



*Universidad de las Ciencias Informáticas  
Facultad 4*

*Módulos “Plan gráfico” y “Test pedagógico” del Sistema para  
la planificación y control del entrenamiento en deportes de  
combate*



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

***Autores:***

*Maday Cristina Larduet Pedroso  
Frank Ernesto Morell Fuentes*

***Tutores:***

*Ing. Karenia Donatien Goliath  
Ing. Roanny Lamas López*

***Co-Tutor:***

*MSc. Esteban Joel Navarro Mora*

*Ciudad de La Habana, Junio 2015  
“Año 57 de la Revolución”*

## Declaración de autoría

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

Para que así conste firmo la presente a los \_\_\_\_ días del mes \_\_\_\_\_ del año 2014.

Autor(es):

---

Frank E. Morell Fuentes

---

Maday C. Larduet Pedroso

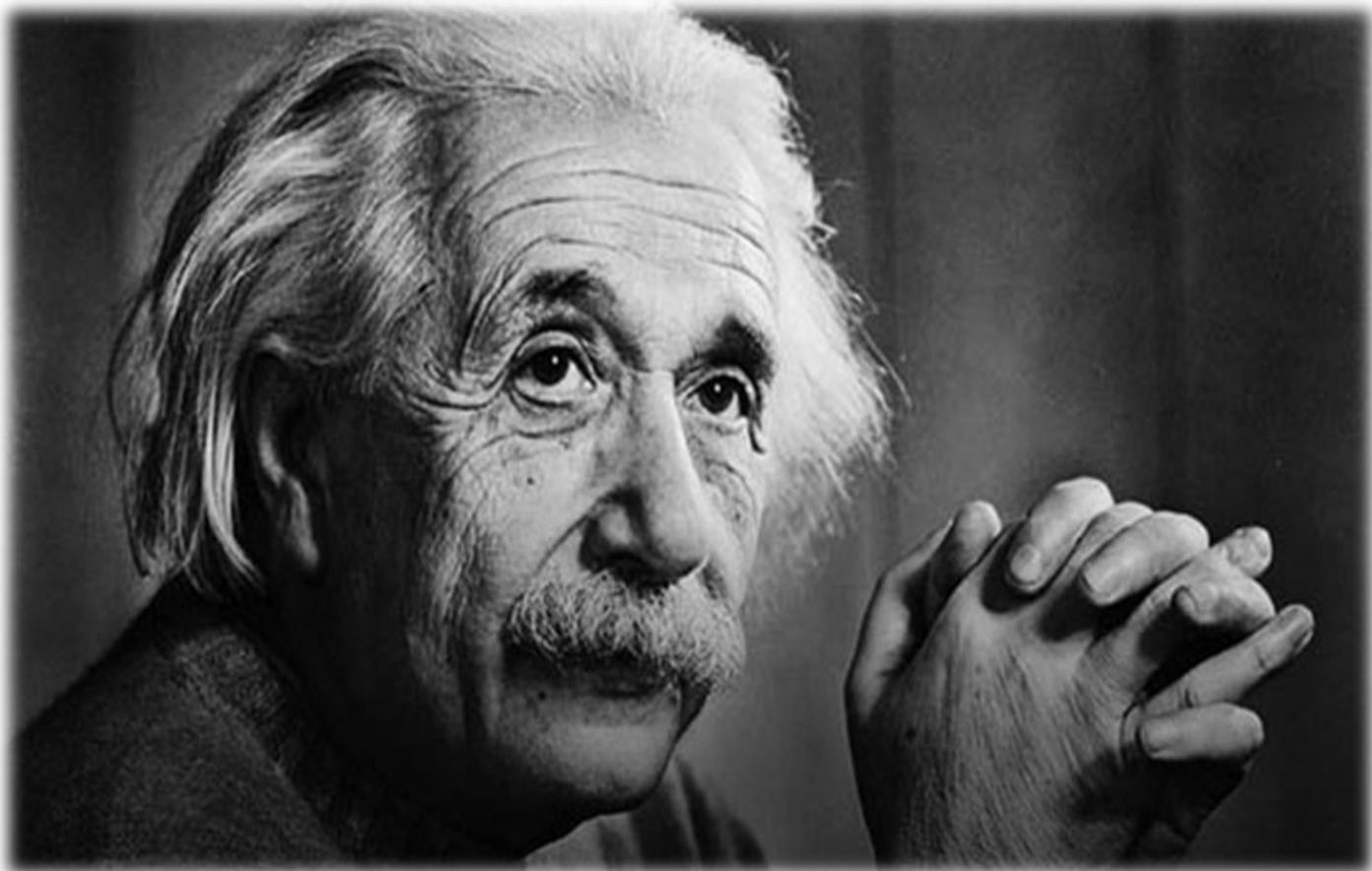
Tutor(es):

---

Ing. Karenia Donatien Goliath

---

Ing. Roanny Lamas López



*“Los grandes espíritus siempre han encontrado violenta oposición de parte de los mediocres. Estos últimos no pueden entender cuando un hombre no sucumbe impensadamente a prejuicios hereditarios sino que, honestamente y con coraje, usa su inteligencia.”*

*Albert Einstein*

*Dedicatorias*

*De Maday Cristina Larduet Pedroso*

*A mi familia por la confianza que han depositado en mí, en especial a mi mamá por darme la vida y por sacrificarse tanto para que hoy yo pueda ser una profesional.*

*De Frank Ernesto Morell Fuentes*

*A mi mamá y mi hermana, por dedicar gran parte de su vida por verme realizar mi sueño de convertirme en ingeniero.*

## *Agradecimientos*

### *De Maday Cristina Larduet Pedroso*

*A mi mamá por ser mi motor impulsor, mi guía, por la dedicación, la confianza, el amor y por hacer que mi vida sea mejor cada día.*

*A mi papá por el amor y la dedicación.*

*A mi tío por darme la inspiración de un nivel superior y por brindarme su apoyo.*

*A mis abuelas Hilda y Martha por el cariño que me han dado toda la vida.*

*A mi prima Olainy la pequeña de la familia, que le sirva de inspiración para seguir estudiando.*

*A mi madrina Elba por abrirme las puertas de su casa siempre que fue necesario y por la ayuda que me ha brindado.*

*A mi amiga Esther María por la amistad incondicional de toda la vida, por ser como una hermana.*

*A los tutores Roanny por las largas horas dedicadas hasta muy tarde en la noche y a Karenia por orientarnos durante todos estos meses de trabajo. A ambos gracias por sus sugerencias.*

*A Yoel por sacar tiempo del poco que tenía libre para atendernos.*

*A Yanela, Alejandro, Olainy por estar presentes en una gran parte de mi vida, por permitirme disfrutar de su amistad.*

*A todas aquellas personas que formaron parte de esta maravillosa etapa de mi vida que ahora termina, que me han hecho disfrutar de su amistad, en especial a Yasiel, Taire, Remberto, Yuniel, Yuselís, Roberto, Yaneisi, Yusniel, nunca los olvidaré.*

*A las amistades de mi mamá que siempre estuvieron al pendiente de mi formación durante estos años.*

*De Frank Ernesto Morell Fuentes*

*A las dos personas que han dedicado prácticamente toda su existencia por verme realizar mi sueño de convertirme en ingeniero, a mi mamá y a kiki mi hermanita, quienes sin dudar estuvieron ahí, dándome su comprensión, amor y apoyo incondicional.*

*A mi novia Mary por ser la fuente de amor y confianza que me dio las fuerzas para seguir adelante. Por ser tan especial conmigo, darme su apoyo, soportar el mal humor y hacerme muy feliz.*

*A Yaiset, Damita, Lucho, José Miguel, David y Parsons que siempre estuvimos unidos sobre todo para hacer café, amistad que surgió desde hace mucho tiempo y llegamos a convertirnos en familia, al punto de querernos y tratarnos como hermanos. Siempre que hizo falta estuvimos los unos para los otros para lo que fuese necesario.*

*Al broo Reinier, que siempre estuvo al tanto de todo, aconsejándome en las buenas y en las malas. Gracias por tu apoyo de hermano.*

*A Yailin y a toda su familia por ser tan atentos y brindarme su apoyo incondicional, especialmente a la tía Isa que la quiero mucho.*

*A los tutores Roanny y Karenia por la dedicación brindada para que este trabajo saliera adelante, por ser tan exigentes y preocupados.*

## Resumen

En el último lustro<sup>1</sup>, el desarrollo de las Tecnologías de la Información y las Comunicaciones en Cuba ha ido en aumento. La Universidad de la Ciencias Informáticas es partícipe de esto, con la creación de soluciones informáticas que agilizan la gestión de los procesos de distintas instituciones. El Sistema para la planificación y control del entrenamiento en deportes de combate creado por estudiantes y profesores de la facultad 4 de dicha institución, es un ejemplo de esto, al mismo se le desarrollaron los módulos “Plan gráfico” y “*Test* pedagógicos”, ya que estas actividades hasta el momento se realizaban de forma manual, lo que hace al trabajo más complicado para los entrenadores. La implementación de los módulos al igual que la del sistema que los soporta, estuvo sustentada por tecnologías libres y fue guiada por la metodología de desarrollo Programación Extrema, generándose así la documentación correspondiente al ciclo de vida propuesto por la metodología. En cuanto a los lenguajes de programación se utilizaron, del lado del cliente: HTML, JavaScript apoyado en la librería JQuery y CSS a través del *framework* de diseño Twitter Bootstrap; del lado del servidor: PHP mediante el *framework* de desarrollo Symfony. El entorno de desarrollo integrado utilizado fue NetBeans IDE, el servidor *web* fue Apache y PostgreSQL como gestor de bases de datos. De esta forma se obtuvieron dos módulos integrados a esta plataforma, capaces de representar de forma plana y lineal la planificación del entrenamiento deportivo. Además se pueden crear y modificar las pruebas que se les realizan a los atletas para saber si están aptos para competir.

**Palabras claves:** entrenamiento deportivo, módulo, planificación, plan gráfico, *test* pedagógico.

---

<sup>1</sup> Período de 5 años.

# Índice

Introducción .....	1
<b>Capítulo 1 Fundamentación Teórica.....</b>	<b>5</b>
1.1    Conceptos generales relacionados con la investigación .....	5
1.2    Análisis de soluciones existentes.....	10
1.2.1    Soluciones similares existentes a nivel internacional .....	10
1.2.2    Soluciones similares existentes a nivel nacional .....	11
1.3    Estudio de las metodologías y estándares para el desarrollo de <i>software</i> .....	13
1.4    Tendencias y tecnologías actuales. Selección de las herramientas y lenguajes de desarrollo.....	16
1.4.1    Estudio del framework de desarrollo en el servidor .....	17
1.4.2    Estudio de la librería o framework JavaScript .....	18
1.4.3    Estudio del framework CSS .....	18
1.4.4    Lenguajes de programación.....	19
1.4.5    Entorno de desarrollo integrado.....	20
1.4.6    Herramienta CASE de modelado con UML.....	21
1.4.7    Servidor de base de datos .....	22
1.4.8    Servidor Web.....	22
1.5    Conclusiones .....	23
<b>Capítulo 2 Descripción de la propuesta de solución.....</b>	<b>24</b>
2.1    Descripción de los módulos a desarrollar.....	24
2.2    Usuarios del sistema.....	24
2.3    Modelo conceptual del negocio.....	25
2.4    Funcionalidades de la propuesta de solución .....	26
2.5    Exploración .....	28
2.5.1    Historia de usuario (HU) de la propuesta de solución .....	29
2.5.2    Prototipos de interfaz de usuario .....	33
2.6    Planificación .....	36
2.6.1    Plan de entregas.....	36
2.6.2    Iteraciones.....	38
2.7    Descripción de la arquitectura.....	40
2.8    Patrones de diseño.....	42

2.8.1	Patrones Generales de Software para Asignar Responsabilidades .....	42
2.8.2	Patrones Gang of Four .....	43
2.9	Seguridad en el sistema .....	44
2.10	Conclusiones .....	45
<b>Capítulo 3 Implementación y pruebas .....</b>		<b>46</b>
3.1	Tarjetas Clase-Responsabilidades-Colaboradores (CRC) .....	46
3.2	Diagrama Entidad Relación (DER).....	48
3.3	Estilos y estándares de codificación .....	51
3.4	Tareas de ingeniería .....	51
3.5	Pruebas .....	54
3.5.1	Pruebas unitarias.....	55
3.5.2	Pruebas de aceptación .....	56
3.6	Análisis de los resultados de las pruebas .....	60
3.7	Conclusiones .....	63
<b>Conclusiones generales y Recomendaciones .....</b>		<b>64</b>
<b>Referencias bibliográficas .....</b>		<b>65</b>



## **Introducción**

El siglo XXI está determinado por el auge y desarrollo de las tecnologías a nivel mundial.

*“Las tecnologías de información y comunicaciones (TIC) es un término que contempla toda forma de tecnología usada para crear, almacenar, intercambiar y procesar información en sus varias formas, tales como datos, conversaciones de voz, imágenes fijas o en movimiento, presentaciones multimedia y otras formas, incluyendo aquéllas aún no concebidas. En particular, las TIC están íntimamente relacionadas con computadoras, software y telecomunicaciones. Su objetivo principal es la mejora y el soporte a los procesos de operación y negocios para incrementar la competitividad y productividad de las personas y organizaciones en el tratamiento de cualquier tipo de información.”* (Tello Leal, 2008, p.3)

Cuba es un país asediado por el recrudescido bloqueo que vive la isla hace más de 5 décadas. Esta política que aplica el gobierno de Estados Unidos de América a la isla no ha logrado su propósito de entorpecer entre otras cosas el desarrollo científico-tecnológico del país. La sociedad cubana actual reconoce el papel desempeñado por las TIC, como motor impulsor de una transformación multidimensional que experimentan sectores priorizados como son la salud, la educación y el deporte.

La madurez que se observa en el desarrollo deportivo en Cuba, evidencia el nivel y la calidad con que cuenta el deporte cubano. Lo que queda demostrado con la estela de buenos resultados alcanzados históricamente en cada una de las competiciones en las que Cuba ha participado, manteniéndose hasta nuestros días, sobre todo en los deportes de combate como el judo femenino, la lucha grecorromana, el boxeo, la lucha libre y el taekwondo.

Esto ha sido posible, gracias a la idea del Comandante en Jefe Fidel Castro Ruz de crear un organismo con la misión de impulsar y promover el deporte de excelencia en Cuba, materializada el 23 de febrero de 1961 con la fundación del Instituto Nacional de Deportes, Educación Física y Recreación (INDER). El trabajo de este organismo se extiende hasta las instituciones de deporte de alto rendimiento, con especial atención en los deportistas y su entrenamiento. (Machado et al., 2010)

El entrenamiento deportivo es un factor importante en el proceso de preparación de los deportistas, permitiendo el desarrollo de la condición física individual de cada cual. Por ello es de vital importancia

## *Módulos “Plan gráfico” y “Test pedagógico” del Sistema para la planificación y control del entrenamiento en deportes de combate*

---



lograr que el proceso de planificación y control del mismo sea lo más claro y preciso posible, ya que de su correcta aplicación dependerá en gran medida el éxito del trabajo.

Para comprender y mejorar el rendimiento deportivo, se desarrollaron en un inicio una serie de estudios científicos con la ayuda de programas informáticos. La intención de los mismos era aportar un mayor rigor metodológico y precisión en la toma de datos, gestionar la información en competiciones deportivas, servir además como herramienta de ayuda en el tratamiento de los resultados y permitir el análisis de los movimientos técnicos de los deportistas. (Pons, 2010) Estas predicciones son hoy día una realidad, y un ejemplo de ello lo constituye el Sistema para la planificación y control del entrenamiento en deportes de combate, creado por estudiantes y profesores de la Facultad 4 de la Universidad de las Ciencias Informáticas (UCI).

Esta plataforma cuenta con tres módulos desarrollados: Administración, el cual permiten la gestión de las entidades básicas del proceso; Biorritmo que posibilita graficar el biorritmo de los atletas facilitando a los instructores planificar el entrenamiento individual y colectivo de su equipo; por último con el fin de apoyar las actividades se cuenta con el módulo Archivo multimedia que sirve como medio de comunicación, observación, análisis y reflexión. Dos elementos importantes dentro del proceso son el plan gráfico y los *test*<sup>2</sup> pedagógicos ya que el primero sirve de guía para plasmar detalladamente eventos y actividades, por lo que mejora la calidad de los entrenamientos obteniendo así los resultados pronosticados; el segundo está encaminado a ofrecer una valoración objetiva del estado del deportista, siendo el vehículo para medir y observar el rendimiento de los atletas en apoyo a la toma de decisiones.

Actualmente el plan gráfico y los *test* pedagógicos se realizan manualmente y el Sistema para la planificación y control del entrenamiento en deportes de combate no es capaz de enfrentar la gestión de estos elementos. Esto propicia que el procesos de elaboración de los mismos sea muy engorroso y complejo para el colectivo técnico de los diferentes deportes de combate, pues para su confección es necesario hacer un análisis de varios indicadores, realizar una serie de cálculos<sup>3</sup> con los resultados de las observaciones, además de la búsqueda de información que muchas veces se encuentra almacenada en

---

<sup>2</sup> Prueba.

<sup>3</sup> Son cálculos estadísticos de media y moda aritmética, necesarios para confeccionar un plan gráfico basando en los resultados obtenidos con el cumplimiento de los planes anteriores. Además se calculan los porcentos para saber la dosificación de los meses de entrenamiento por etapas y los volúmenes de ejercicios para formar las habilidades y capacidades en los atletas.

# Módulos “Plan gráfico” y “Test pedagógico” del Sistema para la planificación y control del entrenamiento en deportes de combate



documentos que no están digitalizados. Estos factores provocan inconsistencias en los resultados obtenidos por los atletas en los diferentes *test*, que la planificación de los prolongados ciclos de entrenamiento no sea exacta en cuanto a la división racional de su tiempo de duración y que la consulta de los mismos se realice con lentitud.

Teniendo en cuenta la situación problemática existente surge el **problema de investigación**: La carencia de la gestión del proceso de elaboración del plan gráfico y los *test* pedagógico, en el sistema de planificación y control del entrenamiento en deportes de combate.

Como **objetivo general** se propone, implementar los módulos “Plan gráfico” y “Test pedagógico” en el Sistema para la planificación y control del entrenamiento en deportes de combate.

Los **objetivos específicos** a cumplir son:

- Elaborar el marco teórico conceptual a partir del estudio de las principales tendencias y herramientas existentes relacionadas con la creación del Sistema para la planificación y control del entrenamiento en deportes de combate.
- Identificar los requerimientos con que debe cumplir la propuesta de solución.
- Diseñar e implementar los módulos “Plan gráfico” y “Test pedagógico”.
- Realizar pruebas de *software* a la propuesta de solución.

La investigación se enmarca en el **objeto de estudio**: el proceso de informatización de la planificación y control del entrenamiento deportivo, teniendo como **campo de acción**: los sistemas informáticos para la elaboración del plan gráfico y los *test* pedagógicos.

**Defendiendo como idea**: si se implementan los módulos “Plan gráfico” y “Test pedagógico” en el Sistema para la planificación y control del entrenamiento en deportes de combate, se facilitará el trabajo de los entrenadores.

Los **métodos científicos** utilizados en la búsqueda y procesamiento de la información fueron:

**A nivel teórico:**

**Analítico - sintético**: Se utilizó para realizar un análisis bibliográfico de los diferentes conceptos relacionados con la planificación y control del entrenamiento deportivo y hacer un estudio de las características de la metodología, las tecnologías y herramientas que sustentan el proceso de desarrollo



**A nivel empírico:**

**Experimental:** se utilizó para demostrar que se obtuvo un producto que funciona según las necesidades del cliente, a partir de la realización de pruebas unitarias y de aceptación.

**Entrevista:** Se utilizó como técnica de recopilación de la información, que mediante la conversación permitió identificar los parámetros e indicadores reales que recogen los artefactos Plan gráfico y Test pedagógicos, con los que trabajan los equipos técnicos de los deportes de combate.

Para un mejor entendimiento de la investigación, la misma se estructuró en tres capítulos:

**Capítulo 1: Fundamentación Teórica,** aborda los principales conceptos relacionados con el plan gráfico y los test pedagógicos. Se realiza un estudio del estado del arte de las herramientas que permiten trabajar con los elementos antes mencionados. Se definen las particularidades de las distintas herramientas, tecnologías y metodologías a utilizar.

**Capítulo 2: Descripción de la propuesta de solución,** se detalla el flujo existente de los procesos involucrados en el objeto de estudio planteado. Se describe la estructura interna de los módulos mediante los artefactos generados por la metodología de desarrollo utilizada.

**Capítulo 3: Implementación y pruebas,** contiene los temas referentes al proceso de implementación y las estrategias de prueba a utilizar.



## Capítulo 1 Fundamentación Teórica

Las nuevas tecnologías se aplican en el deporte desde hace algún tiempo. La utilización de los sistemas informáticos en la planificación y control del entrenamiento deportivo posibilita organizar adecuadamente el entrenamiento y mantener una constante interacción entre deportista y entrenador. El desarrollo de soluciones informáticas que permitan el trabajo con los elementos plan gráfico y *test pedagógico* tiene particular relevancia en el ámbito del entrenamiento, puesto que constituyen documentos rectores dentro del proceso de planificación y control del entrenamiento. En el presente capítulo se recogen los principales conceptos relacionados con estos temas. Se describe el proceso de desarrollo con enfoque ágil guiado por la metodología XP, las herramientas y las tecnologías utilizadas en el diseño y desarrollo de la propuesta de solución.

### 1.1 Conceptos generales relacionados con la investigación

Para lograr un mejor entendimiento de la investigación que se lleva a cabo, se desglosan a continuación, los conceptos a los cuales se hace mención en la misma.

**Planificación:** *“... es el proceso de preparación de un conjunto de decisiones para actuar en el futuro, orientado a lograr los fines con medios óptimos.”* (Roche, 1998, p.14)

**Planificación del entrenamiento:** Es un proceso dinámico, en el que se compara lo planificado con lo ejecutado y se realizan las pertinentes modificaciones al diseño inicial, adaptándose a las necesidades reales del entrenamiento. (Ibáñez et al., 2001)

**Control del entrenamiento:** *“El control del entrenamiento en la formación de rendimiento a largo plazo incluye todas las medidas que se aplican para planificar e influir sistemáticamente en la ejecución del entrenamiento y cuyo fin es asegurar el cumplimiento de los objetivos propuestos y planificados de cada una de las etapas de formación y áreas de entrenamiento...”* (Nicolaus, 2004, p.227)

**Entrenamiento deportivo:** Proceso basado en principios científicos, especialmente pedagógico del perfeccionamiento deportivo, en el cual se la suma de todos los esfuerzos (estímulos) efectuados en intervalos de tiempo determinados, con el objetivo de un aumento del rendimiento, y responsables de una serie de cambios funcionales y morfológicos del organismo.” (Martin 2007, p.16)

# Módulos “Plan gráfico” y “Test pedagógico” del Sistema para la planificación y control del entrenamiento en deportes de combate



**Plan escrito:** es el documento rector de la planificación del entrenamiento. En él se plasman todos los análisis previos, individuales y colectivos de los deportistas. Se determinan y formulan los objetivos generales para el macrociclo de entrenamiento y se planifican las tareas que permitirán el logro de dichos objetivos. (García Carvajal, 2011) Aunque el plan escrito tiene un indiscutible valor en la labor de los entrenadores, no se considerará en la presente investigación.

**Plan gráfico:** Es un documento importante para la planificación del proceso de entrenamiento deportivo. Posee la característica de reflejar con mayor rapidez y evidencia la planificación de prolongados ciclos de entrenamiento ya que ofrece la posibilidad de abarcar de un solo golpe de vista varios indicadores y de reconocer relaciones de una manera más rápida y sencilla. (Vargas, 2007, p.160)

Macrociclo	Preparatorio																								Competitivo										
Periodos	Preparación General												Preparación especial								Pre - competitivo				Competitivo										
Etapas	Desarrollador						Estabilizador						Desarrollador				Estabilizador				Pre - competitivo		Competitivo												
Mesociclos	Septiembre			Octubre			Noviembre			Diciembre			Enero				Febrero				Marzo		Abril												
Nº. de micro	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33		
Calendario	14	21	28	5	12	19	26	2	9	16	23	30	7	14	21	28	4	11	18	25	1	8	15	22	1	8	15	22	29	5	12	19	26		
Característica	o	o	r	o	ch	o	ch	o	ch	r	o	ch	o	ch	o	r	o	ch	ch	o	r	o	ch	ch	r	o	ch	o	a	a	c	c	C		
Frecuencia	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	2	2	2		
Hr. micro	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	
Test. Pedag.																																			
Test. Psicol.																																			
Test. medico																																			
Comp. Oficial																																			
Total juego				2	2	2	2	2	2	2	2	2	3	2	X		2	3	2	3	4	2	3	4	2	2	4	5	6	4	Competencia				
Característica	Resistencia			Resistencia muscular						Fza. Máxima			Fuerza explosiva				Fza. Máxima				Potencia				Mantener										
Objetivo	Adaptar			Multiplicar miofibrillas						Hipertrofia			Hipertrofia				Hipertrofia				Hipertrofia														
Sist.energía	Aerobio			Aerobio						Anaeróbico			Anaeróbico				Anaeróbico				Anaeróbico														
Trabajo	1 min			3 min.						10 seg.			3 min.				10 seg.				3 - 5 seg.														
Descanso	2 - 3 min			2 - 3 min.						3 - 6 min.			1 - 2 min.				3 - 6 min.				3 - 5 min.														
% de la Fza.	Fza. natural			40 - 60 %						80 - 95 %			60 - 80 %				80 - 95 %				60 - 80 %														
Ejercicios	6 - 8			8 - 10						3 - 5			6 - 8				3 - 5				6 - 8														
Series	126			324						192			252				96				90				45										
Repeticiones	1134			3240						288			2268				144				315				150										

Imagen 1 Representación de un Plan gráfico.

La planificación y control del entrenamiento deportivo está determinado por los principios o las leyes que rigen dicho proceso. Precisamente, uno de estos principios, el de carácter cíclico del proceso de entrenamiento, plantea que el mismo se estructura por ciclos, los cuales quedan debidamente definidos en el plan gráfico y se especifican a continuación: (García Carvajal, 2011)

**Macrociclos:** “... es el término utilizado para describir un ciclo de entrenamiento largo que incluye una fase competitiva. Su duración va de los tres a los doce meses. Se compone por tres períodos o tres ciclos (mesociclos).” (Manzano Moreno, 2004) Está constituido por tres estadios biológicos y pedagógicos que



se corresponden entre sí, cuyas fases han sido denominadas como: adquisición de forma deportiva, desarrollo de la forma deportiva y pérdida temporal de la forma deportiva.

En la fase de preparación, a la adquisición de la forma se le denomina período preparatorio; la de mantenimiento período de competición, y la de pérdida, período de transición. (Raposo, 200, p.134)

Los aspectos que deben desarrollarse en la planificación de los períodos son: los objetivos y tareas de los diferentes aspectos de la preparación y el volumen por actividades a desarrollar. (García Carvajal, 2011)

**Período preparatorio (P.P.):** *“es la unidad estructural más larga del macrociclo de entrenamiento; éste período permite construir los fundamentos funcionales necesarios para la ejecución de los volúmenes importantes del trabajo específico (motriz, mental y táctico) centrado en la preparación para la competición.”*(García Carvajal, 2011)

**Período competitivo (P.C.):** Tiene como objetivo mantener y eventualmente elevar el nivel de la preparación específica y favorecer a la plena utilización de la misma durante el transcurso de la competencia. (García Carvajal, 2011)

**Período transitorio (P.T.):** *“... tiene como objetivo aportar un reposo satisfactorio después de los esfuerzos de entrenamiento y de competiciones del año o del macrociclo precedente, manteniendo un nivel de entrenamiento satisfactorio para que el macrociclo siguiente se inicie en buenas condiciones.”* (García Carvajal, 2011)

*“Los períodos se estructuran sobre determinados propósitos que responden a objetivos imprescindibles en el proceso de preparación del deportista. Se denominan etapas.”* (García Carvajal, 2011)

El Período preparatorio está compuesto por dos etapas:

**Etapas de preparación general (E.P.G.):** su objetivo es preparar al deportista desarrollando todas sus capacidades. Predominan los medios generales de la preparación, el deportista crea base para empeños mayores. (Carvajal, 2011)

**Etapas de preparación especial (E.P.E.):** En esta etapa se desarrollan los componentes específicos del deporte sobre las bases creadas en la etapa de PG. (García Carvajal, 2011)

El período competitivo está compuesto por dos etapas:

# Módulos “Plan gráfico” y “Test pedagógico” del Sistema para la planificación y control del entrenamiento en deportes de combate



**Etapa precompetitiva (E.P.):** “... se diseñan, se determina y se pone a punto todo lo que se ha realizado para alcanzar el objetivo fundamental que nos hemos trazado. Como su nombre lo indica es adecuar el trabajo en condiciones muy similares a la competencia.” (García Carvajal, 2011)

**Etapa competitiva (E.C.):** es la etapa definitiva, en ella se van a poner en evidencia el trabajo planificado y desarrollado con el deportista. En ésta etapa el deportista debe haber alcanzado la forma deportiva óptima. (García Carvajal, 2011)

**Mesociclos:** Son una estructura que integra el macrociclo, y guarda muy estrecha relación con las diferentes etapas del entrenamiento y adaptan de forma general sus nombres, abarcan un período de tiempo de 3 a 6 semanas (no hay una regla fija). Los mesociclos deben tener objetivos definidos con el propósito de sistematizar dicho proceso en función del objetivo primordial del período o de la etapa de preparación; se sugiere manejar una dinámica de carga ondulante constante, con el propósito de asegurar resultados óptimos de ésta, y la combinación racional de los diferentes medios y métodos de la preparación. (García Carvajal, 2011)

Mesociclos- 9- OFD	12/5-6/6-0`8				
	O	A	A	R	Totales
Microciclos					
Resist .(A)	10	10	9	5	35
Rap.de traslc	17	17	14	8	55
Fuerza - Rapid	27	27	23	13	90
Resist.a laRap.	31	31	26	16	104
Movil. Articular	19	19	16	9	62
<b>PFG</b>	<b>104</b>	<b>104</b>	<b>87</b>	<b>52</b>	<b>347</b>
Fuerza - Rapid	58	58	49	29	194
Resist.a laRap.	73	73	61	36	243
Cap.Cood.Espel	24	24	20	12	81
Rap.Especial	87	87	73	44	291
<b>PFE</b>	<b>243</b>	<b>243</b>	<b>202</b>	<b>121</b>	<b>809</b>
Técnica de Pie	16	16	13	8	52
Técnica de 4Pto	16	16	13	8	52
Perfec. Téc-Tác	87	87	73	44	291
Tope Est - Ent.	56	56	47	28	187
Tope de Comp.	106	106	88	53	353
Complejo K	31	31	26	16	104

Imagen 2 Representación de un mesociclo de entrenamiento.

**Microciclos:** Son la estructura mínima de la planificación a corto plazo. Por lo general tiene una duración de una semana en las que, en función de los objetivos, se puede entrenar desde una sesión hasta dos o tres sesiones al día según las necesidades y el momento en la planificación. (Manzano Moreno, 2004)

# Módulos “Plan gráfico” y “Test pedagógico” del Sistema para la planificación y control del entrenamiento en deportes de combate



Microciclo # 36 (O)													
Direcciones	Totales			Lunes		Martes		Miercoles		Jueves		Viernes	
	P	R	%	P	R	P	R	P	R	P	R	P	R
Resist .(A)	10	0	0			10							
Rap.de traslc	17	0	0	5				6				6	
Fuerza - Rapid	27	0	0	9				9				9	
Resist.a laRap.	31	0	0			15				16			
Movil. Articular	19	0	0	6				7				6	
<b>PFG</b>	<b>104</b>	<b>0</b>	<b>0</b>	<b>20</b>	<b>0</b>	<b>25</b>	<b>0</b>	<b>22</b>	<b>0</b>	<b>16</b>	<b>0</b>	<b>21</b>	<b>0</b>
Fuerza - Rapid	58	0	0	10		12		12		12		12	
Resist.a laRap.	73	0	0	13		15		15		15		15	
Cap.Cood.Espel	24	0	0	4		5		5		5		5	
Rap.Especial	87	0	0	17		18		17		18		18	
<b>PFE</b>	<b>243</b>	<b>0</b>	<b>0</b>	<b>44</b>	<b>0</b>	<b>50</b>	<b>0</b>	<b>49</b>	<b>0</b>	<b>50</b>	<b>0</b>	<b>50</b>	<b>0</b>
Técnica de Pie	16	0	0	4				4		8			
Técnica de 4Pto	16	0	0	4				4		8			
Perfec. Téc-Tác	87	0	0	17		35		17				18	
Tope Est - Ent.	56	0	0			28				28			
Tope de Comp.	106	0	0	35				36				35	
Complejo K	31	0	0							31			

Imagen 3 Representación de un microciclo de entrenamiento.

Para realizar un trabajo exitoso en el deporte, se hace imprescindible acciones de control y valorar la información de las influencias de los diferentes estímulos que recibe el organismo del deportista durante el proceso pedagógico del entrenamiento deportivo, lo que se hace posible con la ayuda de los *test* pedagógicos. Estos permiten tomar las dediciones correctivas necesarias, según la situación y fase de la preparación o cualquier ciclo de la planificación. (Guerra, 2008)

**Test pedagógico:** Son las mediciones realizadas a un atleta con el objetivo de determinar el estado o desarrollo del deportista en cuanto a capacidad de trabajo, nivel físico, técnico, táctico, teórico. (González, 2003; Guerra, 2008)

Desde el punto de vista metodológico, los *test* pedagógicos se clasifican en: (Guerra, 2008)

**Teóricos:** “Encaminados a evaluar el nivel de conocimiento del contenido del deporte practicado con énfasis en el aprendizaje de los diferentes elementos técnicos, tácticos, así como el nivel de maestría según el nivel deportivo...” (Guerra, 2008)

**Prácticos:** “Encaminados a evaluar el nivel alcanzado en las acciones motrices tanto en las acciones técnico-tácticas como de rendimiento general.” (Guerra, 2008)



De los dos tipos que existen de *test* pedagógicos, esta investigación presta atención a práctico. Estos *test* que se realizan para determinar el estado físico del deportista, están previamente diseñados, en dependencia del objetivo que se desea trabajar, dígase, elasticidad, potencia, velocidad, resistencia; es válido aclarar que estos pueden ser modificados a consideración del entrenador y/o realizar otros nuevos.

Durante los diferentes mesociclos y microciclos, se deben intercalar diferentes *test* para evaluar el desarrollo según los objetivos de cada ciclo que permita de forma correctiva y oportuna realizar cambios necesarios. Estos permitirán evaluar desde el estado inicial hasta las transformaciones urgentes provocadas por la acumulación de la carga, permitiendo hacer posibles pronósticos de un resultado. (Guerra, 2008)

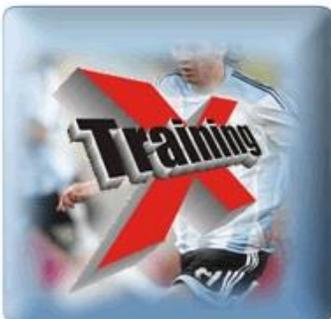
**Módulo:** “... es un fragmento de un programa que se desarrolla de forma independiente del resto del programa...” (Gallardo Ruiz y García López, 2014)

## **1.2 Análisis de soluciones existentes**

La planificación y el buen control del entrenamiento deportivo, tributan a mejorar la condición física e intelectual de los atletas y sus resultados deportivos. El desarrollo científico y tecnológico es uno de los factores más influyentes sobre la sociedad contemporánea y tiene su impacto en la práctica del deporte y las actividades físicas. Los descubrimientos científicos que confirman la eficacia de la práctica deportiva, fueron posibles gracias a la utilización de la tecnología para evaluar los estados funcionales del organismo antes, durante y después de la práctica regular de los mismos. (Bagnara y Pacheco, 2011)

Hoy en día existe un volumen creciente de *software* diseñado y desarrollado para servir de ayuda al técnico deportivo, con aplicaciones específicas orientadas a la valoración técnico-táctica de los deportistas y al control del entrenamiento. A continuación se muestra un análisis realizado de algunas de las soluciones existentes. (Pons, 2010)

### **1.2.1 Soluciones similares existentes a nivel internacional**



**X-Training:** Es un sistema informático para la planificación, periodización y control del entrenamiento deportivo está diseñado para todos los entrenadores de deportes colectivos, especialmente orientado al fútbol, baloncesto, voleibol, balonmano, fútbol sala, rugby, hockey sobre césped y sobre patines y polo



acuático. Cuenta con herramientas para optimizar el tiempo, por ejemplo la de planificación del entrenamiento, la cual permite confeccionar planificaciones para un deportista individual, o bien para un equipo de cualquier deporte o categoría, además visualiza la planificación completa de una semana de entrenamiento y organiza un macrociclo de una manera sencilla y rápida, contemplando todas las cualidades físicas y pudiendo realizar listados, resúmenes y gráficos en forma inmediata. Presenta además un generador de evaluaciones individuales y grupales, es decir de manera general esta herramienta permite diseñar *test* o pruebas para distintas cualidades físicas y diseñar evaluaciones para diferentes deportes cuyo resultado sea un dato numérico, de forma tal que se puedan realizar análisis de la condición del deportista o del equipo. (Informática& Deportes, 2005)



**X-Medalist:** Es un sistema informático enfocado al entrenamiento de fuerza, actividades para la salud, y entrenadores de deportes individuales como atletismo, tenis, natación, ciclismo, patín, carrera, remo, pruebas combinadas y deportes de aventura. Referente a la planificación cuenta con herramientas prácticas que permiten diseñar planes de entrenamiento, macrociclos, mesociclos, microciclos; mostrando a partir de esta información un gráfico interactivo. Permite confeccionar y almacenar pruebas asociadas a las actividades planificadas en el programa,

permitiendo evaluar a los atletas en cualquier aspecto de la condición física. (Informática & Deportes, 2007)

### 1.2.2 Soluciones similares existentes a nivel nacional



**Sistema para la planificación del entrenamiento del judo**

**femenino:** Este sistema está orientado al judo femenino, brinda la posibilidad a los entrenadores del judo femenino de llevar un control sobre la planificación del entrenamiento de este deporte. Cuenta

con cuatro módulos principales. El módulo Administración es donde se gestionan los usuarios. El módulo Plan gráfico permite realizar la planificación de las fechas de las pruebas médicas, competencias y campos de entrenamiento. Esta planificación se divide en macrosistemas, mesosistemas y microsistemas para una mayor organización. El módulo *Test* pedagógicos permite evaluar pruebas de fuerza, resistencia, velocidad, flexibilidad, prueba teórica y técnica. En el módulo *Atleta* se gestionan los datos de todas las



atletas. Este sistema está diseñado de manera genérica por lo que presenta limitantes como: es para un solo deporte y una sola categoría, el judo femenino, además para confeccionar un nuevo *test* hay que hacerlo mediante el código, por lo que sería difícil para un especialista en deporte que no tiene conocimientos de programación. (Donatien, 2011)



**Visplan 3.0:** Este sistema informático permite economizar y rentabilizar el tiempo empleado para planificar y controlar el entrenamiento. Está orientado al deporte de base y el de alto rendimiento. Cuenta con dos módulos Presentación que presenta el logotipo así como el título del programa y Principal que almacena los datos personales de los atletas, las pruebas de control pedagógico realizadas, los macrociclos, mesociclos, microciclos y cuenta con herramientas para la planificación y evaluación de la intensidad de la carga. Ha sido realizado teniendo en cuenta las formas tradicionales de planificación empleadas en Cuba; no obstante puede ser adaptado a las formas contemporáneas que hoy se emplean en el ámbito de la preparación del deportista. (Flores Zamora y Alba Martínez, 2013)

El estudio realizado de las aplicaciones existentes en el ámbito nacional e internacional, permitió concluir lo siguiente: estos sistemas constituyen soluciones a la medida del problema que resuelven, muy difícil de adaptar al entorno de los deportes de combates.

El estudio de las soluciones similares existentes a nivel internacional, aportó los siguientes elementos para la investigación, los cuales serán incorporados a la propuesta de solución en el proceso de implementación de la misma:

- Posibilitan la división de la planificación en macrociclos, mesociclos y microciclos.
- Posibilitan exportar el plan gráfico en formato de documento (pdf, excel).
- Posibilitan la evaluación de los atletas a partir de los resultados obtenidos.

Aunque estos sistemas permiten realizar una planificación del entrenamiento deportivo, no representan de forma plana y lineal un plan gráfico, de manera que se puedan visualizar todos elementos por los que está compuesto el mismo. Además al generar la evaluación de los *test* pedagógicos la escala de evaluación es invariable, de manera que a todos los atletas se les evalúa de la misma manera independientemente de la



división a la que pertenezca. Estas deficiencias serán elementos importantes a tener en cuenta en la propuesta de solución.

En el caso de los sistemas analizados a nivel nacional, están realizados con el objetivo de satisfacer necesidades concretas de un deporte. El estudio de estas aplicaciones informáticas permitió encontrar funcionalidades en común con los módulos a implementar, pero también limitaciones, que serán solucionadas con el desarrollo de la propuesta de solución, convirtiéndola en potencialidades para el Sistema para la planificación y control del entrenamiento en deportes de combate. Además, no es posible integrar estas aplicaciones al sistema, ni se puede reutilizar su código, en el proceso de implementación de los módulos.

### **1.3 Estudio de las metodologías y estándares para el desarrollo de software**

*“Una metodología es una colección de procedimientos, técnicas, herramientas y documentos auxiliares que ayudan a los desarrolladores de software en sus esfuerzos por implementar nuevos sistemas de información. Una metodología está formada por fases, cada una de las cuales se puede dividir en sub-fases, que guiarán a los desarrolladores de sistemas a elegir las técnicas más apropiadas en cada momento del proyecto y también a planificarlo, gestionarlo, controlarlo y evaluarlo.” (Carvajal Riola, 2008, p.59)*

Actualmente el interés por el uso de una metodología que guie y garantice en gran medida el éxito en el desarrollo de *software* se ha vuelto una inminente necesidad, puesto que estas proveen pautas para la aplicación de notaciones de modelado y el uso de herramientas. Las metodologías ágiles surgen principalmente por la dificultad de las metodologías tradicionales de ajustarse totalmente a proyectos donde el entorno del sistema es muy cambiante y en donde resulta necesario reducir el tiempo de desarrollo manteniendo la alta calidad y por la excesiva documentación que estas generan. El término ágil aplicado al desarrollo de *software*, tiene poco tiempo de surgimiento, es por esto que se crea la organización *The Agile Alliance*<sup>4</sup>, encargada de promover los conceptos relacionados con el desarrollo

---

<sup>4</sup> Sitio Oficial: <http://www.agilealliance.com>



ágil, recogidos en el documento Manifiesto Ágil<sup>5</sup>, que resume lo relacionado con esta filosofía de trabajo en los siguientes valores: (Canós, Letelier y Penadés, 2003)

- Simplicidad en las soluciones implementadas.
- Comunicación fluida entre los participantes.
- Retroalimentación continua entre el cliente y el equipo de desarrollo.
- Coraje para enfrentar los cambios.

### **Programación Extrema (XP)**

Existen varias metodologías ágiles entre ellas Programación Extrema (XP<sup>6</sup>), su creador Kent Beck expone en esta frase el motivo principal por el cual surge: (Rendón Gallón, 2011)

*“Todo en el software cambia. Los requisitos cambian. El diseño cambia. El negocio cambia. La tecnología cambia. El equipo cambia. Los miembros del equipo cambian. El problema no es el cambio en sí mismo, puesto que sabemos que el cambio va a suceder; el problema es la incapacidad de adaptarnos a dicho cambio cuando éste tiene lugar.”*

XP es muy reconocida entre los enfoque ágiles. Nació a partir de la idea de establecer un conjunto de prácticas que facilitasen la finalización de los proyectos. El término extreme es debido a que las prácticas que se utilizaron, fueron llevadas hasta el extremo.

El ciclo de vida de XP está conformado por 6 fases: (Carvajal Riola, 2008, p.111-115)

- Fase de Exploración.

El cliente escribe las tarjetas de historia con aquellas funcionalidades que ellos quieren que sean añadidas al programa en su primera versión, mientras que el equipo de desarrollo se enfoca en estudiar las tecnologías y herramientas que van a utilizar.

- Fase de planificación.

Se establece la prioridad para cada historia y se estima por parte de los desarrolladores el tiempo dedicado a cada una y el contenido de la primera entrega.

---

<sup>5</sup> Sitio oficial: <http://www.agilemanifesto.org/>

<sup>6</sup> Por sus siglas en inglés: Extreme Programming.



- Fase de iteraciones.

En esta fase la planificación realizada en la etapa anterior se divide en varias iteraciones de duración variable. Son los usuarios los que deciden qué historia hacer en cada iteración, al final de cada iteración se realiza una prueba funcional creada por el cliente, de manera tal que para el final de esta fase, se obtenga una versión lista para producción.

- Fase de producción.

Se realizan las pruebas necesarias antes de entregar el producto al cliente. Esta fase se divide en varias iteraciones y se decide si realizar cambios importantes al producto en caso de surgir alguno, o dejarlo para posteriores versiones.

- Fase de mantenimiento.

Una vez entregada la primera versión del *software* al cliente, debe permanecer en un entorno de producción, mientras hayan iteraciones en esa fase.

- Fase de cierre del proyecto.

Es la fase en que los clientes ya no tienen más historias que deban ser implementadas y todas sus necesidades quedaron satisfechas. La documentación del proyecto se realiza en esta fase, ya que ni la arquitectura, ni el diseño, ni el código sufrirán cambio alguno.

Se utiliza XP para dirigir el desarrollo de la propuesta de solución, pues es interés del cliente estandarizar la documentación que genera dicha metodología, en consonancia con la obtenida después de haberla aplicado al proceso de desarrollo del Sistema para la planificación y control del entrenamiento en deportes de combate. Además entre los enfoques ágiles, XP es la que más se ajusta a las características del proyecto.

### **Lenguaje de modelado (UML)**

Es un Lenguaje de Modelado Unificado (UML<sup>7</sup>) y se define como “...un lenguaje que permite especificar, visualizar y construir los artefactos de los sistemas de software” (Larman, 1999, p.15)

---

<sup>7</sup> Por sus siglas en inglés: Unified Modeling Language.



Está compuesto por una notación muy específica que describe clases, componentes, nodos, actividades, flujos de trabajo, casos de uso, objetos, estados y cómo modelar la relación entre esos elementos. Soporta además extensiones personalizadas, a través de estereotipos. Permite a los desarrolladores visualizar los resultados de su trabajo en esquemas o diagramas estandarizados. El equipo de desarrollo vio en UML el lenguaje idóneo para modelar el negocio en cuestión, por ser un estándar internacional, por las características antes presentadas y por el nivel de conocimiento que se tiene acerca del mismo.

#### **1.4 Tendencias y tecnologías actuales. Selección de las herramientas y lenguajes de desarrollo**

La *web* es la plataforma que alberga numerosas aplicaciones con fines diferentes. Las aplicaciones *web* han revolucionado la manera en que se utiliza el internet. (Peña, Córcoles y Casado, 2006) Se puede definir estos sistemas como: “...aplicaciones basadas en el modelo Cliente/Servidor que gestionan datos almacenados en un servidor *web*, y que utilizan como interfaz páginas en formato HTML, conteniendo datos hipermedia. El usuario se comunica con la aplicación desde cualquier cliente conectado a la red.” (Cuesta Morales, 1999)

Una vez analizado este concepto se puede concluir que el Sistema para la planificación y control del entrenamiento en deportes de combate cumple con los requisitos para una aplicación *web*. Puesto que a esta plataforma le serán integrados dos módulos, las tecnologías a utilizar para la realización de los mismos están determinadas por las utilizadas en dicho sistema.

Las siguientes ventajas que presentan las aplicaciones *web* la convierten en parte importante de la llamada “sociedad de la información” en la que se encuentra inmerso el mundo, debido al acelerado avance de las comunicaciones y las nuevas tecnologías: (Cuesta Morales, 1999)

- **Facilidad de manejo:** La interacción con el usuario se establece mediante elementos a los que está cada vez más acostumbrado, como las páginas *web*, que le permiten conocer la funcionalidad del sistema con poco esfuerzo.
- **Accesibilidad:** Son accesibles desde cualquier punto de la red, lo cual significa que un usuario autorizado (se pueden establecer controles de acceso) puede utilizarla si dispone de cualquier conexión a Internet.



- **Facilidad de desarrollo:** En este sentido, hay dos aspectos a destacar, en primer lugar, la descomposición intrínseca en componentes, así como al hecho de que en cierta medida, algunos aspectos que tienen que ver con el carácter distribuido de la aplicación están resueltos de antemano (por ejemplo el protocolo HTTP controla el acceso a datos en el servidor).

#### 1.4.1 Estudio del *framework* de desarrollo en el servidor

Las tecnologías de tipo *framework* son muy útiles en el desarrollo de aplicaciones *web*, puesto que simplifican el trabajo con la implementación de patrones que sirven para resolver tareas comunes, permitiendo a los desarrolladores enfocarse en las particularidades de la aplicación; proporcionan una estructura al código caracterizándolo por su limpieza y legibilidad; por lo que se hace fácil su mantenimiento. (Potencier y Zaninotto, 2015)

Un *framework* diseñado con el objetivo de optimizar el desarrollo de aplicaciones *web* es Symfony<sup>8</sup> que será utilizado para el desarrollo de la propuesta de solución en su versión 1.4.20. Entre las características que lo sitúan a Symfony en la élite de los *framework* dedicados al desarrollo *web* se encuentran: separa la lógica de negocio, la lógica de servidor y la presentación de la aplicación *web*, es fácil de instalar y configurar en la mayoría de las plataformas tanto Unix, Linux, Windows, es independiente del Sistema gestor de Bases de Datos (SGBD), es suficientemente estable para desarrollar aplicaciones a largo plazo, permite su integración con otras librerías ya desarrolladas como es el caso de JQuery que será empleada en el desarrollo de la propuesta de solución. (Potencier y Zaninotto, 2015)

Symfony está probado con proyectos reales con millones de usuarios como Yahoo Bookmarks, Yahoo Answers. Desde su primera versión ha sido creado para PHP 5, desechando la versión PHP 4. (Cisneros y Tupe, 2008)

Symfony está diseñado para que pueda ser utilizado por programadores expertos o principiantes, aunque la calidad de su código fuente y a la gran cantidad de documentación disponible son dos ventajas significativas, la decisión de emplear el *framework* recae en el tamaño del proyecto, o sea que para una aplicación *web* compleja con mucha lógica de negocio es recomendable utilizar Symfony, puesto que el

---

<sup>8</sup> Sitio web en español: <http://www.symfony.es/>



mismo constituye un *framework* maduro, bien documentado y con una gran comunidad que lo apoye. (Potencier y Zaninotto, 2015)

#### **1.4.2 Estudio de la librería o *framework* JavaScript**

JQuery<sup>9</sup> es una librería de código JavaScript, la misma ha alcanzado gran popularidad debido a su facilidad de uso y su gran potencia. jQuery simplifica la manera de interactuar con los documentos HTML, manipular el árbol DOM (*Document Object Model*), el manejo de eventos, el desarrollo de y agregar interacción con la técnica AJAX en páginas *web*.

jQuery hace énfasis en la creación de componentes que son compatibles con el mayor número de herramientas que se ejecutan en un entorno *web*. Constituye un producto serio, estable, bien documentado y con un gran equipo de desarrolladores a cargo de la mejora y actualización del *framework*. Cuenta con una aceptación por parte de los programadores muy buena y un grado de penetración en el mercado muy amplio, lo que lo convierte en una de las mejores opciones. Tiene licencia para uso en cualquier tipo de plataforma, personal o comercial. (Álvarez, 2009)

La versión a utilizar será la 1.8.3. Se utilizó además la biblioteca de componentes para JQuery denominada JQuery UI en su versión 1.10.0, la cual permite contar con un conjunto de componentes muy útiles en la construcción de interfaces de usuario.

#### **1.4.3 Estudio del *framework* CSS**

Twitter Bootstrap es una colección de herramientas de *software* libre, creado en agosto del 2011 por los desarrolladores de Twitter, para la creación de sitios y aplicaciones *web*. La versión a utilizar de este *framework* es la 2.2.2, puesto que está perfectamente adaptado a la mayoría de los navegadores (Chrome, Firefox, Safari, IE 7 y Opera) y a los dispositivos móviles y tabletas. Es fácil de aprender y de modificar, posee una documentación muy detallada y abundante. Contiene plantillas de diseño basadas en HTML y CSS con tipografías, formularios, botones, gráficos, barras de navegación y demás componentes de interfaz, así como extensiones opcionales de JavaScript. (Kaplan, 2014)

---

<sup>9</sup> Sitio Oficial: <http://jquery.com/>



#### **1.4.4 Lenguajes de programación**

El empleo de estos lenguajes de programación en la implementación de los módulos, no solo estuvo determinado por su utilización en el Sistema para la planificación y control del entrenamiento en deportes de combate, sino también por el dominio que tiene el equipo de desarrollo de la propuesta de solución sobre los mismos, por su difusión y uso en la universidad a lo largo de la carrera.

**PHP**<sup>10</sup>: *“es un lenguaje de código abierto... especialmente adecuado para el desarrollo web y que puede ser incrustado en HTML.”* (Achour et al. 2015)

Lo que distingue a PHP de las tecnologías del cliente como Javascript es que el código es ejecutado en el servidor, generando HTML y enviándolo al cliente. Es simple de utilizar para los principiantes, pero a su vez ofrece muchas características avanzadas para los programadores profesionales.

PHP puede emplearse en los principales sistemas operativos, incluyendo Linux, muchas variantes de Unix (incluyendo HP-UX, Solaris y OpenBSD), Microsoft Windows, Mac OS X, RISC OS y probablemente otros más. PHP admite la mayoría de servidores *web* de hoy en día, incluyendo Apache, IIS, y muchos otros.

Una de las características más potentes y destacables de PHP es su soporte para un amplio abanico de bases de datos. Tiene útiles características de procesamiento de texto, las cuales incluyen extensiones y herramientas para el acceso y análisis de documentos XML. (Achour et al. 2015)

**JavaScript**: Es un lenguaje de programación para ser creativo, su principal característica es la de ser un lenguaje de scripting y el muy usado, particularidad que lo convierte en seguro y fiable. Se utiliza principalmente para crear páginas *web* dinámicas. Es un lenguaje interpretado en el cliente por el navegador al momento de cargarse la página, es multiplataforma, orientado a eventos con manejo de objetos, cuyo código se incluye directamente en el mismo documento HTML. jQuery se encuentra. (Eguiluz, 2015)

**HTML**<sup>11</sup>: Se define como un Lenguaje de Marcado de Hipertexto, es una implementación del standard SGML<sup>12</sup> (*Standard Generalized Markup Language*), permite definir lenguajes de diseño descriptivos;

---

<sup>10</sup> Acrónimo recursivo de PHP: Hypertext Preprocessor.

<sup>11</sup> HyperText Markup Language

<sup>12</sup> Estándar internacional para la definición de texto electrónico independiente de dispositivos, sistemas y aplicaciones.



proporciona un medio de codificar documentos hipertexto cuyo destino sea el intercambio directo entre sistemas o aplicaciones.

La evolución del lenguaje se impone con el desarrollo de las tecnologías, de ahí que:

*“... HTML 5, ya que no es simplemente una nueva versión del lenguaje de marcación HTML, sino una agrupación de diversas especificaciones concernientes al desarrollo web. Es decir, HTML 5 no se limita sólo a crear nuevas etiquetas, atributos y eliminar aquellas marcas que están en desuso o se utilizan inadecuadamente, sino que va mucho más allá.” (Álvarez, 2009)*

**CSS<sup>13</sup>**: Las Hojas de Estilo en Cascada es un lenguaje creado para controlar el aspecto o presentación de los documentos electrónicos definidos con HTML y XHTML. CSS es la mejor forma de separar los contenidos y su presentación y es imprescindible para crear páginas web complejas.

El gran impulso de los lenguajes de hojas de estilos se produjo con el boom de Internet y el crecimiento exponencial del lenguaje HTML para la creación de documentos electrónicos. Con CSS se mejora la accesibilidad del documento, reduce la complejidad de su mantenimiento y permite visualizar el mismo documento en infinidad de dispositivos diferentes.

CSS ya va por la versión 3 del producto, que trae nuevas novedades para los desarrolladores. Avanza un poco más en la dirección, de aportar mayor control sobre el estilo con el que se muestran los elementos de las páginas. El empleo de este lenguaje para realizar la aplicación viene dado por su integración con el *framework* de CSS Twitter Bootstrap en su versión 2.2.2. (Eguiluz, 2015)

#### **1.4.5 Entorno de desarrollo integrado**

**NetBeans IDE**: Es un Entorno de Desarrollo Integrado (IDE<sup>14</sup>), hecho esencialmente para el lenguaje de programación Java. Es un producto libre y gratuito para su uso.

Un IDE está compuesto por un conjunto de herramientas de programación, que ha sido empaquetado como un programa de aplicación; es decir, que consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica (GUI).

---

<sup>13</sup> Sus siglas en inglés: Cascading Style Sheets.

<sup>14</sup> Por sus siglas en inglés: *Integrated Development Environment*



Entre las características que destacan a NetBeans se encuentran: suele dar soporte a casi todas las novedades en el lenguaje Java, cuenta con asistentes para la creación y configuración de distintos proyectos, incluida la elección de algunos *framework*, con un buen editor de código, multilinguaje, con el habitual coloreado y sugerencias de código, además permite simplificar la gestión de grandes proyectos con el uso de diferentes vistas, asistentes de ayuda, y estructurando la visualización de manera ordenada. Presenta herramientas para depurado de errores, el *debugger* que incluye el IDE es bastante útil para encontrar dónde fallan las cosas. Por todas las particularidades que encierra este IDE y el dominio que tiene el equipo de programadores de la propuesta de solución, es que se decide usar NetBeans 8.0 en el proceso de implementación de los módulos.

#### **1.4.6 Herramienta CASE de modelado con UML**

**Visual Paradigm for UML:** Es una herramienta para la Ingeniería de *Software* Asistida por Computadora (CASE<sup>15</sup>) que soporta el modelado mediante UML y proporciona asistencia a los analistas, ingenieros de *software* y desarrolladores, durante todas las etapas del ciclo de vida de desarrollo de una aplicación informática. Es de carácter comercial, con una versión libre para la comunidad “*Visual Paradigm for UML Community Edition*” y el resto con versiones de evaluación. (Visual Paradigm, 2014)

Dentro de las principales facilidades que ofrece el uso de esta herramienta se encuentran: (Visual Paradigm, 2014)

- Permite generar informes en varios formatos, entre los que se encuentran PDF, HTML y MS WORD. Ello permite estandarizar la documentación.
- Permite la generación de código a partir de diagramas para lenguajes de programación, entre los que se encuentran Java, C# y PHP.
- Permite realizar ingeniería inversa a partir de bases de datos, soportando una amplia gama de bases de datos, entre las que se encuentran Oracle, MySQL y PostgreSQL.
- Se integra con los IDE NetBeans, Eclipse, Visual Studio e IntelliJIDEA.
- Posee una sencilla e intuitiva interfaz para el modelado de los distintos diagramas que soporta.

---

<sup>15</sup> Por sus siglas del inglés *Computer Aided Software Engineering*



La versión a utilizar será la 8.0, lanzada el 16 de agosto de 2010, la misma es la más actualizada en los servidores de la UCI.

#### **1.4.7 Servidor de base de datos**

**PostgreSQL:** Sus características técnicas la hacen una de las bases de datos más potentes y robustas. Su desarrollo comenzó hace más de 16 años, durante este tiempo, la estabilidad, potencia, robustez, facilidad de administración e implementación de estándares han sido las características que más se han tenido en cuenta durante su desarrollo. PostgreSQL funciona muy bien con grandes cantidades de datos y una alta concurrencia de usuarios accediendo a la vez al sistema. (Martínez, 2015)

Algunas de las características más importantes de PostgreSQL

- Es una base de datos completamente ACID<sup>16</sup>.
- Soporta distintos tipos de datos: además del soporte para los tipos base, también soporta datos de tipo fecha, monetarios, elementos gráficos, datos sobre redes (MAC, IP) y cadenas de bits. También permite la creación de tipos propios.
- Copias de seguridad en caliente.
- Amplia documentación.
- Disponible para Linux y UNIX en todas sus variantes (AIX, BSD, HP-UX, SGI IRIX, Mac OS X, Solaris, Tru64) y Windows 32/64bit.

Por estas características se decide utilizar la versión 9.1 de este servidor de base de datos para el desarrollo de los módulos.

#### **1.4.8 Servidor Web**

**Apache:** Es el servidor *web* escogido por excelencia, calidad de servicios, robustez y estabilidad. Es un *software* libre y de código abierto para las plataformas Windows, Mac OS y UNIX (GNU, BSD), en la cual se implementa el protocolo HTTP 1.1 y la noción de sitios virtuales.

---

<sup>16</sup> Atomicity, Consistency, Isolation and Durability (en español: Atomicidad, Consistencia, Aislamiento y Durabilidad)



El éxito de Apache<sup>17</sup> hace que muchos usuarios y servidores renueven la elección a este servicio, es por ello que el equipo de desarrollo de la propuesta de solución escogió este servidor de aplicaciones en su versión 2.0. Esto viene dado por las características que presenta, tales como: (Martín Maldonado, 2008)

- Funciona en una multitud de plataformas y Sistemas Operativos.
- Ofrece tecnología libre y de código abierto, otorgándole una transparencia y dando la posibilidad de conocer que es lo que realmente estamos instalando.
- Es altamente configurable y de diseño modular, capaz de ampliar su funcionalidad y calidad de servicios.
- Trabaja en conjunto con gran cantidad de lenguajes de programación interpretados como PHP, Perl, soporte con CGI (*Common Gateway Interface*), Java, JSP (*Java Server Pages*) y otros lenguajes de script, el complemento ideal para los sitios *web* dinámicos que vemos en la actualidad.

## **1.5 Conclusiones**

Con la investigación realizada acerca del plan gráfico y los *test* pedagógico, se logra una mejor comprensión del papel que desempeñan los mismos en el proceso de planificación y control del entrenamiento deportivo. Además se seleccionan las posibles funcionalidades que tendrán estos módulos que se integrarán al Sistema para la planificación y control del entrenamiento en deportes de combate. El estudio de las soluciones existentes permitió la extracción de rasgos característicos que son importantes adaptados a los deportes de combate. El sistema que soporta a la propuesta de solución fue realizado con las tecnologías libres Symfony, JQuery y Twitter Bootstrap, el análisis que se hizo de las mismas permitió dominar mejor el ambiente de desarrollo sobre el cual se trabajará en la implementación de los módulos, además de cumplir con las políticas de la universidad de potenciar el uso de *software* libre, puesto que también las herramientas a utilizar cumplen esta condicionante. La metodología que dirigió el proceso de desarrollo de los módulos fue XP.

---

<sup>17</sup> Sitio web oficial: <http://httpd.apache.org/>



## **Capítulo 2 Descripción de la propuesta de solución**

En XP las funcionalidades se gestionan de manera sencilla, el cliente escribe y prioriza las historias de usuario que expresan las necesidades del sistema. Los programadores estiman el esfuerzo asociado a cada una de ellas. Para cumplir con las expectativas del cliente, se realiza una planificación general de las funcionalidades que se les presentarán a los interesados en cada entrega del producto.

### **2.1 Descripción de los módulos a desarrollar**

El Sistema para la planificación y control del entrenamiento en deportes de combate, en su segunda versión contará además con los módulos “Plan gráfico” y “Test pedagógico”, en apoyo a la toma de decisiones y a la planificación en la preparación de los deportistas y sus entrenadores, en los diferentes deportes de combate. Los usuarios del sistema tendrán los siguientes roles: Administrador, Jefe de colectivo técnico, Entrenador, Psicólogo.

En el módulo “Plan gráfico”, después de creado un deporte el Jefe de colectivo técnico es el autorizado para elaborar el plan gráfico de su equipo. Además podrá realizar todas las funcionalidades que tengan que ver con el mismo como son: Crear, visualizar y eliminar plan gráfico, adicionar, mostrar, modificar y eliminar macrociclos, períodos, etapas, mesociclos y microciclos. El usuario con rol Entrenador solo podrá consultar el plan gráfico y después de seleccionar el que desea analizar exportarlo a PDF si lo considera necesario.

En el módulo “Test pedagógico”, el usuario con rol Administrador será el encargado de gestionar los atletas para luego poder asociarle los test pedagógicos. Ya en este módulo este usuario tiene la posibilidad de crear, modificar, eliminar test pedagógico y escalas, así como registrar los resultados con lo que se van a evaluar a los atletas.

### **2.2 Usuarios del sistema**

Los usuarios del sistema son aquellas personas que interactúan con la aplicación. Cada uno tiene diferentes restricciones de acuerdo al rol que desempeñan. A continuación se describen los usuarios del sistema con sus responsabilidades.



Tabla 1 Usuarios del sistema.

Usuarios	Responsabilidades
Administrador	Tiene acceso a los módulos “Plan gráfico” y “Test pedagógico”. Solo a ver la información de los mismos.
Jefe de colectivo técnico	Tiene acceso a los módulos “Plan gráfico” y “Test pedagógico”. Es válido aclarar que tiene todos los privilegios, pero solamente sobre la información almacenada por el mismo.
Entrenador	Tendrá acceso a los módulos “Plan gráfico”, en el cual solo tendrá privilegios de visualizar la información y “Test pedagógico” en el que podrá tener todos los privilegios sobre la información almacenada por el mismo.
Psicólogo	Solo tendrá acceso al módulo “Test pedagógico”.

## 2.3 Modelo conceptual del negocio

Utilizando la notación UML y considerando los resultados de la investigación, el diagrama conceptual construido permite la representación del conocimiento que se ha adquirido durante la conformación del proyecto. Ofrece un mejor entendimiento de la relación que existe entre los conceptos asociados al negocio. (Zapata Jaramillo y Arango Isaza, 2005)

Para un mejor entendimiento de lo que se desea desarrollar se confeccionó este artefacto después de haber identificado los conceptos y objetos relacionados con el plan gráfico y los test pedagógicos en el proceso de planificación y control del entrenamiento deportivo.

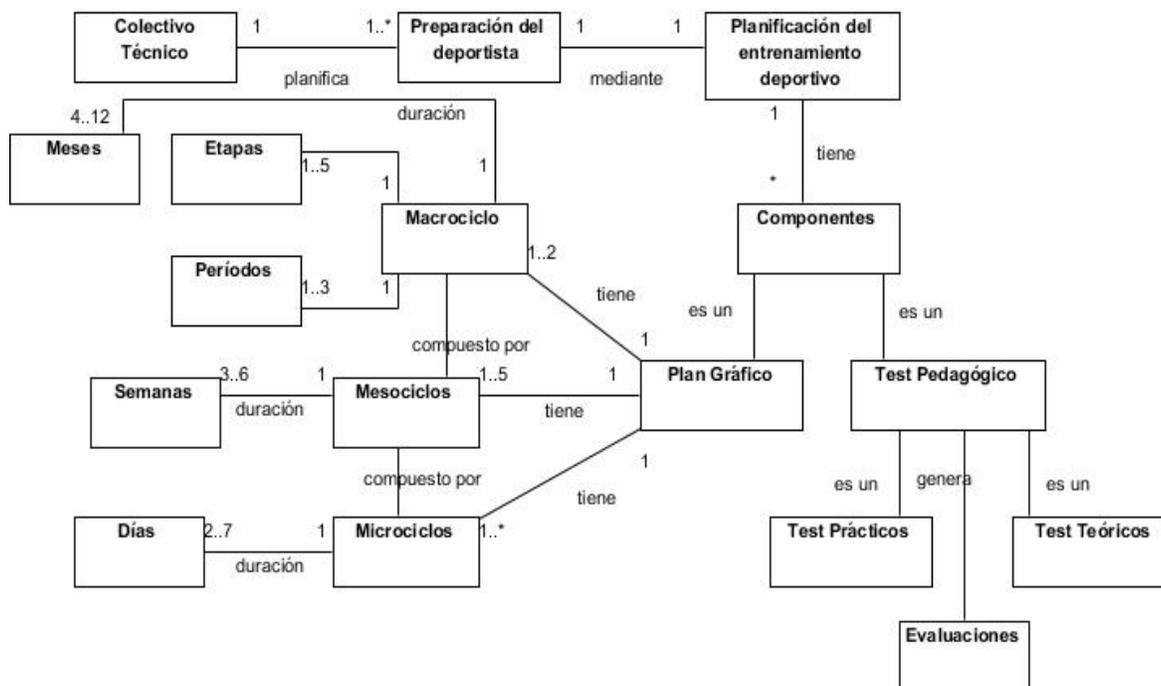


Imagen 4 Modelo conceptual del negocio.

## 2.4 Funcionalidades de la propuesta de solución

La funcionalidad del sistema (FS) se refiere a la condición o la capacidad que debe cumplir el mismo. Las FS identificadas para llevar a cabo el desarrollo de la propuesta de solución son:

### Módulo “Plan gráfico”

**FS. 1:** Gestionar Plan gráfico: El sistema debe permitir adicionar un plan gráfico, para después ver toda la planificación realizada en un formato de tabla, además de exportarlo y eliminarlo si el usuario así lo desea.

**FS. 1.1:** Crear Plan gráfico.

**FS. 1.2:** Visualizar tabla.

**FS. 1.3:** Exportar a PDF.

**FS. 1.4:** Eliminar Plan gráfico.

**FS. 2:** Gestionar macrociclo: El sistema debe permitir adicionarle a un plan gráfico, los macrociclos correspondientes para luego ver la información almacenada, modificarla o eliminarla en caso de ser necesario.

**FS. 2.1:** Adicionar macrociclo.



**FS. 2.2:** Mostrar macrociclo.

**FS. 2.3:** Modificar macrociclo.

**FS. 2.4:** Eliminar macrociclo.

**FS. 3:** Gestionar período: El sistema debe permitir adicionarle a un macrociclo, los períodos correspondientes para luego ver la información almacenada, modificarla o eliminarla en caso de ser necesario.

**FS. 3.1:** Adicionar período.

**FS. 3.2:** Mostrar período.

**FS. 3.3:** Modificar período.

**FS. 3.4:** Eliminar período.

**FS. 4:** Gestionar etapa: El sistema debe permitir adicionarle a un período, las etapas correspondientes para luego ver la información almacenada, modificarla o eliminarla en caso de ser necesario.

**FS. 4.1:** Adicionar etapa.

**FS. 4.2:** Mostrar etapa.

**FS. 4.3:** Modificar etapa.

**FS. 4.4:** Eliminar etapa.

**FS. 5:** Gestionar mesociclo: El sistema debe permitir adicionarle a una etapa, los mesociclos correspondientes para luego ver la información almacenada, modificarla o eliminarla en caso de ser necesario.

**FS. 5.1:** Adicionar mesociclo.

**FS. 5.2:** Mostrar mesociclo.

**FS. 5.3:** Modificar mesociclo.

**FS. 5.4:** Eliminar mesociclo.

**FS. 6:** Gestionar microciclo: El sistema debe permitir adicionarle a una etapa, los mesociclos correspondientes para luego ver la información almacenada, modificarla o eliminarla en caso de ser necesario.

**F. 6.1:** Adicionar microciclo.

**FS. 6.2:** Mostrar microciclo.

**FS. 6.3:** Modificar microciclo.

**FS. 6.4:** Eliminar microciclo.



### **Módulo “Test pedagógico”**

**FS. 1:** Gestionar *test* pedagógico: El sistema debe permitir crear un *test* pedagógico, para luego modificar la información almacenada, exportarla a PDF o eliminarla si el usuario así lo desea.

**FS. 1.1:** Crear *test* pedagógico.

**FS. 1.2:** Modificar *test* pedagógico.

**FS. 1.3:** Eliminar *test* pedagógico.

**FS. 1.4:** Exportar *test* pedagógico.

**FS. 2:** Gestionar escala: El sistema debe permitir adicionar una escala de evaluación, para después modificar la información almacenada o eliminarla si el usuario así lo desea. Además de adicionar uno o varios rangos a una escala determinada.

**FS. 2.1:** Crear escala de evaluación de *test* pedagógico.

**FS. 2.2:** Modificar escala de evaluación.

**FS. 2.3:** Eliminar escala de evaluación.

**FS. 2.4:** Adicionar rango.

**FS. 3:** Registrar resultados de *test* pedagógico: El sistema debe permitir asignarle un *test* a un atleta, además de almacenar los resultados de los *test* realizados por cada uno de ellos para generar una evaluación.

**FS. 4:** Evaluar *test* pedagógico: El sistema debe permitir mostrar un listado por atleta de los *test* que ha realizado con sus respectivos resultados y evaluaciones, además de eliminarle a un atleta un *test* que se le haya asignado.

## **2.5 Exploración**

Es la primera de las etapas de la metodología XP definidas en la investigación. En esta fase, el cliente define los requerimientos que necesita mediante la redacción de las historias de usuarios, basado en esto los programadores estiman los tiempos de desarrollo. Tiene una duración de dos semanas aproximadamente. Por lo general esta fase tiene una duración de dos de semanas aproximadamente. (Joskowicz, 2008, p.8-11)



### 2.5.1 Historia de usuario (HU) de la propuesta de solución

Las HU son escritas por el cliente, en su propio lenguaje, como descripciones cortas de lo que el sistema debe realizar. Las historias de usuario deben estar muy bien detalladas para que los programadores puedan realizar una estimación poco riesgosa del tiempo que llevará su desarrollo. En cualquier momento del desarrollo las HU pueden ser modificadas, o incluso añadir nuevas. (Joskowicz, 2008, p.8-11)

XP no propone ningún formato específico para las historias de usuario, dejando los detalles a consideración del equipo de desarrollo, por lo que se definió una plantilla que contendrá, el nombre, el número, el usuario que realiza la acción dentro del sistema, la estimación del tiempo de desarrollo, el nivel de prioridad en el negocio, el riesgo en caso de no realizarse, la iteración en la que será implementada, una breve descripción y un apartado para agregar alguna observación si es necesario.

**Tabla 2 HU- Crear Plan gráfico.**

Historia de usuario	
<b>Número:</b> 1	<b>Nombre:</b> Crear Plan gráfico
<b>Usuario:</b> Jefe de colectivo técnico	
<b>Prioridad:</b> Alta	<b>Riesgo:</b> Alto
<b>Estimación:</b> 2 días	<b>Iteración:</b> 1
<p><b>Descripción:</b> Al marcar la opción Plan Gráfico de la interfaz principal del módulo se muestra un botón Nuevo que conduce a una vista con los datos generales para crear un plan gráfico, los mismos se muestran a continuación:</p> <ul style="list-style-type: none"> <li>- Nombre</li> <li>- Lugar de la competencia</li> <li>- Año</li> <li>- Tipo de preparación</li> <li>- Deporte</li> <li>- Pronóstico</li> </ul> <p>Una vez introducidos los datos, se selecciona el botón Guardar para que los datos queden registrados y se muestra una lista actualizada con el nuevo plan gráfico.</p>	
<p><b>Observación:</b> El usuario deberá estar autenticado con alguno de los siguientes roles: Jefe de colectivo técnico o Administrador. El sistema le indicará los campos que sean obligatorios.</p>	

**Tabla 3 HU- Adicionar macrociclo**

Historia de usuario	
<b>Número:</b> 2	<b>Nombre:</b> Adicionar macrociclo
<b>Usuario:</b> Jefe de colectivo técnico	
<b>Prioridad:</b> Alta	<b>Riesgo:</b> Alto
<b>Estimación:</b> 2 día	<b>Iteración:</b> 1
<p><b>Descripción:</b> De la lista de planes gráficos, se selecciona el plan gráfico deseado y se marca el ícono de macrociclos que aparece en la columna de acciones. A continuación se muestra un listado de los macrociclos asociados a ese plan, al marcar el botón Nuevo de la interfaz, se</p>	



muestra una vista con los datos necesarios para crear un macrociclo, los mismos se muestran a continuación:

- Nombre
- Objetivo
- Nombre de entrenador
- Modalidad
- Rama

Una vez introducidos los datos, se selecciona el botón Guardar para que los datos queden registrados y se muestra una lista actualizada con el nuevo macrociclo.

**Observación:** El sistema le indicará los campos que sean obligatorios.

Tabla 4 HU-Adicionar período.

Historia de usuario	
<b>Número:</b> 6	<b>Nombre:</b> Adicionar período
<b>Usuario:</b> Jefe de colectivo técnico	
<b>Prioridad:</b> Alta	<b>Riesgo:</b> Alto
<b>Estimación:</b> 1 día	<b>Iteración:</b> 1
<b>Descripción:</b> De la lista de macrociclos, se selecciona uno y se marca el ícono de períodos que aparece en la columna de acciones. A continuación se muestra un listado de los períodos asociados a ese macrociclo, al marcar el botón Nuevo de la interfaz, se muestra una vista con los datos necesarios para crear un período, los mismos se muestran a continuación:	
<ul style="list-style-type: none"> <li>- Nombre</li> <li>- Descripción</li> </ul>	
Una vez introducidos los datos, se selecciona el botón Guardar para que los datos queden registrados y se muestra una lista actualizada con el nuevo período.	
<b>Observación:</b> El sistema le indicará los campos que sean obligatorios.	

Tabla 5 HU- Visualizar tabla.

Historia de usuario	
<b>Número:</b> 22	<b>Nombre:</b> Visualizar tabla
<b>Usuario:</b> Entrenador, Jefe de colectivo técnico	
<b>Prioridad:</b> Alta	<b>Riesgo:</b> Alto
<b>Estimación:</b> 5 días	<b>Iteración:</b> 2
<b>Descripción:</b> Al seleccionar un plan gráfico de la lista, se marca la opción ver tabla que se aparece en la columna de acciones y se muestra una vista con la representación del plan gráfico.	
<b>Observación:</b> El usuario deberá estar autenticado con alguno de los siguientes roles: Jefe de colectivo técnico o Entrenador.	

Tabla 6 HU-Exportar Plan Gráfico a PDF

Historia de usuario	
<b>Número:</b> 23	<b>Nombre:</b> Exporta Plan gráfico a PDF
<b>Usuario:</b> Entrenador, Jefe de colectivo técnico	
<b>Prioridad:</b> Baja	<b>Riesgo:</b> Medio
<b>Estimación:</b> 3 días	<b>Iteración:</b> 2



<b>Tiempo Real:</b> 3 días
<b>Descripción:</b> De una lista de planes gráficos se selecciona el que se desea exportar, se escoge la opción exportar a PDF que se muestra en la columna de acciones y se guarda en la ruta especificada por el usuario.
<b>Observación:</b> El usuario deberá estar autenticado con alguno de los siguientes roles: Jefe de colectivo técnico o Entrenador.

Tabla 7 HU- Crear Test pedagógico.

Historia de usuario	
<b>Número:</b> 25	<b>Nombre:</b> Crear Test pedagógico
<b>Usuario:</b> Entrenador	
<b>Prioridad:</b> Alta	<b>Riesgo:</b> Alto
<b>Estimación:</b> 2 días	<b>Iteración:</b> 3
<p><b>Descripción:</b> Al marcar la opción Test Pedagógico de la interfaz principal del módulo se muestra un botón Nuevo que conduce a una vista con los datos generales para crear un test pedagógico de tipo práctico, los mismos se muestran a continuación:</p> <ul style="list-style-type: none"> <li>- Nombre</li> <li>- Objetivo</li> <li>- Ejecución</li> <li>- Materia</li> <li>- Anotación</li> </ul> <p>Una vez introducidos los datos, se selecciona el botón Guardar para que los datos queden registrados y se muestra una lista actualizada con el nuevo test pedagógico.</p>	
<b>Observación:</b> El usuario deberá estar autenticado con alguno de los siguientes roles: Entrenador.	

Tabla 8 HU- Crear escala de evaluación.

Historia de usuario	
<b>Número:</b> 27	<b>Nombre:</b> Crear escala de evaluación
<b>Usuario:</b> Entrenador	
<b>Prioridad:</b> Alta	<b>Riesgo:</b> Alto
<b>Estimación:</b> 2 días	<b>Iteración:</b> 3
<p><b>Descripción:</b> Al marcar la opción Escalas de la interfaz principal del módulo se muestra un botón Nuevo que conduce a una vista con los datos generales para crear una escala, los mismos se muestran a continuación:</p> <ul style="list-style-type: none"> <li>- División</li> <li>- Rango de evaluación</li> <li>- Evaluación</li> </ul> <p>Una vez introducidos los datos, se selecciona el botón Guardar para que los datos queden registrados y se muestra una lista actualizada con la nueva escala.</p>	
<b>Observación:</b> El usuario deberá estar autenticado con alguno de los siguientes roles: Entrenador.	

Tabla 9 HU- Registrar resultados de test pedagógico.

Historia de usuario	
<b>Número:</b> 30	<b>Nombre:</b> Registrar resultados de test pedagógico



<b>Usuario:</b> Entrenador	
<b>Prioridad:</b> Alta	<b>Riesgo:</b> Medio
<b>Estimación:</b> 2 días	<b>Iteración:</b> 3
<p><b>Descripción:</b> Al marcar la opción Registrar resultados de la interfaz principal del módulo se muestra una vista con los campos necesarios a llenar para recoger los resultados, los mismos se muestran a continuación:</p> <ul style="list-style-type: none"> <li>- Asignar atleta</li> <li>- Asignar <i>test</i></li> <li>- Resultados</li> <li>- Observaciones</li> </ul> <p>Una vez introducidos los datos, se selecciona el botón Guardar para que los datos queden registrados y si se desea agregar más resultados se selecciona el botón Nuevo de la misma interfaz.</p>	
<p><b>Observación:</b> El usuario deberá estar autenticado con alguno de los siguientes roles: Entrenador.</p>	

Tabla 10 HU- Evaluar *test* pedagógico.

Historia de usuario	
<b>Número:</b> 31	<b>Nombre:</b> Evaluar <i>test</i> pedagógico
<b>Usuario:</b> Entrenador	
<b>Prioridad:</b> Alta	<b>Riesgo:</b> Medio
<b>Estimación:</b> 2 días	<b>Iteración:</b> 3
<p><b>Descripción:</b> Al marcar la opción Atleta <i>test</i> de la interfaz principal del módulo se muestra una lista con los atletas a los que se les ha aplicado algún <i>test</i>, se selecciona uno, se debe marcar el ícono ver evaluaciones que aparece en la columna de acciones y se muestra una tabla con todas las evaluaciones que obtuvo el atleta.</p>	
<p><b>Observación:</b> El usuario deberá estar autenticado con alguno de los siguientes roles: Entrenador</p>	

Tabla 11 HU- Exportar *test* pedagógico.

Historia de usuario	
<b>Número:</b> 34	<b>Nombre:</b> Exportar <i>test</i> pedagógico
<b>Usuario:</b> Entrenador	
<b>Prioridad:</b> Alta	<b>Riesgo:</b> Medio
<b>Estimación:</b> 1 día	<b>Iteración:</b> 3
<p><b>Descripción:</b> De una lista de atletas se selecciona el que se desea exportar, se escoge la opción exportar a PDF que se muestra en la columna de acciones y se guarda en la ruta especificada por el usuario.</p>	
<p><b>Observación:</b> El usuario deberá estar autenticado con alguno de los siguientes roles: Entrenador.</p>	

Fueron descritas por parte del cliente un total de 34 HU, en este epígrafe se muestran las principales, las restantes se encuentran en el anexo 1 de la investigación.



## 2.5.2 Prototipos de interfaz de usuario

Los prototipos de interfaz no son más que una forma de representar en una maqueta ciertos aspectos del diseño del producto, propiciando la retroalimentación de los usuarios acerca del sistema. (Kendall y Kendall, 2005, p.178)

Aunque la elaboración de los prototipos no siempre es necesario, se debe observar que hay dos ventajas principales relacionadas con su uso: (Kendall y Kendall, 2005, p.178)

- La aptitud para cambiar a tiempo el sistema en su desarrollo.
- La posibilidad de desarrollar un sistema que se ajuste más a las necesidades y expectativas de los usuarios.

Con el fin de ajustarse a las necesidades del usuario, para el desarrollo de la propuesta de solución se diseñaron los siguientes prototipos:

The screenshot shows a web browser window titled "Módulo PLan Gráfico". The page header includes "Plataforma de Gestión del Entrenamiento Deportivo" and "Bienvenido Jefe Colectivo Técnico" with a "Logout" button. A left sidebar menu contains: "Módulo Plan Gráfico", "Insertar Plan Gráfico", "Consultar Plan Gráfico", "Interpretación Gráfica", and "Modificar Plan Gráfico". The main content area is titled "Nuevo Plan Gráfico" and has two tabs: "Datos Generales" (selected) and "Periodos". The form contains the following fields:

Nombre del plan Gráfico	Competencia Fundamental	Año
<input type="text"/>	<input type="text"/>	<input type="text"/>
[ Tipo de Preparación ] ▾	[ Seleccione el Deporte ] ▾	[ Seleccione el País ] ▾
Pronósticos		
Oros	Platas	Bronces
<input type="text"/>	<input type="text"/>	<input type="text"/>

Imagen 5 Prototipo de interfaz de usuario Crear plan gráfico.



Adicionar Mesociclo

Nombre del Mesociclo:

Descripción:

Características del Mesociclo:

Adicionar Microciclo:

Imagen 6 Prototipo de interfaz de usuario Adicionar mesociclo.

Adicionar Microciclo

Número del Microciclo:

Dia Inicio:   Dia Fin:

Tipo de Característica:

Tipo de Competencia:

Frecuencia:

Horas Microciclo:

Total de Juegos:

Volumen:

Tests

- Test Pedagógico
- Test Psicológico
- Test Médico

Imagen 7 Prototipo de interfaz de usuario Adicionar microciclo.



Plataforma para la planificación y control del entrenamiento en deportes de combate

Test Pedagógicos / Test pedagógico

Test Pedagógicos

Test pedagógico

Crear escala

Registrar resultado

Atletas

Datos del test pedagógico

Nombre

Fecha

Objetivo

Objetivo

Material

Listado

Guardar

Imagen 8 Prototipo de interfaz de usuario Crear Test pedagógico.

Plataforma para la planificación y control del entrenamiento en deportes de combate

Test Pedagógico / Registrar resultado

Test Pedagógicos

Test pedagógico

Crear escala

Registrar resultado

Atletas

Registrar resultado

Atleta

Seleccione el atleta

Test

Seleccione el test

Resultado

Observaciones

Guardar

Imagen 9 Prototipo de interfaz de usuario Registrar resultados.



El diseño final de estos prototipos de interfaz se encuentra en el anexo 5. En este se pueden encontrar otros diseños de interfaz pertenecientes a la propuesta de solución.

## 2.6 Planificación

La planificación es una fase corta, en la que se acuerda entre el cliente y el equipo de desarrollo el orden en que deberán implementarse las historias de usuario y, asociadas a éstas, las entregas. En esencia esta fase consiste en un dialogo continuo entre los involucrados en el proyecto. (Joskowicz, 2008, p.8-11)

Esta fase como resultado genera artefactos de suma importancia para el desarrollo del proyecto, como el Plan de Iteraciones y el Plan de Entregas, los cuales se muestran a continuación.

### 2.6.1 Plan de entregas

El cronograma de entregas establece qué historias de usuario serán agrupadas para conformar una entrega y el orden de las mismas. El plan de entregas se realiza en base a las estimaciones de tiempos de desarrollo realizadas por los desarrolladores. Luego de una reunión en la que el cliente planteó sus prioridades junto al equipo de desarrollo de la propuesta de solución se definió el siguiente plan de entregas:

**Tabla 12 Plan de entregas.**

Entregable	1ra entrega(2 <sup>da</sup> semana de Abril)	2da entrega(3 <sup>ra</sup> semana de Mayo)
Módulos “Plan gráfico” integrado al Sistema para la planificación y control del entrenamiento en deportes de combate.	Versión 2.0	-
Módulos “Test pedagógico” integrado al Sistema para la planificación y control del entrenamiento en deportes de combate.	-	Versión 2.1

**Tabla 13 Funcionalidades disponibles por entrega del producto.**

Historia de Usuario	1ra entrega	2da entrega

*Módulos “Plan gráfico” y “Test pedagógico” del Sistema para la planificación y control del entrenamiento en deportes de combate*



HU1. Crear Plan gráfico	X	-
HU2. Adicionar macrociclo	X	-
HU3. Mostrar macrociclo	X	-
HU4. Modificar macrociclo	X	-
HU5. Eliminar macrociclo	X	-
HU6. Adicionar período	X	-
HU7. Mostrar período	X	-
HU8. Modificar período	X	-
HU9. Eliminar período	X	-
HU10. Adicionar etapa	X	-
HU11. Mostrar etapa	X	-
HU12. Modificar etapa	X	-
HU13. Eliminar etapa	X	-
HU14. Adicionar mesociclo	X	-
HU15. Mostrar mesociclo	X	-
HU16. Modificar mesociclo	X	-
HU17. Eliminar mesociclo	X	-
HU18. Adicionar microciclo	X	-
HU19. Mostrar microciclo	X	-
HU20. Modificar microciclo	X	-
HU21. Eliminar microciclo	X	-
HU23. Visualizar tabla	X	-
HU22. Exportar a PDF	X	-
HU24. Eliminar Plan gráfico	X	-
HU25. Crear <i>Test</i> pedagógico	-	X
HU26. Modificar <i>test</i> pedagógico	-	X
HU27. Crear escala de evaluación	-	X
HU28. Modificar escala de evaluación	-	X



HU29. Adicionar rango	-	X
HU30. Registrar resultados de <i>test</i> pedagógico	-	X
HU31. Evaluar <i>test</i> pedagógico	-	X
HU32. Eliminar <i>test</i> pedagógico	-	X
HU33. Eliminar escala de evaluación	-	X
HU34. Exportar <i>test</i> pedagógico	-	X

### 2.6.2 Iteraciones

Esta es la fase principal en el ciclo de desarrollo de XP. Las funcionalidades son desarrolladas en esta fase, generando al final de cada una un entregable funcional que implementa las historias de usuario asignadas a la iteración. Las iteraciones son también utilizadas para medir el progreso del proyecto. Una iteración terminada sin errores es una medida clara de avance. (Joskowicz, 2008, p. 8-11)

Las historias de usuarios seleccionadas para cada entrega son desarrolladas y probadas en un ciclo de iteración, de acuerdo al orden preestablecido.

Tabla 14 Planificación de las iteraciones.

Iteraciones	Orden de las HU a implementar	Cantidad de tiempo de trabajo
Iteración 1	HU1. Crear plan gráfico HU2. Adicionar macrociclo HU3. Mostrar macrociclo HU4. Modificar macrociclo HU5. Eliminar macrociclo HU6. Adicionar período HU7. Mostrar período HU8. Modificar período	4 semanas <sup>18</sup>

<sup>18</sup> En las semanas se desarrolla los días comprendidos de lunes a viernes.

Módulos “Plan gráfico” y “Test pedagógico” del Sistema para la planificación y control del entrenamiento en deportes de combate



	<p>HU9. Eliminar período</p> <p>HU10. Adicionar etapa</p> <p>HU11. Mostrar etapa</p> <p>HU12. Modificar etapa</p> <p>HU13. Eliminar etapa</p>	
Iteración 2	<p>HU14. Adicionar mesociclo</p> <p>HU15. Mostrar mesociclo</p> <p>HU16. Modificar mesociclo</p> <p>HU17. Eliminar mesociclo</p> <p>HU18. Adicionar microciclo</p> <p>HU19. Mostrar microciclo</p> <p>HU20. Modificar microciclo</p> <p>HU21. Eliminar microciclo</p> <p>HU22. Visualizar tabla</p> <p>HU23. Exportar a PDF</p> <p>HU24. Eliminar plan gráfico</p>	4 semanas
Iteración 3	<p>HU25. Crear <i>test</i> pedagógico</p> <p>HU26. Modificar <i>test</i> pedagógico</p> <p>HU27. Crear escala de evaluación</p> <p>HU28. Modificar escala de evaluación</p> <p>HU29. Adicionar rango</p> <p>HU30. Registrar resultados de <i>test</i> pedagógico</p> <p>HU31. Evaluar <i>test</i> pedagógico</p> <p>HU32. Eliminar <i>test</i> pedagógico</p> <p>HU33. Eliminar escala de evaluación</p> <p>HU34. Exportar <i>test</i> pedagógico</p>	3 semanas



## 2.7 Descripción de la arquitectura

Symfony es el *framework* sobre el cual se desarrolla la propuesta de solución y está basado en patrón arquitectónico Modelo-Vista-Controlador (MVC). La arquitectura MVC divide una aplicación interactiva en tres componente: (Potencier y Zaninotto, 2015)

- **Modelo:** contiene la funcionalidad básica y los datos. Esta capa tiene dos funciones importantes el acceso a los datos y la de abstracción de la base de datos, es aquí donde actúa el ORM Doctrine. Las clases y archivos relacionados con el modelo se guardan en el directorio lib/model/ del proyecto.
- **Vista:** muestran la información al usuario, por lo que se consigue un mantenimiento más sencillo de las aplicaciones<sup>19</sup>. Esta capa se divide en layout: para los elementos que se mantienen invariable en una aplicación y las plantillas: estas sólo se encargan de visualizar las variables definidas en el controlador. Todas las plantillas son guardadas en varios directorios templates/ dentro de las aplicaciones del proyecto.
- **Controlador:** maneja la entrada del usuario, o sea se encarga de obtener los datos del modelo y pasárselos a la vista, procesar las peticiones, las sesiones de los usuarios, la autenticación y de otras acciones en dependencia de la aplicación que se desea construir. Es por esto que esta capa tiene un controlador frontal, que es único para cada aplicación y las acciones, que incluyen el código específico del controlador de cada página.

Symfony toma lo mejor de la arquitectura MVC y la implementa de forma que el desarrollo de aplicaciones sea rápido y sencillo. En la siguiente imagen se puede apreciar la representación en concreto del patrón arquitectónico MVC en la propuesta de solución.

---

<sup>19</sup> Por ejemplo, si se debe mostrar una misma aplicación en varios dispositivos, solamente se crearía una vista para cada uno, manteniendo el modelo y el controlar.

# Módulos “Plan gráfico” y “Test pedagógico” del Sistema para la planificación y control del entrenamiento en deportes de combate

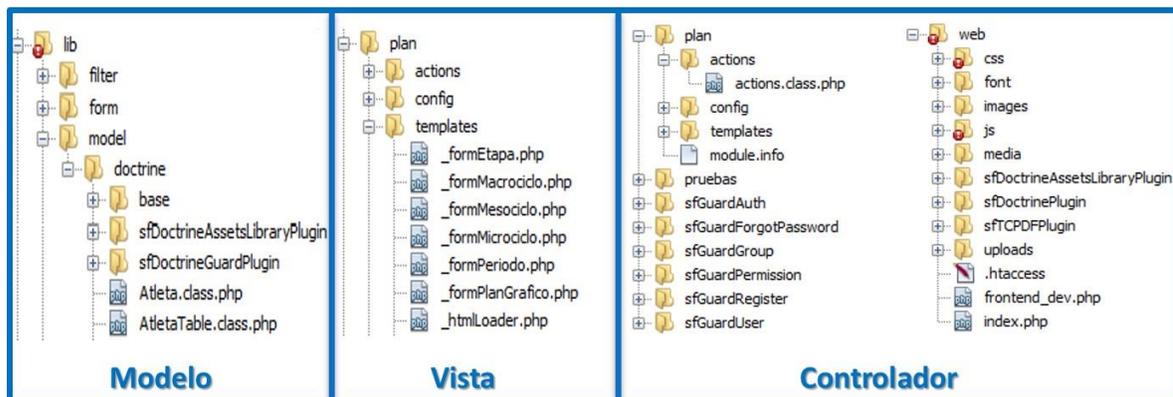


Imagen 10 Representa del patrón MVC en la propuesta de solución.

De esta forma, cuando un usuario accede al sistema, ya sea a través de una red interna o internet, es atendido por un controlador frontal. Los controladores frontales son los únicos puntos de acceso a la aplicación, estos redireccionan al usuario a la acción solicitada del módulo correspondiente. Las acciones piden al modelo toda información solicitada por el usuario, en caso de ser necesario. Esta información se le suministra a la plantilla asociada a la acción solicitada. La plantilla es insertada dentro del Layout de la aplicación, y posteriormente el controlador frontal devuelve al usuario la vista solicitada. Este flujo descrito puede apreciarse en la siguiente imagen.

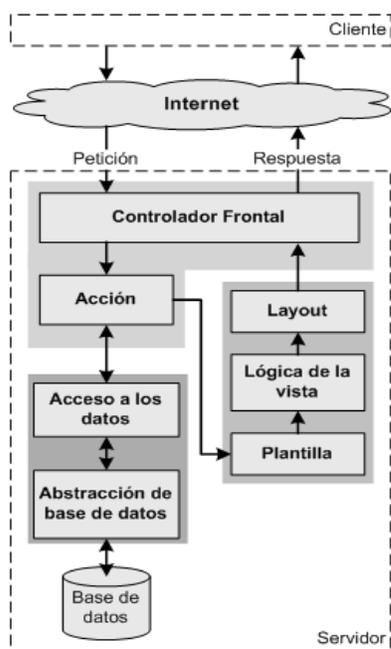


Imagen 11 Representación del patrón arquitectónico MVC en Symfony 1.4.



## 2.8 Patrones de diseño

“Un patrón de diseño es una descripción de clases y objetos comunicándose entre sí adaptada para resolver un problema de diseño general en un contexto particular” (Gamma et al., 1995) Los patrones de diseño intentan codificar el conocimiento, las expresiones y los principios ya existentes. El *framework* Symfony trae implícito en su implementación un conjunto de patrones de diseño que fueron definidos por sus desarrolladores para dar solución a problemas específicos del diseño orientado a objetos durante el flujo de ejecución de una petición. Los patrones que se describen a continuación son los que implementa el *framework* utilizado para el desarrollo de la propuesta de solución.

### 2.8.1 Patrones Generales de Software para Asignar Responsabilidades<sup>20</sup>

“Los patrones GRASP describen los principios fundamentales de la asignación de responsabilidades a objetos, expresados en forma de patrones.” (Larman, 1999, p.191)

**Experto:** Este patrón tiene como solución “asignar una responsabilidad al experto en información: la clase que cuenta con la información necesaria para realizar la responsabilidad.” (Larman, 1999, p.193) Este patrón es utilizado por Symfony en la capa de abstracción en el modelo con la inclusión de la librería Doctrine para mapear la base de datos. Symfony mediante esta librería genera las clases con todas las funcionalidades comunes de las entidades. (Potencier y Zaninotto, 2015)

**Creador:** Este patrón “...guía la asignación de responsabilidades relacionadas con la creación de objetos...” (Larman, 1999, p.198) Es utilizado en los controladores, en ellos se encuentran las acciones definidas para el sistema. En la implementación de las acciones se crean instancias de las clases del modelo y de los formularios que representan a estas clases. (Potencier y Zaninotto, 2015)

**Alta Cohesión:** Este patrón tiene como solución “asignar una responsabilidad de modo que la cohesión siga siendo alta.” (Larman, 1999, p.203) En el marco del diseño orientado a objetos, la cohesión es una medida de cuán relacionadas y enfocadas están las responsabilidades de una clase. (Larman, 1999) La alta cohesión se evidencia en los controladores que poseen un conjunto de funcionalidades, existiendo estrecha relación entre algunas. Ejemplo de ello lo constituyen las acciones create y update, que al crear o

---

<sup>20</sup> Del inglés General Responsibility Assignment Software Patterns.



actualizar un objeto, realizan las validaciones mediante la acción processForm. (Potencier y Zaninotto, 2015)

**Bajo Acoplamiento:** Este patrón tiene como solución “asignar una responsabilidad para mantener bajo acoplamiento.” (Larman, 1999, p.200) El acoplamiento no es más que una medida de la fuerza con que una clase está conectada a otras clases, con que las conoce y con que recurre a ellas. (Larman, 1999, p.200) En Symfony las clases que implementan la lógica del negocio y de acceso a datos se encuentran en el modelo y no tienen asociaciones con las de la vista o el controlador, lo que propicia que la dependencia en este caso sea baja. (Potencier y Zaninotto, 2015)

**Controlador:** Este ofrece una guía para elegir los controladores que manejen los eventos del sistema. Un controlador sirve como intermediario entre una interfaz y la acción que se desee ejecutar. (Larman, 1999) En Symfony este patrón se evidencia en la capa Controlador del patrón arquitectónico MVC en las clases sfFrontController, sfWebFrontController, sfContext, y todas las peticiones son manipuladas por un solo controlador frontal que es único punto de entrada de toda la aplicación. (Potencier y Zaninotto, 2015)

## **2.8.2 Patrones Gang of Four<sup>21</sup>**

**Singleton:** Es un patrón de tipo creación a nivel de objetos, ya que abstrae el proceso de creación de instancias. Con el mismo se garantiza que una clase solo tenga una única instancia, propiciando un punto de acceso global a la misma. (Larman, 1999) Una implementación de este patrón en Symfony se ve presente en sfRouting, clase que es utilizada por el controlador frontal sfWebFrontController, para enrutar todas las peticiones hechas a la aplicación, mediante el método getInstance. (Potencier y Zaninotto, 2015)

**Decorador:** Es un patrón de tipo estructura, ya que describen la forma en que diferentes tipos de objetos pueden ser organizados para trabajar unos y otros. Su propósito es el de añadir la funcionalidad de un objeto dinámicamente. En la capa Vista del patrón arquitectónico MVC que aplica Symfony la clase abstracta sfView es el padre de todas las vistas. El archivo layout.php que es la plantilla global que guarda el código HTML que es usual en todas las páginas del sistema, para no tener que repetirlo en cada página. (Potencier y Zaninotto, 2015)

---

<sup>21</sup> Reciben su nombre por el colectivo de autores denominado Banda de los Cuatro



**Command:** Es de tipo comportamiento a nivel de objetos, ya que ayuda a definir la comunicación e interacción entre los objetos del sistema, reduciendo el acoplamiento entre los mismos. La esencia de este patrón es encapsular en un objeto la acción que satisface una petición. (Larman, 1999) En el *framework* este patrón se observa en la clase `sfWebFrontController` que mediante el método `dispatch()` se encarga de establecer el módulo y la acción que se va a usar según la petición del usuario. (Potencier y Zaninotto, 2015)

**Registry:** Este patrón está orientado a compartir datos y objetos en la aplicación sin la necesidad de preocuparse por conservar numerosos parámetros o hacer uso de variables globales. El *framework* Symfony aplica este patrón en clase `sfConfig`, que se encargada de acumular todas las variables de uso global en el sistema. (Potencier y Zaninotto, 2015)

## 2.9 Seguridad en el sistema

Un tema importante a tener en cuenta en el desarrollo de una aplicación *web* es la seguridad informática, con el fin de garantizar la disponibilidad, confidencialidad, integridad y buen uso de la información que se maneja en un los módulos “Plan gráfico” y “Test pedagógico”. Por tal motivo se diseñaron un grupo de procedimientos en conjunto con los que provee el *framework* Symfony para lograr que el procesamiento de los datos en la aplicación proceda de manera segura.

Uno de los procedimientos empleados fue la autenticación del sistema mediante usuario y contraseña, cada usuario deberá autenticarse previamente para poder realizar cualquier acción en los módulos “Plan gráfico” y “Test pedagógico”. Para su implementación se utilizó el componente `sfGuardPlugin` donde los roles y credenciales de los usuarios se manejan mediante varios grupos de usuarios y permisos asociados al usuario o al grupo de usuarios. Además este mecanismo se apoya en ficheros de configuración YAML, esta sintaxis utilizada en el archivo `security.yml` permite restringir el acceso a usuarios que tienen una combinación de credenciales, usando asociaciones de tipo AND y OR. De esta manera los usuarios son capaces de acceder solo a la información que se le está permitida en dependencia del rol establecido en el sistema, garantizando así la confidencialidad de la información. (Potencier y Zaninotto, 2015)

Un ataque típico de las aplicaciones *web* es el de XSS (*Cross Site Scripting*), que pone en riesgo la integridad del sistema. XSS es un ataque de inyección de código malicioso para su posterior ejecución. (Aguilar Domínguez, 2012)



Para evitar ataques de este tipo durante el trabajo con los módulos, se hizo empleo de los mecanismos de escape que incluye Symfony, para las variables disponibles en las vistas evitando la ejecución de código JavaScript malicioso. Se aplica a todos los datos mostrados mediante las variables de las plantillas. Este mecanismo se encuentra activado por defecto mediante el archivo settings.yml de la aplicación. (Potencier y Zaninotto, 2015)

## **2.10 Conclusiones**

La realización del presente capítulo permitió definir las funcionalidades a implementar mediante el levantamiento de requisitos funcionales y no funcionales, logrando una visión concreta en la descripción de los mismos. Igualmente se dejaron claras las iteraciones por las cuales va a transitar la aplicación, señalando las historias de usuario que serán implementadas en cada una de ellas. Con la descripción de los patrones de arquitectura y diseño utilizados, se logró una guía para la construcción de la propuesta de solución.



## Capítulo 3 Implementación y pruebas

Para planificar el trabajo desde el punto de vista técnico las historias de usuario son divididas en tareas para las cuales también se realiza una estimación. Los desarrolladores definen las dependencias que existen entre las funcionalidades. Las pruebas de *software* forman parte de las fases que propone XP, con el objetivo de lograr una herramienta que cumpla con cada uno de los requerimientos previamente identificados. XP divide las pruebas de *software* en dos grupos, las pruebas unitarias y las pruebas de aceptación. Las pruebas unitarias son las encargadas de verificar el código y son diseñadas por los programadores. Las pruebas de aceptación están destinadas a evaluar si al final de una iteración se consiguió la funcionalidad que se esperaba, estas pruebas usualmente son diseñadas por el usuario o cliente final. En el presente capítulo se describe la fase de codificación y la validación de la herramienta desarrollada.

### 3.1 Tarjetas Clase-Responsabilidades-Colaboradores (CRC)

Las tarjetas CRC fueron presentadas por Beck y Cunningham en 1989 para la enseñanza de la Programación Orientada a Objetos (OOP). (Beck y Cunningham, 1989) Esta técnica se usa para el modelado conceptual y diseño de sistemas OO. La simple elaboración de las mismas y el formato físico que presentan permiten la interacción entre los participantes del proyecto, siendo este uno de los principios de la metodología empleada para el desarrollo de la aplicación. En las tarjetas CRC se representan las entidades del sistema, a las cuales se les asignan responsabilidades y colaboraciones. En la implementación del sistema, estas tarjetas CRC se convierten en clases con métodos, atributos, relaciones de herencia, composición o dependencia. A continuación se muestra un fragmento de las tarjetas CRC correspondientes a los módulos plan gráfico y *test* pedagógico. (Fowler y Scott, 1999, p.8)

Tabla 15 Tarjeta CRC: planActions

Nombre de la clase: planActions	
Responsabilidades	Colaboraciones
executeIndex()	
executePermitirGuardarContenido()	
Gestionar plan gráfico	
executeIndexPlanGrafico()	
executeListarPlanGrafico()	Plan_Grafico, Plan_GraficoTable,
executeShowPlanGrafico()	Plan_Grafico, Plan_GraficoTable
executeNewPlanGrafico()	Plan_Grafico, Plan_GraficoTable, formPlanGrafico

*Módulos “Plan gráfico” y “Test pedagógico” del Sistema para la planificación y control del entrenamiento en deportes de combate*



executeCreatePlanGrafico()	Plan_Grafico, Plan_GraficoTable, formPlanGrafico
executeEditPlanGrafico()	Plan_Grafico, Plan_GraficoTable, formPlanGrafico
executeUpdatePlanGrafico()	Plan_Grafico, Plan_GraficoTable, formPlanGrafico
executeDeletePlanGrafico()	Plan_Grafico, Plan_GraficoTable
processFormPlanGrafico()	formPlanGrafico
<b>Gestionar macrociclo</b>	
executeIndexMacrociclo()	
executeListarMacrociclo()	Macrociclo, MacrocicloTable,
executeShowMacrociclo()	Macrociclo, MacrocicloTable
executeNewMacrociclo()	Macrociclo, MacrocicloTable, formMacrociclo
executeCreateMacrociclo()	Macrociclo, MacrocicloTable, formMacrociclo
executeEditMacrociclo()	Macrociclo, MacrocicloTable, formMacrociclo
executeUpdateMacrociclo()	Macrociclo, MacrocicloTable, formMacrociclo
executeDeleteMacrociclo()	Macrociclo, MacrocicloTable,
processFormMacrociclo()	formMacrociclo
<b>Gestionar período</b>	
executeIndexPeriodo()	
executeListarPeriodo()	Periodo, PeriodoTable
executeShowPeriodo()	Periodo, PeriodoTable
executeNewPeriodo()	Periodo, PeriodoTable, formPeriodo
executeCreatePeriodo()	Periodo, PeriodoTable, formPeriodo
executeEditPeriodo()	Periodo, PeriodoTable, formPeriodo
executeUpdatePeriodo()	Periodo, PeriodoTable, formPeriodo
executeDeletePeriodo()	Periodo, PeriodoTable,
processFormPeriodo()	formPeriodo

**Tabla 16 Tarjetas CRC: testActions.**

<b>Nombre de la clase: testActions</b>	
<b>Responsabilidades</b>	<b>Colaboraciones</b>
executeIndex()	
executePermitirGuardarContenido()	
<b>Gestionar test pedagógico</b>	
executeIndexTestPedagogico()	
executeListarTestPedagogico()	Test_Pedagogico, Test_PedagogicoTable
executeShowTestPedagogico()	Test_Pedagogico, Test_PedagogicoTable
executeNewTestPedagogico()	Test_Pedagogico, Test_PedagogicoTable, formTestPedagogico
executeCreateTestPedagogico()	Test_Pedagogico, Test_PedagogicoTable, formTestPedagogico
executeEditTestPedagogico()	Test_Pedagogico, Test_PedagogicoTable, formTestPedagogico
executeUpdateTestPedagogico ()	Test_Pedagogico, Test_PedagogicoTable, formTestPedagogico
executeDeleteTestPedagogico()	Test_Pedagogico, Test_PedagogicoTable



processFormTestPedagogico ()	formTestPedagogico
<b>Gestionar resultados</b>	
executeIndexAtletaTest()	
executeListarAtletaTest()	Atleta_Test, Atleta_TestTable
executeShowAtletaTest()	Atleta_Test, Atleta_TestTable
executeNewAtletaTest()	Atleta_Test, Atleta_TestTable, formAtletaTest
executeCreateAtletaTest()	Atleta_Test, Atleta_TestTable, formAtletaTest
executeEditAtletaTest()	Atleta_Test, Atleta_TestTable, formAtletaTest
executeUpdateAtletaTest()	Atleta_Test, Atleta_TestTable, formAtletaTest
executeDeleteMesociclo()	Atleta_Test, Atleta_TestTable
processFormAtletaTest()	formAtletaTest

### 3.2 Diagrama Entidad Relación (DER)

El DER que se propone a continuación representa el modelo físico correspondiente a la base de datos del módulo “Plan gráfico” y “Test Pedagógico” respectivamente, expresando las entidades más relevantes del negocio así como su interrelación y propiedades.

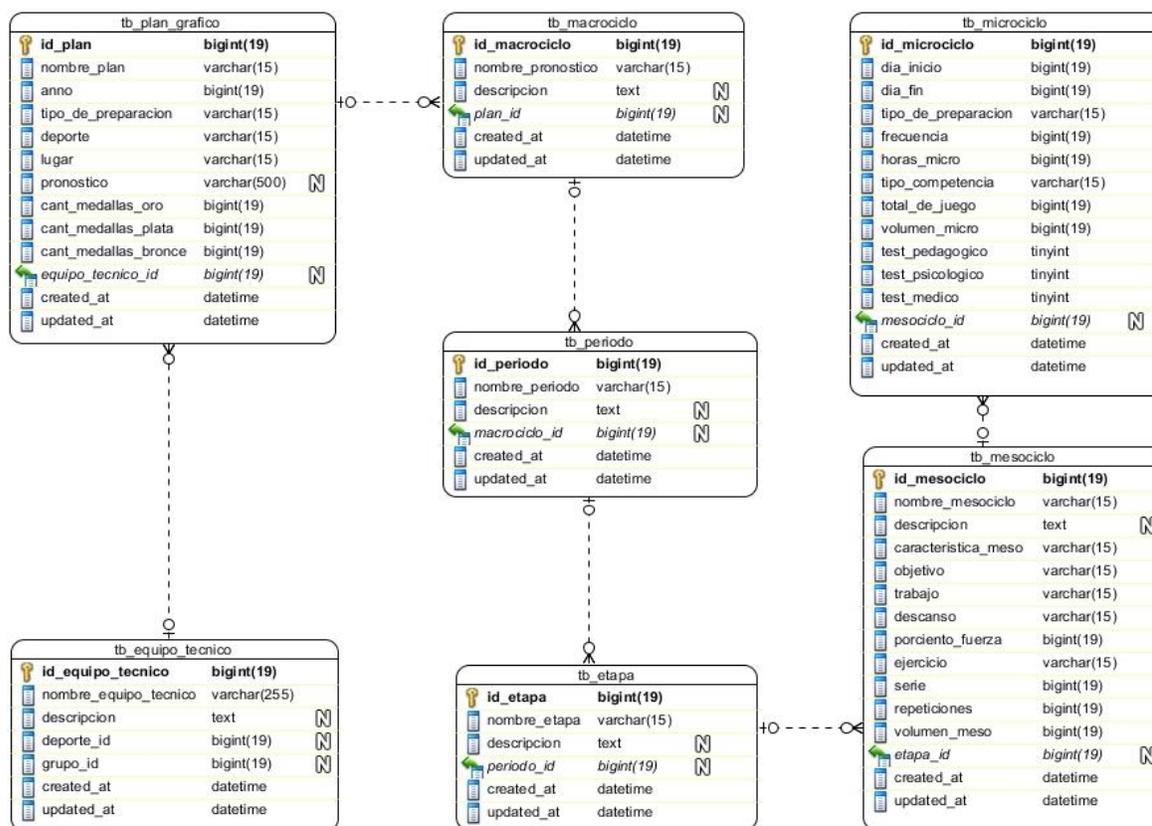


Imagen 12 Diagrama Entidad Relación correspondiente al módulo “Plan gráfico”.

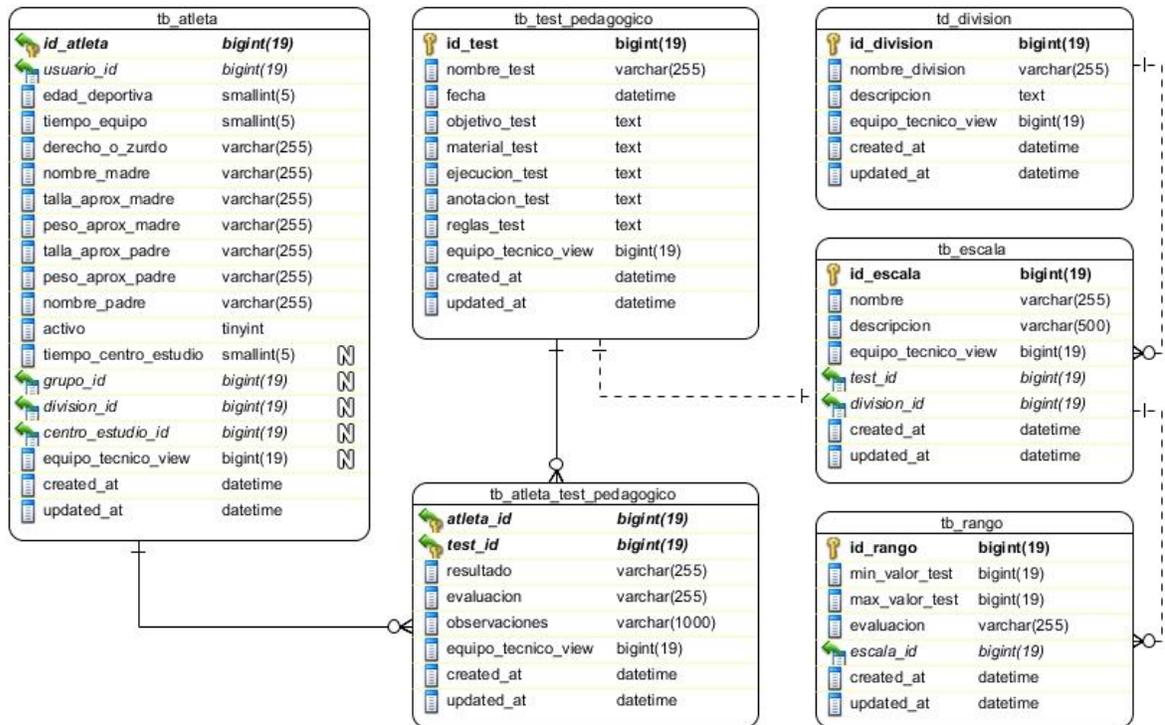


Imagen 13 Diagrama Entidad Relación correspondiente al módulo “Test pedagógico”.

A continuación se muestra una descripción de las tablas que conforma la base de datos que la aplicación utiliza para hacer persistir la información que gestiona:

- **tb\_plan\_grafico:** Es la encargada de almacenar la información general de un plan gráfico, el cual pertenece a un equipo técnico. La misma está compuesta por uno o varios macrociclos, por lo que su relación con esta será de uno a muchos.
- **tb\_macro ciclo:** Pertenece a un plan gráfico y presenta una relación con la tabla *tb\_periodes* de uno a muchos. Es la encargada de almacenar la información básica de un macrociclo.
- **tb\_periodes:** Almacena información referente a los diferentes periodos. Cada período pertenece a un macrociclo y tiene una relación de uno a muchos con la tabla *tb\_etapas*.
- **tb\_etapas:** Cada etapa pertenece a un determinado periodo. En ella se almacena la información referente a las etapas. Existen etapas que pueden tener uno o varios mesociclos, por lo que su relación es de uno a muchos.



- **tb\_mesociclo:** Es la encargada de almacenar toda la información de los mesociclos, cada mesociclo pertenece a una etapa. A su vez pueden tener varios microciclos, por lo que su relación con la tabla tb\_microciclos será de uno a muchos.
- **tb\_microciclo:** Esta tabla guarda todos los datos de los microciclos, cada microciclo pertenecerá a un mesociclo.
- **tb\_equipo\_tecnico:** En ella se guardan los datos de los equipos técnicos. Un equipo técnico está compuesto por varios miembros de equipo técnico y presenta uno o varios planes gráficos. El equipo técnico y los miembros de equipo técnico tienen que poseer el mismo deporte dentro de sus datos. Posee una relación de uno a muchos con la tabla tb\_miembro\_equipo\_tecnico y la tabla tb\_plan\_grafico.
- **tb\_plan\_grafico:** Es la encargada de almacenar la información general de un plan gráfico, el cual pertenece a un equipo técnico. La misma está compuesta por uno o varios macrociclos, por lo que su relación con esta será de uno a muchos.
- **tb\_macro ciclo:** Pertenece a un plan gráfico y presenta una relación con la tabla tb\_periodo de uno a muchos. Es la encargada de almacenar la información básica de un macrociclo.
- **tb\_periodo:** Almacena información referente a los diferentes periodos. Cada periodo pertenece a un macrociclo y tiene una relación de uno a muchos con la tabla tb\_etapa.
- **tb\_etapa:** Cada etapa pertenece a un determinado periodo. En ella se almacena la información referente a las etapas. Existen etapas que pueden tener uno o varios mesociclos, por lo que su relación es de uno a muchos.
- **tb\_mesociclo:** Es la encargada de almacenar toda la información de los mesociclos, cada mesociclo pertenece a una etapa. A su vez pueden tener varios microciclos, por lo que su relación con la tabla tb\_microciclos será de uno a muchos.
- **tb\_microciclo:** Esta tabla guarda todos los datos de los microciclos, cada microciclo pertenecerá a un mesociclo.
- **tb\_equipo\_tecnico:** En ella se guardan los datos de los equipos técnicos. Un equipo técnico está compuesto por varios miembros de equipo técnico y presenta uno o varios planes gráficos. El equipo técnico y los miembros de equipo técnico tienen que tener el mismo deporte como dato. Posee una relación de uno a muchos con la tabla tb\_miembro\_equipo\_tecnico y la tabla tb\_plan\_grafico.



### **3.3 Estilos y estándares de codificación**

A lo largo del ciclo de vida de un *software* un aspecto importante es la codificación basada en estándares de estilos que ayuden a asegurar la calidad del mismo. La codificación de los módulos a desarrollar cumple con los siguientes estándares:

- Se le aplica indentación al código, mediante la combinación de teclas ALT+SHIFT+F del NetBeans IDE.
- Los comentarios multilíneas se escriben comenzando con los caracteres “/\*” y terminando con “\*/”, los comentarios de una sola línea comienzan con los caracteres “/”.
- Las líneas de código que debido a su tamaño ocupan más del espacio visible que ofrece el NetBeans IDE para escribir código (se refiere al ancho de la ventana maximizada), deben ser divididas en varias, preferiblemente después de una coma o un operador.
- Los nombres de las clases del modelo utilizan el estándar UpperCamelCase, las clases controladoras y los métodos utilizan en sus nombres el estándar lowerCamelCase.
- Los nombres de las tablas son en singular y en minúscula, comenzando por el prefijo “tb\_”.
- Todos los atributos de las clases del modelo están encapsulados, son privados y el acceso a ellos se hace por medio de los métodos getAtributo y/o setAtributo.
- Las variables deben ser explícitas, aunque se pueden usar abreviaturas siempre y cuando no se viole este principio.

Con la utilización de estos estándares de codificación en el desarrollo de la propuesta de solución, se logra una mayor homogeneidad con el código que conforma el Sistema para la planificación y control del entrenamiento en deportes de combate, lo que facilitará el mantenimiento futuro de la aplicación. Además como resultado de haberlos aplicado se obtendrá un código con más limpieza, claridad y organización propiciando un mejor entendimiento del mismo.

### **3.4 Tareas de ingeniería**

Con el objetivo de facilitar el proceso de implementación a los desarrolladores, XP propone dividir cada HU en tareas de ingeniería, las cuales tendrán una duración de entre uno y tres días aproximadamente. Para las HU de la propuesta de solución se elaboraron una serie de tareas, de las cuales a continuación se muestran las que están asociadas a las HU del capítulo anterior.



Tabla 17 Tarea #1.

Tarea	
<b>Número:</b> 2	<b>Número de HU:</b> 1
<b>Nombre:</b> Crear Plan gráfico	
<b>Tipo de tarea:</b> Desarrollo	<b>Estimación:</b> 1 días
<b>Fecha inicio:</b> 17 de Febrero de 2015	<b>Fecha fin:</b> 17 de Febrero de 2015
<b>Programador responsable:</b> Maday C. Larduet Pedroso, Frank E. Morell Fuentes	
<b>Descripción:</b> Implementación de la funcionalidad para adicionar los datos generales de un plan gráfico.	

Tabla 18 Tarea #3.

Tarea	
<b>Número:</b> 4	<b>Número de HU:</b> 2
<b>Nombre:</b> Adicionar macrociclo	
<b>Tipo de tarea:</b> Desarrollo	<b>Estimación:</b> 1 días
<b>Fecha inicio:</b> 19 de Febrero de 2015	<b>Fecha fin:</b> 19 de Febrero de 2015
<b>Programador responsable:</b> Maday C. Larduet Pedroso, Frank E. Morell Fuentes	
<b>Descripción:</b> Implementación de la funcionalidad para adicionar un macrociclo al plan gráfico.	

Tabla 19 Tarea #8.

Tarea	
<b>Número:</b> 8	<b>Número de HU:</b> 6
<b>Nombre:</b> Adicionar período	
<b>Tipo de tarea:</b> Investigación	<b>Estimación:</b> 1 día
<b>Fecha inicio:</b> 25 de Febrero de 2015	<b>Fecha fin:</b> 25 de Febrero de 2015
<b>Programador responsable:</b> Maday C. Larduet Pedroso, Frank E. Morell Fuentes	
<b>Descripción:</b> Investigar la información que maneja un período de entrenamiento.	

Tabla 20 Tarea #29.

Tarea	
<b>Número:</b> 29	<b>Número de HU:</b> 22
<b>Nombre:</b> Visualizar tabla	
<b>Tipo de tarea:</b> Desarrollo	<b>Estimación:</b> 5 días
<b>Fecha inicio:</b> 31 de Marzo de 2015	<b>Fecha fin:</b> 3 de Abril de 2015
<b>Programador responsable:</b> Maday C. Larduet Pedroso, Frank E. Morell Fuentes	



**Descripción:** Implementación de la funcionalidad para mostrar la tabla que representa el plan gráfico con toda la información que se fue generando anteriormente.

**Tabla 21 Tarea #30.**

Tarea	
<b>Número:</b> 30	<b>Número de HU:</b> 23
<b>Nombre:</b> Exportar a PDF	
<b>Tipo de tarea:</b> Desarrollo	<b>Estimación:</b> 5 días
<b>Fecha inicio:</b> 6 de Abril de 2015	<b>Fecha fin:</b> 6 de Abril de 2015
<b>Programador responsable:</b> Maday C. Larduet Pedroso, Frank E. Morell Fuentes	
<b>Descripción:</b> Implementación de la funcionalidad para guardar en un documento en formato PDF el plan gráfico.	

**Tabla 22 Tarea #33.**

Tarea	
<b>Número:</b> 33	<b>Número de HU:</b> 25
<b>Nombre:</b> Crear <i>Test</i> pedagógico	
<b>Tipo de tarea:</b> Desarrollo	<b>Estimación:</b> 1 día
<b>Fecha inicio:</b> 21 de Abril de 2015	<b>Fecha fin:</b> 21 de Abril de 2015
<b>Programador responsable:</b> Maday C. Larduet Pedroso, Frank E. Morell Fuentes	
<b>Descripción:</b> Implementación de la funcionalidad para adicionar un <i>test</i> pedagógico.	

**Tabla 23 Tarea #36.**

Tarea	
<b>Número:</b> 36	<b>Número de HU:</b> 27
<b>Nombre:</b> Crear escala de evaluación de <i>test</i> pedagógico	
<b>Tipo de tarea:</b> Desarrollo	<b>Estimación:</b> 3 día
<b>Fecha inicio:</b> 24 de Abril de 2015	<b>Fecha fin:</b> 24 de Abril de 2015
<b>Programador responsable:</b> Maday C. Larduet Pedroso, Frank E. Morell Fuentes	
<b>Descripción:</b> Implementación de la funcionalidad para adicionar una nueva escala de evaluación.	

**Tabla 24 Tarea #39.**

Tarea	
<b>Número:</b> 39	<b>Número de HU:</b> 31
<b>Nombre:</b> Registrar resultados de <i>test</i> pedagógico	
<b>Tipo de tarea:</b> Desarrollo	<b>Estimación:</b> 2 días
<b>Fecha inicio:</b> 29 de Abril de 2015	<b>Fecha fin:</b> 30 de Abril de 2015
<b>Programador responsable:</b> Maday C. Larduet Pedroso, Frank E. Morell Fuentes	



**Descripción:** Implementación de la funcionalidad para reunir los resultados de todos los test aplicados a un atleta.

Tabla 25 Tarea #41.

Tarea	
<b>Número:</b> 41	<b>Número de HU:</b> 33
<b>Nombre:</b> Evaluar test pedagógico	
<b>Tipo de tarea:</b> Desarrollo	<b>Estimación:</b> 1 día
<b>Fecha inicio:</b> 4 de Mayo de 2015	<b>Fecha fin:</b> 4 de Mayo de 2015
<b>Programador responsable:</b> Maday C. Larduet Pedroso, Frank E. Morell Fuentes	
<b>Descripción:</b> Implementación de la funcionalidad para emitir una evaluación en dependencia de los resultados recogidos.	

Tabla 26 Tarea #44.

Tarea	
<b>Número:</b> 44	<b>Número de HU:</b> 23
<b>Nombre:</b> Exportar a PDF	
<b>Tipo de tarea:</b> Desarrollo	<b>Estimación:</b> 5 días
<b>Fecha inicio:</b> 7 de Mayo de 2015	<b>Fecha fin:</b> 7 de Mayo de 2015
<b>Programador responsable:</b> Maday C. Larduet Pedroso, Frank E. Morell Fuentes	
<b>Descripción:</b> Implementación de la funcionalidad para guardar en un documento en formato PDF los atletas con sus evaluaciones.	

### 3.5 Pruebas

Uno de los pilares de XP es el proceso de pruebas. XP anima a probar constantemente tanto como sea posible. Esto permite aumentar la calidad de los sistemas reduciendo el número de errores no detectados y disminuyendo el tiempo transcurrido entre la aparición de un error y su detección. (Gutiérrez et al. 2015) Es por ello que una de las características más importantes de XP es su marcado énfasis en la práctica de TDD<sup>22</sup> (Desarrollo dirigido por las pruebas).

XP divide las pruebas del sistema en dos grupos: pruebas unitarias, encargadas de verificar el código y diseñada por los programadores, y pruebas de aceptación o pruebas funcionales destinadas a evaluar si al final de una iteración se consiguió la funcionalidad requerida por el cliente. (Gutiérrez et al. 2015)

<sup>22</sup> Del inglés Test Driven Development



### 3.5.1 Pruebas unitarias

Las pruebas unitarias constituyen un factor importante para la metodología XP. Todos los módulos deben de pasar las pruebas unitarias antes de ser liberados o publicados. Las pruebas unitarias son establecidas antes de escribir el código y son ejecutadas constantemente ante cada modificación del sistema. (Joskowicz, 2008, p.15)

Las pruebas unitarias aseguran que un determinado módulo cumpla con un comportamiento esperado en forma aislada antes de ser integrado al sistema. Existen herramientas automatizadas para realizar este tipo de pruebas en diferentes lenguajes de programación como por ejemplo Java con JUnit y PHP con PHPUnit, por su parte Symfony incluye su propio *framework* para realizar esta tarea, llamado Lime. Este permite ejecutar los archivos de prueba en un entorno independiente para evitar interferencias entre las diferentes pruebas, además son fáciles de leer al igual que sus resultados; consta únicamente de un archivo, llamado lime.php, y no tiene ninguna dependencia. (Potencier y Zaninotto, 2015)

Para la escritura de estas pruebas fueron utilizados los distintos métodos que provee Lime, los cuales se muestran a continuación:

Tabla 27 Métodos del objeto lime\_test para las pruebas unitarias. (Potencier y Zaninotto, 2015)

Método	Descripción
<b>diag</b> (\$mensaje)	Muestra un comentario, pero no ejecuta ninguna prueba
<b>ok</b> (\$condicion[, \$mensaje])	Si la condición que se indica es true, la prueba tiene éxito
<b>is</b> (\$valor1, \$valor2[, \$mensaje])	Compara dos valores y la prueba pasa si los dos son iguales (==)
<b>isnt</b> (\$valor1, \$valor2[, \$mensaje])	Compara dos valores y la prueba pasa si no son iguales
<b>like</b> (\$cadena, \$expresionRegular[, \$mensaje])	Prueba que una cadena cumpla con el patrón de una expresión regular
<b>unlike</b> (\$cadena, \$expresionRegular[, \$mensaje])	Prueba que una cadena no cumpla con el patrón de una expresión regular
<b>cmp_ok</b> (\$valor1, \$operador, \$valor2[, \$mensaje])	Compara dos valores mediante el operador que se indica



<b>isa_ok</b> (\$variable, \$tipo[, \$mensaje])	Comprueba si la variable que se le pasa es del tipo que se indica
<b>isa_ok</b> (\$objeto, \$clase[, \$mensaje])	Comprueba si el objeto que se le pasa es de la clase que se indica
<b>can_ok</b> (\$objeto, \$metodo[, \$mensaje])	Comprueba si el objeto que se le pasa dispone del método que se indica
<b>is_deeply</b> (\$array1, \$array2[, \$mensaje])	Comprueba que dos arreglos tengan los mismos valores
<b>include_ok</b> (\$archivo[, \$mensaje])	Valida que un archivo existe y que ha sido incluido correctamente
<b>fail</b> ([\$mensaje])	Provoca que la prueba siempre falle (es útil para las excepciones)
<b>pass</b> ([\$mensaje])	Provoca que la prueba siempre se pase (es útil para las excepciones)
<b>skip</b> ([\$mensaje, \$numeroPruebas])	Cuenta como si fueran \$numeroPruebas pruebas (es útil para las pruebas condicionales)
<b>todo</b> ([\$mensaje])	Cuenta como si fuera una prueba (es útil para las pruebas que todavía no se han escrito)
<b>comment</b> (\$mensaje)	Muestra el comentario indicado pero no cuenta como ninguna prueba
<b>error</b> (\$mensaje)	Muestra el mensaje de error indicado pero no cuenta como ninguna prueba
<b>info</b> (\$mensaje)	Muestra el mensaje informativo indicado pero no cuenta como ninguna prueba

### 3.5.2 Pruebas de aceptación

Las pruebas de aceptación se crearán a partir de las historias de usuario. Una historia puede tener una o varias pruebas, según sea la funcionalidad a probar. Las pruebas de aceptación son consideradas como pruebas de caja negra. Los clientes son responsables de verificar que los resultados de estas pruebas sean correctos. Una historia de usuario no se puede considerar terminada hasta tanto pase correctamente todas las pruebas de aceptación. (Joskowicz, 2008, p.15)



Para validar la solución implementada se definieron un total de 34 casos de prueba de aceptación, mostrando a continuación los que están relacionados con las tareas de ingeniería de tipo desarrollo que aparecen en el epígrafe 3.5:

Tabla 28 Caso de prueba de aceptación #1

Caso de prueba de aceptación	
<b>Código:</b> HU1_CP1	<b>Historia de usuario:</b> 1
<b>Nombre:</b> Crear Plan gráfico	
<b>Condiciones de ejecución:</b> El usuario debe estar autenticado con el rol Jefe de colectivo técnico.	
<b>Entrada/ Pasos de ejecución:</b> Entrada: Los datos del plan gráfico que el usuario debe introducir para que sea insertado en la base de datos. Pasos de ejecución: Para insertar los datos, debe ir al menú y seleccionar la opción “Plan Gráfico”. En la vista que se va a mostrar debe presionar en el botón “Nuevo”. Una vez introducidos los datos se presiona en el botón “Guardar” y seguido se le muestra el plan gráfico añadido.	
<b>Resultado esperado:</b> El plan gráfico es adicionado correctamente.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria.	

Tabla 29 Caso de prueba de aceptación #2.

Caso de prueba de aceptación	
<b>Código:</b> HU2_CP2	<b>Historia de usuario:</b> 2
<b>Nombre:</b> Adicionar macrociclo	
<b>Condiciones de ejecución:</b> El usuario debe estar autenticado con el rol Jefe de colectivo técnico y además debe haber algún plan gráfico insertado en la base de datos.	
<b>Entrada/ Pasos de ejecución:</b> Entrada: Los datos del macrociclo que el usuario debe introducir para que sea insertado en la base de datos. Pasos de ejecución: Para insertar los datos, debe ir a la columna de acciones de la tabla de planes gráfico, marcar uno y presionar en el ícono que indica adicionar macrociclo. En la vista que se va a mostrar debe presionar en el botón “Nuevo”. Una vez introducidos los datos se presiona en el botón “Guardar” y seguido se le muestra el macrociclo añadido.	
<b>Resultado esperado:</b> El plan gráfico es adicionado correctamente.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria	

Tabla 30 Caso de prueba de aceptación #6.

Caso de prueba de aceptación	
<b>Código:</b> HU6_CP6	<b>Historia de usuario:</b> 6
<b>Nombre:</b> Adicionar período	
<b>Condiciones de ejecución:</b> El usuario debe estar autenticado con el rol Jefe de colectivo	



técnico y además debe haber algún macrociclo insertado en la base de datos.
<b>Entrada/ Pasos de ejecución:</b> Entrada: Los datos del período que el usuario debe introducir para que sea insertado en la base de datos. Pasos de ejecución: Para insertar los datos, debe ir a la columna de acciones de la tabla de macrociclos, marcar uno y presionar en el ícono que indica adicionar período. En la vista que se va a mostrar debe presionar en el botón “Nuevo”. Una vez introducidos los datos se presiona en el botón “Guardar” y seguido se le muestra el período añadido.
<b>Resultado esperado:</b> El período es adicionado correctamente.
<b>Evaluación de la prueba:</b> Prueba satisfactoria

**Tabla 31 Caso de prueba de aceptación #22**

Caso de prueba de aceptación	
<b>Código:</b> HU22_CP22	<b>Historia de usuario:</b> 22
<b>Nombre:</b> Visualizar tabla	
<b>Condiciones de ejecución:</b> El usuario debe estar autenticado con alguno de los siguientes roles: Jefe de colectivo técnico o Entrenador y además debe haber algún plan gráfico insertado en la base de datos.	
<b>Entrada/ Pasos de ejecución:</b> Pasos de ejecución: Para graficar la tabla, debe ir a la columna de acciones de la tabla de plan gráfico, marcar uno y presionar en el ícono que indica mostrar tabla. En la vista que se va a mostrar se le mostrará una tabla.	
<b>Resultado esperado:</b> El plan gráfico con todos los aspectos introducidos anteriormente.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria	

**Tabla 32 Caso de prueba de aceptación #23.**

Caso de prueba de aceptación	
<b>Código:</b> HU23_CP23	<b>Historia de usuario:</b> 23
<b>Nombre:</b> Exportar a PDF	
<b>Condiciones de ejecución:</b> El usuario debe estar autenticado con alguno de los siguientes roles: Jefe de colectivo técnico o Entrenador y además debe haber algún plan gráfico insertado en la base de datos.	
<b>Entrada/ Pasos de ejecución:</b> Pasos de ejecución: Para guardar la tabla en formato de documento, debe ir a la columna de acciones de la tabla de plan gráfico, marcar uno y presionar en el ícono que indica exportar a PDF.	
<b>Resultado esperado:</b> Un documento PDF que tiene como contenido un plan gráfico.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria	

**Tabla 33 Caso de prueba de aceptación #25.**

Caso de prueba de aceptación
------------------------------



<b>Código:</b> HU25_CP25	<b>Historia de usuario:</b> 25
<b>Nombre:</b> Crear Test pedagógico	
<b>Condiciones de ejecución:</b> El usuario debe estar autenticado con el rol Entrenador.	
<b>Entrada/ Pasos de ejecución:</b> Entrada: Los datos del test pedagógico que el usuario debe introducir para que sea insertado en la base de datos. Pasos de ejecución: Para insertar los datos, debe ir al menú y seleccionar la opción “Test Pedagógico”. En la vista que se va a mostrar debe presionar en el botón “Nuevo”. Una vez introducidos los datos se presiona en el botón “Guardar” y seguido se le muestra el test pedagógico añadido.	
<b>Resultado esperado:</b> El test pedagógico es adicionado correctamente.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria	

Tabla 34 Caso de prueba de aceptación #27.

Caso de prueba de aceptación	
<b>Código:</b> HU27_CP27	<b>Historia de usuario:</b> 27
<b>Nombre:</b> Crear escala de evaluación de test pedagógico	
<b>Condiciones de ejecución:</b> El usuario debe estar autenticado con el rol Entrenador.	
<b>Entrada/ Pasos de ejecución:</b> Entrada: Los datos de la escala que el usuario debe introducir para que sea insertada en la base de datos. Pasos de ejecución: Para insertar los datos, debe ir al menú y seleccionar la opción “Escala”. En la vista que se va a mostrar debe presionar en el botón “Nuevo”. Una vez introducidos los datos se presiona en el botón “Guardar” y seguido se le muestra la escala añadida.	
<b>Resultado esperado:</b> La escala es adicionado correctamente.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria	

Tabla 35 Caso de prueba de aceptación #31.

Caso de prueba de aceptación	
<b>Código:</b> HU31_CP31	<b>Historia de usuario:</b> 31
<b>Nombre:</b> Registrar resultados de test pedagógico	
<b>Condiciones de ejecución:</b> El usuario debe estar autenticado con el rol Entrenador y además debe haber algún test pedagógico insertado en la base de datos.	
<b>Entrada/ Pasos de ejecución:</b> Entrada: Los resultados de los test que el usuario debe introducir para que sean registrados en la base de datos. Pasos de ejecución: Para insertar los datos, debe ir al menú y seleccionar la opción “Registrar resultados”. En la vista que se va a mostrar se introducen los datos y se debe presionar en el botón “Nuevo” para agregar tantos resultados como se desee. Una vez introducidos los datos se presiona en el botón “Guardar”.	
<b>Resultado esperado:</b> Los resultados quedaron recogidos correctamente.	



Evaluación de la prueba: Prueba satisfactoria

Tabla 36 Caso de prueba de aceptación #32.

Caso de prueba de aceptación	
Código: HU32_CP32	Historia de usuario: 32
Nombre: Evaluar <i>test</i> pedagógico	
<b>Condiciones de ejecución:</b> El usuario debe estar autenticado con el rol Entrenador y además debe haber algún <i>test</i> pedagógico y alguna escala insertada en la base de datos.	
<b>Entrada/ Pasos de ejecución:</b> Pasos de ejecución: Para obtener las evaluaciones de los <i>test</i> aplicados a los atletas, se debe ir al menú y seleccionar la opción “Atleta <i>test</i> ”. En la vista que se va a mostrar debe ir a la columna de acciones de la tabla de atletas con evaluaciones, marcar uno y presionar en el ícono que indica mostrar. En la vista que se va a mostrar se le mostrará un resumen con todos los <i>test</i> que ha hecho ese atleta con sus respectivos resultados y evaluaciones.	
<b>Resultado esperado:</b> El <i>test</i> recibe una evaluación correctamente.	
Evaluación de la prueba: Prueba satisfactoria	

Tabla 37 Caso de prueba de aceptación #34.

Caso de prueba de aceptación	
Código: HU34_CP34	Historia de usuario: 34
Nombre: Exportar a PDF	
<b>Condiciones de ejecución:</b> El usuario debe estar autenticado con el rol Entrenador y además debe haber algún <i>test</i> pedagógico insertado en la base de datos.	
<b>Entrada/ Pasos de ejecución:</b> Pasos de ejecución: Para guardar el resumen de las evaluaciones de un <i>test</i> en formato de documento, debe ir a la columna de acciones de la tabla de atleta con evaluaciones, marcar uno y presionar en el ícono que indica exportara PDF.	
<b>Resultado esperado:</b> Un documento PDF que tiene como contenido un plan gráfico.	
Evaluación de la prueba: Prueba satisfactoria	

### 3.6 Análisis de los resultados de las pruebas

Durante el ciclo de vida del software, se realizaron las pruebas pertinentes para garantizar la calidad producto, las tablas que se muestran a continuación son el resultado de dichas pruebas.

Tabla 38 Cantidad de NC detectadas por iteración en las pruebas de aceptación.

	Iteración 2	Iteración 3
Cantidad	19	9

A partir de la realización de las pruebas de aceptación se detectaron no conformidades (NC) no significativas, que se centraron en errores ortográficos como: omisiones de tildes y exceso de mayúsculas.

## *Módulos “Plan gráfico” y “Test pedagógico” del Sistema para la planificación y control del entrenamiento en deportes de combate*



Las significativas encontradas consistieron en errores de validación, aceptar caracteres alfabéticos donde se esperaban valores numéricos, errores en la visualización de los planes gráficos y en la generación de un plan gráfico a formato de documento en PDF. Una vez concluida cada iteración de este tipo de prueba el equipo de desarrollo analizó las no conformidades obtenidas para solucionar los errores.

**Tabla 39 Resultado de la ejecución de las pruebas unitarias.**

Historia de Usuario	Cantidad de ejecuciones	Fallidas	Satisfactorias
HU1. Crear Plan gráfico	45	15	30
HU2. Adicionar macrociclo	45	10	35
HU3. Mostrar macrociclo	45	12	33
HU4. Modificar macrociclo	45	15	30
HU5. Eliminar macrociclo	45	27	18
HU6. Adicionar período	45	5	40
HU7. Mostrar período	45	5	40
HU8. Modificar período	45	5	40
HU9. Eliminar período	45	5	40
HU10. Adicionar etapa	45	9	36
HU11. Mostrar etapa	45	9	36
HU12. Modificar etapa	45	9	36
HU13. Eliminar etapa	45	9	36
HU14. Adicionar mesociclo	30	4	26
HU15. Mostrar mesociclo	30	4	26
HU16. Modificar mesociclo	30	4	26
HU17. Eliminar mesociclo	30	4	26
HU18. Adicionar microciclo	30	7	23
HU19. Mostrar microciclo	30	7	23
HU20. Modificar microciclo	30	7	23
HU21. Eliminar microciclo	30	7	23
HU23. Visualizar tabla	30	22	8



HU22. Exportar a PDF	30	15	15
HU24. Eliminar Plan gráfico	30	8	22
HU25. Crear Test pedagógico	15	5	10
HU26. Modificar test pedagógico	15	3	12
HU27. Crear escala de evaluación	15	3	12
HU28. Modificar escala de evaluación	15	3	12
HU29. Adicionar rango	15	6	9
HU30. Registrar resultados de test pedagógico	15	2	13
HU31. Evaluar test pedagógico	15	10	5
HU32. Eliminar test pedagógico	15	2	13
HU33. Eliminar escala de evaluación	15	2	13
HU34. Exportar test pedagógico	15	11	4

Fueron escritas 34 pruebas y realizadas al finalizar cada funcionalidad. Cada error encontrado en las pruebas unitarias fue corregido inmediatamente, y se tomaron precauciones para que errores similares no volvieran a ocurrir. Además, se generaron nuevas pruebas para verificar que el error haya sido resuelto.

En las siguientes imágenes se representan dos ejemplos de los resultados de las pruebas realizadas al código.

```
"C:\wamp\bin\php\php5.4.12\php.exe" "C:\wamp\www\plataforma\symfony" "--color" "test:unit"
pruebaTest.....errors
Looks like you planned 2 tests but only ran 1.
Errors:
- Doctrine_Connection_Exception: There is no open connection
Failed Test          Stat Total  Fail  Errors  List of Failed
-----
pruebaTest          255    1    0    1
Failed 1/1 test scripts, 0.00% okay. 1/2 subtests failed, 50.00% okay.
Done.
```

Imagen 14 Resultado de una prueba una prueba unitaria fallida.

Esta salida es el resultado de utilizar el método isa\_ok() al cual se le pasó como parámetro un objeto y la clase Atleta\_TestTable, para comprobar si este objeto era instancia de dicha clase, resultando fallida. Una vez detectado el error fue resuelto y comprobado nuevamente arrojado un resultado satisfactorio.



```
"C:\wamp\bin\php\php5.4.12\php.exe" "C:\wamp\www\plataforma\symfony" "--color" "test:unit" "prueba"
1..2
ok 1 - Este test siempre pasa.
ok 2 - Comprueba que sea una instancia de la clase pasada por parámetro.
# Looks like everything went fine.
Done.
```

Imagen 15 Resultado de una prueba unitaria satisfactoria.

### 3.7 Conclusiones

La conclusión de este capítulo permitió asegurar la calidad del producto y la conformidad del cliente con respecto al mismo, mediante la aplicación de pruebas que garantizaron la corrección a tiempo de las no conformidades detectadas. Además fueron definidas las pautas de codificación que enmarcan la estructura que tendrá el código fuente de la solución implementada.



## **Conclusiones generales y Recomendaciones**

### **Conclusiones generales**

Una vez culminada la presente investigación se arribaron a las siguientes conclusiones:

- Se hizo un estudio de los sistemas para la planificación y control del entrenamiento específicamente de las funcionalidades plan gráfico y *test* pedagógico, el cual sirvió de apoyo en la definición de las funcionalidades del sistema.
- Se desarrollaron dos módulos integrados al Sistema para la planificación y control del entrenamiento en deportes de combate, que brindará la posibilidad a los entrenadores de gestionar un plan gráfico y los *test* pedagógicos referente a su deporte.
- Se realizaron las pruebas al sistema, de aceptación y unitarias según define la metodología XP, detectando un total de 28 no conformidades en dos iteraciones, las cuales fueron resueltas al finalizar cada iteración, mejorando la calidad del sistema.
- El cliente quedó satisfecho con la solución entregada, la cual podrá ser implantada en el INDER para facilitar a los equipos técnicos, la realización del plan gráfico y los *test* pedagógicos como parte del proceso de planificación y control del entrenamiento deportivo.

### **Recomendaciones**

Como resultado de la investigación y del proceso de desarrollo de los módulos “Plan gráfico” y “*Test* pedagógico”, han surgido ideas que serían recomendables para futuras versiones del sistema y el perfeccionamiento del mismo.

- Agregar al módulo “Plan gráfico” nuevas funcionalidades que posibiliten crear un plan escrito, propiciando mayor nivel de detalle en el proceso de planificación y control del entrenamiento.
- Agregar al módulo “*Test* pedagógico” nuevas funcionalidades de manera que permita controlar las evaluaciones de las pruebas psicológicas de los atletas.



## Referencias bibliográficas

1. **ACHOUR, et al.** *Manual de PHP*. [En línea]. 16 Enero 2015. [Accedido 22 Enero 2015]. Disponible en: <https://php.net/manual/es/intro-what-is.php>
2. **AGUILAR DOMINGUEZ, Angie.** *¿Qué es y cómo opera un ataque de Cross-Site Scripting (XSS)?* [En línea]. 18 January 2012. [Accedido 12 Mayo 2015]. Disponible en: <http://www.seguridad.unam.mx/documento/?id=35>
3. **ÁLVAREZ, Miguel Angel.** *Introducción a jQuery*. [En línea]. 25 Marzo 2009. [Accedido 23 Enero 2015]. Disponible en: <http://www.desarrolloweb.com/articulos/introduccion-jquery.html>
4. **ÁLVAREZ, Miguel Angel.** *Qué es HTML 5*. [En línea]. 14 Octubre 2009. [Accedido 23 Enero 2015]. Disponible en: <http://www.desarrolloweb.com/articulos/que-es-html5.html>
5. **BAGNARA, Ivan Carlos y PACHECO GARCÍA, C. Israel.** *El desarrollo tecnológico en las actividades físicas y el deporte*. [En línea]. Agosto 2011. [Accedido 19 Marzo 2015]. Disponible en: <http://www.efdeportes.com/efd159/el-desarrollo-tecnologico-en-el-deporte.htm>
6. **CANÓS, José H., LETELIER, Patricio y PENADÉS, Ma. Carmen.** *Metodologías ágiles en el desarrollo de software.pdf*. In : . DSSIC- Universidad Politécnica de Valencia, 2003.
7. **CARVAJAL RIOLA, José Carlos.** *Metodologías ágiles: herramientas y modelo de desarrollo para aplicaciones Java EE como metodología empresarial*. [En línea]. 2008. [Accedido 4 Diciembre 2014]. P.59. Disponible en: <http://upcommons.upc.edu/handle/2099.1/5608>
8. **CARVAJAL RIOLA, José Carlos.** Ob. cit. P.111-115.
9. **CUESTA MORALES, Pedro.** *Desarrollo de Aplicaciones Distribuidas basadas en Tecnologías Web*. [En línea]. Octubre 1999. [Accedido 21 Enero 2015]. Disponible en: <http://trevinca.ei.uvigo.es/~pcuesta/publicaciones/TecWeb.pdf>
10. **DONATIEN GOLIATH, Karenia.** *Sistema para la planificación del entrenamiento deportivo de judo femenino*. [En línea]. Junio 2011. [Accedido 1 Diciembre 2014]. Disponible en: <http://publicaciones.uci.cu/index.php/SC/article/view/703>
11. **EGUILUZ, Javier.** *Introducción a CSS*. [En línea]. [Accedido 23 Enero 2015]. Disponible en: [http://librosweb.es/css/capitulo\\_1.html](http://librosweb.es/css/capitulo_1.html)



12. **EGUILUZ, Javier.** *Introducción a JavaScript.* [En línea]. [Accedido 23 Enero 2015]. Disponible en: [http://librosweb.es/javascript/capitulo\\_1.html](http://librosweb.es/javascript/capitulo_1.html)
13. **FLORES ZAMORA, Argelio Camilo and ALBA MARTÍNEZ, Dixan.** *Visplan: software para la planificación y control del entrenamiento deportivo.* [En línea]. Septiembre 2013. [Accedido 1 Diciembre 2014]. Disponible en: <http://www.efdeportes.com/efd184/visplan-software-para-entrenamiento-deportivo.htm>
14. **FOWLER, Martin and SCOTT, Kendall.** *UML gota a gota.* Pearson Educación, 1999. ISBN 9789684443648. P.8. Disponible en: [https://books.google.com/cu/books?hl=es&lr=&id=AL0YkFeaHwIC&oi=fnd&pg=PT14&dq=tarjetas+crc+esta+tecnica+se+usa&ots=Fv\\_PM25gWq&sig=6eqUo6JRWVwDUi-rBYPhwQoweFw#v=onepage&q=tarjetas%20crc%20esta%20tecnica%20se%20usa&f=false](https://books.google.com/cu/books?hl=es&lr=&id=AL0YkFeaHwIC&oi=fnd&pg=PT14&dq=tarjetas+crc+esta+tecnica+se+usa&ots=Fv_PM25gWq&sig=6eqUo6JRWVwDUi-rBYPhwQoweFw#v=onepage&q=tarjetas%20crc%20esta%20tecnica%20se%20usa&f=false)
15. **GALLARDO RUIZ, José E. and GARCÍA LÓPEZ, Carmen M.** *tema10 Diseño Modular.pdf* [En línea]. [Accedido 29 Noviembre 2014]. Disponible en: <http://www.lcc.uma.es/~pepeg/modula/temas/tema10.pdf>
16. **GAMMA, Erich, HELM, Richard, JOHNSON, Ralph y VLISSIDES, John.** *Design Patterns: Elements of Reusable Object-Oriented Software - DesignPatterns.pdf* [En línea]. 1995. [Accedido 14 Marzo 2015]. ISBN 0201633612. Disponible en: <http://www.uml.org.cn/c++/pdf/DesignPatterns.pdf>
17. **GARCÍA CARVAJAL, Oscar Alfredo.** *Aplicación práctica para la planificación del entrenamiento en la modalidad deportiva del lanzamiento del tejo.* [En línea]. Diciembre 2011. [Accedido 28 Noviembre 2014]. Disponible en: <http://www.efdeportes.com/efd163/entrenamiento-del-lanzamiento-del-tejo.htm>
18. **GONZÁLEZ CATALÁ, Silvio Antonio.** *Los test pedagógicos en la lucha deportiva.* [En línea]. Julio 2003. [Accedido 30 Noviembre 2014]. Disponible en: <http://www.efdeportes.com/efd62/lucha.htm>
19. **GUERRA, MsC. Edecio Perez.** *Las pruebas o tests en el deporte.* [En línea]. November 2008. [Accedido 30 Noviembre 2014]. Disponible en: <http://www.efdeportes.com/efd126/las-pruebas-o-tests-en-el-deporte.htm>
20. **GUTIÉRREZ, J. J., ESCALONA, M. J., MEJÍAS, M. y TORRES, J.** *Pruebas del Sistema en Programación Extrema.* Departamento de Lenguajes y Sistemas Informáticos. Universidad de



- Sevilla [En línea]. [Accedido 14 Abril 2015]. Disponible en: [http://www.lsi.us.es/~javierj/investigacion\\_ficheros/PSISEXTREMA.pdf](http://www.lsi.us.es/~javierj/investigacion_ficheros/PSISEXTREMA.pdf)
21. **IBÁÑEZ, S. J., MACÍAS, M., PÉREZ, M. A. y FEU, S.** *Planificación y control del entrenamiento en baloncesto*. PYC-Basket, software aplicado. Aportaciones al proceso de enseñanza y entrenamiento del baloncesto. 2001. P. 193–201. Disponible en: <http://cib2012.es/anteriores/2001/ibanez.pdf>
22. **INFORMATICA & DEPORTES.** *X-Medalist | Nuevo Software para entrenadores personales*. [En línea]. [Accedido 3 Diciembre 2014]. Disponible en: <http://www.x-medalist.com.ar/>
23. **INFORMÁTICA & DEPORTES.** *X-Training FUSSION | Software para Planificación, Periodización y Control del Entrenamiento Deportivo*. [En línea]. [Accedido 3 Diciembre 2014]. Disponible en: <http://www.x-trainingfussion.com.ar/>
24. **JOSKOWICZ, José.** *Reglas y prácticas en eXtreme Programming*. Universidad de Vigo. 2008. P. 8-11. Disponible en: <http://iie.fing.edu.uy/~josej/docs/XP%20-%20Jose%20Joskowicz.pdf>
25. **JOSKOWICZ, José.** Ob. cit. P.15.
26. **KAPLAN, Gabriel.** *¿Qué es Bootstrap?* [En línea]. 26 Febrero 2014. [Accedido 28 Febrero 2015]. Disponible en: <http://nebaris.com/post/126/que-es-bootstrap>
27. **KENDALL, Kennet E. y KENDALL, Julie E.** *Análisis y diseño de sistemas* [En línea]. Pearson Educación. 2005. [Accedido 2 Marzo 2015]. ISBN 9702605776. P.178. Disponible en: [https://books.google.com/cu/books?id=5-rZA0FggusC&pg=PT207&lpg=PT207&dq=metodologia+xp+prototipo+de+interfaz&source=bl&ots=1D2MFHx2A6&sig=0\\_2A662bTt-pYsfJiVhKZTwJaPc&hl=es&sa=X&ei=F5zzVlbuOlq0sAThrILYAQ&ved=0CCQQ6AEwAQ#v=onepage&q=metodologia%20xp%20prototipo%20de%20interfaz&f=false](https://books.google.com/cu/books?id=5-rZA0FggusC&pg=PT207&lpg=PT207&dq=metodologia+xp+prototipo+de+interfaz&source=bl&ots=1D2MFHx2A6&sig=0_2A662bTt-pYsfJiVhKZTwJaPc&hl=es&sa=X&ei=F5zzVlbuOlq0sAThrILYAQ&ved=0CCQQ6AEwAQ#v=onepage&q=metodologia%20xp%20prototipo%20de%20interfaz&f=false)
28. **LARMAN, Graig.** *UML y Patrones. Introducción al análisis y diseño orientado a objetos*. 1999. ISBN 9701702611. P.15. Disponible en: [http://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=books&cd=1&cad=rja&uact=8&ved=0C CAQFjAA&url=http%3A%2F%2Fbiblioteca.uazuay.edu.ec%2Fopac\\_css%2Findex.php%3Fvl%3Da uthor\\_see%26id%3D33647&ei=hoRLVZnXD\\_PmsATk74CwBw&usg=AFQjCNE9kuCQEKn\\_ZjMR M5DAk3pgUQ70bQ&bvm=bv.92765956,d.cWc](http://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=books&cd=1&cad=rja&uact=8&ved=0C CAQFjAA&url=http%3A%2F%2Fbiblioteca.uazuay.edu.ec%2Fopac_css%2Findex.php%3Fvl%3Da uthor_see%26id%3D33647&ei=hoRLVZnXD_PmsATk74CwBw&usg=AFQjCNE9kuCQEKn_ZjMR M5DAk3pgUQ70bQ&bvm=bv.92765956,d.cWc)
29. **LARMAN, Graig.** Ob. cit. P.191-203.



30. **MACHADO BARÓ, MsC. Juan Gualberto, et al.** *El proceso de institucionalización de la cultura física a partir del triunfo de la revolución cubana (1959)*. 16 Enero 2010. Disponible en: <http://www.portaldeportivo.cl/articulos/FE.0017.pdf>
31. **MANZANO MORENO, J. Ignacio.** *1065698.pdf. Clarificación de conceptos relacionados con el entrenamiento deportivo* [En línea]. 2004. [Accedido 29 Noviembre 2014]. Disponible en: <http://dialnet.unirioja.es/descarga/articulo/1065698.pdf>
32. **MARTIN, Dietrich.** *Manual de metodología del entrenamiento deportiva*. Editorial Paidotribo, 2007. ISBN 9788480195195. P.16. Disponible en: [http://www.google.com/cu/books?hl=en&lr=&id=\\_ehXzkJzpQIC&oi=fnd&pg=PT14&dq=DEFINE+planificacion+deportiva&ots=1LDm5zqqu0&sig=vufa3eKK1Yh4SVm9IOgx7Va78c&redir\\_esc=y#v=onepage&q=DEFINE%20planificacion%20deportiva&f=false](http://www.google.com/cu/books?hl=en&lr=&id=_ehXzkJzpQIC&oi=fnd&pg=PT14&dq=DEFINE+planificacion+deportiva&ots=1LDm5zqqu0&sig=vufa3eKK1Yh4SVm9IOgx7Va78c&redir_esc=y#v=onepage&q=DEFINE%20planificacion%20deportiva&f=false)
33. **MARTÍN MALDONADO, Daniel.** *Apache, el servidor Web más reconocido | Empresa y economía*. [En línea]. 20 April 2008. [Accedido 23 Enero 2015]. Disponible en: <http://www.empresayeconomia.es/aplicaciones-para-empresas/apache-el-servidor-web-mas-reconocido.html>
34. **MARTINEZ, Rafael.** *Portal en español sobre PostgreSQL*. [En línea]. [Accedido 7 Febrero 2015]. Disponible en: <http://www.postgresql.org.es/>
35. **NICOLAUS, Jürgen.** *Metodología general del entrenamiento infantil y juvenil*. Editorial Paidotribo, 2004. ISBN 9788480197922. P.227. Disponible en: <http://books.google.com/cu/books?id=bqwJLdKzJlcC&pg=PA227&lpg=PA227&dq=define+control+de+entrenamiento&source=bl&ots=Chb2lvdNzU&sig=9XqRQwfaNXPumP12spD8CDvxSoQ&hl=es&sa=X&ei=PT55VLuWEsWkNueBg9gF&ved=0CEkQ6AEwBw#v=onepage&q=define%20control%20de%20entrenamiento&f=false>
36. **PEÑA, Ismael, CÓRCOLES, César Pablo y CASADO, Carlos.** *El Profesor 2.0: docencia e investigación desde la Red*. [En línea]. 2006. Vol. n. 3. [Accedido 21 Enero 2015]. Disponible en: <http://ullviu.blog.cat/gallery/12161/12161-68417.pdf>
37. **PONS, Juan de Pablos.** *Entrenamiento deportivo y nuevas tecnologías*. [En línea]. 2010. [Accedido 19 Marzo 2015]. Disponible en: [http://recursos.cnice.mec.es/edfisica/publico/articulos/articulo17/articulo\\_17.php](http://recursos.cnice.mec.es/edfisica/publico/articulos/articulo17/articulo_17.php)



38. **POTENCIER, Fabien y ZANINOTTO, François.** *Symfony 1.4, la guía definitiva.* [En línea]. [Accedido 22 Enero 2015]. Disponible en: [http://librosweb.es/symfony\\_1.4/index.html](http://librosweb.es/symfony_1.4/index.html)
39. **RAPOSO, A. Vasconcelos.** *Planificación y organización del entrenamiento deportivo.* Editorial Paidotribo, 2000. P.134.
40. **RENDÓN GALLÓN, Álvaro.** *Programación Extrema (XP).* [En línea] Universidad del Cauca Abril 2011. [Accedido 14 Abril 2015] Disponible en: <http://dtm.unicauca.edu.co/esptelematica-cali/Arquitecturas/transp/1.2-XP.pdf>
41. **ROCHE, Fernando Paris.** *La planificación estratégica en las organizaciones deportivas.* Editorial Paidotribo, 1998. ISBN 9788480192842. P.14. Disponible en: [http://www.google.com/cu/books?hl=en&lr=&id=qYKvdeXgbjgC&oi=fnd&pg=PA9&dq=DEFINE+planificacion&ots=eEDyxduDhp&sig=RxLsQ3fePonbPstZfKLnVDeBDIA&redir\\_esc=y#v=onepage&q=DEFINE%20planificacion&f=false](http://www.google.com/cu/books?hl=en&lr=&id=qYKvdeXgbjgC&oi=fnd&pg=PA9&dq=DEFINE+planificacion&ots=eEDyxduDhp&sig=RxLsQ3fePonbPstZfKLnVDeBDIA&redir_esc=y#v=onepage&q=DEFINE%20planificacion&f=false)
42. **SHELLY, Gary y ROSENBLATT, Harry J.** *Systems Analysis and Design, Video Enhanced* [En línea]. Cengage Learning, 2010. ISBN 1111793077. Disponible en: [https://books.google.com/cu/books?id=YRg8AAAAQBAJ&pg=PA144&lpg=PA144&dq=agilian+standard+edition+caracteristicas&source=bl&ots=NklGRW5Ysj&sig=DTB\\_xGzq0NTta2mWHGOMyFKDdpo&hl=es&sa=X&ei=DjjWVKjzCcyYyATqy4DoDw&ved=0CDoQ6AEwAw#v=onepage&q=agilian%20standard%20edition%20caracteristicas&f=false](https://books.google.com/cu/books?id=YRg8AAAAQBAJ&pg=PA144&lpg=PA144&dq=agilian+standard+edition+caracteristicas&source=bl&ots=NklGRW5Ysj&sig=DTB_xGzq0NTta2mWHGOMyFKDdpo&hl=es&sa=X&ei=DjjWVKjzCcyYyATqy4DoDw&ved=0CDoQ6AEwAw#v=onepage&q=agilian%20standard%20edition%20caracteristicas&f=false)
43. **TELLO LEAL, Edgar.** *Las tecnologías de la información y comunicaciones (TIC) y la brecha digital.* Revista de Universidad y Sociedad del conocimiento [En línea]. Enero 2008. [Accedido 17 Marzo 2015]. P.3. Disponible en: <http://rusc.uoc.edu/>
44. **VARGAS, René.** *Diccionario de Teoría Del Entrenamiento Deportivo* [En línea]. UNAM, 2007. [Accedido 3 Diciembre 2014]. ISBN 9789703247905. P.160. Disponible en: [http://books.google.com/cu/books?id=QulyPvzBhIC&pg=PA160&dq=plan+grafico&hl=es&sa=X&ei=7Rt\\_VJewOoX4yQS1joG4CA&ved=0CB0Q6AEwAA#v=onepage&q=plan%20grafico&f=false](http://books.google.com/cu/books?id=QulyPvzBhIC&pg=PA160&dq=plan+grafico&hl=es&sa=X&ei=7Rt_VJewOoX4yQS1joG4CA&ved=0CB0Q6AEwAA#v=onepage&q=plan%20grafico&f=false)
45. **Visual Paradigm - Udemy.** *Visual Paradigm Essential* [En línea]. [Accedido 30 Noviembre 2014]. Disponible en: <https://www.udemy.com/visual-paradigm-essential/#>
46. **ZAPATA JARAMILLO, Carlos Mario and ARANGO ISAZA, Fernando.** *Los modelos verbales en lenguaje natural y su utilización en la elaboración de esquemas conceptuales para el desarrollo de software: una revisión crítica - 839-2503-1-PB.pdf.* 2005. Vol. 41, p. 77–95. 137