línea] 2004. [Citado el: 12 de febrero de 2010.]
<http://www.adictosaltrabajo.com/index.php>.
- Chapela Martínez, Jairo. 2007.** Introducción al entorno de desarrollo Eclipse. *www-gris.det.uvigo.es*. [En línea] 26 de septiembre de 2007. [Citado el: 12 de mayo de 2010.]
http://www-gris.det.uvigo.es/wiki/pub/Main/MiscResources/Manual_Eclipse.pdf.
- Hernández León, R. A. y Coello González, A. 2002.** *El paradigma cuantitativo de la investigación científica*. La Habana: s.l. : Universitaria, 2002.
- Hernández Meléndrez, E. 2006.** *Cómo escribir una tesis*. Escuela Nacional de Salud Pública. 2006.
- Hernández Sampieri, Roberto, Hernández Colado, Carlos y Baptista Lucio, Pilar. 2006.** *Metodología de la investigación*. Cuarta edición. México: s.l. : MC Graw Hill, 2006.
- Huarachi, Maritza. 2009.** TRABAJO DE INVESTIGACIÓN Y EXPOSICIÓN. *www.mex.tl*. [En línea] julio de 2009. [Citado el: 2 de marzo de 2010.]
<http://www.mex.tl/images/18149/informe.doc>.
- Itzcoalt Álvarez, M. 2004.** Desarrollo Ágil con SCRUM . *www.sg.com.mx*. [En línea] 2004. [Citado el: 2 de marzo de 2010.]
<http://www.sg.com.mx/sg07/presentaciones/Mejora%20de%20procesos/SG07.P02.Scrum.pdf>.

BIBLIOGRAFÍA

Lou Torrijos, Ricard. 2003. Programación en castellano. *www.programacion.com*. [En línea] 14 de diciembre de 2003. [Citado el: 14 de mayo de 2010.] http://www.programacion.com/articulo/catalogo_de_patrones_de_diseno_j2ee_y_ii:_capas_de_negocio_y_de_integracion_243/8.

Manzanedo del Campo, Miguel Ángel, y otros. 1998. Guía de iniciación al Lenguaje Java en formato HTML. Comparativa con otros L.P.O.O. *pisuerga.inf.ubu.es*. [En línea] 1998. [Citado el: 14 de febrero de 2010.] http://pisuerga.inf.ubu.es/lsi/Invest/Java/Tuto.Oct98/1_Intro/4_Compar.htm.

González Campos, S., Fernández Martínez, L.F. 2006 Programación Extrema: Prácticas, Aceptación y Controversia. [En línea] 2006. [Citado el: 7 de marzo de 2010.] www2.uacj.mx/IIT/CULCYT/mayo-agosto2006/8ArtProg.pdf.

Guerrero, F. 2004 MailxMail.com. [En línea] 2004. [Citado el: 3 de febrero de 2010.] <http://www.mailxmail.com/curso-lenguaje-c-guia-programadores>.

Grupo de investigación de ingeniería de software. 2005. Tertulia de Ingeniería de software. La importancia de la arquitectura en el de desarrollo del software de calidad. *www.eafit.edu.co*. [En línea] 17 de febrero de 2005. [Citado el: 6 de febrero de 2010.] <http://www.eafit.edu.co/NR/rdonlyres/223A8F47-27B5-4EB8-B695-4097F745D701/0/Arquitectura.pdf>.

REFERENCIAS BIBLIOGRÁFICAS

Alvarez, S. 2007 desarrollo web.com. [En línea] 2007. [Citado el: 11 de febrero de 2010.] <http://www.desarrolloweb.com/articulos/sistemas-gestores-bases-datos.html>.

Arce Morell, Domingo Andrés. 2009. Buscador basado en contenido XML de las páginas de centros y departamentos de la UV. [En línea] 2009. [Citado el: 15 de marzo de 2010.] <http://www.google.com/cu/url?sa=t&source=web&ct=res&cd=7&ved=0CDAQFjAG&url=http%3A%2F%2Fvusep2.googlecode.com%2Ffiles%2FPFC%2520Buscador%2520basado%2520en%2520contenido%2520XML.pdf&rct=j&q=%22Es+el+motor+de+base+de+datos+relacional+m%C3%A1s+usado%22&ei=AK>.

Ayala, Claudia P., y otros. 2001. Análisis Comparativo de Lenguajes de Modelado Orientados a Objetivos basados en i*. *www.lsi.upc.edu*. [En línea] 2001. [Citado el: 15 de enero de 2010.] <http://www.lsi.upc.edu/~cayala/Papers/JIISIC.pdf>.

Barrios Núñez, Juan Manuel. 2003. INVESTIGACIÓN DE LA PLATAFORMA J2EE Y SU APLICACIÓN PRÁCTICA. *www.dcc.uchile.cl*. [En línea] enero de 2003. [Citado el: 13 de mayo de 2010.] <http://www.dcc.uchile.cl/~jbarrios/J2EE/MemoriaJ2EE.pdf>.

Bascón Pantoja, Ernesto. 2004. El patrón de diseño Modelo-Vista-Controlador (MVC) y su implementación en Java Swing. *www.ucbcb.edu.bo*. [En línea] 2004. [Citado el: 12 de mayo de 2010.] <http://www.ucbcb.edu.bo/Publicaciones/revistas/actanova/documentos/v2n4/v2.n4.bascon.pdf>.

Beck, Kent. 1999. *Extreme Programming Explained. Embrace Change.* 1999. http://www.freedownloadmanager.org/es/downloads/Paradigma_Visual_para_UML_%28M%C3%8D%29_14720_p/.

Brown, B. 2009 Volleyball Scores. [En línea] 2009. [Citado el: 4 de noviembre de 2009.] <http://www.volleyballscores.co.uk/about-this-site/vis.html>.

Buenas Tareas. 2010. Buenas Tareas. *www.buenastareas.com*. [En línea] 2010. [Citado el: 10 de febrero de 2010.] <http://www.buenastareas.com/ensayos/Gestores-De-Bases-De-Datos/12666.html>.

Caballero Franco, Miriam, y otros. 2009. Tecnología web 2.0 para servicios por videoconferencia de la SOMECE. *itzamna.bnct.ipn.mx:8080*. [En línea] 2009. [Citado el: 13 de mayo de 2010.] <http://itzamna.bnct.ipn.mx:8080/dspace/bitstream/123456789/4724/1/C7.1393.pdf>.

REFERENCIAS BIBLIOGRÁFICAS

- Camaleón. 2010.** Camaleón. *www.centrocamaleon.com*. [En línea] 21 de enero de 2010. [Citado el: 14 de mayo de 2010.] <http://www.centrocamaleon.com/v1/?p=233>.
- Catalani, Exequiel. 2007.** exequielc.wordpress.com. [En línea] 20 de agosto de 2007. [Citado el: 27 de mayo de 2010.] <http://exequielc.wordpress.com/2007/08/20/arquitectura-modelovistacontrolador/>.
- Chapela Martínez, Jairo. 2007.** Introducción al entorno de desarrollo Eclipse. *www-gris.det.uvigo.es*. [En línea] 26 de septiembre de 2007. [Citado el: 12 de mayo de 2010.] http://www-gris.det.uvigo.es/wiki/pub/Main/MiscResources/Manual_Eclipse.pdf.
- Díaz, Luis Carlos, Carrillo, Angela y Alvarado, Deicy. 2009.** IS, Procesos de Software y UML en el Contexto de ADOO. *sophia.javeriana.edu.co*. [En línea] enero de 2009. [Citado el: 10 de enero de 2010.] <http://sophia.javeriana.edu.co/~lfranky/adoo/c03-ProcesosIngSoft-relacionUML.pdf>.
- Espinoza, Humberto. 2005.** PostgreSQL Una Alternativa de DBMS Open Source. *www.lgs.com.ve*. [En línea] 2005. [Citado el: 27 de febrero de 2010.] http://www.lgs.com.ve/pres/PresentacionES_PSQL.pdf.
- Gilfillan, Ian. 2003.** *La biblia de MySQL*. s.l. : Anaya Multimedia, 2003..
- Giraldo, Luis y Zapata, Yuliana. 2005.** Herramientas de desarrollo de Ingeniería de Software para Linux. *hugolopez.phi.com.co*. [En línea] 24 de septiembre de 2005. [Citado el: 25 de febrero de 2010.] http://hugolopez.phi.com.co/docs/download/file=Giraldo-Zapata-Herramientas%20de%20ISW.pdf,_id=17.
- GPI. 2009.** GPI. *www.gpilatam.com*. [En línea] 2009. [Citado el: 9 de febrero de 2010.] <http://www.gpilatam.com/ComoTrabajamos/DisciplinaTrabajo.aspx>.
- Hibernate. 2009.** Hibernate. *www.juntadeandalucia.es*. [En línea] 2009. [Citado el: 15 de abril de 2010.] <http://www.juntadeandalucia.es/xwiki/bin/view/MADEJA/Hibernate#caracteristicas>.
- Integra Tecnología y Comunicación. 2009.** Integra. Tecnología y Comunicación. *www.integracanarias.com*. [En línea] 2009. [Citado el: 9 de febrero de 2010.] http://www.integracanarias.com/index.php?option=com_content&view=article&id=30&Itemid=36
- Jacobson, Ivar, Booch, Grady y Rumbaugh, James. 2000.** *El proceso unificado de desarrollo de software*. Madrid : Series Editors, 2000.
- JasperReports. 2001.** JasperReports. *www.cepeu.edu.py*. [En línea] 2001. [Citado el: 11 de abril de 2010.] http://www.cepeu.edu.py/LIBROS_ELECTRONICOS_3/lpcu089%20-%202001.pdf.
- Larman, Craig. 1999.** *UML y Patrones. Introducción al análisis y diseño orientado a objetos*. México : s.n., 1999.

REFERENCIAS BIBLIOGRÁFICAS

- Letelier, Patricio y Penadés, M^a Carmen. 2006.** Metodologías ágiles para el desarrollo de software: eXtreme Programming (XP). *www.willydev.net*. [En línea] 2006. [Citado el: 2 de febrero de 2010.] <http://www.willydev.net/descargas/masyxp.pdf>.
- Luciano. 2008** Entornos de Desarrollo Integrado para Java. [En línea] 2008. [Citado el: 11 de febrero de 2010.] <http://luauf.com/2008/05/13/entornos-de>
- Masip, D. 2002** desarrollo web.com. [En línea] 2002. [Citado el: 16 de febrero de 2010.] <http://www.desarrolloweb.com/arti>
- Molpeceres, A. 2003** *Procesos de desarrollo: RUP, XP, FDD*. 2003.
- Mosqueda, Wilmer. 2008.** iuttol.wmsistemas.net. [En línea] 29 de julio de 2008. [Citado el: 13 de mayo de 2010.] <http://iuttol.wmsistemas.net/?p=6>.
- Pressman, Roger S. 2002.** *Ingeniería del Software: Un enfoque práctico*. s.l. : McGraw-Hill, 2002.
- Ruiz Valdez, Patricia. 2009.** Artículo científico. *jeff.comlu.com*. [En línea] 7 de julio de 2009. [Citado el: 5 de febrero de 2010.] <http://jeff.comlu.com/documento.pdf>.
- Rodríguez Guerra, Marcia Leidis y Aguilera Acosta, Alcides. 2009.** *Análisis, Diseño e Implementación del Sistema Estudio de Contrarios para el Voleibol*. La Habana : s.n., 2009.
- universal, Dubai EFE El. 2008** El universal.com.mx Deportes. [En línea] 2008. [Citado el: 5 de marzo de 2010.] <http://www.eluniversal.com.mx/notas/515239.html>.
- Xinox Software. 2009** JCreator. El entorno de desarrollo integrado para Java más potente. [En línea] 2009. [Citado el: 11 de febrero de 2010.] <http://jcreator.malavida.com/>

GLOSARIO DE TÉRMINOS

- K1: Abreviatura de la palabra técnica complejo estratégico 1, que agrupa la combinación de la recepción, colocación y remate.
- K2: Abreviatura de la palabra técnica complejo estratégico 2, que agrupa la combinación de la defensa, acomodo y ataque.
- RF: Abreviatura de la palabra requerimiento funcional.
- CU: Abreviatura de la palabra casos de uso.
- Estadísticas: forma de expresar las vicisitudes del juego en números.
- Bytecode: código intermedio más abstracto que el código máquina. Habitualmente es tratado como un fichero binario que contiene un programa ejecutable similar a un módulo objeto, que es un fichero binario producido por el compilador cuyo contenido es el código abierto o código máquina.
- USB: El Universal Serial Bus (bus universal en serie) o Conductor Universal en Serie (CUS), abreviado comúnmente USB, es un puerto que sirve para conectar periféricos a una computadora (impresoras, cámara digital, teclado, ratones, disco duro).
- String: Tipo de dato básico en el lenguaje Java que representa una cadena de caracteres.
- Plug-in: Es un módulo de hardware o software que añade una característica o un servicio específico a un sistema más grande.
- Metodología: Una metodología de desarrollo de software es un conjunto de pasos, técnicas y procedimientos que deben seguirse para desarrollar software en cuanto a Quién debe hacer, Qué, Cuándo y Cómo debe hacerlo.
- IBM: International Business Machines, es una empresa multinacional que fabrica y comercializa herramientas, programas y servicios relacionados con la informática.
- XML: siglas en inglés de Extensible Markup Language (lenguaje de marcas extensible). Es un metalenguaje extensible de etiquetas, una manera de definir lenguajes para diferentes necesidades.
- Kit: tecnología en general.
- JUnit: Es un conjunto de bibliotecas creadas por Erich Gamma y Kent Beck que son utilizadas en programación para hacer pruebas unitarias de aplicaciones Java.

Anexo 1: Descripción del Caso de Uso Gestionar Evento.

Nombre CU	Gestionar Evento
Actor	Estadístico
Propósito	Insertar, modificar, abrir o ver un evento.
Resumen	El caso de uso se inicia cuando el estadístico solicita insertar un nuevo evento, modificar uno ya existen, abrir o ver un evento. Una vez realizada la acción que desee el estadístico el caso de uso termina.
Referencia	RF 1
Precondiciones	El estadístico debe estar registrado en el sistema.
Prioridad	Crítico
Flujo Normal de Eventos	
Acciones del Actor	Respuesta del Sistema
1. El CU se inicia cuando el estadístico selecciona Gestionar Evento.	2. Brinda la posibilidad de realizar las acciones: <ul style="list-style-type: none"> • Insertar Evento. • Modificar Evento. • Abrir Evento. • Ver Evento.
Sección "Insertar Evento": Flujo Normal de los Eventos.	
1. Selecciona la opción Insertar Evento.	2. Brinda la posibilidad de introducir: <ul style="list-style-type: none"> • Nombre • Porciento k1 masculino • Porciento k1 femenino • Porciento k2 masculino

ANEXOS

	<ul style="list-style-type: none"> • Porciento k2 femenino • Porciento saque masculino • Porciento saque femenino <p>Y seleccionar:</p> <ul style="list-style-type: none"> • Fecha • Categoría • Zona • Sede <p>Y permite :</p> <ul style="list-style-type: none"> • Insertar el evento. • Salir de la vista actual.
3. Introduce los datos del evento y presiona el botón Insertar.	4. Valida los datos. 5. Registra los datos. 6. El caso de uso termina.
Flujos Alternos	
4.a Si quedó algún campo vacío	
	4. a.1 El sistema envía un mensaje (“Verifique que no haya algún campo vacío.”) 4. a.2 Regresa al paso 3 del Flujo Normal de los Eventos.
4.b Existe algún dato incorrecto	
	4.b.1 El sistema envía un mensaje (“Verifique los datos”) 4. b.2 Regresa al paso 3 del Flujo Normal de los Eventos.
2.a.El estadístico selecciona la opción Salir de la vista actual	
	2. a.1. Regresa a la vista anterior. 2. a.2. El caso de uso termina.

ANEXOS

Sección “Modificar Evento”: Flujo Normal de Eventos	
Acciones del actor	Respuesta del sistema
1. Selecciona opción Modificar Evento.	2. Permite seleccionar: <ul style="list-style-type: none"> • Evento
3. Selecciona el evento.	4. Muestra datos del evento. <ul style="list-style-type: none"> • Nombre • Por ciento k1 masculino • Por ciento k1 femenino • Por ciento k2 masculino • Por ciento k2 femenino • Por ciento saque masculino • Por ciento saque femenino Permite seleccionar: <ul style="list-style-type: none"> • Fecha • Categoría • Zona • Sede Y permite: <ul style="list-style-type: none"> • Modificar los datos del evento • Cerrar
5. Modifica los datos y presiona el botón Modificar.	6. Valida los datos. 7. Actualiza los datos del evento. 8. Muestra mensaje (“Modificado correctamente”). 9. El caso de uso termina.
Flujos Alternos	
6.a Si quedó algún campo vacío	
	6.a.1 El sistema envía un mensaje (“Verifique que no haya algún campo vacío”) 6. a.2 Regresa al paso 4 del Flujo Normal de Eventos.

ANEXOS

6.b Existe algún dato incorrecto	
	6.b.1 El sistema envía un mensaje (“Verifique los datos”) 6. b.2 Regresa al paso 4 del Flujo normal de Eventos.
4.a Selecciona la opción de Cerrar	
	4.2 Regresa a la vista anterior. 4.3 El caso de uso termina.
Sección “Abrir Evento”: Flujo Normal de Evento	
1. Selecciona opción Abrir Evento.	2. Permite seleccionar: <ul style="list-style-type: none"> • Evento Y permite: <ul style="list-style-type: none"> • Abrir Evento. • Cerrar Evento. • Salir de la vista actual.
3. Selecciona el evento y presiona el botón Abrir.	4. Pone al estado del evento abierto. 5. Envía un mensaje (“Evento abierto correctamente”). 6. El caso de uso termina.
Flujos Alternos	
3.a Selecciona la opción Salir de la vista actual	
	3. a.1 Regresa a la vista anterior. 3. a.2 El caso de uso termina.
3.b Selecciona la opción Cerrar Evento	
	3. b.1 Cierra el evento poniendo el estado cerrado. 3.b.2 Envía un mensaje(“ Evento cerrado correctamente”) 3. b.3 El caso de uso termina.
Sección “Ver Evento”: Flujo Normal de Evento	

ANEXOS

1. Selecciona la opción de Ver Evento.	2. Muestra los datos del evento Y permite: <ul style="list-style-type: none"> • Modificar los datos del evento • Cerrar
3. Selecciona la opción Cerrar.	4. Regresa a la vista anterior. 5. El caso de uso termina.
Flujos Alternos	
5.a El actor selecciona la opción de Modificar los datos del evento	
	3. a.1. Brinda la posibilidad de modificar los datos del evento. Ver Sección “Modificar evento”. 3. a.2. El caso de uso termina.
Poscondiciones	Se insertó, modificó, abrió o vio un evento.

Tabla 12: Descripción del Caso de Uso Gestionar Evento

Anexo 2: Descripción del Caso de Uso Gestionar Jugador.

Nombre CU	Gestionar Jugador
Actor	Estadístico
Propósito	Insertar, modificar, eliminar o ver un jugador.
Resumen	El caso de uso se inicia cuando el estadístico decide insertar un nuevo jugador, modificar los datos de uno ya existente, eliminar o ver uno del sistema. Una vez realizada la operación deseada el caso de uso termina.
Referencia	RF 2
Precondiciones	El estadístico debe tener los permisos para realizar las acciones necesarias en el sistema.
Prioridad	Crítico

ANEXOS

Flujo Normal de Eventos	
Acciones del Actor	Respuesta del Sistema
1. El CU se inicia cuando el estadístico selecciona Gestionar jugador.	2. Brinda la posibilidad de realizar las acciones: <ul style="list-style-type: none">• Insertar jugador.• Modificar jugador.• Eliminar jugador.• Ver jugador.
Sección "Insertar jugador": Flujo Normal de Eventos	
3. Selecciona la opción Insertar jugador.	4. Brinda la posibilidad de insertar los datos siguientes: <ul style="list-style-type: none">• Nombre• Apellidos• CI• Peso• Altura Permite seleccionar: <ul style="list-style-type: none">• Sexo• Evento• Equipo Y permite: <ul style="list-style-type: none">• Adicionar• Modificar• Eliminar• Insertar• Cerrar
5. Introduce los datos y presiona el botón Adicionar.	6. Valida los datos.
7. Selecciona Insertar.	8. Registra los datos. 9.El sistema envía mensaje ("Insertado correctamente") 10. El caso de uso termina.

ANEXOS

Flujos Alternos	
*.a Existen campos vacíos	
	.a.1 El sistema envía un mensaje("Verifique que no haya algún campo vacío ") .2 Regresa al paso 5 del Flujo Normal de Eventos.
*.b Existen datos incorrectos	
	.b.1 Envía mensaje ("Verifique los datos") . b.2 Regresa al paso 5 del Flujo Normal de Eventos.
*.c El evento está cerrado	
	c.1 Envía un mensaje ("El evento está cerrado, no pude realizar ninguna acción")
4.a Selecciona la opción de Cerrar	
	4. a.1 Regresa a la vista anterior. 4. a.2 El caso de uso termina.
4.b Selecciona la opción de Modificar	
4. b.1 Selecciona el jugador.	4. b.2 Muestra los datos del jugador.
4. b.3. Modifica los datos y presiona el botón Modificar.	4.b.4 Valida los datos 4. b.5. Actualiza los datos. 4.b.5 Envía un mensaje ("Modificado correctamente")
4.c Selecciona la opción de Eliminar	
4. c.1 Selecciona el jugador. 4. c.1.1 El estadístico presiona el botón Eliminar.	4.c.2 Muestra mensaje de confirmación y permite seleccionar: <ul style="list-style-type: none"> • Si • No
2. c.3. Selecciona la opción Si.	2c.4. Elimina el jugador. 2.c.5.Envía un mensaje("Eliminado correctamente")

ANEXOS

	2.3.6. El caso de uso termina.
4.c.2.1 Selecciona la opción No	
	4. c.2.1 Regresa a la vista anterior.
	4. c.2.2 El caso de uso termina.
Sección “Modificar Jugador”: Flujo Normal de Evento	
Acciones del actor	Respuesta del sistema
1. Selecciona opción Modificar jugador.	2. Brinda la posibilidad de seleccionar: <ul style="list-style-type: none"> • Evento • Equipo
3. Selecciona el evento y el equipo.	4. Permite modificar: <ul style="list-style-type: none"> • Nombre • Apellidos • CI • Peso • Altura • Sexo Y permite: <ul style="list-style-type: none"> • Modificar • Cerrar
5. Modifica los datos del jugador y presiona el botón Modificar.	6. Valida los datos 7. Actualiza los datos del jugador. 8. Envía mensaje (“Modificado correctamente”) 9. El caso de uso termina.
Sección “Eliminar Jugador”: Flujo Normal de Evento	
1. Selecciona opción Eliminar jugador.	2. Brinda la posibilidad de seleccionar: <ul style="list-style-type: none"> • Evento • Equipo
3. Selecciona el evento y equipo.	4. Muestra listado de jugadores. Y permite: <ul style="list-style-type: none"> • Eliminar • Cerrar

ANEXOS

5. Selecciona el jugador y presiona el botón Eliminar.	6. Muestra mensaje de confirmación y permite seleccionar: <ul style="list-style-type: none"> • Si • No
7. Selecciona Si	8. Elimina el jugador. 9. Envía mensaje (“Eliminado correctamente”) 10. El caso de uso termina.
Flujos Alternos	
6.a Selecciona la opción No	
	6. a.1 Regresa a la vista anterior.
	6. a.2 El caso de uso termina.
4.a Selecciona la opción Cerrar	
	4.a.1 Regresa a la vista anterior 4.3 El caso de uso termina.
Sección “Ver Jugador”: Flujo Normal de Eventos	
1. Selecciona la opción de Ver los datos de un jugador.	2. Muestra los datos del jugador Y permite: <ul style="list-style-type: none"> • Cerrar • Modificar los datos del jugador. • Eliminar a un jugador.
3. Selecciona la opción de Cerrar	4. Regresa a la vista anterior. 5. El caso de uso termina.
Flujo Alternos	
3.a El actor selecciona la opción de Modificar jugador	
	3. a.1. Brinda la posibilidad de modificar los datos de un jugador. Ver Sección “Modificar jugador”.
Poscondiciones	Se insertó, modificó, vio o eliminó un jugador.

Tabla 13: Descripción del Caso de Uso Gestionar Jugador

Anexo 3: Descripción del Caso de Uso Consultar Evento.

Nombre del CU	Consultar Evento
Actores	Estadístico
Propósito	Obtener los datos de un evento por zona, sede, categoría o fecha.
Resumen	El caso de uso se inicia cuando el estadístico decide consultar los datos de un evento por la fecha, categoría o zona. Una vez obtenido los resultados puede modificar los datos, terminando así el caso de uso.
Referencias	RF7.
Precondiciones	Deben estar registrados los datos del evento.
Prioridad	Secundario
Flujo Normal de Evento	
Acción del Actor	Respuesta del Sistema
1. El caso de uso se inicia cuando el estadístico accede a la opción Consultar Evento.	2. Brinda la posibilidad de seleccionar : <ul style="list-style-type: none"> • Fecha • Categoría • Zona • Sede Y permite: <ul style="list-style-type: none"> • Realizar una búsqueda a partir de

ANEXOS

	<p>los datos.</p> <ul style="list-style-type: none"> • Cerrar
<p>3. Selecciona los datos que considera para realizar una búsqueda y presiona el botón buscar.</p>	<p>4. Muestra una lista de posibles coincidencias con :</p> <ul style="list-style-type: none"> • Nombre • Fecha • Categoría • Zona • Sede • Porciento k1 masculino • Porciento k1 femenino • Porciento k2 masculino • Porciento k2 femenino • Porciento saque masculino • Porciento saque femenino <p>Y permite:</p> <ul style="list-style-type: none"> • Realizar una nueva búsqueda. • Ver un evento de la lista mostrada. • Modificar los datos de un evento de la lista.
<p>5. Consulta los datos de los eventos de la lista de coincidencia.</p>	
<p>6. El caso de uso termina.</p>	
<p>Poscondiciones</p>	<p>Se consultó los datos de un juego coincidentes con uno o varios criterios de búsqueda.</p>
<p>Flujos Alternos</p>	

ANEXOS

*.a El estadístico presiona el botón Cerrar	
	a.1 Regresa a la vista anterior.
	a.2 El caso de uso termina.
4. a El actor no seleccionó ningún dato de búsqueda.	
	4. a.1 Muestra el mensaje de información “Debe especificar un criterio de búsqueda”.
	4. a.2 Regresa al paso 2 del Flujo Básico.
4. b El sistema no encuentra ninguna coincidencia.	
	4. b.1 Muestra el mensaje “No se encontraron coincidencias” y permite realizar una nueva búsqueda.
	4. b.2 Regresa al paso 2 del Flujo Básico.
4. c El estadístico selecciona la opción de realizar una nueva búsqueda.	
	4. c.1 Regresa al paso 2 del Flujo Básico.
4. d El estadístico selecciona la opción de ver los datos de un evento de la lista mostrada.	
	4. d.1 Muestra los datos del evento. Ver CU Gestionar Evento.
	4. d.2 El caso de uso termina.
4.e El estadístico selecciona la opción de modificar los datos de un evento de la lista	
	4. e.1 Muestra los datos del evento permitiendo cambiar sus valores. Ver CU Gestionar Evento.
	4. e.2 El caso de uso termina.

Tabla 14: Descripción del Caso de Uso Consultar Evento

Anexo 4: Descripción del Caso de Uso Consultar Jugador.

Nombre del CU	Consultar Jugador
----------------------	--------------------------

ANEXOS

Actores	Estadístico
Propósito	Obtener los datos de un jugador por la zona, evento, equipo, categoría, sexo o CI.
Resumen	El caso de uso se inicia cuando el estadístico decide consultar los datos de un jugador por el evento, equipo, zona, sexo o CI. Una vez obtenido los resultados puede modificar los datos o eliminar un jugador, terminando así el caso de uso.
Referencias	RF 8
Precondiciones	Se debe haber insertado los datos de los jugadores.
Prioridad	Secundario
Flujo Normal de Evento	
Acción del Actor	Respuesta del Sistema
1. El caso de uso se inicia cuando el estadístico accede a la opción Consultar jugador.	2. Brinda la posibilidad de introducir: <ul style="list-style-type: none"> • CI Y seleccionar los elementos de búsqueda: <ul style="list-style-type: none"> • Evento • Equipo • Categoría • Zona • Sexo Y permite:

ANEXOS

	<ul style="list-style-type: none"> • Realizar una búsqueda a partir de los datos. • Cerrar
<p>3. Introduce y selecciona los datos necesarios para realizar una búsqueda y presiona el botón Buscar.</p>	<p>4. Valida los datos.</p> <p>5. Muestra una lista de posibles coincidencias con:</p> <ul style="list-style-type: none"> • Nombre • Apellidos • CI • Peso • Altura • Evento • Equipo • Categoría • Zona • Sexo <p>Y permite:</p> <ul style="list-style-type: none"> • Realizar una nueva búsqueda. • Ver un jugador de la lista mostrada. • Modificar los datos de un jugador de la lista. • Eliminar un jugador.
<p>6. Consulta los datos de los jugadores de la lista de coincidencia.</p>	
<p>7. El caso de uso termina.</p>	
<p>Poscondiciones</p>	<p>Se consultó los datos del Jugador coincidentes con uno o varios criterios de búsqueda.</p>

ANEXOS

Flujos Alternos	
3.a El estadístico selecciona la opción de Cerrar	
	3. a.1 Regresa a la vista anterior.
	3. a.2 El caso de uso termina.
4.a Existe algún dato incorrecto	
	4.a.1 Envía mensaje (“Verifique los datos”) 4. a.2 Regresa al paso 3 del Flujo Normal de Eventos.
5. a El estadístico no seleccionó ningún dato de búsqueda.	
	5. a.1 Muestra el mensaje de información “Debe especificar un criterio de búsqueda”.
	5. a.2 Regresa al paso 2 del Flujo Básico.
5. b El sistema no encuentra ninguna coincidencia.	
	5. b.1 Muestra el mensaje “No se encontraron coincidencias” y permite realizar una nueva búsqueda.
	5. b.2 Regresa al paso 2 del Flujo Básico.
5. c El estadístico selecciona la opción de realizar una nueva búsqueda.	
	5. c.1 Regresa al paso 2 del Flujo Básico.
5. d El estadístico selecciona la opción de ver los datos de un jugador de la lista mostrada.	
	5. d.1 Muestra los datos del Jugador. Ver Sección “Ver jugador” del caso de uso Gestionar Jugador.
	5. d.2 El caso de uso termina.
5. e El estadístico selecciona la opción de modificar los datos de un jugador de la lista mostrada.	
	5. e.1 Muestra los datos del jugador

ANEXOS

	permitiendo cambiar sus valores. Ver Sección “Modificar Jugador” del caso de uso Gestionar Jugador.
	5. e.2 El caso de uso termina.
5.f El estadístico selecciona la opción de eliminar un jugador de la lista mostrada	
5. f.1 Selecciona el jugador y presiona en botón Eliminar.	5. f.2 Elimina al jugador de la lista y envía un mensaje (“Eliminado correctamente”). 5. f.3 El caso de uso termina.

Tabla 15: Descripción del Caso de Uso Consultar Jugador.

Anexo 5: Descripción del Caso de Uso Consultar Juego.

Nombre del CU	Consultar Juego
Actores	Estadístico
Propósito	Obtener los datos de un juego por zona, evento, fecha, categoría, sexo, equipo ganador, equipo perdedor.
Resumen	El caso de uso se inicia cuando el estadístico necesita consultar los datos de un juego, permitiéndole el sistema obtenerlo por el evento, la fecha o el equipo. Una vez obtenido los resultados puede modificar los datos o eliminar un juego determinado, terminando así el caso de uso.
Referencias	RF 8
Precondiciones	Se encuentran registrados los datos de un juego.
Prioridad	Secundario
Flujo Normal de Eventos	
Acción del Actor	Respuesta del sistema
1. El caso de uso se inicia cuando el estadístico accede a la opción Consultar juego.	2. Brinda la posibilidad de seleccionar: <ul style="list-style-type: none"> • Zona • Evento • Equipo ganador

ANEXOS

	<ul style="list-style-type: none"> • Equipo perdedor • Fecha • Sexo • Categoría <p>Y permite:</p> <ul style="list-style-type: none"> • Realizar una búsqueda a partir de los datos mostrados. • Cerrar
3. Selecciona los datos que considere para realizar la búsqueda y presiona el botón buscar coincidencias.	4. Muestra una lista de posibles coincidencias con: <ul style="list-style-type: none"> • Numero del juego. • Equipo ganador. • Sets ganados del equipo ganador. • Equipo perdedor. • Sets ganados del equipo perdedor. <p>Y permite:</p> <ul style="list-style-type: none"> • Realizar una nueva búsqueda. • Ver un juego de la lista mostrada. • Modificar los datos de un juego de la lista. • Eliminar un juego.
5. Consulta los datos de los juego de la lista de coincidencia.	
6. El caso de uso termina.	
Poscondiciones	Se consultó los datos de un juego coincidentes con uno o varios criterios de búsqueda.
Flujos Alternos	
3. a El estadístico selecciona la opción de Cerrar.	
	3. a.1 Regresa a la vista anterior. 3. a.2 El caso de uso termina.

ANEXOS

4. a El estadístico no seleccionó ningún dato de búsqueda.	
	4. a.1 Muestra el mensaje de información “Debe especificar un criterio de búsqueda”.
	4. a.2 Regresa al paso 2 del Flujo Básico.
4. b El sistema no encuentra ninguna coincidencia.	
	4. b.1 Muestra el mensaje “No se encontraron coincidencias” y permite realizar una nueva búsqueda.
	4. b.2 Regresa al paso 2 del Flujo Básico.
4. c El estadístico selecciona la opción de realizar una nueva búsqueda.	
	4. c.1 Regresa al paso 2 del Flujo Básico.
4. d El estadístico selecciona la opción de ver los datos de un juego de la lista mostrada.	
	4. d.1 Muestra los datos del juego. Ver Sección Ver juego del caso de uso Gestionar Juego.
	4. d.2 El caso de uso termina.
4.e El estadístico selecciona la opción de modificar los datos de un juego de la lista mostrada.	
	4. e.1 Brinda la posibilidad de modificar los datos de un juego. Ver Sección “Modificar Juego” del caso de uso Gestionar Juego.
4.f El estadístico selecciona la opción de eliminar un jugo de la lista mostrada	
	4. f.2 Brinda la posibilidad de eliminar un juego. Ver Sección “Eliminar juego” del caso de uso Gestionar juego.
	4. f.3 El caso de uso termina.

Tabla 16: Descripción del Caso de Uso Consultar Juego

Anexo 6: Descripción del Caso de Uso Consultar Usuario.

Nombre del CU	Consultar usuario
Actores	Administrador

ANEXOS

Propósito	Obtener los datos de un usuario por nombre, usuario o categoría.
Resumen	El caso de uso se inicia cuando el administrador necesita consultar los datos de un usuario, permitiéndole el sistema obtenerlo por el nombre, apellidos, la categoría. Una vez obtenido los resultados puede modificar los datos o eliminar un usuario determinado, terminando así el caso de uso.
Referencias	RF 9
Precondiciones	Deben estar registrados los datos de los usuarios.
Prioridad	Secundario
Flujo Normal de Eventos	
Acción del Actor	Respuesta del sistema
1. El caso de uso se inicia cuando el estadístico accede a la opción Consultar usuario.	2. Brinda la posibilidad de seleccionar: <ul style="list-style-type: none"> • Categoría Permite introducir: <ul style="list-style-type: none"> • Nombre (s) • Apellidos • Usuario Y permite: <ul style="list-style-type: none"> • Realizar una búsqueda a partir de los datos mostrados. • Cerrar
3. Selecciona los datos que considere para realizar la búsqueda y presiona el botón buscar coincidencias.	4. Muestra una lista de posibles coincidencias con: <ul style="list-style-type: none"> • Nombre (s) y Apellidos • Usuario • Categoría Y permite: <ul style="list-style-type: none"> • Realizar una nueva búsqueda.

ANEXOS

	<ul style="list-style-type: none"> • Ver un usuario de la lista mostrada. • Modificar los datos de un usuario de la lista. • Eliminar un usuario
5. Consulta los datos del usuario de la lista de coincidencia.	
6. El caso de uso termina.	
Poscondiciones	Se consultó los datos de un usuario coincidentes con uno o varios criterios de búsqueda.
Flujos Alternos	
3. a El administrador selecciona la opción de Cerrar.	
	3. a.1 Regresa a la vista anterior. 3. a.2 El caso de uso termina.
4. a El administrador no seleccionó ningún dato de búsqueda.	
	4. a.1 Muestra el mensaje de información "Debe especificar un criterio de búsqueda".
	4. a.2 Regresa al paso 2 del Flujo Básico.
4. b El sistema no encuentra ninguna coincidencia.	
	4. b.1 Muestra el mensaje "No se encontraron coincidencias" y permite realizar una nueva búsqueda.
	4. b.2 Regresa al paso 2 del Flujo Básico.
4. c El administrador selecciona la opción de realizar una nueva búsqueda.	
	4. c.1 Regresa al paso 2 del Flujo Básico.
4. d El administrador selecciona la opción de ver los datos de un usuario de la lista mostrada.	
	4. d.1 Muestra los datos del juego. Ver Sección "Ver juego" del caso de uso Gestionar Juego.

ANEXOS

	4. d.2 El caso de uso termina.
4.e El administrador selecciona la opción de modificar los datos de un juego de la lista mostrada.	
	4. e.1 Brinda la posibilidad de modificar los datos de un juego. Ver Sección “Modificar Usuario” del caso de uso Gestionar Usuario.
4.f El administrador selecciona la opción de eliminar un usuario de la lista mostrada	
	4. f.2 Brinda la posibilidad de eliminar un juego. Ver Sección “Eliminar Usuario” del caso de uso Gestionar Usuario. 4. f.3 El caso de uso termina.

Anexo 7: Descripción del Caso de Uso Consultar Reporte Resumen Estadístico de cada Equipo.

Nombre del CU	Consultar Reporte Resumen Estadístico de cada Equipo
Actores	Estadístico
Propósito	Consultar por parte del estadístico el reporte resumen estadístico de cada equipo.
Resumen	El caso de uso se inicia cuando el estadístico necesita consultar el reporte resumen estadístico de cada equipo. El sistema muestra los datos del reporte y termina así el caso de uso.
Referencias	Requerimiento funcional número 10.
Precondiciones	Se debe haber registrado todos los datos de los juegos.
Prioridad	Crítico
Flujo Normal de Eventos	
Acciones del Actor	Respuesta del sistema
1. Selecciona la opción Consultar reporte resumen estadístico de cada equipo.	2. Muestra interfaz y permite seleccionar: <ul style="list-style-type: none"> • Evento

ANEXOS

	<ul style="list-style-type: none"> • Sexo <p>Y permite:</p> <ul style="list-style-type: none"> • Ver reporte • Cerrar
3. Selecciona el evento, sexo y presiona el botón Ver reporte.	4. Muestra el reporte.
5. Consulta el reporte. 6. El caso de uso termina.	
Poscondiciones	Se consultó un reporte estadístico de cada equipo.
Flujo Alterno	
2. a Selecciona la opción de Cerrar.	
	2. a.1 Regresa a la vista anterior. 2. a.2 El caso de uso termina.

Tabla 17: Descripción del Caso de Uso Consultar Reporte Resumen Estadístico de cada Equipo

Anexo 8: Descripción del Caso de Uso Consultar Reporte Resultados de los Objetivos Pedagógicos.

Nombre del CU	Consultar Reporte Resultados de los Objetivos Pedagógicos
Actores	Estadístico
Propósito	Consultar por parte del estadístico el reporte resultados de los objetivos pedagógicos.
Resumen	El caso de uso se inicia cuando el estadístico necesita consultar el reporte resultados de los objetivos pedagógicos. El sistema muestra los datos del reporte y termina así el caso de uso.
Referencias	RF 11

ANEXOS

Precondiciones	Se debe haber registrado todos los datos de los juegos.
Prioridad	Crítico
Flujo Normal de Eventos	
Acciones del Actor	Respuesta del sistema
1. Selecciona la opción Consultar reporte resultados de los objetivos pedagógicos.	2. Brinda la posibilidad de seleccionar: <ul style="list-style-type: none"> • Evento Y permite: <ul style="list-style-type: none"> • Ver reporte. • Cerrar
3. Selecciona el evento y presiona el botón Ver reporte.	4. Muestra el reporte.
5. Consulta el reporte. 6. El caso de uso termina.	
Flujo Alternativo	
2. a El estadístico selecciona la opción de Cerrar.	
	2. a.1 Regresa a la vista anterior. 2. a.2 El caso de uso termina.
Poscondiciones	Se consultó un reporte resultados de los objetivos pedagógicos.

Tabla 18: Descripción del Caso de Uso Consultar Reporte Resultados de los Objetivos Pedagógicos

Anexo 9: Descripción del Caso de Uso Consultar Reporte Resultados por Provincia.

Nombre del CU	Consultar Reporte Resultados por Provincia
Actores	Estadístico
Propósito	Consultar los resultados por cada una de las

ANEXOS

	provincias.
Resumen	El caso de uso se inicia cuando el estadístico necesita consultar el reporte resultados por provincia. El sistema muestra los datos del reporte y termina así el caso de uso.
Referencias	RF 12
Precondiciones	Se debe haber registrado todos los datos de los juegos.
Prioridad	Critico
Flujo Normal de Eventos	
Acciones del Actor	Respuesta del sistema
1. Selecciona la opción Consultar reporte resultados por provincia.	2. Brinda la posibilidad de seleccionar: <ul style="list-style-type: none"> • Evento Y permite: <ul style="list-style-type: none"> • Ver reporte. • Cerrar
3. Selecciona el evento y la opción Ver reporte.	4. Muestra el reporte.
5. Consulta el reporte. 6. El caso de uso termina.	6. Regresa a la vista anterior. 7. El caso de uso termina
Flujo Alterno	
2. a El estadístico selecciona la opción de Cerrar.	
	2. a.1 Regresa a la vista anterior. 2. a.2 El caso de uso termina.
Poscondiciones	Se consultó un reporte resultados por las provincias.

Tabla 19: Descripción del Caso de Uso Consultar Reporte Resultados por Provincia

Anexo 10: Descripción del Caso de Uso Consultar Reporte Resultados Alcanzados por las Provincias.

ANEXOS

Nombre del CU	Consultar reporte resultados alcanzados por las provincias
Actores	Estadístico
Propósito	Consultar los resultados alcanzados por cada una de las provincias.
Resumen	El caso de uso se inicia cuando el estadístico necesita consultar el reporte resultados alcanzados por cada una de las provincias. El sistema los datos del reporte. El caso de uso termina.
Referencias	RF 13
Precondiciones	Deben estar registrados todos los datos de los juegos.
Prioridad	Crítico
Flujo Normal de Eventos	
Acciones del Actor	Respuesta del sistema
1. Selecciona la opción Consultar reporte resultados alcanzados por las provincias.	2. Brinda la posibilidad de seleccionar: <ul style="list-style-type: none"> • Evento Y permite: <ul style="list-style-type: none"> • Ver reporte. • Cerrar
3. Selecciona el evento y presiona el botón Ver reporte.	4. Muestra el reporte.
5. Consulta el reporte. 6. El caso de uso termina.	
Flujo Alternativo	
2. a El estadístico selecciona la opción de Cerrar.	

ANEXOS

	2. a.1 Regresa a la vista anterior.
	2. a.2 El caso de uso termina.
Poscondiciones	Se consultó un reporte con los resultados alcanzados por las provincias.

Tabla 20: Descripción del Caso de Uso Consultar Reporte Resultados Alcanzados por las Provincias

Anexo 11: Descripción del Caso de Uso Consultar Reporte Equipo Ideal.

Nombre del CU	Consultar Reporte Equipo Ideal
Actores	Estadístico
Propósito	Consultar cual es el equipo que mejor promedio de talla tiene.
Resumen	El caso de uso se inicia cuando el estadístico necesita consultar el reporte equipo ideal. El sistema muestra los datos del reporte y termina así el caso de uso.
Referencias	RF 14
Precondiciones	Deben estar registrados todos los jugadores de cada equipo.
Prioridad	Crítico
Flujo Normal de Eventos	
Acciones del Actor	Respuesta del sistema
1. Selecciona la opción Consultar reporte equipo ideal.	2. Brinda la posibilidad de seleccionar: <ul style="list-style-type: none"> • Evento Y permite: <ul style="list-style-type: none"> • Ver reporte. • Cerrar
3. Selecciona el evento y presiona el botón	4. Muestra el reporte.

ANEXOS

Ver reporte.	
5. Consulta el reporte. 6. El caso de uso termina.	
2. a El estadístico selecciona la opción de Cerrar.	
	2. a.1 Regresa a la vista anterior.
	2. a.2 El caso de uso termina.
Poscondiciones	Se consultó un reporte con los resultados de los equipos por talla.

Tabla 21: Descripción del Caso de Uso Consultar Reporte Equipo Ideal

Anexo12: Descripción del Caso de Uso Consultar Reporte Control Competitivo.

Nombre del CU	Consultar Reporte Control Competitivo
Actores	Estadístico
Propósito	Consultar el control competitivo de del evento.
Resumen	El caso de uso se inicia cuando el estadístico necesita consultar el reporte control competitivo. El sistema muestra los datos del reporte y termina así el caso de uso.
Referencias	RF 14
Precondiciones	Deben estar registrados los datos de cada uno de los juegos.
Prioridad	Crítico
Flujo Normal de Eventos	
Acciones del Actor	Respuesta del sistema
1. Selecciona la opción Consultar reporte control competitivo.	2. Brinda la posibilidad de seleccionar: <ul style="list-style-type: none"> • Evento Y permite: <ul style="list-style-type: none"> • Ver reporte.

ANEXOS

	<ul style="list-style-type: none"> • Cerrar
3. Selecciona el evento y presiona el botón Ver reporte.	4. Muestra el reporte.
5. Consulta el reporte.	
6. El caso de uso termina.	
2. a El estadístico selecciona la opción de Cerrar.	
	2. a.1 Regresa a la vista anterior.
	2. a.2 El caso de uso termina.
Poscondiciones	Se consultó un reporte con el control competitivo del evento.

Tabla 22: Descripción del Caso de Uso Consultar Reporte Control Competitivo

Anexo 13: Descripción del Caso de Uso Consultar Reporte Historial de Usuario.

Nombre del CU	Consultar Reporte Historial de Usuario
Actores	Estadístico
Propósito	Consultar el historial de cada usuario.
Resumen	El caso de uso se inicia cuando el estadístico necesita consultar el reporte historial de usuario. El sistema muestra los datos del reporte y termina así el caso de uso.
Referencias	RF 16
Precondiciones	El estadístico debe estar autenticado.
Prioridad	Crítico
Flujo Normal de Eventos	
Acciones del Actor	Respuesta del sistema
1. Selecciona la opción Consultar reporte historial de usuario.	2. Brinda la posibilidad de seleccionar: <ul style="list-style-type: none"> • Usuario • Fecha

ANEXOS

	Y permite <ul style="list-style-type: none"> • Ver reporte
3. Selecciona el usuario y la fecha y presiona el botón Ver reporte.	4. Muestra el reporte
5. Consulta el reporte. 6. El caso de uso termina.	
Poscondiciones	Se consultó un reporte con el historial de cada usuario.

Tabla 23: Descripción del Caso de Uso Consultar Reporte Historial de Usuario

Anexo14: Descripción del Caso de Uso Autenticar Usuario.

Nombre del CU	Autenticar Usuario
Actores	Usuario
Propósito	Verificar que el usuario pertenece al sistema.
Resumen	El caso de uso se inicia cuando el usuario decide acceder a la aplicación, por lo que debe autenticarse para garantizar la seguridad del sistema, inserta el usuario y la contraseña para acceder al sistema, se verifican los datos y finaliza cuando el sistema le otorga los privilegios según el rol que desempeña.
Referencias	RF7
Precondiciones	El actor debe ser un usuario registrado en el sistema.
Prioridad	Crítico.

ANEXOS

Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El usuario accede a la aplicación.	2. Brinda la posibilidad de introducir los datos usuario y contraseña. Y permite: <ul style="list-style-type: none"> • Guardar los datos.
3. El usuario introduce usuario y contraseña y selecciona la opción de guardar los datos.	4. Valida los datos. 5. Brinda privilegios de acceso. 6. El caso de uso termina.
Flujo Alternativo	
4.a Si quedó algún campo vacío	
	4.a.1 El sistema envía mensaje (“Debe llenar todos los campos”) 4. a.2 Regresa al paso 3 del Flujo Normal de eventos.
4.b Si existe algún dato incorrecto	
	4.b1 El sistema envía mensaje (“Verifique los datos”) 4. b.2 Regresa al paso 3 del Flujo Normal de eventos.
Poscondiciones	Se autenticó un usuario en el sistema.

Tabla 24: Descripción del Caso de Uso Autenticar Usuario

Anexo 15: Descripción del Caso de Uso Gestionar Usuario.

Nombre CU	Gestionar Usuario
Actor	Administrador
Propósito	Insertar, modificar, eliminar o ver un usuario.
Resumen	El caso de uso se inicia cuando el administrador decide insertar un nuevo usuario, modificar algún dato de un usuario ya existente o eliminar un usuario del sistema. Realiza la operación deseada y termina el caso de uso.
Referencia	RF8
Precondiciones	Que exista al menos un usuario en el sistema.
Prioridad	Crítico
Sección “Insertar Usuario”: Flujo Normal de Eventos	
Acciones del Actor	Respuesta del Sistema
1. Selecciona la opción Insertar Usuario.	2. Brinda la posibilidad de introducir los siguientes datos: <ul style="list-style-type: none"> • Nombre (s) y Apellidos • Usuario • Clave • Confirmar Clave Brinda la posibilidad de seleccionar: <ul style="list-style-type: none"> • Categoría Y permite: <ul style="list-style-type: none"> • Agregar el usuario • Cerrar
3. Introduce los datos del usuario y selecciona la opción de Agregar.	4. Valida los datos. 5. Verifica si el usuario esta registrado en el

ANEXOS

	<p>sistema.</p> <p>6. Registra el usuario.</p> <p>7. Muestra mensaje (“Insertado correctamente”)</p> <p>8 El caso de uso termina.</p>
Flujos Alternos	
4.a Si se quedó algún campo vacío	
	<p>4. a.1 El sistema envía un mensaje (“Debe llenar todos los campos”)</p> <p>4. a.2Regresa al paso 3 del Flujo Normal de Evento.</p>
4.b Si existe algún dato incorrecto	
	<p>4.b.2 El sistema envía mensaje (“ Verifique que los datos estén correctos”)</p> <p>4. b.2Regresa al paso 3 del Flujo Normal de Evento</p>
5. a El usuario está registrado en el sistema.	
	<p>5.a.1 Envía un mensaje (“ El usuario que desea insertar ya está registrado”)</p>
Sección “Modificar Usuario”: Flujo Normal de Eventos	
Acciones del actor	Respuesta del sistema
1. Selecciona opción Modificar Usuario.	2. Brinda la posibilidad de seleccionar: <ul style="list-style-type: none"> • Usuario.
3. Selecciona el usuario.	4. Muestra los datos del usuario a modificar: <ul style="list-style-type: none"> • Nombre (s) y Apellidos • Clave <p>Brinda la posibilidad de seleccionar:</p> <ul style="list-style-type: none"> • Categoría <p>Y permite:</p> <ul style="list-style-type: none"> • Modificar los datos • Cerrar
5. Modifica los datos del usuario y presiona	6. Valida los datos.

ANEXOS

el botón Modificar.	7. Verifica si existe un solo administrador. 8. Actualiza los datos del usuario. 9. El caso de uso termina.
Flujos Alternos	
6.a Si existe algún dato incorrecto	
	6.a.1 Muestra mensaje (“Verifique los datos”) 6. a.2 Regresa al paso 5 del Flujo Normal de Evento.
6.a.1 Si existe algún campo vacío	
	6.a.1.1 Muestra mensaje(“ Debe llenar todos los campos”) 6. a.1.2 Regresa al paso 5 del Flujo Normal de Evento.
7. a Si existe un solo administrador y desea modificar su categoría.	
	7.a.1Muestra un mensaje (“Solo existe un administrador, no puede cambiar su categoría ”)
Sección “Eliminar Usuario”: Flujo Normal de Evento	
1. Selecciona opción Eliminar Usuario.	2. Brinda la posibilidad de seleccionar el usuario a eliminar. Y permite: <ul style="list-style-type: none"> • Eliminar • Cerrar
3. Selecciona el usuario y presiona el botón Eliminar.	4. Envía un mensaje de confirmación (“Está seguro que desea eliminar el usuario”) y permite seleccionar (Si, No).
5. Selecciona la opción Si.	6. Elimina el usuario. 7. El caso de uso termina.
Flujos Alternos	
2. a. Selecciona la opción de Cerrar.	
	2. a.1 Regresa a la vista anterior.
	2. a.2 El caso de uso termina.

ANEXOS

4.a. Selecciona la opción No.	
	4. a.1 Regresa a la vista anterior. 4. a.2 El caso de uso termina.
Sección “Ver Usuario”: Flujo Normal de Evento	
1. El administrador selecciona la opción Ver usuario.	2. Muestra los datos del usuario. Y permite: <ul style="list-style-type: none"> • Modificar los datos del usuario • Eliminar un usuario • Cerrar
3. Selecciona la opción Cerrar.	4. Regresa a la vista anterior. 5. El caso de uso termina.
Flujos Alternos	
3.a Selecciona la opción Modificar los datos de un usuario	
	3. a.1 Muestra los datos del usuario. Ver Sección “Modificar Usuario” 3. a.2 El caso de uso termina.
3.b Selecciona la opción Eliminar un usuario	
	3.b.1 Ver Sección “Eliminar Usuario” 3. b.2 El caso de uso termina.
Poscondiciones	Se insertó, modificó, eliminó o vio un usuario.

Tabla 25: Descripción del Caso de Uso Gestionar Usuario

Anexo16: Caso de Prueba del Caso de Uso Gestionar Juego.

Entrada	Resultados	Condiciones
El estadístico selecciona la opción Insertar Juego.	El sistema muestra interfaz con los datos requeridos para insertar un nuevo juego.	
El estadístico introduce los datos del juego dejando algún campo vacío y presiona el botón Insertar.	El sistema envía un mensaje informando que debe llenar todos los campos. Ejemplo: “El campo sets ganados del equipo ganador	No se registren valores vacíos en los campos de entrada para insertar un nuevo juego.

ANEXOS

	está vacío o tiene caracteres incorrectos”.	
<p>El estadístico selecciona: Fecha:”4/05/10” Evento:”Antonio Maceo” Equipo ganador:”Santiago de Cuba” Introduce los siguientes datos: Total de saques: “28” Total k1:”19” Total k2: “15” Cantidad Positivos de saques:”15” Cantidad Positivos k1:”8” Cantidad Positivos k2:”7” Sets Ganados:”3”</p> <p>El estadístico selecciona : Equipo perdedor:”Las Tunas” Introduce los siguientes datos: Total de saques: “19” Total k1:”15” Total k2: “13” Cantidad Positivos de saques:”10” Cantidad Positivos k1:”8” Cantidad Positivos k2:”5” Sets Ganados:”0” Y presiona el botón Insertar</p>	<p>El sistema inserta el nuevo juego, confirmando la operación. Ejemplo: “Insertado correctamente”.</p>	<p>No exista el juego en la aplicación.</p>

ANEXOS

<p>El estadístico selecciona: Fecha:"4/05/10" Evento:"Antonio Maceo" Equipo ganador:"Santiago de Cuba" Introduce los siguientes datos: Total de saques: "28" Total k1:"19" Total k2: "15" Cantidad Positivos de saques:"15" Cantidad Positivos k1:"8" Cantidad Positivos k2:"19" Sets Ganados:"3"</p> <p>El estadístico selecciona : Equipo perdedor:"Las Tunas" Introduce los siguientes datos: Total de saques: "19" Total k1:"15" Total k2: "13" Cantidad Positivos de saques:"21" Cantidad Positivos k1:"17" Cantidad Positivos k2:"5" Sets Ganados:"0" Y presiona el botón Insertar</p>	<p>El sistema notifica al usuario que existe error en los totales efectivos. Ejemplo:"La cantidad total de K2 es menor que los efectivos, por favor, rectifique."</p>	<p>Los totales efectivos sean mayores que los totales.</p>
<p>El estadístico selecciona: Fecha:"4/05/10"</p>	<p>El sistema notifica al usuario que existe un error en los</p>	<p>El equipo perdedor tenga 3 o más sets ganados.</p>

ANEXOS

<p>Evento: "Antonio Maceo" Equipo ganador: "Santiago de Cuba" Introduce los siguientes datos: Total de saques: "28" Total k1: "19" Total k2: "15" Cantidad Positivos de saques: "15" Cantidad Positivos k1: "8" Cantidad Positivos k2: "7" Sets Ganados: "3"</p> <p>El estadístico selecciona : Equipo perdedor: "Las Tunas" Introduce los siguientes datos: Total de saques: "19" Total k1: "15" Total k2: "13" Cantidad Positivos de saques: "10" Cantidad Positivos k1: "8" Cantidad Positivos k2: "5" Sets Ganados: "3" Y presiona el botón Insertar</p>	<p>sets ganados del equipo perdedor. Ejemplo: "La cantidad de sets introducidos del equipo perdedor es incorrecto"</p>	
<p>El estadístico selecciona: Fecha: "4/05/10" Evento: "Antonio Maceo" Equipo ganador: "Santiago</p>	<p>El sistema notifica al usuario que existe un error en los sets ganados del equipo ganador. Ejemplo: "La</p>	<p>El equipo ganador no tenga 3 sets ganados.</p>

ANEXOS

<p>de Cuba” Introduce los siguientes datos: Total de saques: “28” Total k1:”19” Total k2: “15” Cantidad Positivos de saques:”15” Cantidad Positivos k1:”8” Cantidad Positivos k2:”7” Sets Ganados:”2” El estadístico selecciona : Equipo perdedor:”Las Tunas” Introduce los siguientes datos: Total de saques: “19” Total k1:”15” Total k2: “13” Cantidad Positivos de saques:”10” Cantidad Positivos k1:”8” Cantidad Positivos k2:”5” Sets Ganados:”1” Y presiona el botón Insertar</p>	<p>cantidad de sets introducidos del equipo ganador es incorrecto”</p>	
<p>El estadístico selecciona: Fecha:”4/05/09” Evento:”Antonio Maceo” Equipo ganador:”Santiago de Cuba” Introduce los siguientes</p>	<p>El sistema notifica al usuario que existe un error en la fecha del juego. Ejemplo: “La fecha introducida del juego es anterior a la fecha del evento, por favor, compruebe”</p>	<p>La fecha del juego sea posterior a la del evento.</p>

ANEXOS

<p>datos: Total de saques: "28" Total k1:"19" Total k2: "15" Cantidad Positivos de saques:"15" Cantidad Positivos k1:"8" Cantidad Positivos k2:"7" Sets Ganados:"3"</p> <p>El estadístico selecciona : Equipo perdedor:"Las Tunas" Introduce los siguientes datos: Total de saques: "19" Total k1:"15" Total k2: "13" Cantidad Positivos de saques:"10" Cantidad Positivos k1:"8" Cantidad Positivos k2:"5" Sets Ganados:"1" Y presiona el botón Insertar</p>		
<p>El estadístico selecciona la opción de Modificar Juego.</p>	<p>El sistema muestra interfaz para seleccionar el evento y el sexo.</p>	
<p>El estadístico oprime el botón Modificar</p>	<p>El sistema muestra mensaje "Seleccione el juego a ser modificado"</p>	<p>No esté seleccionado el juego.</p>
<p>El estadístico selecciona Evento:"Antonio Maceo"</p>	<p>El sistema muestra listado de juegos.</p>	<p>Debe estar seleccionado el evento y el sexo.</p>

ANEXOS

Sexo:"F"		
El estadístico selecciona Evento:"Antonio Maceo" Sexo:"F" Uno de los juegos	El sistema muestra los datos del juego a modificar.	Debe estar seleccionado el evento, el sexo y uno de los juegos.
El estadístico selecciona la fecha: Fecha:"7/05/10" Modifica los datos del equipo ganador: Total de saques: "30" Total k1:"20" Total k2: "15" Cantidad Positivos de saques:"15" Cantidad Positivos k1:"8" Cantidad Positivos k2:"7" Sets Ganados:"3" Y modifica los datos del equipo perdedor: Total de saques: "23" Total k1:"15" Total k2: "13" Cantidad Positivos de saques:"10" Cantidad Positivos k1:"8" Cantidad Positivos k2:"5" Sets Ganados:"1" Presiona el botón Modificar	El sistema envía un mensaje que los datos fueron modificados correctamente. Ejemplo:" Los datos fueron modificados correctamente"	Los datos modificados estén correctos.
El estadístico modifica los datos del juego dejando algún campo vacío. Y	El sistema envía un mensaje informando que debe llenar todos los campos.	No se registren valores vacíos en los campos de entrada para modificar un

ANEXOS

<p>presiona el botón Modificar</p>	<p>Ejemplo: “El campo sets ganados del equipo ganador está vacío ”</p>	<p>juego.</p>
<p>El estadístico selecciona la fecha: Fecha:”7/05/10” Modifica los datos del equipo ganador: Total de saques: “28” Total k1:”19” Total k2: “15” Cantidad Positivos de saques:”15” Cantidad Positivos k1:”8” Cantidad Positivos k2:”7” Sets Ganados:”3”</p> <p>Y modifica los datos del equipo perdedor: Total de saques: “19” Total k1:”15” Total k2: “13” Cantidad Positivos de saques:”10” Cantidad Positivos k1:”8” Cantidad Positivos k2:”5” Sets Ganados:”3” Presiona el botón Modificar</p>	<p>El sistema notifica al usuario que existe un error en los sets ganados del equipo perdedor. Ejemplo: “La cantidad de sets introducidos del equipo perdedor es incorrecto”</p>	<p>El equipo perdedor tenga 3 o más sets ganados.</p>
<p>El estadístico selecciona la fecha: Fecha:”12/05/10” Modifica los datos del equipo</p>	<p>El sistema notifica al usuario que existe un error en los sets ganados del equipo ganador. Ejemplo: “La</p>	<p>El equipo ganador no tenga 3 sets ganados.</p>

ANEXOS

<p>ganador: Total de saques: "28" Total k1:"19" Total k2: "15" Cantidad Positivos de saques:"15" Cantidad Positivos k1:"8" Cantidad Positivos k2:"7" Sets Ganados:"2" Y modifica los datos del equipo perdedor: Total de saques: "19" Total k1:"15" Total k2: "13" Cantidad Positivos de saques:"10" Cantidad Positivos k1:"8" Cantidad Positivos k2:"5" Sets Ganados:"1" Presiona el botón Modificar</p>	<p>cantidad de sets introducidos del equipo ganador es incorrecto"</p>	
<p>El estadístico selecciona: Fecha:"4/05/09" Modifica los datos del equipo ganador: Total de saques: "28" Total k1:"19" Total k2: "15" Cantidad Positivos de saques:"15" Cantidad Positivos k1:"8" Cantidad Positivos k2:"7" Sets Ganados:"3"</p>	<p>El sistema notifica al usuario que existe un error en la fecha del juego. Ejemplo: "La fecha introducida del juego es anterior a la fecha del evento, por favor, compruebe"</p>	<p>La fecha del juego sea posterior a la del evento.</p>

ANEXOS

<p>Y modifica los datos del equipo perdedor: Total de saques: "19" Total k1:"15" Total k2: "13" Cantidad Positivos de saques:"10" Cantidad Positivos k1:"8" Cantidad Positivos k2:"5" Sets Ganados:"1" Y presiona el botón Modificar</p>		
<p>El estadístico selecciona: Fecha:"4/05/10" Modifica los datos del equipo ganador: Total de saques: "28" Total k1:"19" Total k2: "15" Cantidad Positivos de saques:"15" Cantidad Positivos k1:"8" Cantidad Positivos k2:"19" Sets Ganados:"3" Y modifica los datos del equipo perdedor: Total de saques: "19" Total k1:"15" Total k2: "13" Cantidad Positivos de saques:"21"</p>	<p>El sistema notifica al usuario que existe error en los totales efectivos. Ejemplo:"La cantidad total de K2 es menor que los efectivos, por favor, rectifique."</p>	<p>Los totales efectivos sean mayores que los totales.</p>

ANEXOS

Cantidad Positivos k1:"17" Cantidad Positivos k2:"5" Sets Ganados:"0" Y presiona el botón Modificar		
El estadístico selecciona la opción Eliminar juego.	El sistema permite seleccionar el evento y el sexo.	
El estadístico selecciona: Evento:"Antonio Maceo" Sexo:"M"	El sistema muestra listado de juegos.	Sea seleccionado el evento y el sexo.
El estadístico selecciona: Evento:"Antonio Maceo" Sexo:"M" Y presiona el botón Eliminar.	El sistema muestra mensaje:"Seleccione el juego a eliminar"	No sea seleccionado el juego a eliminar.
El estadístico selecciona el juego	El sistema envía una pregunta para garantizar que no se realice una operación errada. ¿El juego seleccionado se eliminará. Desea continuar? (Si, No)	Sea presionado el botón Eliminar.
El estadístico presiona el botón Si.	El sistema notifica al usuario que el juego fue eliminado correctamente. Ejemplo:"Eliminado correctamente."	Sea presionado el botón Si.
El estadístico presiona el botón No.	El sistema muestra la interfaz anterior.	Sea presionado el botón No.

Tabla 26: Caso de Prueba del Caso de Uso Gestionar Juego

Anexo 17: Caso de Prueba del Caso de Uso Gestionar Evento

Entrada	Resultados	Condiciones
El estadístico selecciona la	El sistema muestra una	

ANEXOS

opción Insertar Evento.	interfaz con los datos requeridos para insertar un nuevo evento.	
El estadístico introduce los datos del evento dejando algún campo vacío.	El sistema envía un mensaje informando que debe llenar todos los campos. Ejemplo: "El campo nombre está vacío o tiene caracteres incorrectos".	No se registren valores vacíos en los campos de entrada para insertar un nuevo evento.
El estadístico introduce el dato siguiente: Nombre: "Antonio Maceo." Y selecciona: Sede: "Santiago de Cuba" Fecha: "4/03/10" Categoría: "Estudiantil" Zona: "Oriental." Equipos: "Guantánamo", "Las Tunas", "Granma"	El sistema inserta el nuevo evento, confirmando la operación. Ejemplo: "Insertado correctamente".	No exista el evento en la aplicación.
El estadístico introduce el dato siguiente: Nombre: "Antonio Mac4eo." Y selecciona: Sede: "Santiago de Cuba" Fecha: "4/03/10" Categoría: "Estudiantil" Zona: "Oriental." Equipos: "Guantánamo", "Las Tunas", "Granma"	El sistema notifica al usuario que hubo un error en el nombre. Ejemplo: "El campo nombre está vacío o tiene caracteres incorrectos".	Se inserten números en el nombre del evento.
El estadístico introduce el dato siguiente: Nombre: "Antonio Mac4eo."	El sistema notifica al usuario que no se han adicionado los equipos.	No se inserten los equipos

ANEXOS

<p>Y selecciona: Sede: "Santiago de Cuba" Fecha: "4/03/10" Categoría: "Estudiantil" Zona: "Oriental." Equipos: ""</p>	<p>Ejemplo: "Adicione los equipos del evento"</p>	
<p>El estadístico selecciona la opción Modificar Evento.</p>	<p>El sistema muestra interfaz para seleccionar el nombre del evento.</p>	
<p>El estadístico selecciona el evento. Evento: "Antonio Maceo"</p>	<p>El sistema muestra los datos del evento a modificar.</p>	<p>Debe estar seleccionado el evento.</p>
<p>El estadístico modifica los datos necesarios: Nombre: "Antonio Maceo." Sede: "Las Tunas" Fecha: "6/03/10" Categoría: "Estudiantil" Zona: "Oriental" Equipos: "Guantánamo", "Las Tunas", "Granma"</p>	<p>El sistema notifica al usuario un mensaje de que los datos fueron modificados correctamente. Ejemplo: "Modificado correctamente."</p>	<p>Los datos estén correctos.</p>
<p>El estadístico modifica los datos del evento dejando algún campo vacío.</p>	<p>El sistema envía un mensaje informando que debe llenar todos los campos. Ejemplo: "El campo nombre está vacío o tiene caracteres incorrectos".</p>	<p>No se registren valores vacíos en los campos de entrada para insertar un nuevo evento.</p>

ANEXOS

<p>El estadístico modifica los datos siguientes: Nombre: "Antonio Mac4eo." Y selecciona: Sede: "Santiago de Cuba" Fecha: "4/03/10" Categoría: "Estudiantil" Zona: "Oriental." Equipos: "Guantánamo", "Las Tunas", "Granma"</p>	<p>El sistema notifica al usuario que hubo un error en el nombre. Ejemplo: "El campo nombre está vacío o tiene caracteres incorrectos".</p>	<p>Se modifiquen números en el nombre del evento.</p>
<p>El estadístico modifica los datos siguientes: Nombre: "Antonio Mac4eo." Y selecciona: Sede: "Santiago de Cuba" Fecha: "4/03/10" Categoría: "Estudiantil" Zona: "Oriental." Equipos: ""</p>	<p>El sistema notifica al usuario que no se han adicionado los equipos. Ejemplo: "Adicione los equipos del evento"</p>	<p>Se eliminen todos los equipos.</p>

Tabla 27: Caso de Prueba del Caso de Uso Gestionar Evento

Anexo 18: Caso de Prueba del Caso de Uso Gestionar Jugador.

Entrada	Resultados	Condiciones
<p>El estadístico selecciona la opción Insertar Jugador.</p>	<p>El sistema muestra una interfaz con los datos requeridos para insertar un nuevo jugador.</p>	
<p>El estadístico introduce los datos del jugador dejando algún campo vacío.</p>	<p>El sistema envía un mensaje informando que debe llenar todos los campos. Ejemplo: "El campo nombre está vacío o tiene caracteres</p>	<p>No se registren valores vacíos en los campos de entrada para insertar un nuevo jugador.</p>

ANEXOS

	incorrectos”.	
<p>El estadístico introduce los datos siguientes: Nombre: “ Ángel” Apellidos:”Danger Labañino” CI: “90092836542” Peso:”56” Altura:”165” Y selecciona: Sexo:”M” Evento:”Antonio Maceo” Equipo:”Santiago de Cuba”</p>	<p>El sistema inserta el nuevo jugador, confirmando la operación. Ejemplo: “Insertado correctamente”.</p>	<p>No exista el jugador en la aplicación.</p>
<p>El estadístico introduce los datos siguientes: Nombre: “ Ángel” Apellidos:”Danger Labañino” CI: “90092836542” Peso:”56” Altura:”165” Permite seleccionar: Sexo:”M” Evento:”Antonio Maceo” Equipo:”Santiago de Cuba”</p>	<p>El sistema informa que el jugador que se pretende insertar ya está registrado, enviando un mensaje. Ejemplo: “En el evento Antonio Maceo hay un jugador con ese CI, por favor, rectifique”.</p>	<p>Existe el jugador en la aplicación.</p>

ANEXOS

<p>El estadístico introduce los datos siguientes: Nombre: " Ángel" Apellidos:"Danger Labañino" CI: "90212836542" Peso:"56" Altura:"1.65" Permite seleccionar: Sexo:"M" Evento:" Antonio Maceo" Equipo:"Santiago de Cuba"</p>	<p>El sistema notifica al usuario que el mes del carnet es incorrecto. Ejemplo: ("El mes es incorrecto")</p>	<p>El mes del CI es incorrecto.</p>
<p>El estadístico introduce los datos siguientes: Nombre: " Ángel" Apellidos:"Danger Labañino" CI: "90093236542" Peso:"56" Altura:"1.65" Permite seleccionar: Sexo:"M" Evento:" Antonio Maceo" Equipo:"Santiago de Cuba"</p>	<p>El sistema notifica al usuario que el día del CI es incorrecto. Ejemplo: ("El día es incorrecto").</p>	<p>El día del CI es incorrecto.</p>

ANEXOS

<p>El estadístico introduce los datos siguientes: Nombre: " Ángel" Apellidos:"Danger Labañino" CI: "900928365429" Peso:"56" Altura:"1.65" Y selecciona: Sexo:"M" Evento:" Antonio Maceo" Equipo:"Santiago de Cuba"</p>	<p>El sistema notifica al usuario que el CI es incorrecto. Ejemplo: ("El CI debe tener 11 dígitos, sobran 2")</p>	<p>El CI tiene más de 11 números.</p>
<p>El estadístico introduce los datos siguientes: Nombre: " Ángel" Apellidos:"Danger Labañino" CI: "90092836545" Peso:"80d" Altura:"1.65" Y selecciona: Sexo:"M" Evento:" Antonio Maceo" Equipo:"Santiago de Cuba"</p>	<p>El sistema notifica al usuario que el peso es incorrecto. Ejemplo: ("El campo peso está vacío o tiene caracteres incorrectos")</p>	<p>El peso tiene caracteres incorrectos.</p>
<p>El estadístico selecciona uno de los jugadores para modificar algunos de sus datos.</p>	<p>Ver Sección "Modificar jugador."</p>	
<p>El estadístico selecciona uno de los jugadores para eliminarlo.</p>	<p>Ver Sección "Eliminar jugador."</p>	
<p>El estadístico selecciona la opción Modificar Jugador.</p>	<p>El sistema muestra interfaz para seleccionar el evento y el equipo.</p>	

ANEXOS

<p>El estadístico selecciona el evento y el equipo Evento: "Antonio Maceo." Equipo: "Santiago de Cuba"</p>	<p>El sistema muestra datos del jugador a modificar.</p>	<p>Debe estar seleccionado el evento y el equipo.</p>
<p>El estadístico selecciona el evento y el equipo. Evento: "Antonio Maceo." Equipo: "Santiago de Cuba" Y presiona el botón Modificar</p>	<p>El sistema muestra mensaje. Ejemplo: "Selecione el jugador a ser modificado"</p>	
<p>El estadístico modifica los datos necesarios: Nombre: "Ángel" Apellidos: "Danger Labañino" Peso: "58" Altura: "1.62" Y selecciona: Sexo: "M" Equipo: "Santiago de Cuba" Presiona el botón Modificar</p>	<p>El sistema envía un mensaje de que fue modificado correctamente. Ejemplo: "Modificado correctamente."</p>	<p>Los datos estén correctos.</p>
<p>El estadístico modifica los datos necesarios: Nombre: "Ángel" Apellidos: "Danger Labañino" Peso: "58kg" Altura: "1.62cm" Y selecciona: Sexo: "M" Equipo: "Santiago de Cuba"</p>	<p>El sistema le notifica al usuario que solo se aceptan números. Ejemplo: "Solo se aceptan números, rectifique."</p>	<p>En el peso y la altura se inserten letras.</p>
<p>El estadístico selecciona la opción Eliminar Jugador.</p>	<p>El sistema muestra interfaz para seleccionar el evento y el equipo.</p>	

ANEXOS

El estadístico selecciona el evento y el equipo. Evento: "Antonio Maceo." Equipo: "Las Tunas."	El sistema muestra listado de jugadores.	Debe estar seleccionado el evento y el equipo.
El estadístico selecciona el jugador y presiona el botón Eliminar.	El sistema envía una pregunta para garantizar que no se realice una operación errada. ¿Está seguro que desea eliminar el jugador? (Si, No)	Sea presionado el botón Eliminar.
El estadístico presiona el botón Si.	El sistema envía mensaje que el jugador fue eliminado correctamente. Ejemplo: "El jugador seleccionado se eliminará. Desea continuar (Si, No)"	Sea presionado el botón Si.
El estadístico selecciona el evento y el equipo: Evento: "Antonio Maceo" Equipo: "Las Tunas" Y presiona el botón Eliminar.	El sistema muestra mensaje "Seleccione el jugador a ser eliminado"	

Tabla 28: Caso de Prueba del Caso de Uso Gestionar Jugador

Anexo 19: Caso de Prueba del Caso de Uso Gestionar Usuario.

Entrada	Resultados	Condiciones
El administrador selecciona la opción Insertar Usuario.	El sistema muestra una interfaz con los datos requeridos para insertar un nuevo usuario.	
El administrador introduce los datos del usuario	El sistema envía un mensaje informando que	No se registren valores vacíos en los campos de entrada para

ANEXOS

dejando algún campo vacío.	debe llenar todos los campos. Ejemplo: "El campo nombre completo está vacío o tiene caracteres incorrectos".	insertar un nuevo usuario.
El administrador introduce los datos siguientes: Nombre Completo:"Ernesto Vargas Rodríguez " Usuario:"ernesto" Clave:"871009" Confirmar Clave:"871009" Categoría:"usuario"	El sistema inserta el nuevo usuario, confirmando la operación. Ejemplo: "Insertado correctamente".	No exista el usuario en la aplicación.
El administrador introduce los datos siguientes: Nombre Completo:"Ernesto Vargas Rodríguez " Usuario:"ernesto" Clave:"870112" Confirmar Clave:"870112" Categoría:"administrador"	El sistema informa que el usuario que se pretende insertar ya está registrado, enviando un mensaje. Ejemplo: " El usuario ya se encuentra registrado"	Existe el usuario en la aplicación.
El administrador introduce los datos siguientes: Nombre Completo:"Ángel Suárez Rodríguez " Usuario:"angel" Clave:"870112" Confirmar Clave:"8701123" Categoría:"administrador"	El sistema le comunica al usuario que las contraseñas introducidas en los campos Clave y Confirmar Clave son incorrectas. Ejemplo: "Verifique la clave."	Las contraseñas son incorrectas.
El administrador selecciona la opción de Modificar Usuario.	El sistema muestra una interfaz con un listado de usuarios.	

ANEXOS

<p>El administrador selecciona el usuario a modificar y modifica alguno de los datos siguientes: Nombre Completo: "Ángel Suárez Rodríguez " Usuario: "angel" Clave: "8701123" Confirmar Clave: "8701123" Categoría: "administrador" Y presiona el botón Modificar</p>	<p>El sistema envía un mensaje de que fue modificado correctamente. Ejemplo: "Modificado correctamente."</p>	<p>Los datos estén correctos.</p>
<p>El administrador selecciona el usuario a modificar y modifica alguno de los datos siguientes: Nombre Completo: "Ángel Suárez Rodríguez " Usuario: "asuauezr" Clave: "8701123" Categoría: "usuario" Y presiona el botón Modificar.</p>	<p>El sistema le comunica al usuario que no puede cambiar la categoría. Ejemplo: "Solo existe un administrador, no puede cambiar su categoría."</p>	<p>Exista un solo administrador en la aplicación.</p>
<p>El administrador selecciona la opción de Eliminar Usuario.</p>	<p>El sistema muestra una interfaz con una lista de usuarios.</p>	
<p>El administrador selecciona el usuario a eliminar y presiona el botón Eliminar.</p>	<p>El sistema envía una pregunta para garantizar que no se realice una operación errada. ¿El usuario está a punto de ser eliminado. Desea</p>	<p>Sea presionado el botón Eliminar.</p>

ANEXOS

	continuar?(Si, No)	
El administrador presiona el botón Si de la pregunta lanzada posteriormente.	El sistema envía mensaje que el usuario fue eliminado correctamente. Ejemplo:"Eliminado correctamente."	Sea presionado el botón Si.

Tabla 29: Caso de Prueba del Caso de Uso Gestionar Usuario

Anexo 20: Caso de Prueba del Caso de Uso Autenticar Usuario.

Entrada	Resultados	Condiciones
El usuario desea acceder a la aplicación por lo que introduce los datos: Usuario:"asuares" Contraseña: "*****" (adrian). Y presiona el botón Aceptar	El sistema muestra interfaz en dependencia del rol que tenga este usuario.	El usuario esté registrado en la base de datos.
El usuario desea acceder al sistema por lo que introduce los datos: Usuario:"adrian" Contraseña:"*****" (administrador). Y presiona el botón Aceptar.	El sistema muestra mensaje Ejemplo:"Error"	El usuario no se encuentre registrado en la base de datos.
El usuario desea acceder al sistema por lo que introduce los datos: Usuario:"asuares" Contraseña:"*****"(admin)	El sistema le notifica al usuario que la contraseña es incorrecta. Ejemplo:"Error."	La contraseña para acceder al sistema es incorrecta.

Tabla 30: Caso de Prueba del Caso de Uso Autenticar Usuario

Anexo 21: Caso de Prueba del Caso de Uso Consultar Reporte Resumen Estadístico de cada Equipo.

ANEXOS

Entrada	Resultados	Condición
El estadístico selecciona la opción Consultar reporte resumen estadístico de cada equipo.	El sistema muestra una interfaz para seleccionar el evento y el sexo.	
El estadístico presiona el botón Ver reporte.	El sistema notifica al usuario que debe escoger las opciones para realizar la búsqueda. Ejemplo: "Escoja las opciones para ver el reporte. "	No existen seleccionadas las opciones para ver el reporte.
El estadístico selecciona Evento:" Antonio Maceo" Sexo:"F" Y presiona el botón Ver reporte.	Muestra el reporte resumen estadístico de cada equipo	Deben existir juegos registrados.
El estadístico selecciona Evento:" Rubén Martínez Villena" Sexo:"M" Y presiona el botón Ver reporte.	El sistema notifica al usuario que no se han registrado juegos. Ejemplo:"No se han registrado juegos masculinos en este evento"	No existan juegos masculinos registrados en el evento seleccionado.

Tabla 31: Caso de Prueba del Caso de Uso Consultar Reporte Resumen Estadístico de cada Equipo

Anexo 22: Caso de Prueba del Caso de Uso Consultar Reporte Resultados por Provincia.

Entrada	Resultados	Condición
El estadístico selecciona la opción Consultar reporte	El sistema muestra una interfaz para seleccionar el	

ANEXOS

resultados por provincias.	evento.	
El estadístico selecciona Evento:" Antonio Maceo" Y presiona el botón Ver reporte.	Muestra el reporte resultados por provincias.	Deben existir juegos registrados.
El estadístico selecciona Evento:" Rubén Martínez Villena" Y presiona el botón Ver reporte.	El sistema notifica al usuario que no se han registrado juegos. Ejemplo:"No se han registrado juegos en este evento"	No existan juegos registrados del evento seleccionado.

Tabla 32: Caso de Prueba del Caso de Uso Consultar Reporte Resultados por Provincia

Anexo 23: Caso de Prueba del Caso de Uso Consultar Reporte Resultados Alcanzados por las Provincias.

Entrada	Resultados	Condición
El estadístico selecciona la opción Consultar reporte resultados alcanzados por las provincias.	El sistema muestra una interfaz para seleccionar el evento.	
El estadístico selecciona Evento:" Antonio Maceo" Y presiona el botón Ver reporte.	Muestra el reporte resultados alcanzados por las provincias.	Deben existir juegos registrados.
El estadístico selecciona Evento:" Rubén Martínez Villena" Y presiona el botón Ver	El sistema notifica al usuario que no se han registrado juegos. Ejemplo:"No se han	No existan juegos registrados del evento seleccionado.

ANEXOS

reporte.	registrado juegos en este evento”	
----------	-----------------------------------	--

Tabla 33: Caso de Prueba del Caso de Uso Consultar Reporte Resultados Alcanzados por las Provincias

Anexo 24: Caso de Prueba del Caso de Uso Consultar Reporte Resultados de los Objetivos Pedagógicos.

Entrada	Resultados	Condición
El estadístico selecciona la opción Consultar reporte resultados de los objetivos pedagógicos.	El sistema muestra una interfaz para seleccionar el evento y el sexo.	
El estadístico selecciona Evento:” Antonio Maceo” Sexo:”F” Y presiona el botón Ver reporte.	Muestra el reporte resultados de los objetivos pedagógicos.	Deben existir juegos registrados.
El estadístico selecciona Evento:” Rubén Martínez Villena” Sexo:”M” Y presiona el botón Ver reporte.	El sistema notifica al usuario que no se han registrado juegos. Ejemplo:”No se han registrado juegos masculinos en este evento”	No existan juegos registrados del evento seleccionado.

Tabla 34: Caso de Prueba del Caso de Uso Consultar Reporte Resultados de los Objetivos Pedagógicos

Anexo 25: Caso de Prueba del Caso de Uso Consultar Reporte Control Competitivo.

Entrada	Resultados	Condición
El estadístico selecciona la opción Consultar reporte control competitivo.	El sistema muestra una interfaz para seleccionar el evento.	
El estadístico selecciona Evento:” Mella” Y presiona el botón Ver reporte.	El sistema notifica al usuario que el evento está abierto. Ejemplo:” El evento está abierto, estará disponible al final de la competencia ”	El estado del evento seleccionado este abierto.
El estadístico selecciona Evento:” Antonio Maceo” Y presiona el botón Ver reporte.	El sistema muestra el reporte control competitivo.	Deben existir juegos registrados.
El estadístico selecciona Evento:” Rubén Martínez Villena” Y presiona el botón Ver reporte.	El sistema notifica al usuario que no se han registrado juegos. Ejemplo:”No se han registrado juegos en este evento”	No existan juegos registrados del evento seleccionado.

Tabla 35: Caso de Prueba del Caso de Uso Consultar Reporte Control Competitivo.

Anexo 26: Caso de Prueba del Caso de Uso Consultar Reporte Equipo Ideal.

Entrada	Resultados	Condición
El estadístico selecciona la opción Consultar reporte	El sistema muestra una interfaz para seleccionar el	

ANEXOS

equipo ideal.	evento.	
El estadístico selecciona Evento:” Antonio Maceo” Y presiona el botón Ver reporte.	El sistema muestra el reporte equipo ideal.	Debe estar registrada la talla de los jugadores.
El estadístico selecciona Evento:” Mella” Y presiona el botón Ver reporte.	El sistema notifica al usuario que el evento está abierto. Ejemplo:” El evento está abierto, estará disponible al final de la competencia ”	El estado del evento seleccionado este abierto.

Tabla 36: Caso de Prueba del Caso de Uso Consultar Reporte Equipo Ideal

Anexo 27: Caso de Prueba del Caso de Uso Consultar Reporte Historial de Usuario.

Entrada	Resultados	Condición
El estadístico selecciona la opción Consultar reporte historial de usuario.	El sistema muestra una interfaz para seleccionar el evento.	
El estadístico presiona el botón Ver reporte.	El sistema notifica al usuario que debe escoger las opciones para realizar la búsqueda. Ejemplo: “Escoja las opciones para ver el reporte. “	No esté seleccionado el evento.
El estadístico selecciona: Evento:” Antonio Maceo” Y presiona el botón Ver reporte.	Muestra el reporte historial de usuario	Deben existir usuarios registrados.

Tabla 37: Caso de Prueba del Caso de Uso Consultar Reporte Historial de Usuario

Anexo 28: Caso de Prueba del Caso de Uso Consultar Evento.

Entrada	Resultados	Condiciones
El estadístico selecciona la opción Consultar Evento.	El sistema muestra los parámetros para realizar la búsqueda.	
El estadístico presiona el botón Buscar.	El sistema notifica al usuario que debe escoger los criterios para realizar la búsqueda. Ejemplo: "Escoja uno de los criterios para realizar la búsqueda. "	No existe ningún criterio seleccionado para realizar la búsqueda.
El estadístico selecciona los criterios para realizar la búsqueda. Zona:"Oriental" Sede:"Las Tunas" Categoría:"Pioneril" Fecha:"05/04/10"	El sistema muestra los datos según los criterios escogidos. Evento:"5" Nombre:"Antonio Maceo" Sede:"Las Tunas" Zona:"Oriental" Categoría:"Pioneril" Fecha:"05/04/010" Estado:"Cerrado"	Debe estar seleccionado al menos uno de los criterios.
El estadístico selecciona las opciones para realizar la búsqueda. Zona:"Oriental" Sede:"Las Tunas" Categoría:"Pioneril" Fecha:"05/04/10"	El sistema notifica al usuario que no existen coincidencias encontradas.	No exista registrado en la aplicación ningún evento con esas características.
El estadístico selecciona uno de los eventos para	El sistema muestra los datos a modificar.	Se selecciona un evento.

ANEXOS

<p>modificar alguno de sus datos.</p>	<p>Evento:"5" Nombre:"Antonio Maceo" Sede:"Las Tunas" Zona:"Oriental" Categoría:"Pioneril" Fecha:"05/04/010"</p>	
<p>El estadístico modifica los datos.</p> <p>Evento:"5" Nombre:"Mella" Sede:"Santiago de Cuba" Zona:"Oriental" Categoría:"Pioneril" Fecha:"05/04/010" Y presiona el botón Modificar</p>	<p>El sistema le notifica al usuario que el evento se encuentra cerrado.</p> <p>Ejemplo:" El evento Antonio Maceo se encuentra cerrado, consulte al administrador"</p>	<p>El estado del evento es cerrado.</p>
<p>El estadístico modifica los datos.</p> <p>Evento:"5" Nombre:"Mella" Sede:"Santiago de Cuba" Zona:"Oriental" Categoría:"Pioneril" Fecha:"05/04/010" Y presiona el botón Modificar</p>	<p>El sistema notifica al usuario que los datos fueron modificados correctamente.</p> <p>Ejemplo:" Evento modificado correctamente."</p>	<p>Los datos estén correctos.</p>

Tabla 38: Caso de Prueba del Caso de Uso Consultar Evento

Anexo 29: Caso de Prueba del Caso de Uso Consultar Juego.

Entrada	Resultados	Condiciones
<p>El estadístico selecciona la opción Consultar Juego.</p>	<p>El sistema muestra los parámetros para realizar la</p>	

ANEXOS

	búsqueda.	
El estadístico presiona el botón Buscar.	El sistema notifica al usuario que debe escoger las opciones para realizar la búsqueda. Ejemplo: "Escoja una de las opciones para realizar la búsqueda. "	No existe ningún criterio seleccionado para realizar la búsqueda.
El estadístico selecciona los criterios para realizar la búsqueda. Zona:"Oriental" Evento:"Mella" Equipo ganador:"Santiago de Cuba" Equipo perdedor:"Guantánamo" Categoría:"Escolar" Sexo:"Masculino" Fecha:"09/04/10"	El sistema muestra los datos a modificar según los criterios escogidos. Juego:"8" Equipo ganador:"Santiago de Cuba" Sets ganados:"3" Equipo perdedor:"Guantánamo" Sets ganados:"1" Fecha:"09/04/10"	Debe estar seleccionado al menos uno de los criterios.
El estadístico selecciona las opciones para realizar la búsqueda. Zona:"Oriental" Evento:"Mella" Equipo ganador:"Santiago de Cuba" Equipo perdedor:"Guantánamo" Categoría:"Escolar" Sexo:"Masculino" Fecha:"05/04/10"	El sistema notifica al usuario que no existen coincidencias encontradas. Ejemplo: "No se encontraron coincidencias."	No exista registrado en la aplicación ningún juego con esas características.
El estadístico selecciona modificar o uno de los juegos para modificar algunos de sus	Ver caso de prueba de la Sección "Modificar juego" del caso de uso Gestionar juego.	

ANEXOS

datos.		
El estadístico selecciona la eliminar o el juego a eliminar y presiona el botón Eliminar.	Ver caso de prueba de la Sección “Eliminar juego” del caso de uso Gestionar juego.	

Tabla 39: Caso de Prueba del Caso de Uso Consultar Juego

Anexo 30: Caso de Prueba del Caso de Uso Consultar Jugador.

Entrada	Resultados	Condiciones
El estadístico selecciona la opción Consultar Jugador	El sistema muestra los criterios para realizar la búsqueda.	
El estadístico presiona el botón Buscar.	El sistema notifica al usuario que debe escoger las opciones para realizar la búsqueda. Ejemplo: “Escoja uno de los criterios para realizar la búsqueda. “	No existe ningún criterio seleccionado para realizar la búsqueda.
El estadístico selecciona los criterios para realizar la búsqueda. Zona:”Occidental” Evento:”Mella” Equipo:”Cienfuegos” Categoría:”Escolar” Sexo:”F” CI:”87100932653”	El sistema muestra los datos según los criterios escogidas. CI:”87100932653” Nombre:”Yaneisis” Apellidos:” Vargas Labañino” Sexo:”F” Peso:”49” Altura:”1.55” Equipo:”Cienfuegos”	Debe estar selecciona al menos uno de los criterios.
El estadístico selecciona las opciones para realizar la búsqueda. Zona:”Occidental” Evento:”Mella” Equipo:”Cienfuegos”	El sistema notifica al usuario que no existen coincidencias encontradas.	No exista registrado en la aplicación ningún evento con esas características.

ANEXOS

Categoría:"Escolar" Sexo:"F" CI:"87100932653"		
El estadístico selecciona uno de los jugadores para modificar alguno de sus datos.	Ver caso de prueba de la Sección "Modificar jugador" del caso de uso Gestionar jugador.	
El estadístico selecciona el jugador a eliminar y presiona el botón Eliminar.	Ver caso de prueba de la Sección "Eliminar jugador" del caso de uso Gestionar jugador.	

Tabla 40: Caso de Prueba del Caso de Uso Consultar Jugador

Anexo 31: Caso de Prueba del Caso de Uso Consultar Usuario.

Entrada	Resultados	Condiciones
El administrador selecciona la opción Consultar usuario	El sistema muestra los criterios para realizar la búsqueda.	
El administrador selecciona la opción Buscar.	El sistema notifica al usuario que debe escoger las opciones para realizar la búsqueda. Ejemplo: "Escoja una de los criterios para realizar la búsqueda. "	No existe ninguna opción seleccionada para realizar la búsqueda.
El administrador selecciona los criterios para realizar la búsqueda. Usuario:"ylabanino" Categoría:"usuario "	El sistema muestra los datos según las opciones escogidas. Nombre (s):" Yaneisis" Apellidos:" Vargas Labañino" Usuario:"ylabanino" Categoría:"usuario "	Debe estar selecciona al menos una de las opciones.

ANEXOS

El administrador selecciona los criterios para realizar la búsqueda. Usuario: "yabanino" Categoría: "usuario "	El sistema notifica al usuario que no existen coincidencias encontradas.	No exista registrado en la aplicación ningún usuario con esas características.
El administrador selecciona uno de los usuarios para modificar alguno de sus datos.	Ver caso de prueba de la Sección "Modificar usuario" del caso de uso Gestionar Usuario.	
El administrador selecciona el usuario a eliminar y presiona el botón Eliminar.	Ver caso de prueba de la Sección "Eliminar usuario" del caso de uso Gestionar Usuario.	

Tabla 41: Caso de Prueba del Caso de Uso Consultar Usuario

Anexo 32: Descripción de la Tabla de la Base de Datos "equipo".

Nombre: equipo		
Descripción: Esta tabla contiene los equipos que pueden participar en un evento.		
Atributo	Tipo	Descripción
id_equipo	Integer	Representa el identificador de la tabla equipo.
nombre	String	Representa el nombre del equipo.
nombre_corto	String	Representa la abreviatura del nombre del equipo.

Tabla 42: Descripción de la Tabla de la Base de Datos "equipo"

Anexo 33: Descripción de la Tabla de la Base de Datos “evento_equipo”.

Nombre: evento_equipo		
Descripción: Esta tabla contiene los datos		
Atributo	Tipo	Descripción
id_evento	Integer	Representa el identificador del evento.
id_equipo	Integer	Representa el identificador del equipo
orden	Integer	Representa el orden en que es insertado los datos.

Tabla 43: Descripción de la Tabla de la Base de Datos "evento_ equipo"

Anexo 34: Descripción de la Tabla de la Base de Datos “jugador”.

Nombre: jugador		
Descripción: Esta tabla contiene los datos del jugador.		
Atributo	Tipo	Descripción
id_jugador	Integer	Representa el identificador de la tabla jugador.
id _ equipo	Integer	Representa el identificador del equipo al que pertenece.
id _ evento	Integer	Representa el identificador del evento al que pertenece.
nombre	String	Representa el nombre del jugador.
apellidos	String	Representa los apellidos del jugador.
sexo	String	Representa el sexo del jugador.
peso	Double	Representa el peso del jugador.
altura	Double	Representa la altura del jugador.

ANEXOS

orden	Integer	Representa el orden con que se insertan los datos.
-------	---------	--

Tabla 44: Descripción de la Tabla de la Base de Datos "jugador"

Anexo 35: Descripción de la Tabla de la Base de Datos "juego".

Nombre: juego		
Descripción: Esta tabla contiene los datos de un juego entre dos equipos.		
Atributo	Tipo	Descripción
id_juego	Integer	Representa el identificador de la tabla juego.
id_evento	Integer	Representa el identificador del evento al que pertenece el juego.
sexo	String	Representa si el juego entre los equipos es femenino o masculino.
fecha	Date	Representa la fecha en que se realizó un juego.
equipo_ganador	Integer	Representa el equipo que fue ganador en un juego.
k1_g	Integer	Representa el total de K1 del equipo ganador de un juego.
k1_efectivos_g	Integer	Representa el total de K1 que fueron efectivos del equipo ganador de un juego.
k2_g	Integer	Representa el total de K2 del equipo ganador de un juego.
k2_efectivos_g	Integer	Representa el total de K1 que fueron efectivos del equipo ganador de un juego.
saques_g	Integer	Representa el total de saques del equipo ganador de un

ANEXOS

		juego.
saques_efectivos_g	Integer	Representa el total de saques que fueron efectivos del equipo ganador de un juego.
set_g	Integer	Representa el número de sets ganados del equipo ganador en un juego.
equipo_perdedor	Integer	Representa el equipo que fue perdedor en un juego.
k1_p	Integer	Representa el total de K1 del equipo perdedor de un juego.
k1_efectivos_p	Integer	Representa el total de K1 que fueron efectivos del equipo perdedor de un juego.
k2_p	Integer	Representa el total de K2 del equipo perdedor de un juego.
k2_efectivos_p	Integer	Representa el total de K2 que fueron efectivos del equipo perdedor de un juego.
saques_p	Integer	Representa el total de saques del equipo perdedor de un juego.
saques_efectivos_p	Integer	Representa el total de saques que fueron efectivos del equipo perdedor de un juego.
set_g	Integer	Representa el número de sets ganados del equipo perdedor en un juego.
orden	Integer	Representa el orden en que se insertan los datos.

Tabla 45: Descripción de la Tabla de la Base de Datos "juego"

Anexo 36: Descripción de la Tabla de la Base de Datos "usuario".

ANEXOS

Nombre: usuario		
Descripción: Esta tabla contiene los datos del usuario.		
Atributo	Tipo	Descripción
id_usuario	Integer	Representa el identificador del usuario.
nombre_usuario	String	Representa el nombre de usuario que va a tener una persona.
clave	String	Representa la clave que va a tener un usuario.
nombre_persona	String	Representa el nombre de la persona a las que se les ha asignado un usuario y clave.
categoría	Integer	Representa la categoría que tiene el usuario.

Tabla 46: Descripción de la Tabla de la Base de Datos "usuario"

Anexo 37: Descripción de la Tabla de la Base de Datos "datosVariables".

Nombre: datosVariables		
Descripción: Esta tabla contiene los parámetros usados para realizar los reportes que contienen objetivos pedagógicos y que los usuarios pueden cambiar según sus necesidades.		
Atributo	Tipo	Descripción
id_datos	Integer	Representa el identificador de la clase.
id_evento	Integer	Representa el identificador del evento al que pertenece.
porcentaje_k1_m	Integer	Representa el porcentaje de puntos de la efectividad del k1 para los equipos masculinos.
porcentaje_k1_f	Integer	Representa el porcentaje de puntos de la efectividad del k1 para los equipos femeninos.

ANEXOS

porciento_k2_m	Integer	Representa el porciento de puntos de la efectividad del k2 para los equipos masculinos.
porciento_k2_f	Integer	Representa el porciento de puntos de la efectividad del k2 para los equipos femeninos.
porciento_saques_m	Integer	Representa el porciento de puntos de la efectividad de los saques para los equipos masculinos.
Porciento_saques_f	Integer	Representa el porciento de puntos de la efectividad de los saques para los equipos femeninos.

Tabla 47: Descripción de la Tabla de la Base de Datos "datosVariables"

Anexo 38: Diagrama de Clases del Análisis del Caso de Uso Autenticar Usuario.



Figura 15: Diagrama de Clases del Análisis del Caso de Uso Autenticar Usuario

Anexo 39: Diagrama de Clases del Análisis del Caso de uso Gestionar Evento.

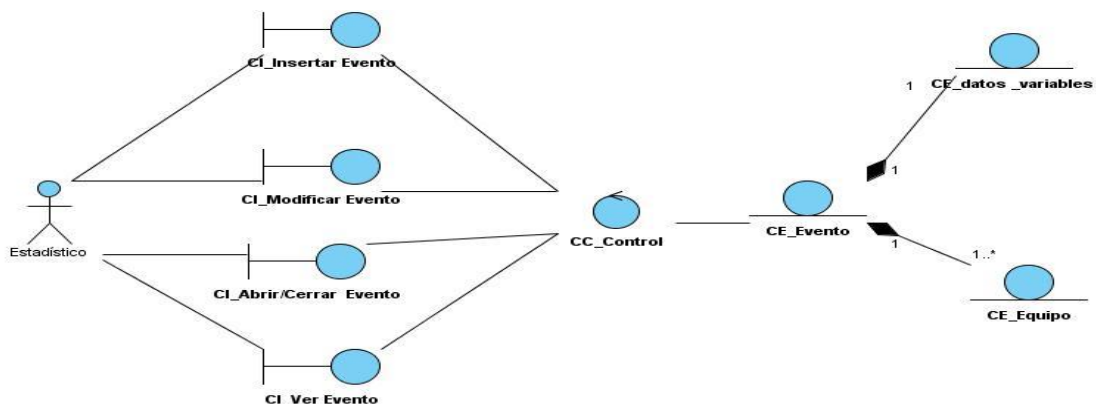


Figura 16: Diagrama de Clases del Análisis del Caso de Uso Gestionar Evento

Anexo 40: Diagrama de Clases del Análisis del Caso de Uso Gestionar Jugador.

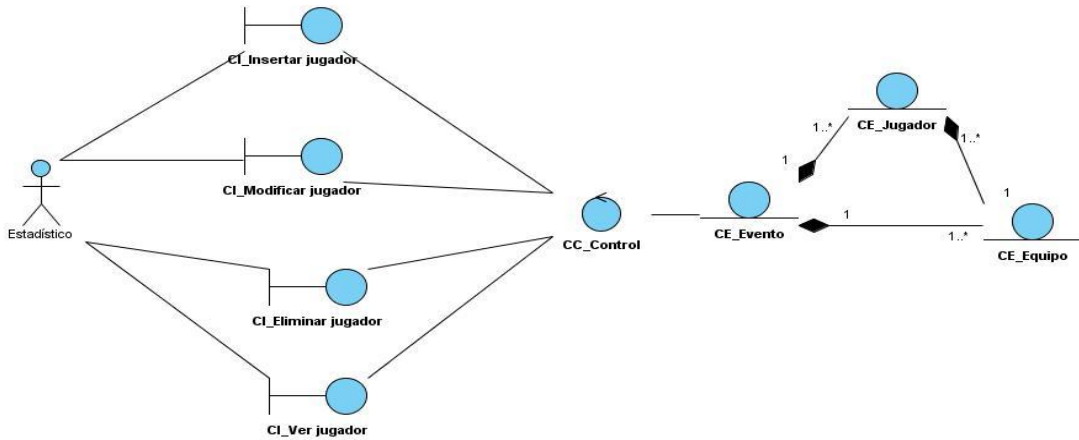


Figura 17: Diagrama de Clases del Análisis del Caso de Uso Gestionar Jugador

Anexo 41: Diagrama de Clases del Análisis del Caso de Uso Gestionar Usuario.

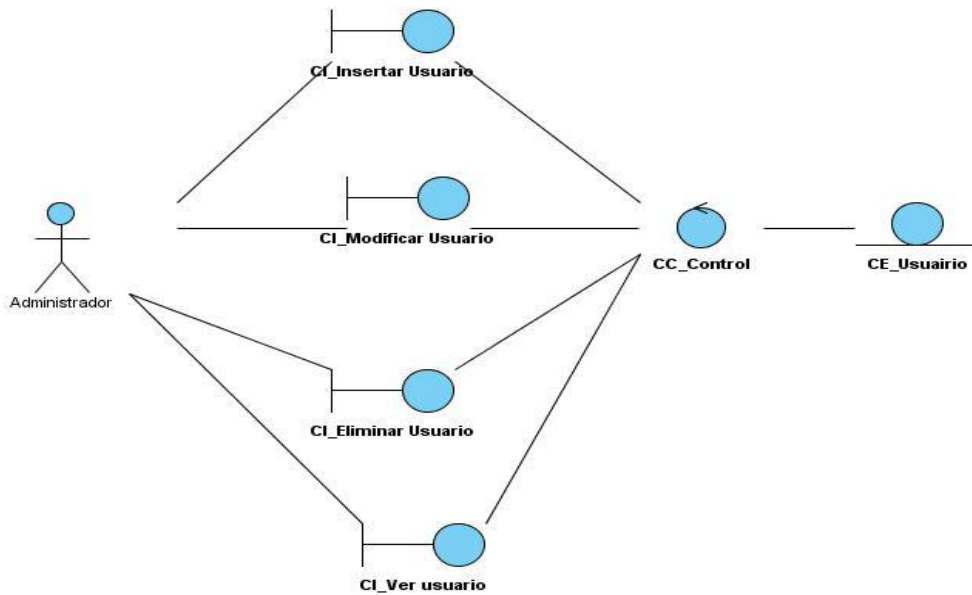


Figura 18: Diagrama de Clases del Análisis del Caso de Uso Gestionar Usuario

Anexo 44: Diagrama de Clases del Análisis del Caso de Uso Consultar Reporte Equipo Ideal.

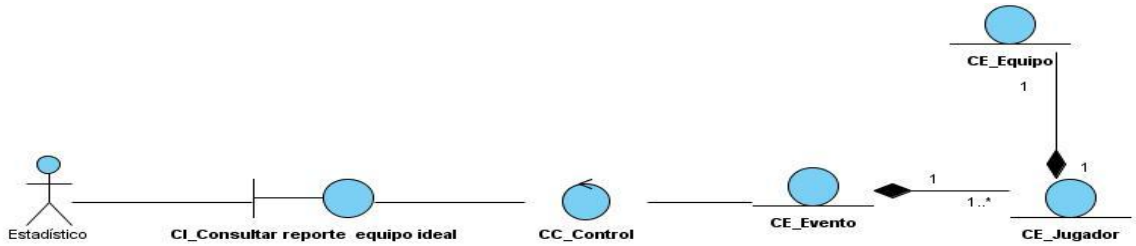


Figura 21: Diagrama de Clases del Análisis del Caso de Uso Consultar Reporte Equipo Ideal.

Anexo 45: Diagrama de Clases del Análisis del Caso de Uso Consultar Reporte Control Competitivo.

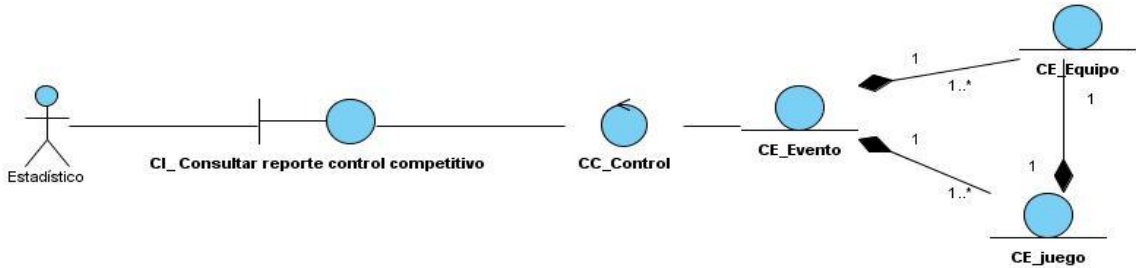


Figura 22: Diagrama de Clases del Análisis del Caso de Uso Consultar Reporte Control Competitivo

Anexo 46: Diagrama de Clases del Análisis del Caso de Uso Consultar Reporte Resultados Alcanzados por Provincias.

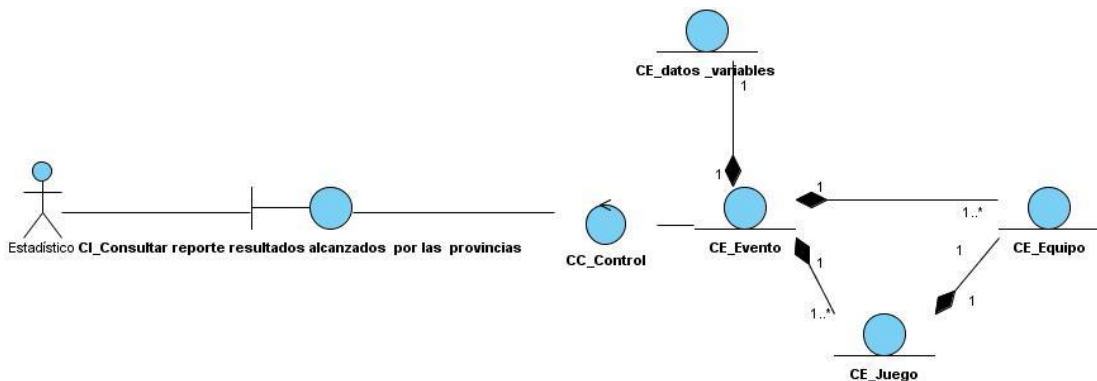


Figura 23: Diagrama de Clases del Análisis del Caso de Uso Consultar Reporte Resultados Alcanzados por las Provincias

Anexo 47: Diagrama de Clases del Análisis del Caso de Uso Consultar Reporte Resultados por Provincias.

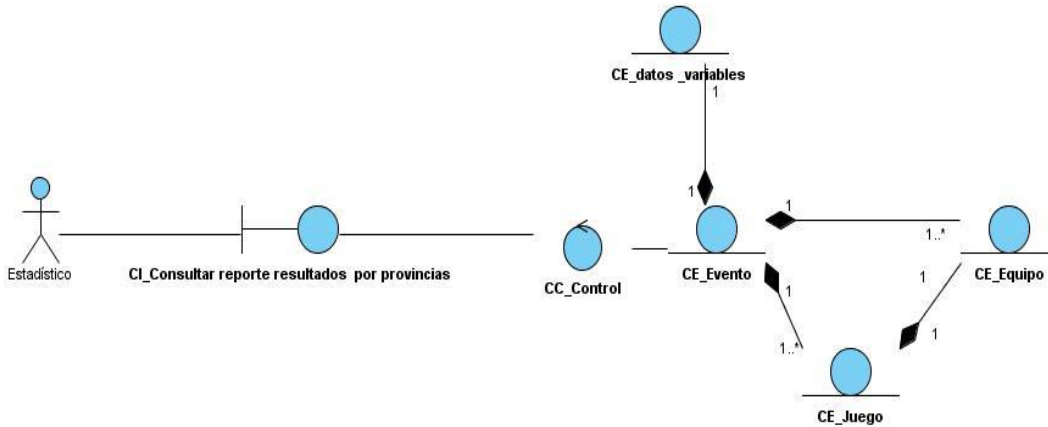


Figura 24: Diagrama de Clases del Análisis del Caso de Uso Consultar Reporte Resultados por Provincias

Anexo 48: Diagrama de Clases del Análisis del Caso de Uso Consultar Reporte Historial de Usuario.



Figura 25: Diagrama de Clases del Análisis del Caso de Uso Consultar Reporte Historial de Usuario

Anexo 49: Diagrama de Clase del Análisis del Caso de Uso Consultar Evento.

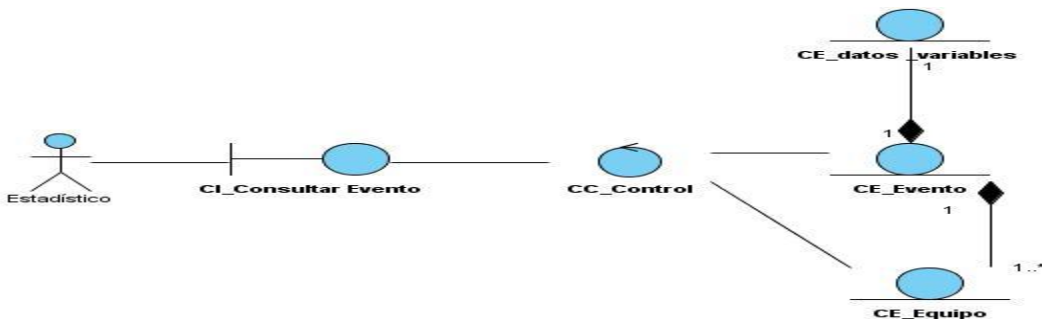


Figura 26: Diagrama de Clase del Análisis del Caso de Uso Consultar Evento

Anexo 50: Diagrama de Clase del Análisis del Caso de Uso Consultar Juego

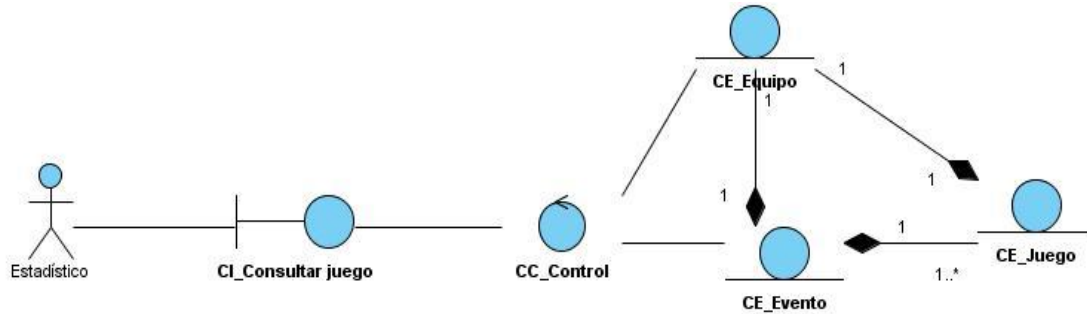


Figura 27: Diagrama de Clase del Análisis del Caso de Uso Consultar Juego

Anexo 51: Diagrama de Clase del Análisis del Caso de Uso Consultar Jugador.

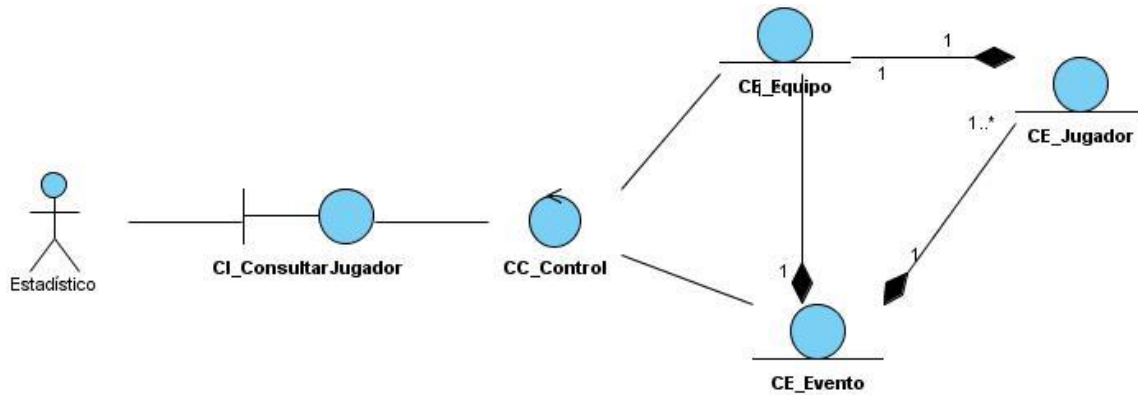


Figura 28: Diagrama de Clase del Análisis del Caso de Uso Consultar Jugador

Anexo 52: Diagrama de Clase del Análisis del Caso de Uso Consultar Usuario.

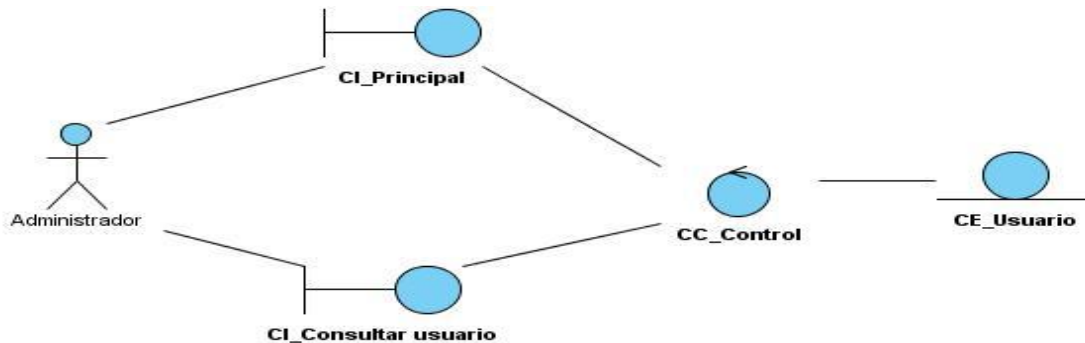


Figura 29: Diagrama de Clase de Análisis del Caso de Uso Consultar Usuario

Anexo 53: Diagrama de Secuencia del Caso de Uso Autenticar Usuario.

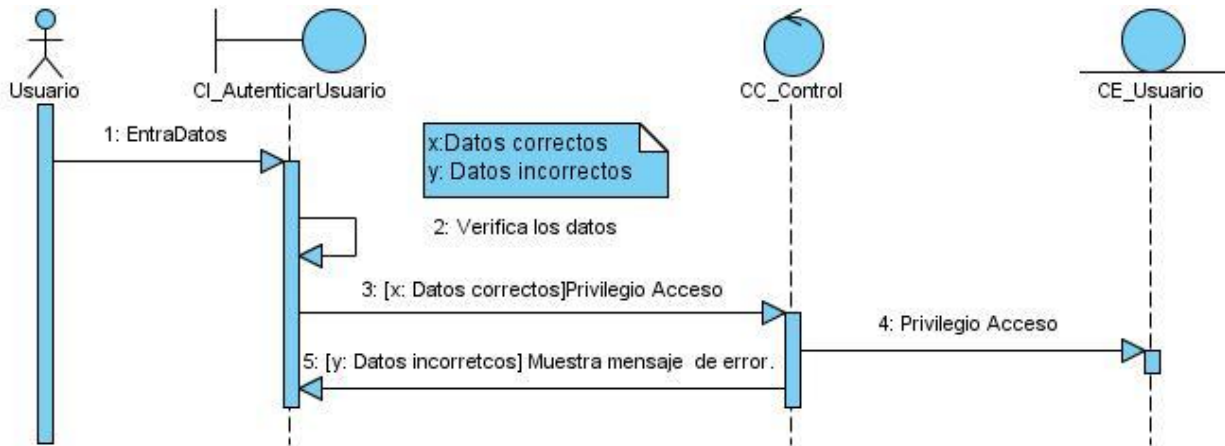


Figura 30: Diagrama de Secuencia del Caso de Uso Autenticar Usuario

Anexo 54: Diagrama de Secuencia del Caso de Uso Gestionar Usuario Sección Insertar Usuario.

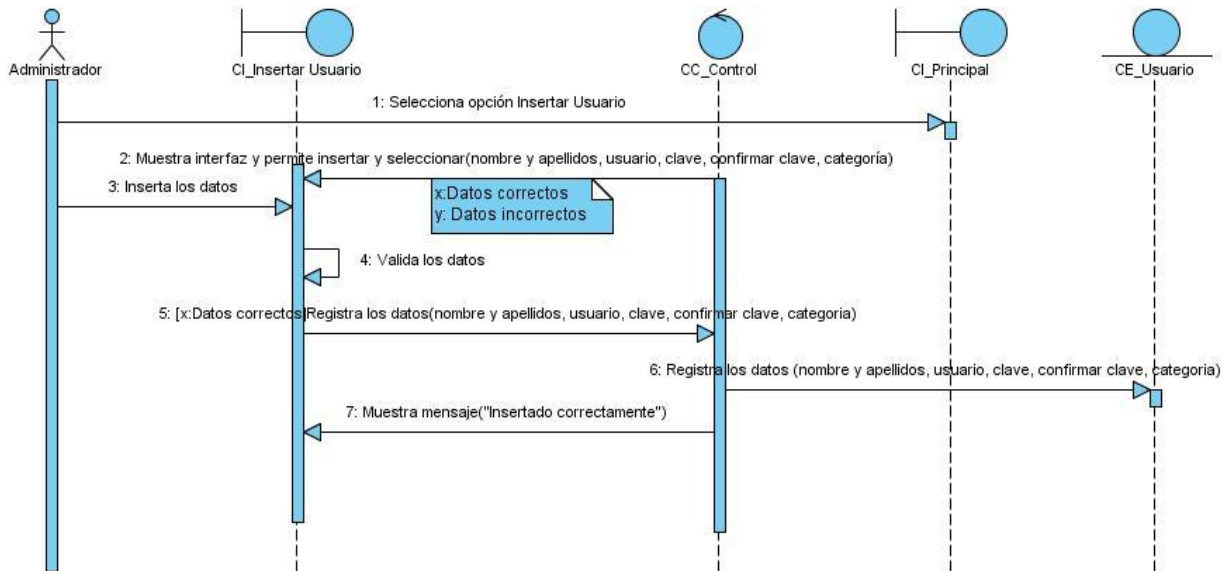


Figura 31: Diagrama de Secuencia del Caso de Uso Gestionar Usuario Sección Insertar Usuario

Anexo 55: Diagrama de Secuencia del Caso de Uso Gestionar Usuario Sección Eliminar Usuario.

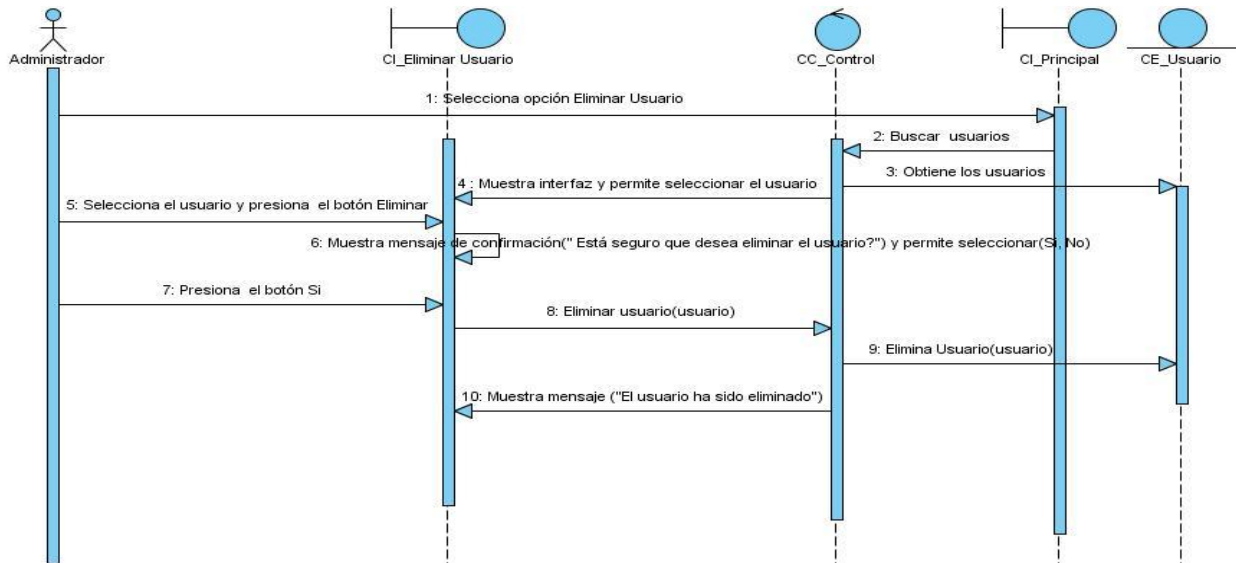


Figura 32: Diagrama de Secuencia del Caso de Uso Gestionar Usuario Sección Eliminar Usuario

Anexo 56: Diagrama de Secuencia del Caso de Uso Gestionar Usuario Sección Modificar Usuario.

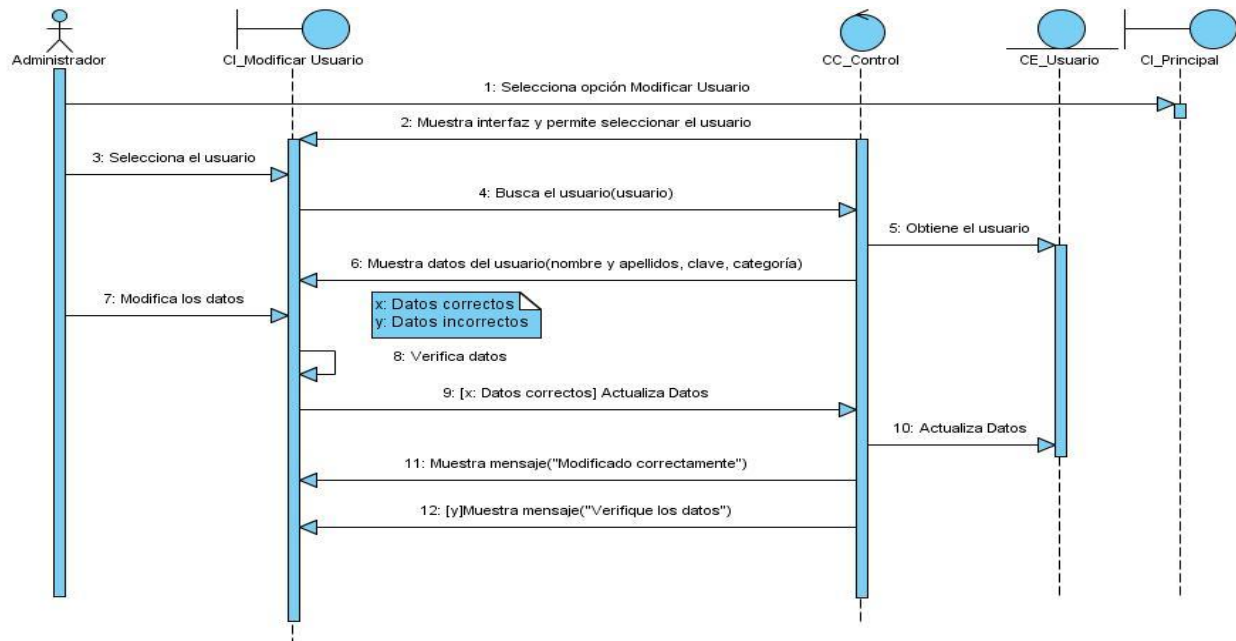


Figura 33: Diagrama de Secuencia del Caso de Uso Gestionar Usuario Sección Modificar Usuario

Anexo 57: Diagrama de Secuencia del Caso de Uso Gestionar Usuario Sección Ver Usuario.

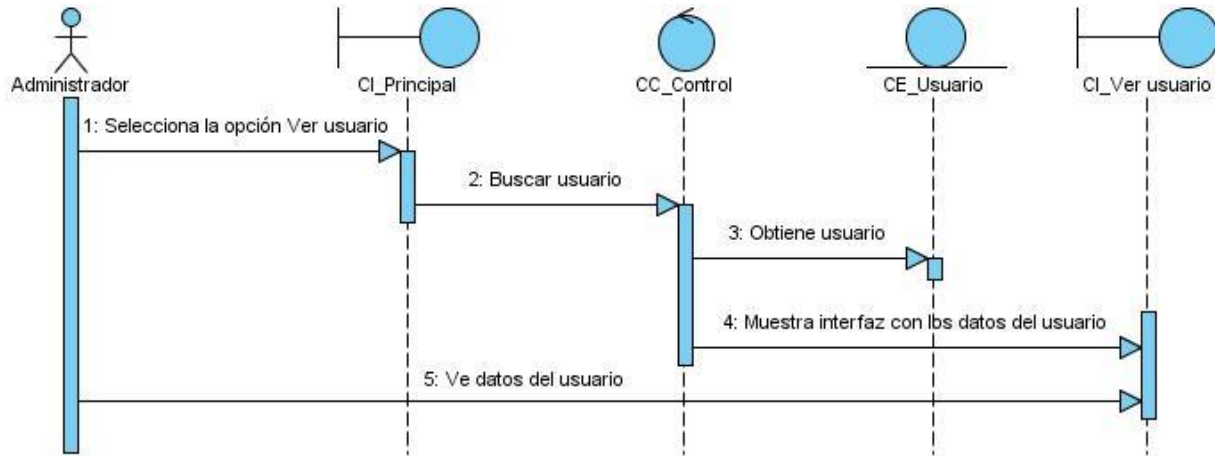


Figura 34: Diagrama de Secuencia del Caso de Uso Gestionar Usuario Sección Ver Usuario

Anexo 58: Diagrama de Secuencia del Caso de Uso Gestionar Usuario Sección Abrir/Cerrar Evento.

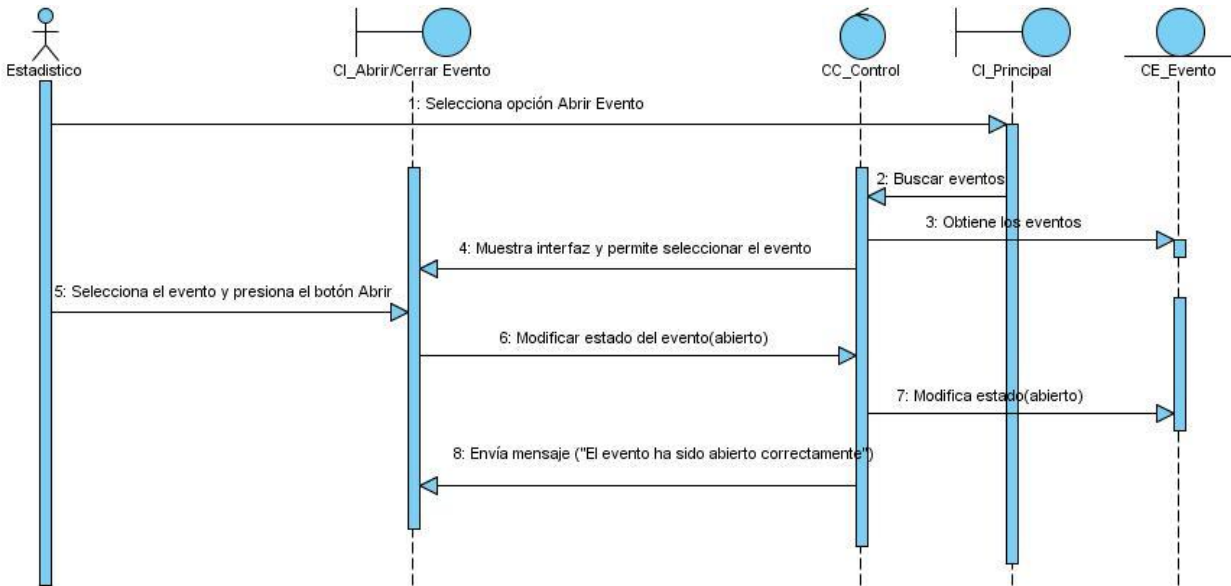


Figura 35: Diagrama de Secuencia del Caso de Uso Gestionar Evento Sección Abrir/Cerrar Evento.

Anexo 59: Diagrama de Secuencia del Caso de Uso Gestionar Evento Sección Insertar Evento.

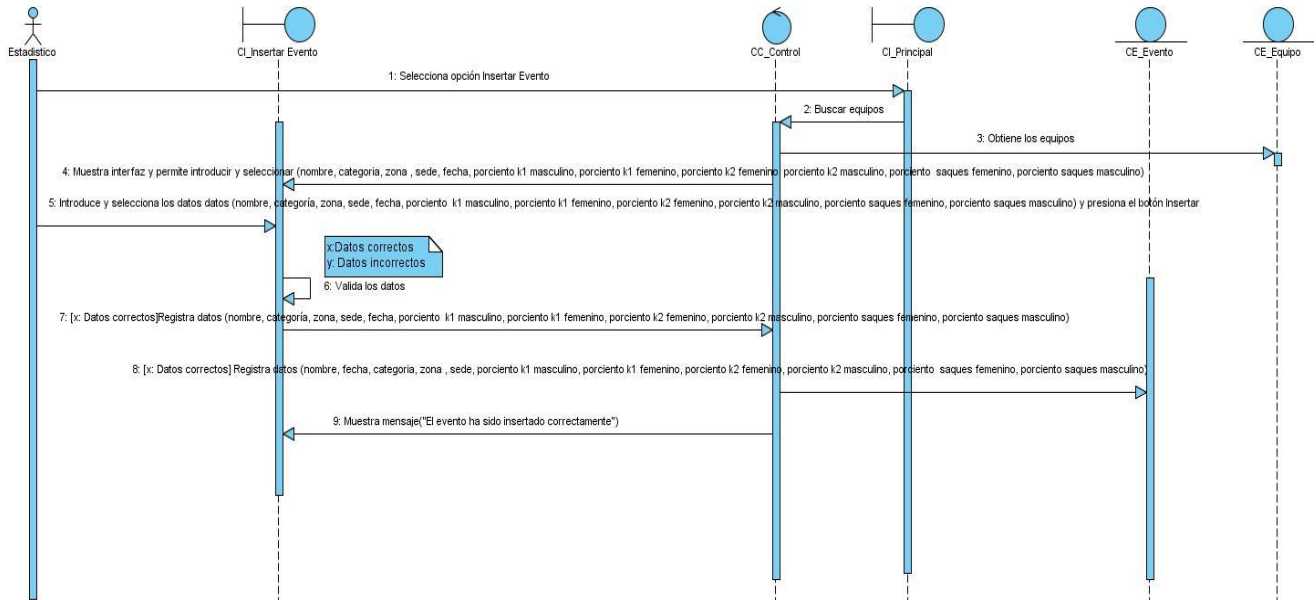


Figura 36: Diagrama de Secuencia del Caso de Uso Gestionar Evento Sección Insertar Evento.

Anexo 60: Diagrama de Secuencia del Caso de Uso Gestionar Evento Sección Modificar Evento.

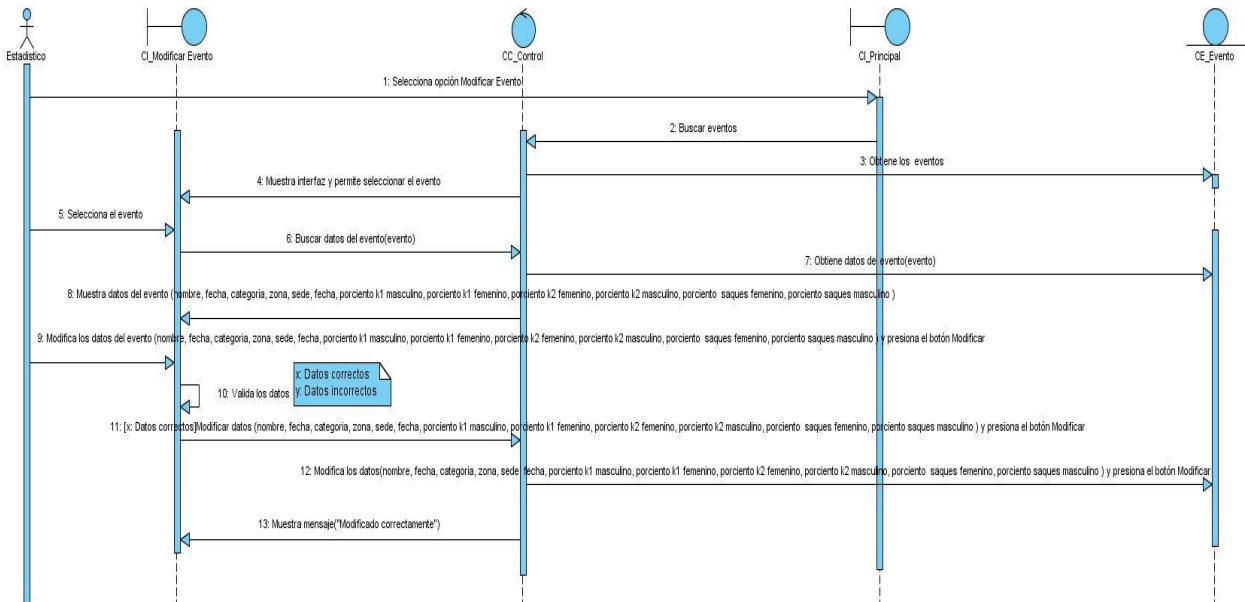


Figura 37: Diagrama de Secuencia del Caso de Uso Gestionar Evento Sección Modificar Evento

Anexo 61: Diagrama de Secuencia del Caso de Uso Gestionar Evento Sección Ver Evento.

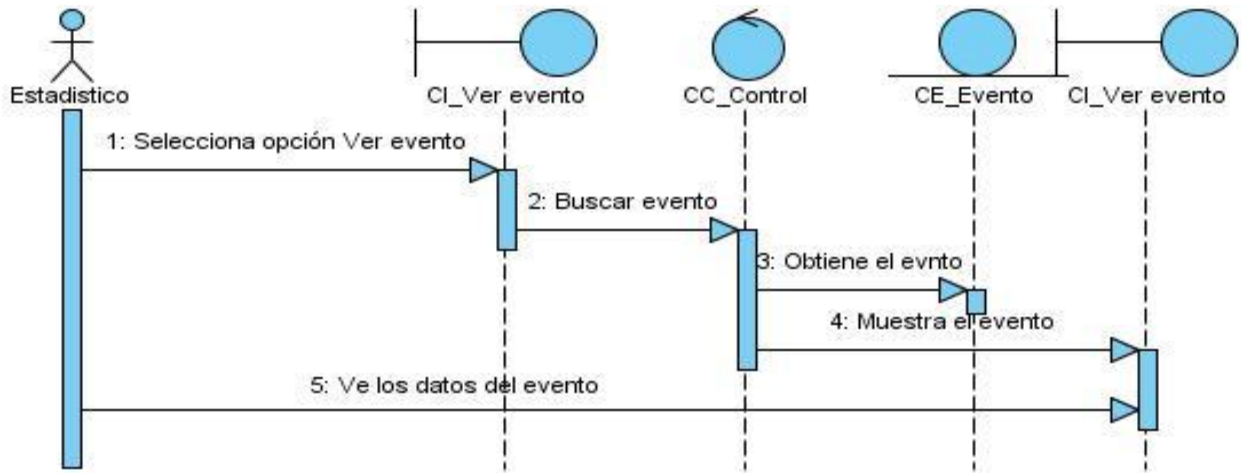


Figura 38: Diagrama de Secuencia del Caso de Uso Gestionar Evento Sección Ver Evento

Anexo 62: Diagrama de Secuencia del Caso de Uso Gestionar Jugador Sección Insertar Jugador.

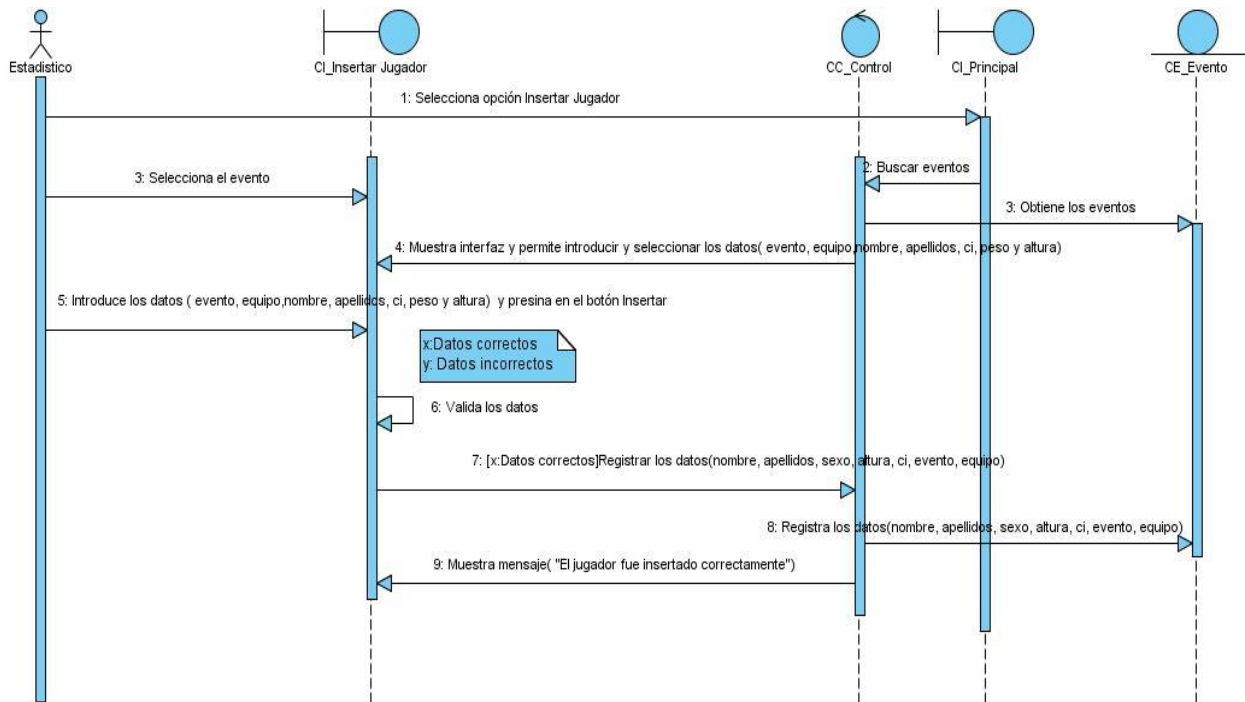


Figura 39: Diagrama de Secuencia del Caso de Uso Gestionar Jugador Sección Insertar Jugador

Anexo 63: Diagrama de Secuencia del Caso de Uso Gestionar Jugador Sección Modificar Jugador.

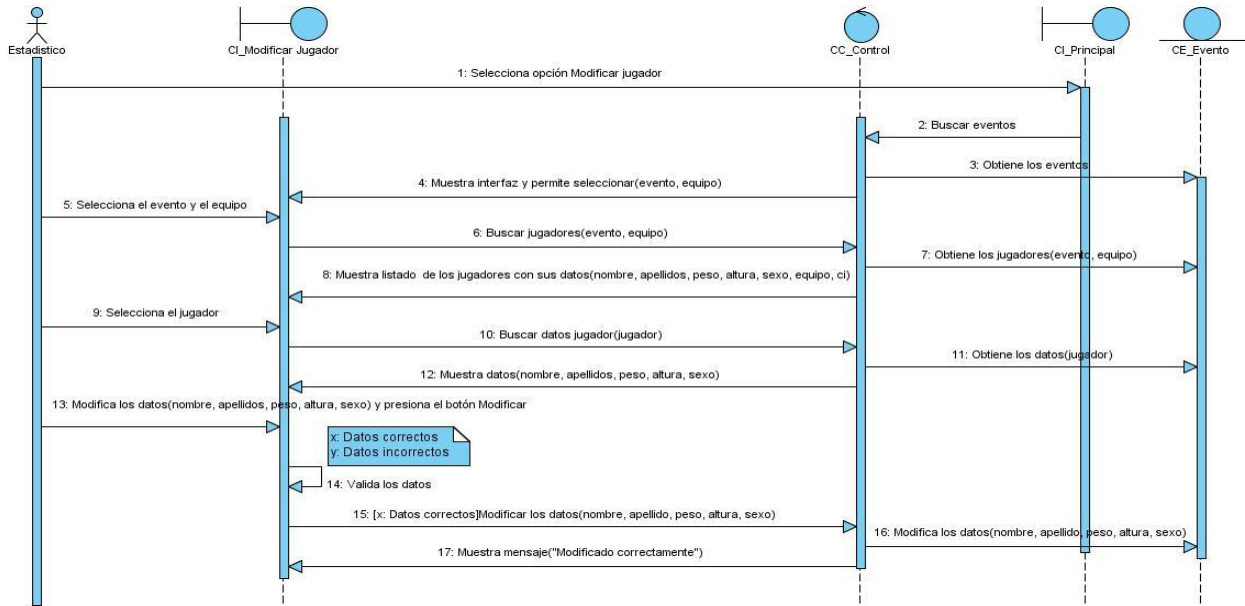


Figura 40: Diagrama de Secuencia del Caso de Uso Gestionar Jugador Sección Modificar Jugador

Anexo 64: Diagrama de Secuencia del Caso de Uso Gestionar Jugador Sección Eliminar Jugador.

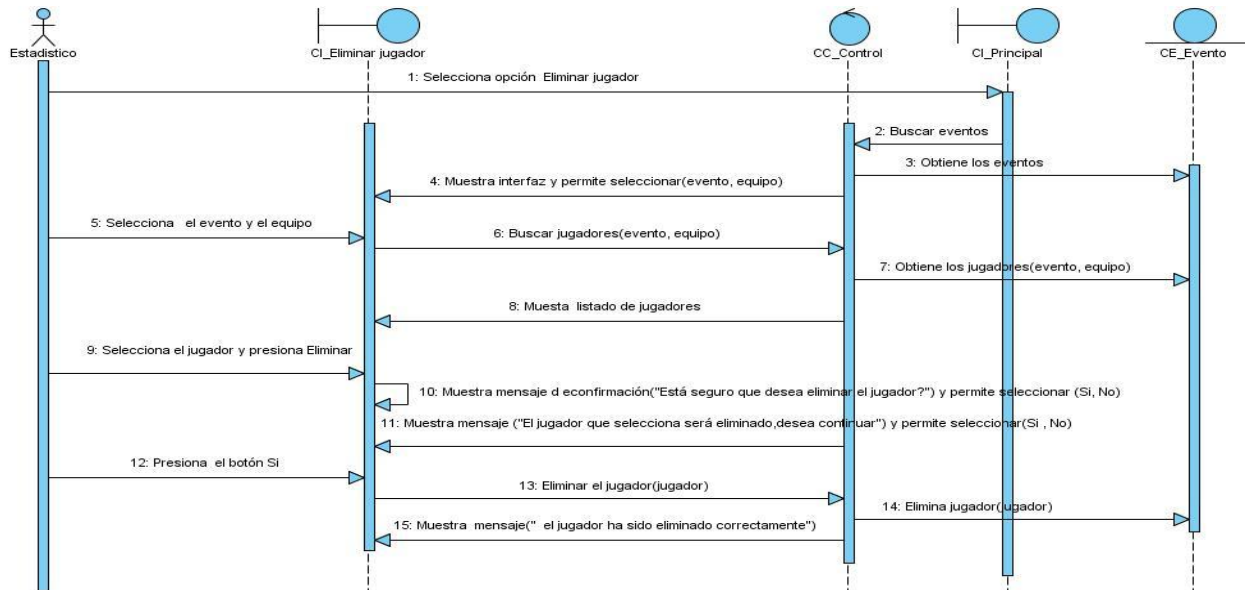


Figura 41: Diagrama de Secuencia del Caso de Uso Gestionar Jugador Sección Eliminar Jugador

Anexo 65: Diagrama de Secuencia del Caso de Uso Gestionar Jugador Sección Ver Jugador.

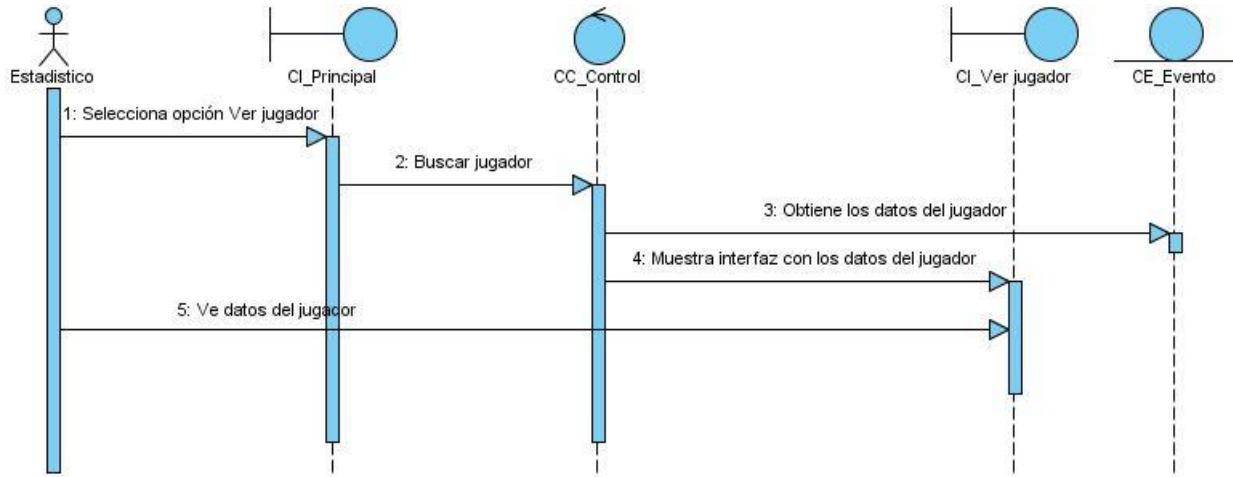


Figura 42: Diagrama de Secuencia del Caso de Uso Gestionar Jugador Sección Ver Jugador

Anexo 66: Diagrama de Secuencia del Caso de Uso Consultar Reporte Boletín Diario.

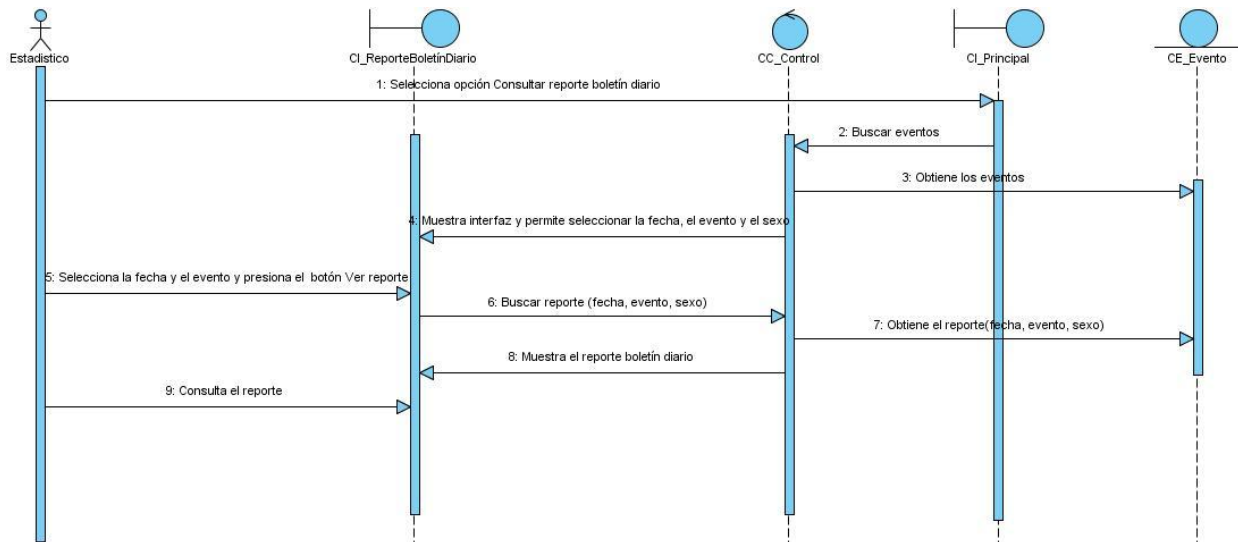


Figura 43: Diagrama de Secuencia del Caso de Uso Consultar Reporte Boletín Diario

Anexo 67: Diagrama de Secuencia del Caso de uso Consultar Reporte Resultados de los Objetivos Pedagógicos.

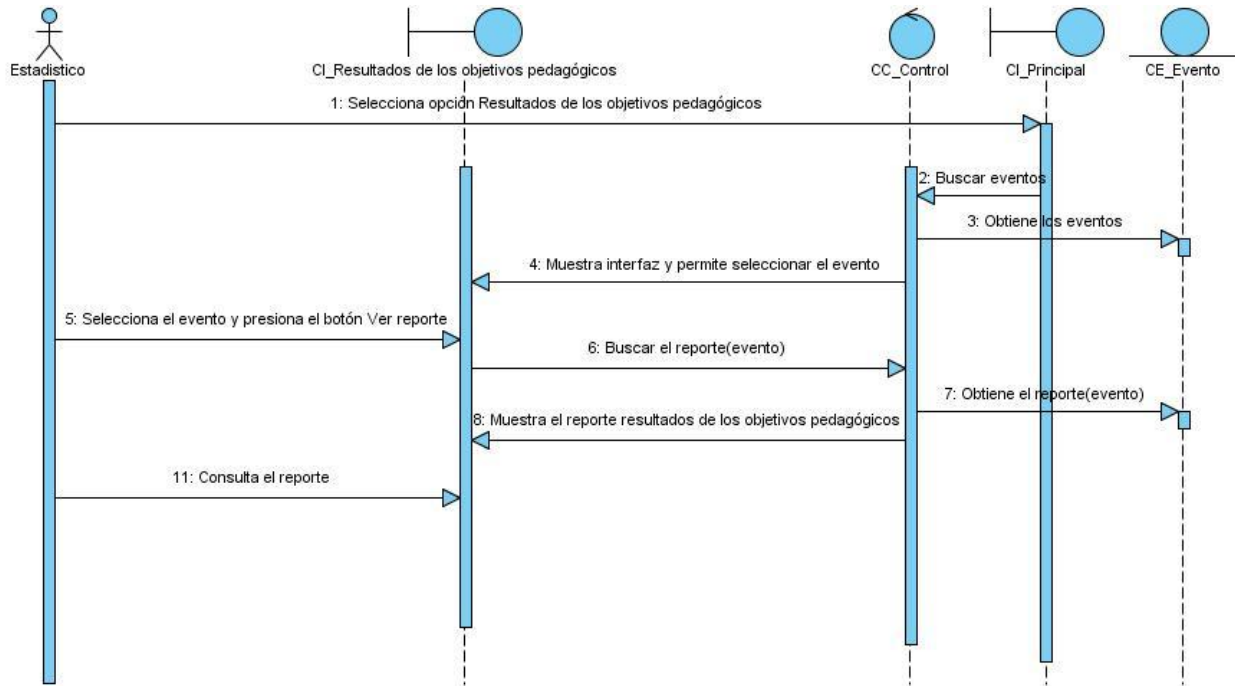


Figura 44: Diagrama de Secuencia del Caso de Uso Consultar Reporte Resultados de los Objetivos Pedagógicos

Anexo 68: Diagrama de Secuencia del Caso de Uso Consultar Reporte Control Competitivo.

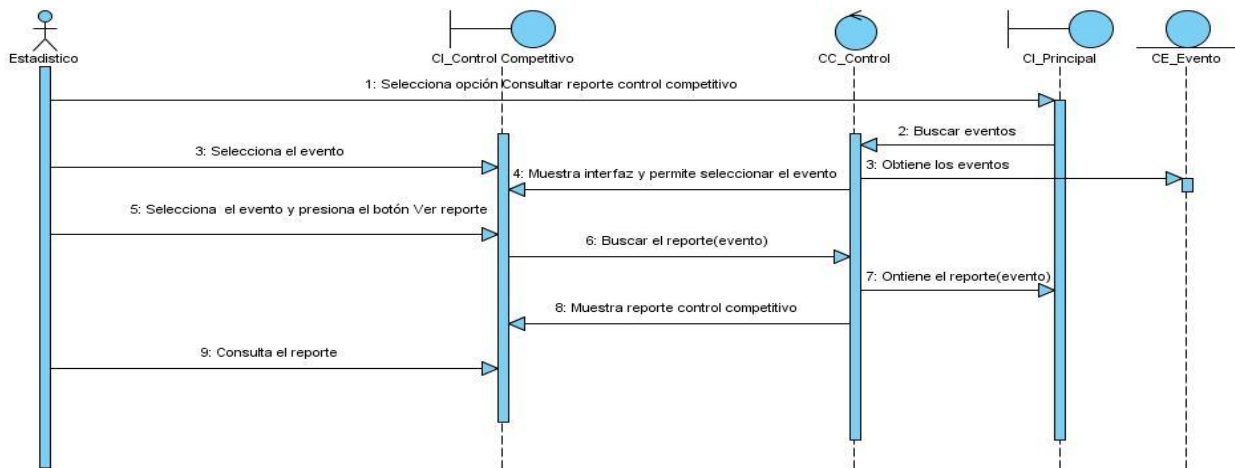


Figura 45: Diagrama de Secuencia del Caso de Uso Consultar Reporte Control Competitivo

Anexo 69: Diagrama de Secuencia del Caso de Uso Consultar Reporte Historial de Usuario.

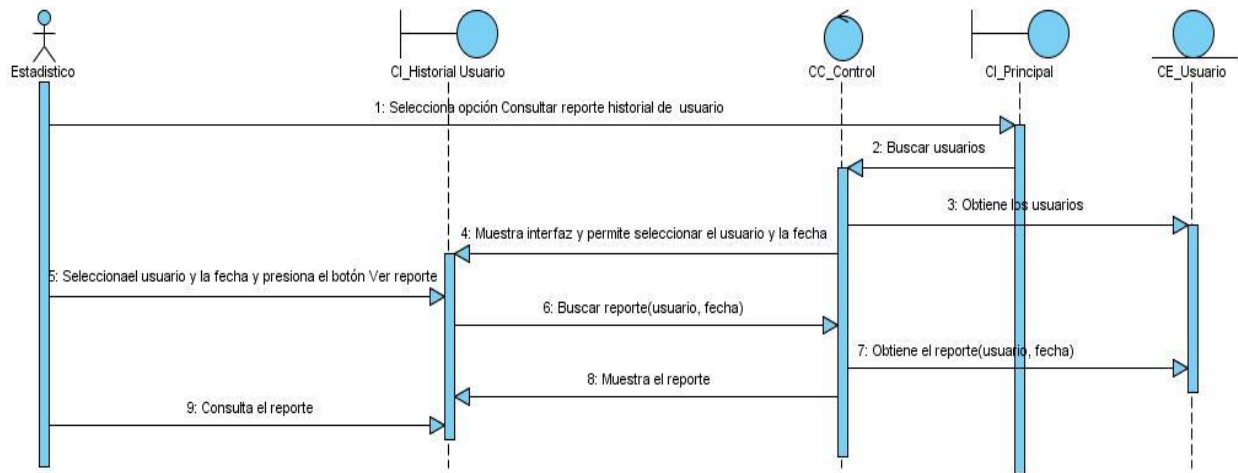


Figura 46: Diagrama de Secuencia del Caso de Uso Consultar Reporte Historial de Usuario

Anexo 70: Diagrama de Secuencia del Caso de Uso Consultar Reporte Resumen Estadístico de cada Equipo.

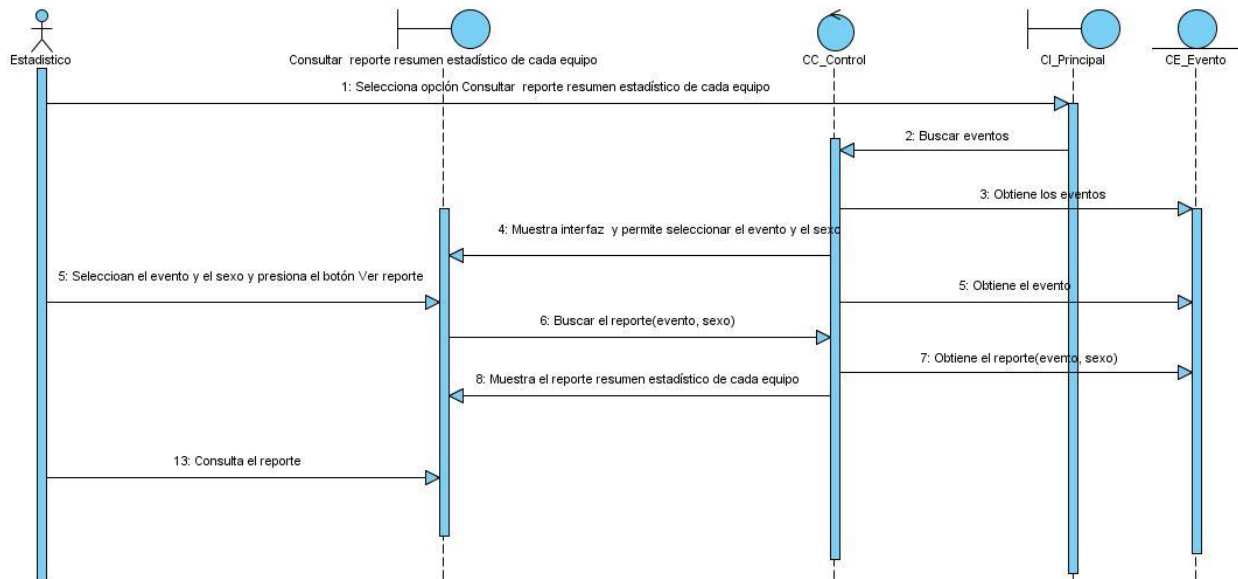


Figura 47: Diagrama de Secuencia del Caso de Uso Consultar Reporte Resumen Estadístico de cada Equipo

Anexo 71: Diagrama de Secuencia del Caso de Uso Consultar Reporte Equipo Ideal.

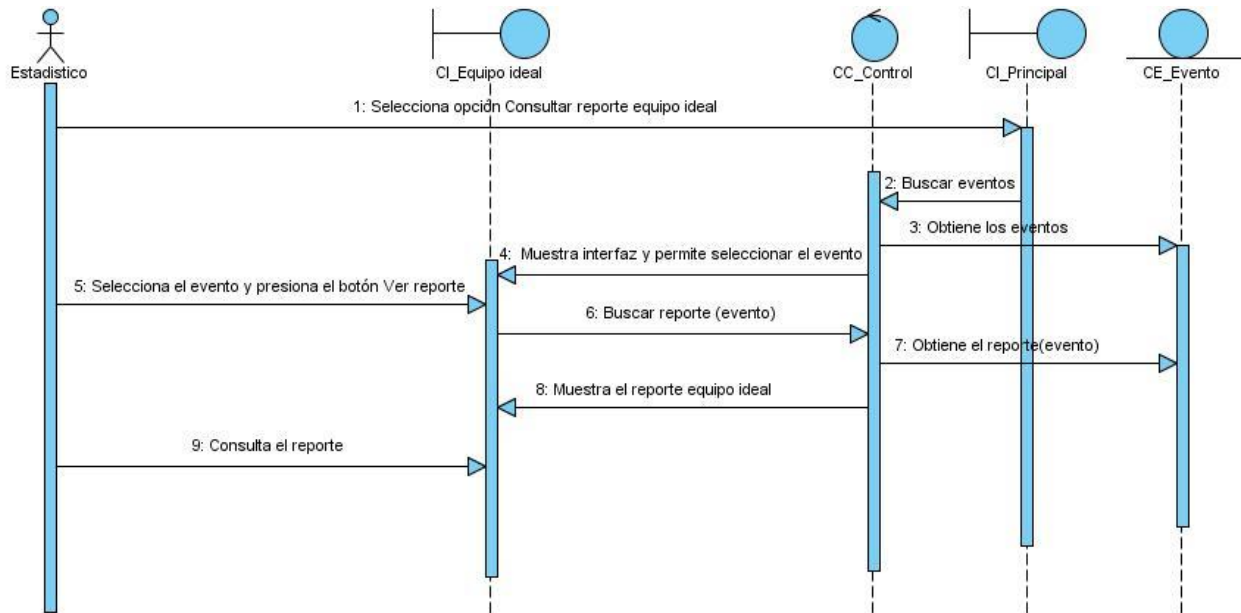


Figura 48: Diagrama de Secuencia del Caso de Uso Consultar Reporte Equipo Ideal

Anexo 72: Diagrama de Secuencia del Caso de Uso Consultar Reporte Resultados Alcanzados por las Provincias.

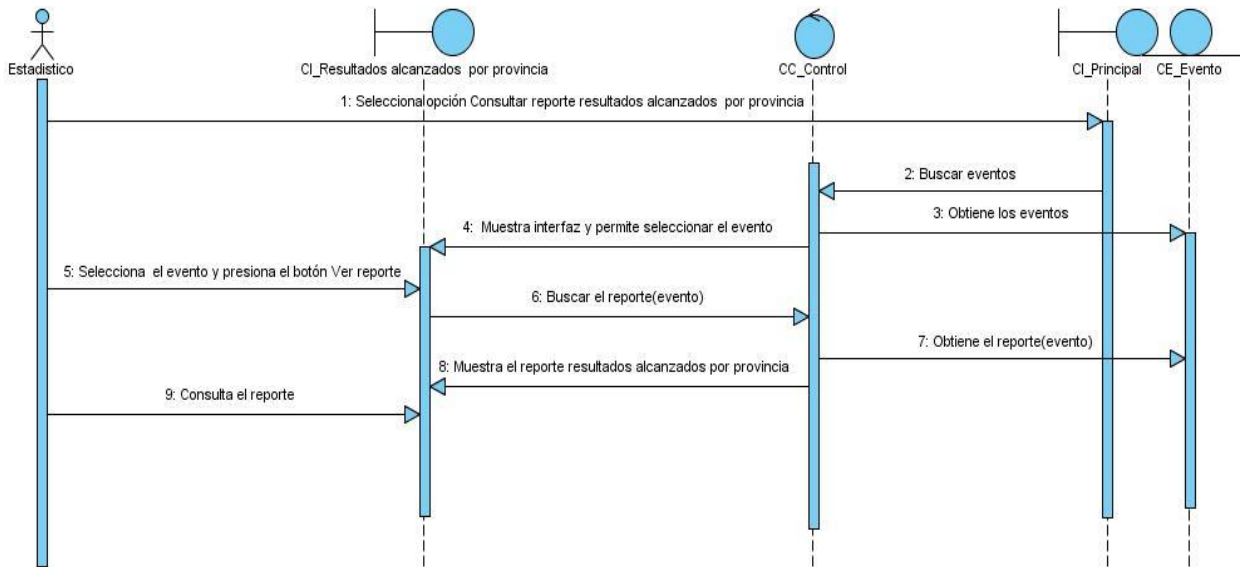


Figura 49: Diagrama de Secuencia del Caso de Uso Consultar Reporte Resultados Alcanzados por las Provincias

Anexo 73: Diagrama de Secuencia del Caso de Uso Consultar Reporte Resultados por Provincias.

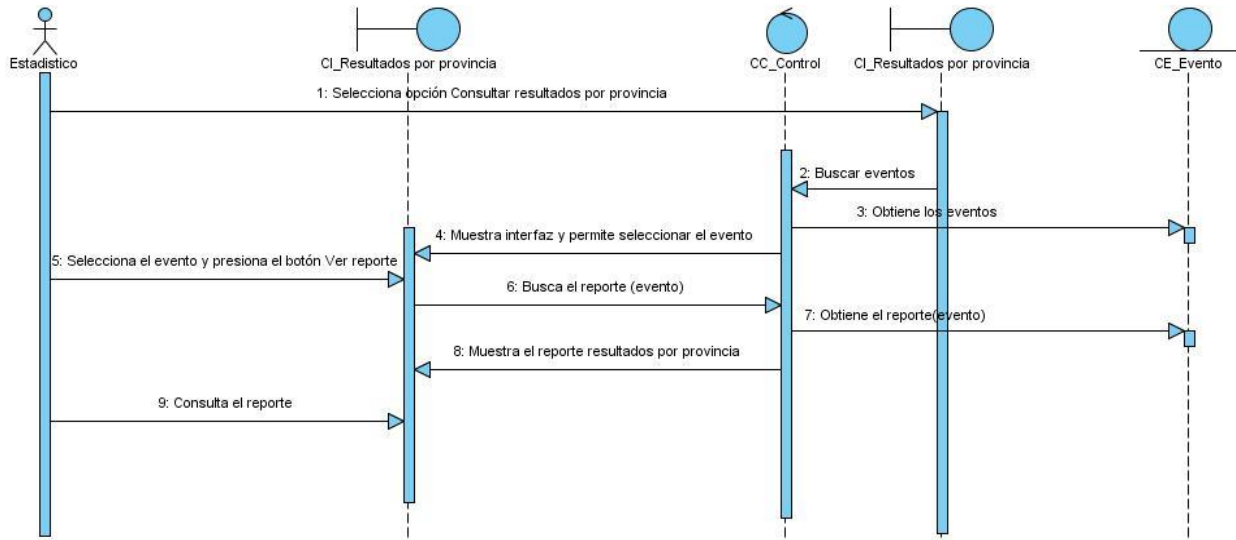


Figura 50: Diagrama de Secuencia del Caso de Uso Consultar Reporte Resultados por Provincias

Anexo 74: Diagrama de Secuencia del Caso de Uso Consultar Evento.

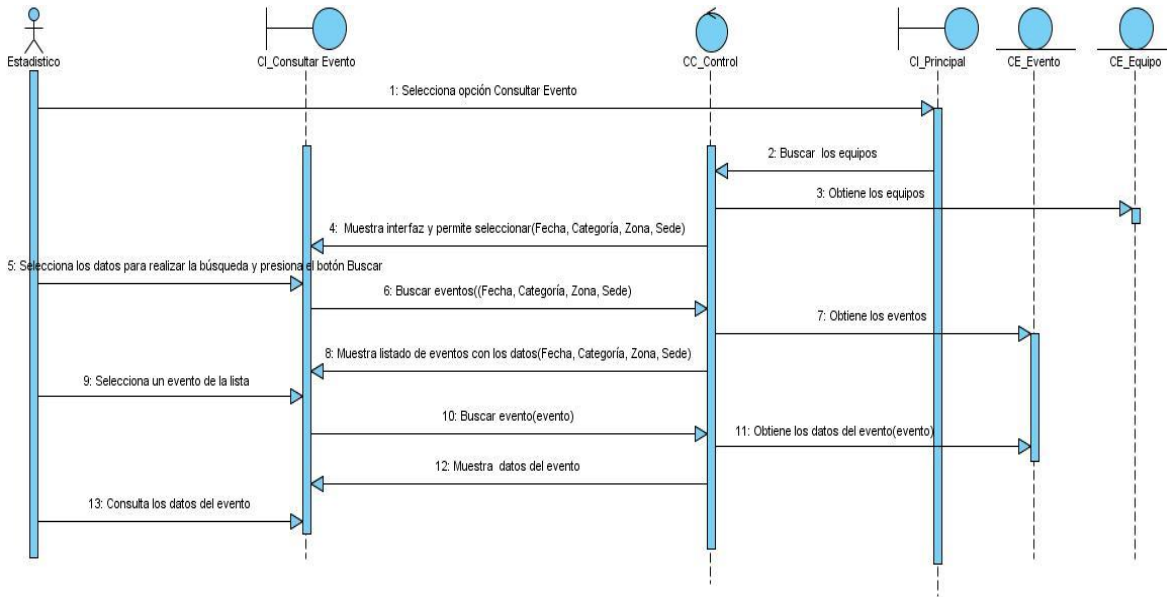


Figura 51: Diagrama de Secuencia del Caso de Uso Consultar Evento

Anexo 75: Diagrama de Secuencia del Caso de Uso Consultar Juego.

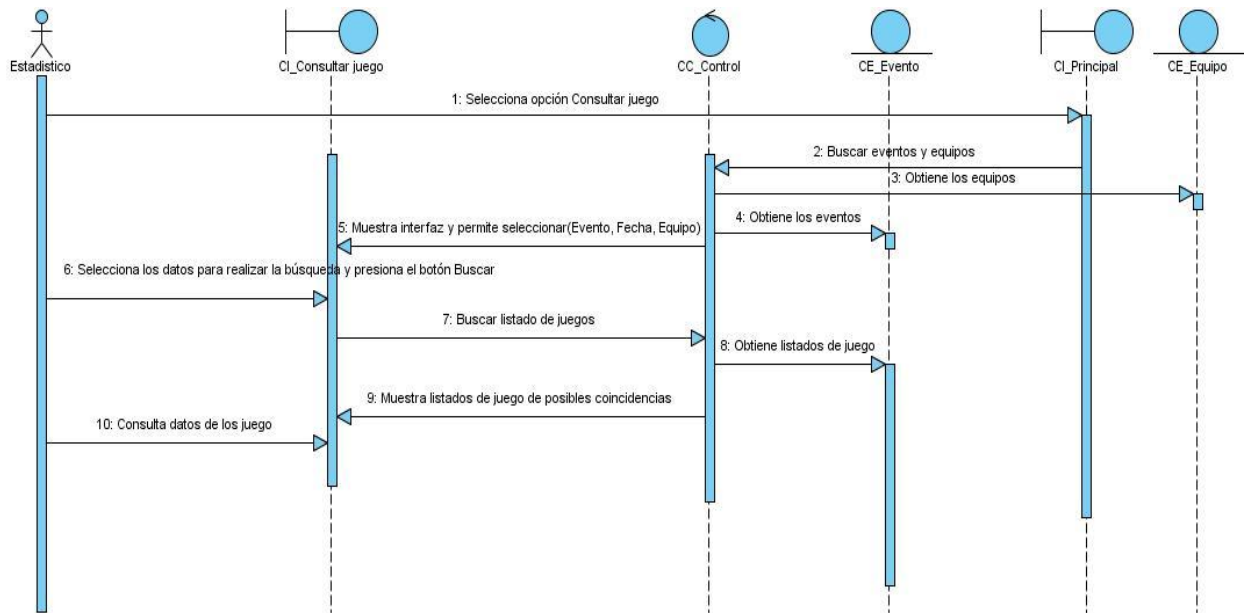


Figura 52: Diagrama de Secuencia del Caso de Uso Consultar Juego

Anexo 76: Diagrama de Secuencia del Caso de Uso Consultar Jugador.

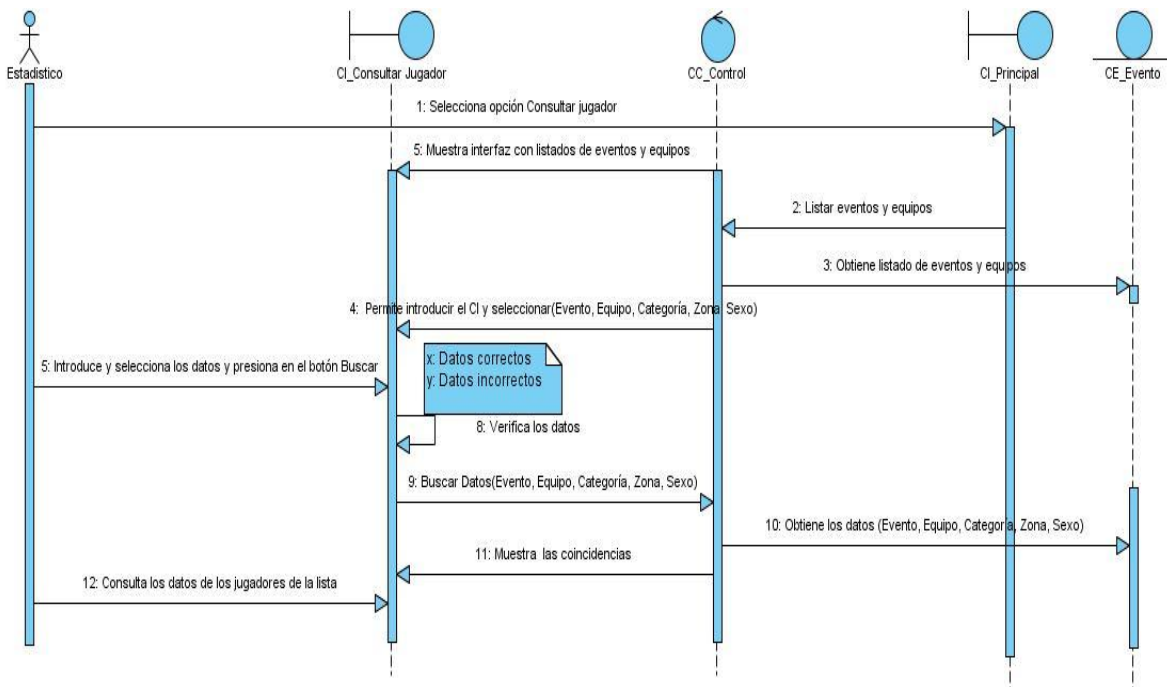


Figura 53: Diagrama de Secuencia del Caso de Uso Consultar Jugador

Anexo 77: Diagrama de Secuencia del Caso de Uso Consultar Usuario.

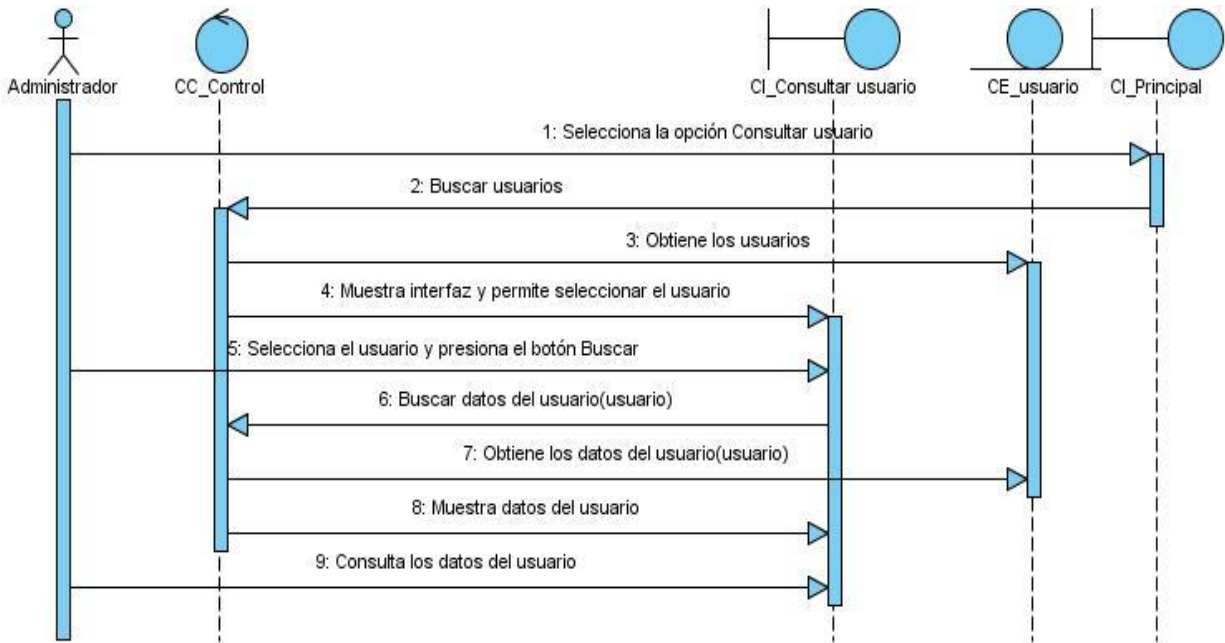


Figura 54: Diagrama de Secuencia del Caso de Uso Consultar Usuario

Anexo 78: Diagrama de Clases del Diseño del Caso de Uso Gestionar Evento.

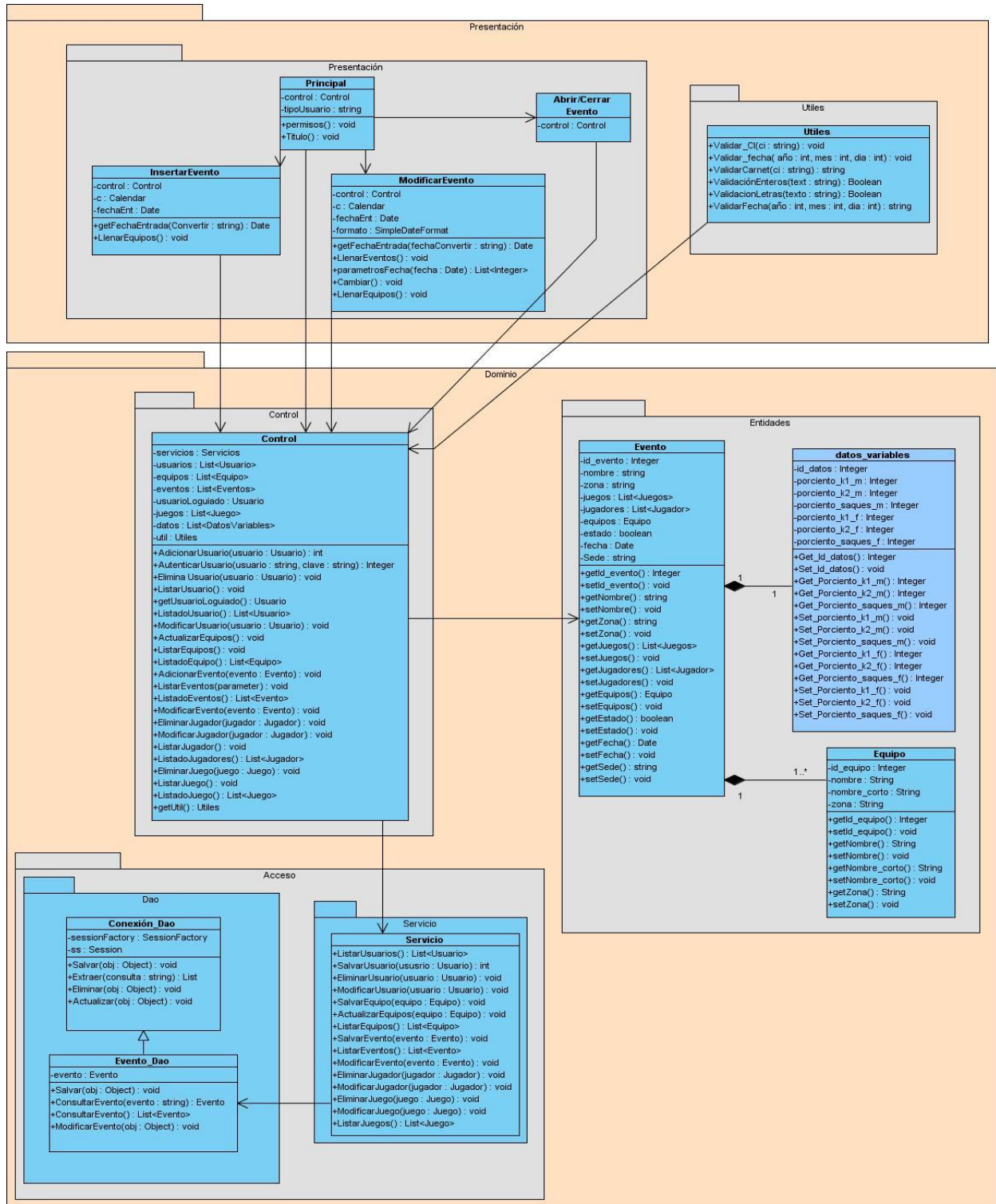


Figura 55: Diagrama de Clases del Diseño del Caso de Uso Gestionar Evento

Anexo 79: Diagrama de Clases del Diseño del Caso de Uso Gestionar Jugador.

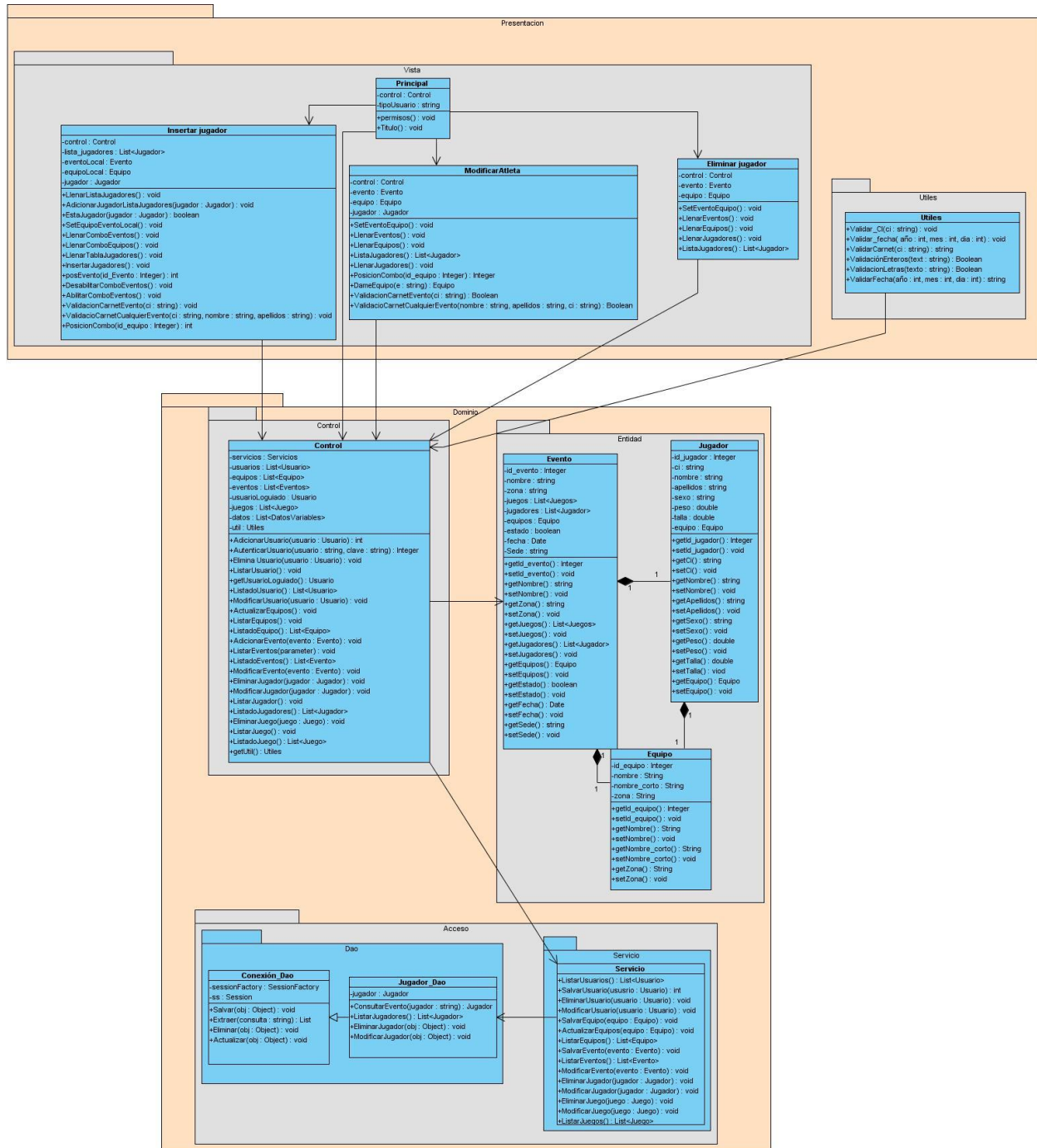


Figura 56: Diagrama de Clases del Diseño del Caso de Uso Gestionar Jugador

Anexo 80: Diagrama de Clases del Diseño del Caso de Uso Gestionar Usuario y Autenticar Usuario.

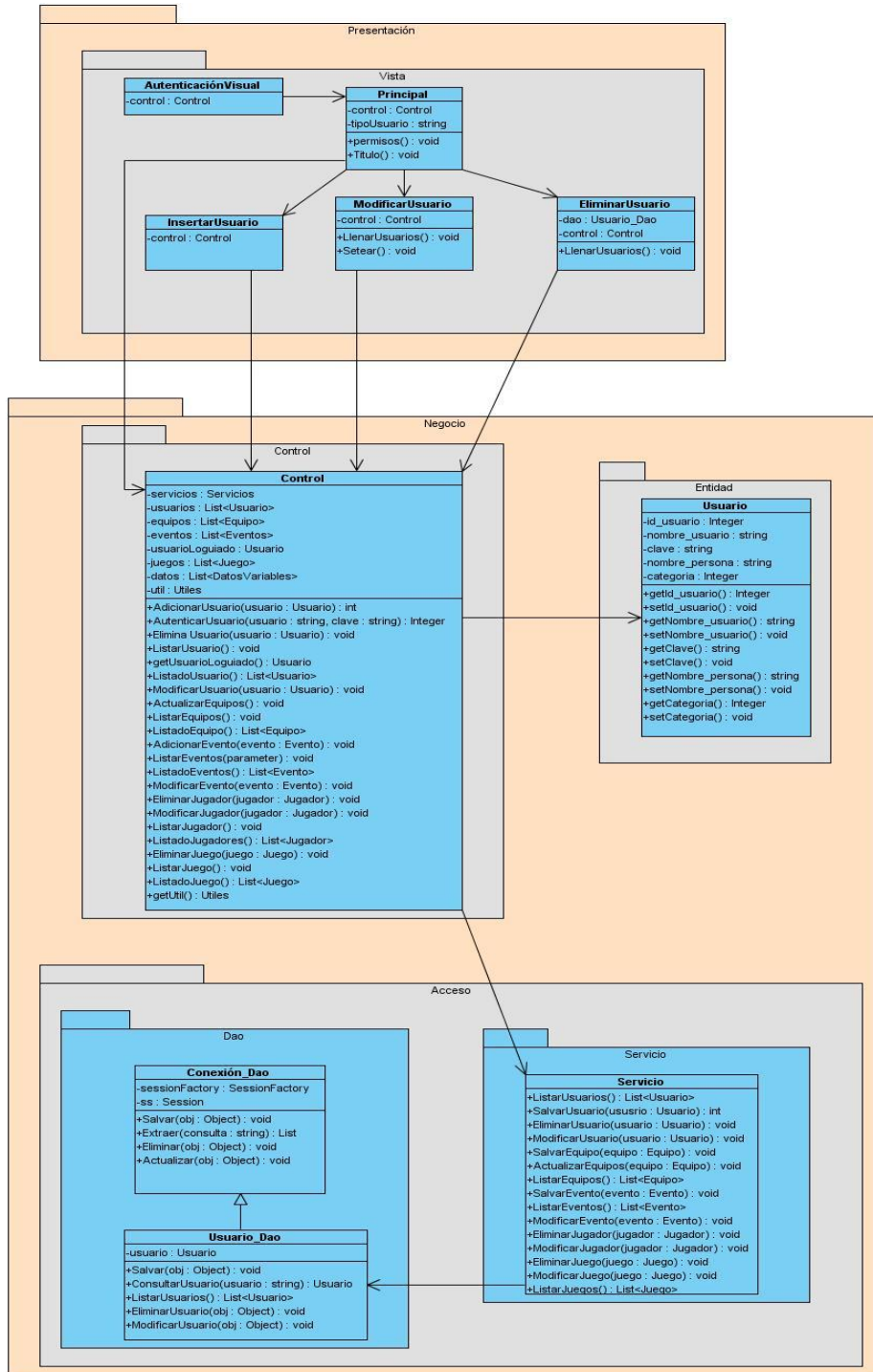


Figura 57: Diagrama de Clases del Diseño del Caso de Uso Gestionar Usuario y Autenticar Usuario

Anexo 81: Diagrama de Clases del Diseño de los Casos de Uso Consultar Reportes.

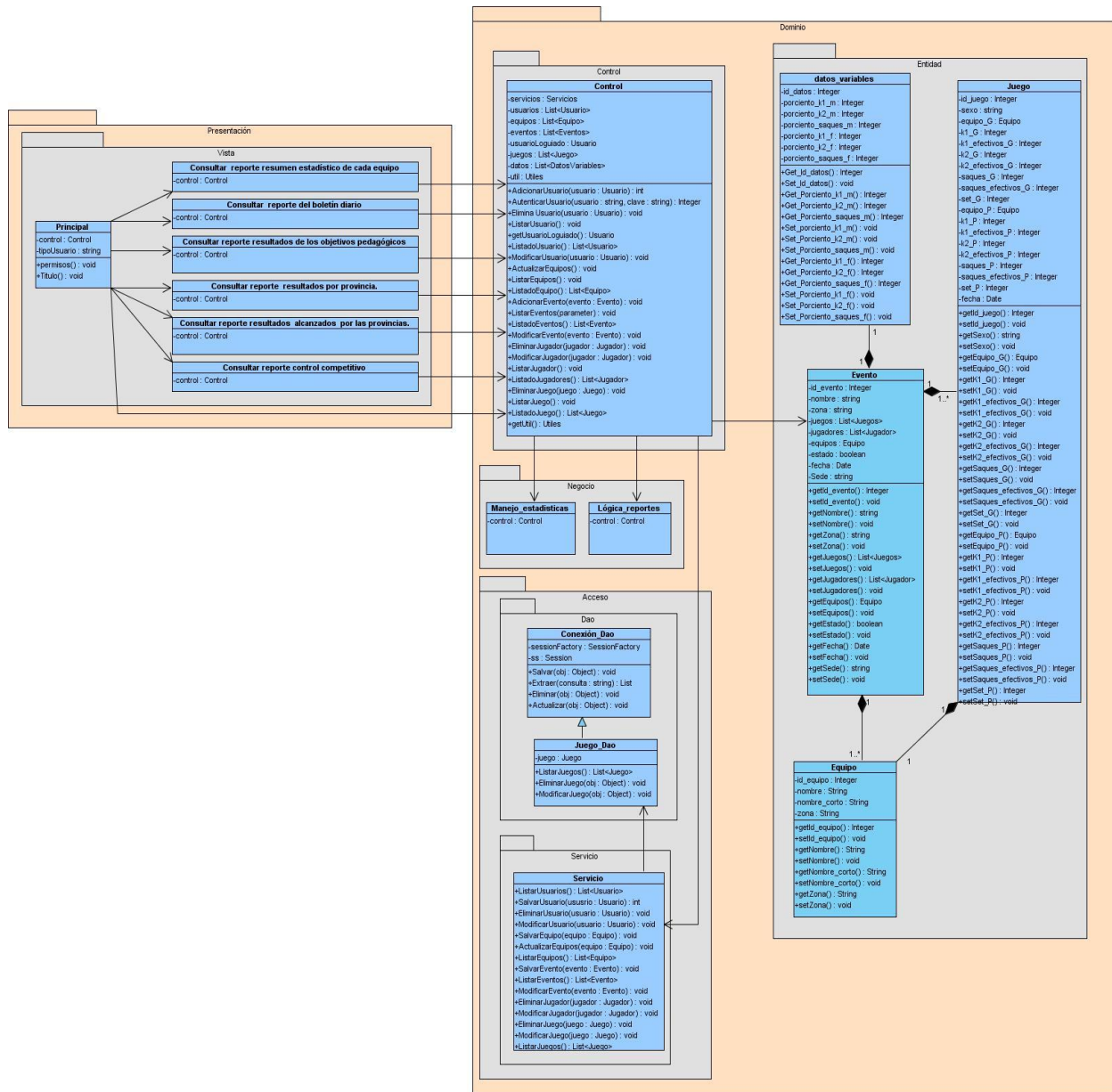


Figura 58: Diagrama de Clases del Diseño del Caso de Uso Consultar Reporte Boletín Diario, Consultar Reporte Resumen Estadístico de cada Equipo, Consultar Reporte Resultados por Provincia, Consultar Reporte Resultados Alcanzados por las Provincias, Consultar Reporte Control Competitivo, Consultar Reporte Resultados de los Objetivos Pedagógicos.

Anexo 82: Diagrama de Clases del Diseño del Caso de Uso Consultar Reporte Equipo Ideal.

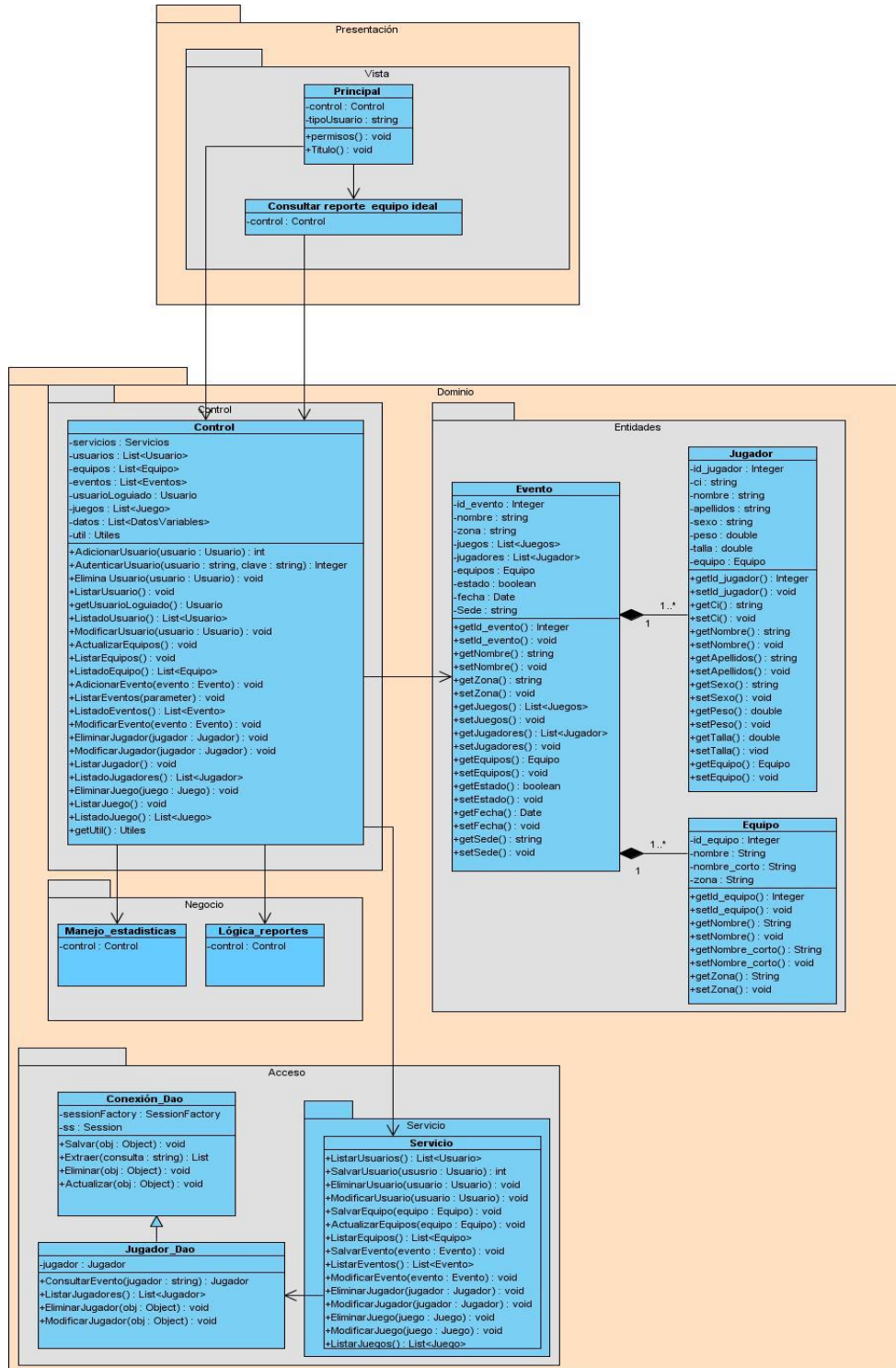


Figura 59: Diagrama de Clases del Diseño del Caso de Uso Consultar Reporte Equipo Ideal

Anexo 83: Diagrama de Clases del Diseño del Caso de Uso Consultar Reporte Historial de Usuario.

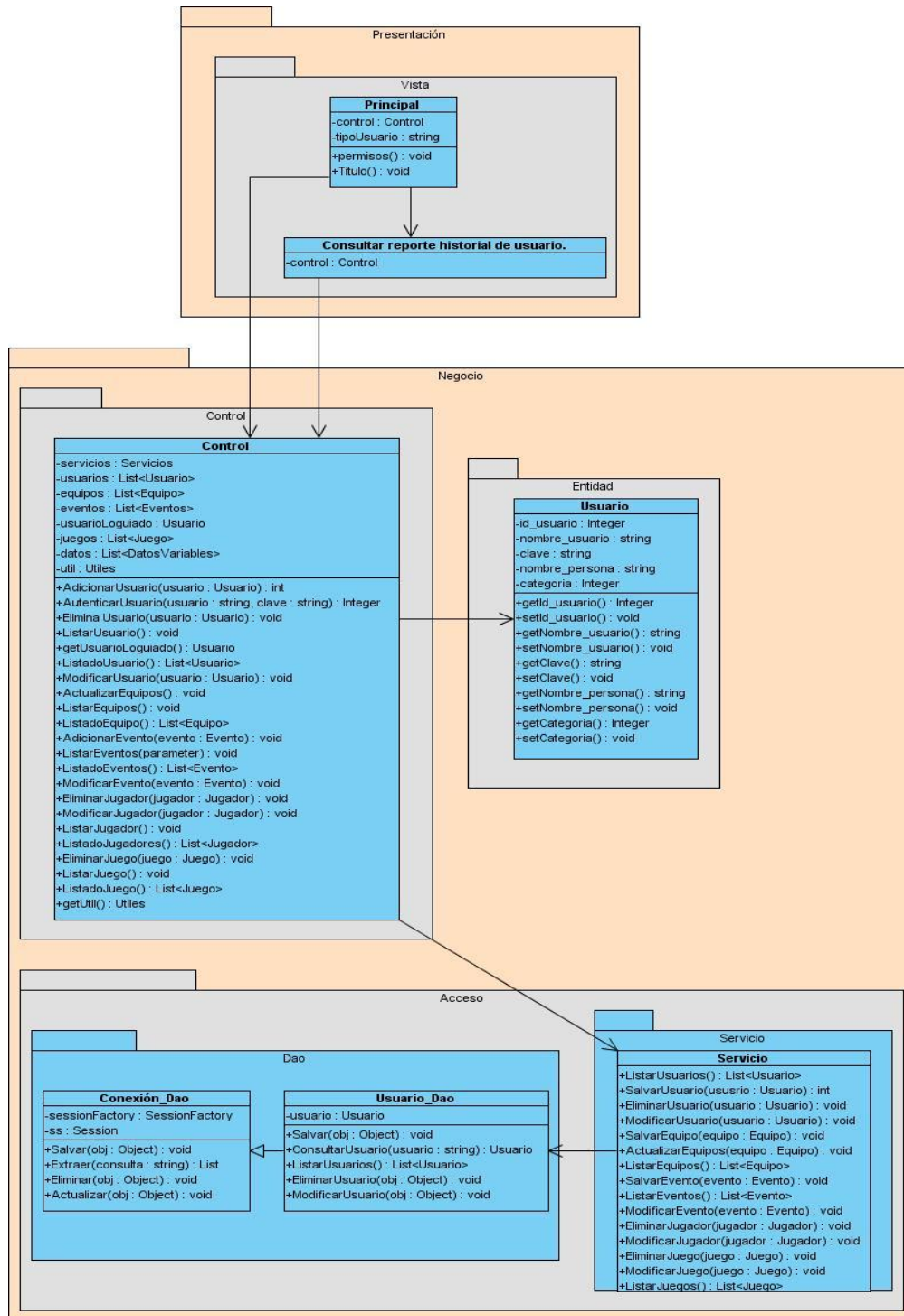


Figura 60: Diagrama de Clases del Diseño del Caso de Uso Consultar Reporte Historial de Usuario.

Anexo 84: Diagrama de Clases del Diseño del Caso de Uso Consultar Jugador.

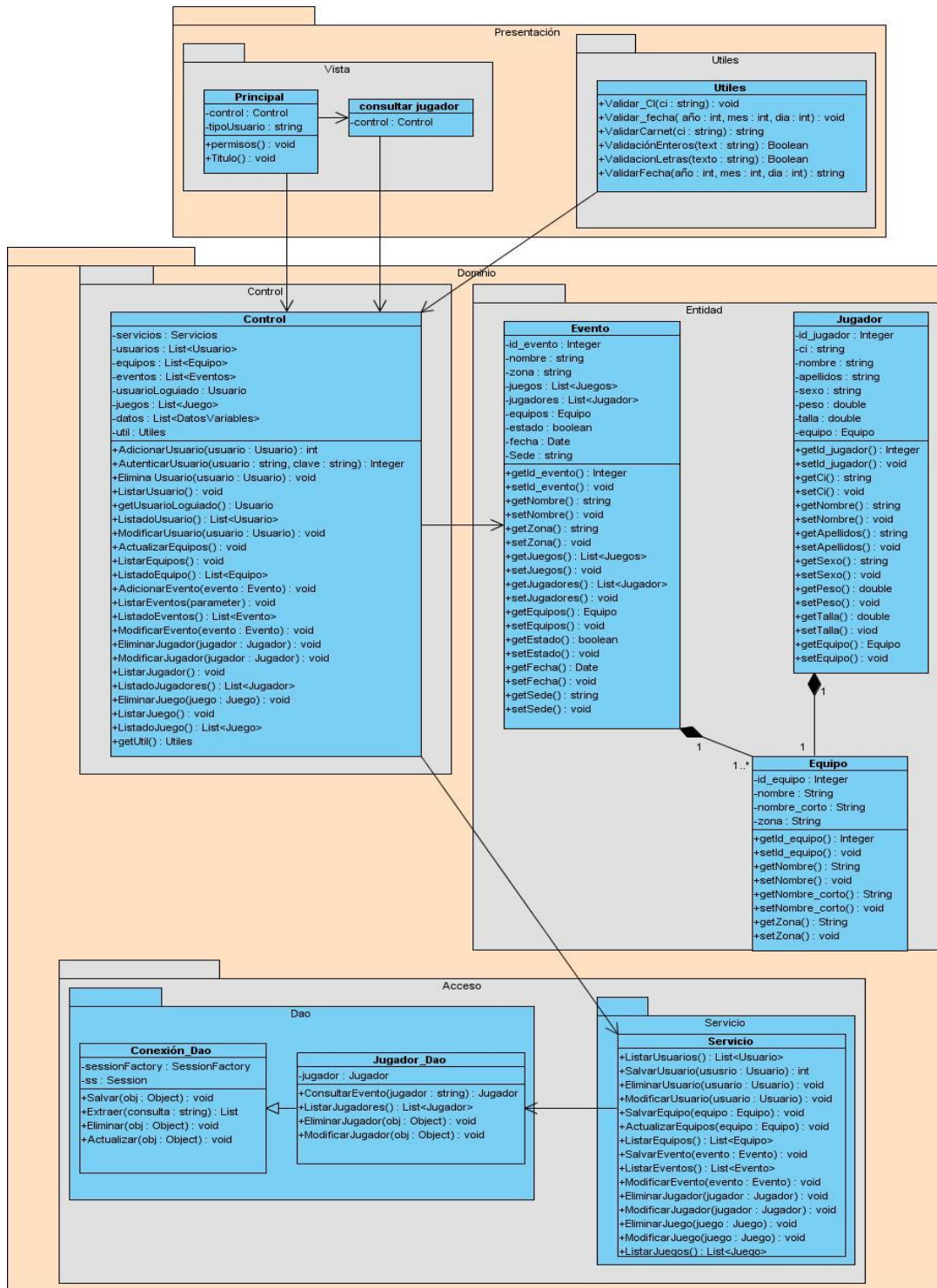


Figura 61: Diagrama de Clases del Diseño del Caso de Uso Consultar Jugador

Anexo 85: Diagrama de Clases del Diseño del Caso de Uso Consultar Juego.

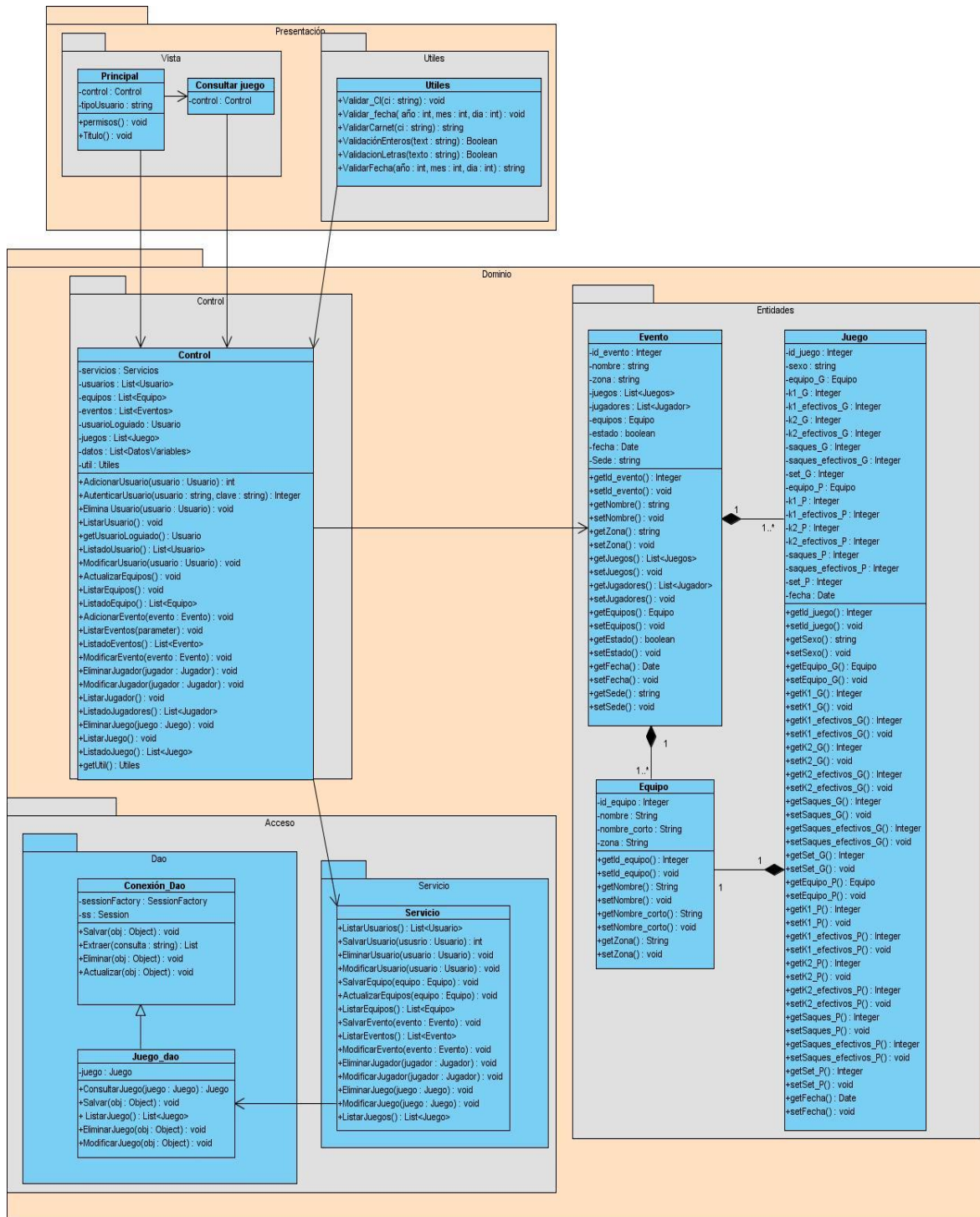


Figura 62: Diagrama de Clases del Diseño del Caso de Uso Consultar Juego

Anexo 86: Diagrama de Clases del Diseño del Caso de Uso Consultar Evento.

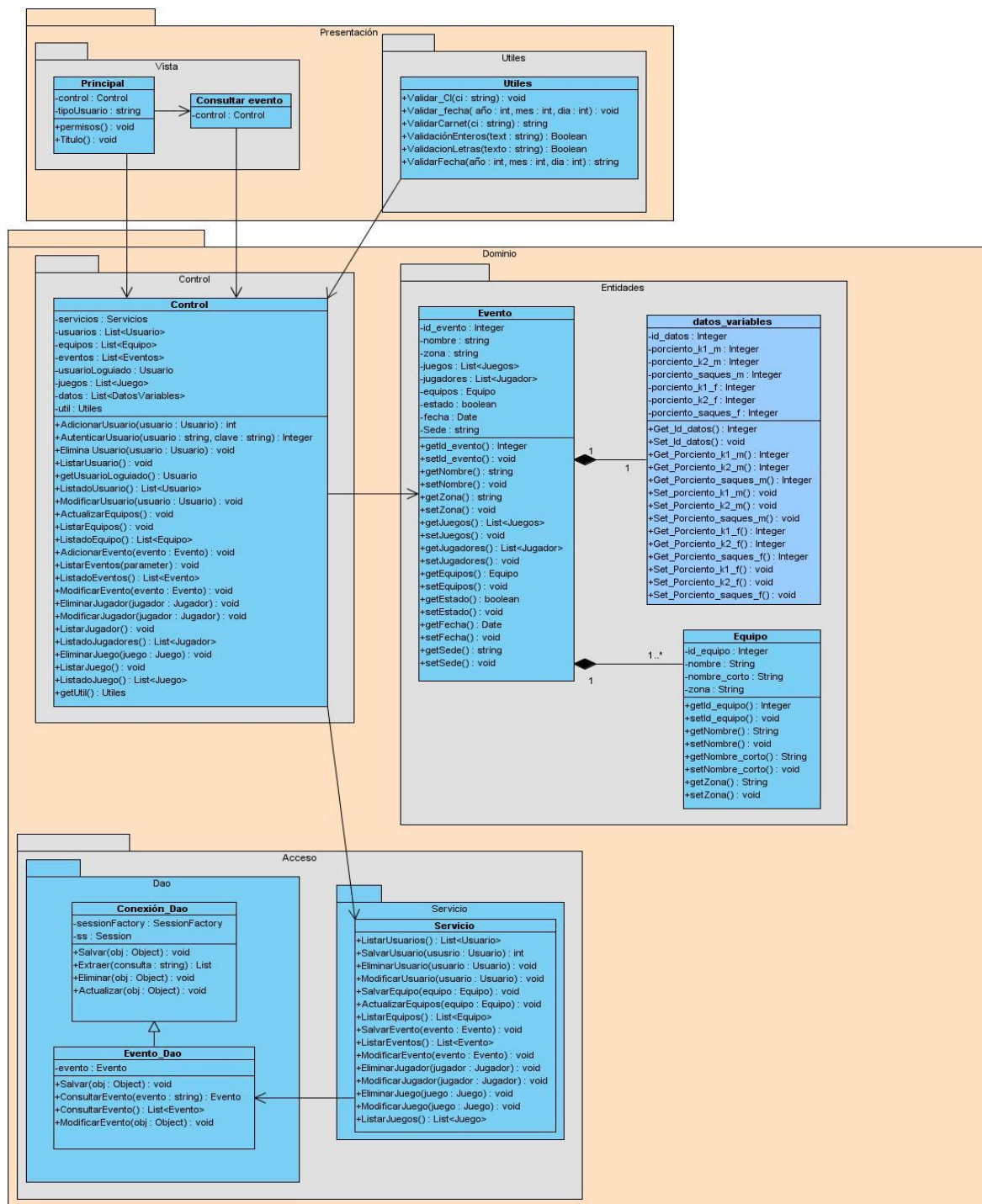


Figura 63: Diagrama de Clases del Diseño del Caso de Uso Consultar Evento

Anexo 87: Diagrama de Clases del Diseño del Caso de Uso Consultar Usuario.

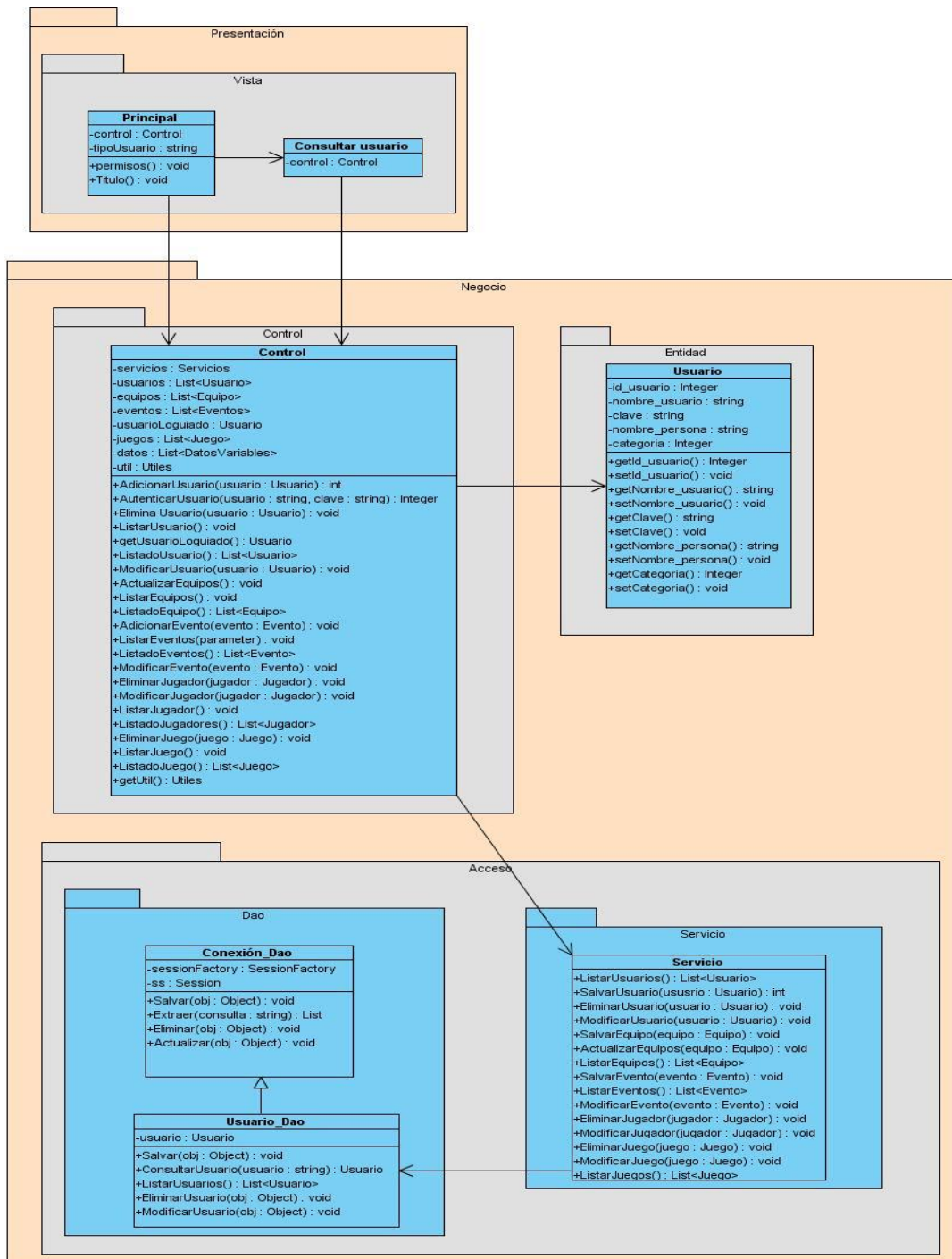


Figura 64: Diagrama de Clases del Diseño del Caso de Uso Consultar Usuario