



**Universidad de las Ciencias Informáticas**

**Facultad 8**

**Sistema informático para el análisis del  
comportamiento de acciones tácticas  
significativas de los equipos de balonmano.**

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

**Autores:**

María de los Angeles Obregón Valdés.

Heimer Lambert Delgado.

**Tutor:**

Ing. Noel Miño Herrera

**Cotutores:**

Lic. Aloy Machado Sánchez

Ing. Humberto Rivero Guevara

Ciudad de La Habana, junio del 2010

“Año 52 de la Revolución”

# Declaración de Autoría

---

## Declaración de Autoría

Declaramos que somos los únicos autores del presente trabajo que lleva por título “Sistema informático para el análisis del comportamiento de acciones tácticas significativas de los equipos de balonmano”, autorizando de esta manera a la Universidad de las Ciencias Informáticas (UCI) a que haga uso de él.

Para que así conste, firmamos la presente a los \_\_\_\_ días del mes de \_\_\_\_\_ del 2010.

**Autores:**

**María de los Angeles Obregón Valdés**

\_\_\_\_\_

**Heimer Lambert Delgado:**

\_\_\_\_\_

**Tutor:**

**Ing. Noel Miño Herrera**

\_\_\_\_\_

**Cotutor:**

**Lic. Aloy Machado Sánchez**

\_\_\_\_\_

**Cotutor:**

**Ing. Humberto Rivero Guevara**

\_\_\_\_\_

# *Dedicatoria*

*A mis padres por haber dedicado su vida a mi buena formación como profesional y como ser humano, porque no se siembran los valores con palabras y conversaciones, sino con acciones y ejemplo y es eso lo que han hecho todo el tiempo con suma elegancia. Es por eso que serán siempre mi meta a alcanzar y la huella que dejen, la que yo pise porque sé que es la más segura y honorable.*

*María de los A. Obregón Valdés*

*A mis padres por haberse esforzado tanto en mi desarrollo como profesional y haberse dedicado a hacer de mí un hombre con valores y capaz de confiar en sí para lograr grandes cosas dentro de ellas este trabajo el cual se lo dedico con todo el corazón.*

*Heimer Lambert Delgado*

# *Agradecimientos Comunes*

*A nuestro Comandante en Jefe Fidel Castro y la Revolución por habernos dado la posibilidad de formarnos como profesionales y estudiar en Universidad de las Ciencias Informáticas, la escuela del futuro.*

*A nuestro tutor Ing. Noel Miño Herrera, cotutores Lic. Aloy Machado Sánchez e Ing. Humberto Rivero Guevara por su ayuda incondicional y paciencia durante todo el proceso de realización del presente trabajo.*

*A nuestros familiares por haber sido la viga fundamental de nuestra formación y brindarnos su apoyo sin reparos en todo los momentos difíciles.*

*A nuestros amigos por haber constituido su amistad un logro exclusivo e irremplazable durante nuestro decursar por la universidad, fue aquí donde encontramos amigos para toda la vida.*

# Agradecimientos

*A mi mamá por ser mi conciencia y darme esos consejos en los momentos más precisos para que me lleguen de la forma que sabe que me hacen reflexionar. Por haber sido la mejor profesora que he tenido. Gracias a ti, mami, llegué a ser "Adulta", siguiendo siempre ese rastro de amor que me vas dejando para que no me pierda en la vida.*

*A mi papá, el hombre más inteligente que he conocido, por hacerme saber que los obstáculos no son tan grandes cuando uno sabe crecerse ante ellos. Por haber sido mi mejor profesor. Gracias a ti, papi, tengo confianza en las cosas que hago porque sé que tú me diste el mejor patrón a seguir y es así como me guío.*

*A mi hermana por ser mi paradigma a seguir. Por enseñarme a hablar y estar ahí en todos los momentos de mi vida e irme dejando a su paso un camino del que estoy orgullosa seguir porque, aunque no llegue a alcanzarte, Miri, sé que ese camino que dejaste es grande.*

*A mis abuelos, Blanquita y Alfredo por tener confianza en mí en todo momento y enseñarme lo grande de esta Revolución y lo afortunada que soy al vivir en este tiempo y por tanto saber aprovechar al máximo las cosas buenas que nos brinda, para de esa manera llegar a formar parte del bando de los hombres que aman y fundan.*

*A mi novio Heimer, por su dedicación para la realización del presente trabajo, por haber estado siempre junto a mí en todos los momentos brindándome apoyo con amor y paciencia y hasta cuando se acababa la última. Te agradezco, mi amor, por estar siempre para mí, porque sé que me has dado de ti lo mejor.*

*María de los A. Obregón Valdés*

# *Agradecimientos*

*Quisiera agradecer a toda mi familia en especial a mi papá Humberto Lambert Lobaina, a mi madre Lidia Delgado Delgado y a mi hermanito querido Humberto Lambert Delgado por la ayuda incondicional que me han dado siempre, por la confianza y seguridad que depositaron en mí, gracias a su fuerza y su amor me han dirigido por la vida y me han dado los consejos que necesitaba para salir adelante en cada uno de los momentos de mi vida. Todo lo que soy, se lo debo a mi familia, atribuyo todos mis éxitos en esta vida a la enseñanza moral, intelectual y física que recibí de ellos.*

*A mi novia María por todo su apoyo durante el desarrollo de este trabajo por ser siempre esa persona alentadora e inspiradora de confianza haciendo grandes cosas con sus acciones, gracias por estar siempre a mí lado en todo momento y recuerda que no soy un santo. Al menos que para ti un santo es un pecador que simplemente sigue esforzándose.*

*Heimer Lambert Delgado*

# Resumen

---

## Resumen

El presente trabajo de diploma que lleva como título: “Sistema informático para el análisis del comportamiento de acciones tácticas significativas de los equipos de balonmano”, tiene como objetivo desarrollar un sistema de registro estadístico informatizado para el análisis del comportamiento de acciones tácticas significativas en los equipos de balonmano. El desarrollo de la aplicación de escritorio fue guiado por los principios y reglas que propone la metodología XP y se obtuvo de esta manera, al final de cada iteración en las que se dividió el proceso de desarrollo del software, un producto. Se muestra además un estudio de tecnologías y herramientas que se utilizan para la realización de software como el que se presenta, que ayudó a hacer una selección justa de las mismas. También se llevó a cabo la realización de pruebas al sistema para garantizarle al cliente la obtención de un producto funcional y con calidad.

El sistema que se presenta mediante este trabajo garantizará que el proceso de recogida de estadísticas deportivas en los juegos de balonmano que se realicen a nivel nacional, se logre con plena funcionalidad a partir de su utilización.

# Índice

---

## Índice

Introducción .....	1
Capítulo 1: Fundamentación teórica .....	5
1. 1    Introducción .....	5
1. 2    Conceptos asociados al problema .....	5
1. 3    Análisis de otras soluciones existentes .....	6
1. 4    Estudio de metodologías .....	10
1.4.1    Proceso Unificado de Desarrollo de Software (RUP) .....	10
1.4.2    Plataforma de Soluciones de Microsoft (MSF) .....	12
1.4.3    SCRUM .....	12
1.4.4    Programación Extrema (XP) .....	13
1.5    Estudio de herramientas .....	15
1. 5. 1    Herramientas de modelado .....	15
1. 5. 2    Herramientas de desarrollo .....	18
1.6    Estudio de lenguajes de programación .....	23
1.7    Estudio de Entornos de Desarrollo Integrado (IDEs) .....	27
1.8    Conclusiones .....	29
Capítulo 2: Desarrollo de la solución .....	31
2.1    Introducción .....	31
2.2    Descripción de las acciones vinculadas al campo de acción .....	31
2.2.1    Flujo actual de eventos .....	31
2.2.2    Objetos de automatización .....	32
2.2.3    Propuesta del Sistema .....	33
2.3    Desarrollo de la solución .....	34
2.4    Personas relacionadas con el sistema .....	35
2.4.1    Historias de usuario (HU) .....	35
2.4.2    Iteraciones .....	39
2.4.3    Estimación de esfuerzo por historia de usuario .....	39
2.5    Plan de entrega .....	40
2.6    Conclusiones .....	41
Capítulo 3: Implementación y pruebas de la solución propuesta .....	42
3.1    Introducción .....	42



# Índice

---

3.2	Diseño del sistema .....	42
3.2.1	Arquitectura y Patrones .....	44
3.3	Fase de Producción.....	47
3.3.1	Iteración 1.....	49
3.3.2	Iteración 2.....	54
3.3.3	Iteración 3.....	57
3.4	Pruebas .....	60
3.4.1	Pruebas de aceptación .....	60
3.5	Conclusiones .....	66
	Conclusiones .....	67
	Recomendaciones .....	68
	Referencias Bibliográficas .....	69
	Glosario de Términos.....	72

# Introducción

---

## Introducción

Actualmente la informática se introduce en muchos ámbitos de la vida del hombre, pues a partir de su surgimiento y evolución se han producido una serie de cambios positivos. Es por esta razón que su desarrollo y utilización son indetenibles en todo el mundo.

Cuba, a pesar de ser un país en vías de desarrollo, ha dedicado un gran esfuerzo al progreso de la informática y uno de sus valiosos resultados es la Universidad de las Ciencias Informáticas (UCI). Entre las características de este centro está la vinculación estudio-producción. Por este precepto existen proyectos productivos en los que trabajan de conjunto estudiantes y profesores que desarrollan software de alta calidad para informatizar los sectores sociales.

En el ámbito deportivo se ha contribuido al desarrollo de las distintas disciplinas. Con la informática se ha propiciado desde la creación de implementos eficientes, hasta sistemas de control tan necesarios a la hora de determinar un resultado. Esto ha permitido, entre otras cosas, ganar en espectáculo y en conocimientos. La Facultad 8 dedica esfuerzos en la creación de aplicaciones vinculadas al deporte, las que viabilizan el manejo de datos que luego son analizados con el objetivo de lograr un mejor rendimiento deportivo.

Entre los deportes desarrollados en Cuba que obtienen menos resultados en competencias internacionales está el balonmano, que no es un deporte masivo en la sociedad cubana. A pesar de estas agravantes el número de aficionados que lo siguen, ha aumentado. Es por ello que la Comisión Nacional de Balonmano ha tomado la tarea de llevar a cabo una investigación en las competencias que se desarrollan en el país con vista a encontrar las posibles causas de los resultados obtenidos.

Los entrenadores de balonmano con el fin de mejorar el rendimiento de cada uno de los jugadores y equipos, desarrollan nuevas técnicas y tácticas de entrenamiento. Pero antes de formular cualquier nueva estrategia se necesita un basamento estadístico de cómo se están comportando cada una de las variables en el juego.

Para lograr dicho objetivo, la investigación plantea recoger en cada uno de los partidos la información de acciones desarrolladas durante el mismo para, al terminar cada competencia sea posible realizar un análisis general del desempeño por equipos y el comportamiento frente a sus contrarios en una competencia o en un ciclo competitivo

# Introducción

---

determinado. Este análisis brinda una solución a parte de la problemática que afronta el balonmano cubano actual, basado en los adelantos científico-técnicos de acuerdo con las tendencias actuales de este deporte a nivel internacional.

La situación antes reflejada hace que en la investigación se manejen numerosas variables correspondientes a las acciones por lo que, su proceso de recogida se torna en ocasiones engoroso. Este resultado se debe a las características tan dinámicas que posee el balonmano. Procesar las mismas al término de las competencias es otra debilidad. Además al llevar a cabo la recogida de los datos de forma manuscrita, hace que la durabilidad y organización de los mismos se vuelvan vulnerables, pues están más expuestos a perderse o a deteriorarse en un corto período de tiempo.

Teniendo en cuenta la situación actual, surge el **problema a resolver** que plantea:

No existe suficiente organización ni durabilidad en los datos que se recogen, de forma manuscrita en los juegos de balonmano.

El **objetivo general** de este trabajo consiste en desarrollar un sistema de registro estadístico informatizado para el análisis del comportamiento de acciones tácticas significativas en los equipos de balonmano.

Por lo planteado anteriormente el **objeto de estudio** es el estudio de las estadísticas que se recogen en aplicaciones relacionadas con el deporte.

En consecuencia el **campo de acción** es el proceso de recogida y almacenamiento de las estadísticas en los juegos de balonmano a nivel nacional.

Los **objetivos específicos** a cumplir son:

1. Realizar el diseño teórico de la investigación.
2. Revisar y estudiar la existencia de otras aplicaciones o soluciones similares.
3. Estudiar el lenguaje y metodología de desarrollo de software a utilizar.
4. Estudiar las herramientas que se van a emplear.
5. Realizar la captura de requisitos del sistema.
6. Elaborar el análisis y diseño del sistema según los requerimientos.
7. Implementar el software en dependencia de las funcionalidades que debe tener.

# Introducción

---

Teniendo en cuenta los aspectos mencionados anteriormente la **idea a defender** es que, con la culminación de este trabajo se logrará disponer de un sistema informatizado cuyo fin es evaluar acciones colectivas del deporte balonmano.

El sistema propuesto permitirá obtener datos específicos de los juegos, necesarios para enriquecer la información a considerar por los entrenadores quienes realizarán una planificación más científica y contextualizada del proceso de preparación deportiva. De esta manera se contribuirá, a la teoría del entrenamiento deportivo mediante la revelación de los componentes esenciales del juego de balonmano.

Al confeccionar el sistema de registro estadístico para el análisis pedagógico del comportamiento de las acciones tácticas, se ofrecerá un instrumento para elevar el nivel teórico y científico, siendo de interés expreso de la Federación Cubana de Balonmano. Por otra parte se mejorará la calidad de los registros estadísticos del juego, lo que a su vez servirá de apoyo a los entrenadores para la planificación y dosificación de los entrenamientos.

## **Los métodos científicos utilizados**

Dentro de los métodos teóricos:

Analítico – Sintético

Son dos procesos inherentes al pensamiento, operaciones lógicas importantes, que permiten, como métodos teóricos, buscar la esencia de los fenómenos, los rasgos que lo caracterizan y los distinguen. Su objetivo en la investigación es analizar las teorías e instrumentos, lo que permite la extracción de los elementos más importantes que se consideran básicos por su frecuencia e importancia para el registro de las acciones de juego en el balonmano.

Análisis histórico-lógico

Para investigar sobre las aplicaciones informáticas de este tipo implementadas en Cuba y el resto del mundo.

Dentro de los empíricos se emplearon:

Entrevistas (de forma no estructurada) para conocer cuáles son las necesidades de los profesionales del INDER y de la Federación Cubana de Balonmano.

## **Estructura capitular**

**Capítulo 1:** Fundamentación teórica.

# Introducción

---

**Capítulo 2:** Desarrollo de la solución.

**Capítulo 3:** Implementación y pruebas de la solución propuesta.

Las **tareas** a realizar para dar solución a los objetivos son:

1. Revisión de la bibliografía técnica del balonmano con el apoyo del método Analítico – Sintético.
2. Realización del estado del arte relacionado con sistemas informáticos existentes dentro de la recogida de resultados estadísticos de deportes basándose en el método Histórico – Lógico.
3. Realización del análisis de diferentes lenguajes y metodologías de desarrollo de software que existen para conocer las ventajas que brindan y de esta manera seleccionar los más convenientes con ayuda del método Analítico – Sintético.
4. Realización de un análisis crítico de las tendencias, tecnologías y herramientas de desarrollo de software a utilizar basándose en el método Analítico – Sintético.
5. Realización de entrevistas con los clientes que desean informatizar el proceso de recogida de acciones significativas en los juegos de balonmano. Para la realización de esta tarea es necesario la técnica de la entrevista.
6. Identificación de los requerimientos del sistema con ayuda del método Analítico – Sintético.
7. Realización del Análisis y Diseño del software mediante el método Analítico – Sintético.
8. Implementación de la aplicación basándose en el método Analítico – Sintético.

## **Capítulo 1: Fundamentación teórica**

### **1.1 Introducción**

Para lograr que el rendimiento del balonmano en Cuba se eleve, es necesario recoger las estadísticas colectivas en las competencias que se realicen de este deporte con el objetivo de analizarlas y corregir errores que se cometieron durante el partido. Para lograr este propósito con más eficiencia y veracidad es necesario informatizar el proceso de recogida de las variables que se manejan en los juegos de balonmano.

En el presente capítulo, se brinda una explicación de los conceptos relacionados con el problema a resolver del presente Trabajo de Diploma, para una mejor comprensión del mismo. Además, se hace un análisis de las estadísticas que se recogen en varias aplicaciones y se realiza un estudio de las herramientas y tecnologías que pueden ser las adecuadas a utilizar para construir el software.

### **1.2 Conceptos asociados al problema**

#### **Comisión Nacional de Balonmano**

En la Resolución 567 del 23 de febrero de 1961 se establecen las funciones específicas de las estructuras del INDER y entre ellas sobre la Comisión Nacional Deportiva plantea:

Aparato auxiliar del órgano deportivo para realizar las funciones estatales de planificar, organizar y desarrollar la política de desarrollo de cada uno de los deportes.

Funciona como una organización administrativa que desarrolla cada deporte bajo el principio del deporte como derecho del pueblo, desde la base hasta el deporte de Alto Rendimiento.

Sus cargos son designados y realiza su labor de conjunto con las Federaciones Deportivas y son dirigidas por la estructura de dirección del INDER.

Ejecuta y controla los recursos humanos y financieros puestos por el INDER para el desarrollo del deporte.

# Capítulo 1

---

Se auxilia en comisiones de trabajo, fundamentalmente la Comisión Técnica que en cada deporte propone las estrategias del Alto Rendimiento. (INDER, 1961)

## **Variables en el juego**

Cuando se tiene una variable y los valores de ella; esa variable puede representar características, propiedades, atributos, fenómenos, conceptos, cualquier aspecto de la realidad.

Para este trabajo las variables en el juego:

Representan las características particulares de cada actividad deportiva, sustentadas en el Reglamento de cada deporte. (Anoceto, 2006)

## **Estadística**

Existen diversos criterios de diferentes autores acerca del concepto de lo que es la estadística:

- Es la ciencia que permite tomar decisiones en presencia de la incertidumbre.
- Se relaciona con los métodos científicos para la recolección, organización, tabulación, presentación y análisis de datos.
- Permite planear, analizar e interpretar los resultados de una investigación.
- Se refiere a un conjunto de hechos, personas, objetos y fenómenos.
- Trabaja con conjuntos numerosos que ocurren en diferentes momentos y bien seleccionados. (Anoceto, 2006)

Para esta obra la estadística:

Es la ciencia que provee de métodos y procedimientos que permiten recolectar, organizar, resumir, presentar y analizar datos relativos a un conjunto de individuos u observaciones y que permiten extraer conclusiones válidas, realizar generalizaciones y tomar decisiones lógicas basadas en dichos análisis. (Omar A. Pérez Jacinto, 2008)

## **1. 3 Análisis de otras soluciones existentes**

### **Sistema de Anotación de Dominó**

El desarrollo del Sistema de Anotación de Dominó tiene por finalidad aplicar las más modernas herramientas tecnológicas al sistema de registro manual inventado por el

# Capítulo 1

---

profesor Valenty Sinecio González Pulgar, natural de Punto Fijo, Estado Falcón, Venezuela y con más de 40 años de dedicación al estudio del deporte del Dominó.

Estos estudios realizados por el profesor Valenty Sinecio y su participación destacada como Atleta de Alta Competencia y Gerente Deportivo, aportan una visión integral de los aspectos a considerar para el desarrollo de esta aplicación.

Es así como el Sistema de Anotación de Dominó cumple con las siguientes características generales:

- Es un sistema confiable dado la importancia de los registros a almacenar y el uso final de los mismos.
- Tiene una interfaz intuitiva y de fácil uso para el operador del equipo.
- El diseño del menú para ingreso de datos en la partida es suficientemente sencillo y rápido para que el operador del equipo pueda registrar las secuencias de las jugadas sin alterar el normal desenvolvimiento de la mano.
- El sistema es fácil de manejar, de manera que puede ser operado por cualquier persona.
- Se registran las jugadas una por una con la finalidad de poder reconstruir las partidas completamente y a la vez generar un sin fin de estadísticas hasta ahora no registradas formalmente.
- El sistema fortalece el reconocimiento de manera general del Dominó como deporte. (González, 2007)

Por las características del Dominó, el software fue desarrollado para que funcione en Equipos Móviles (pocket PC) sin necesidad de uso de cables (wireless) de manera de darle completa movilidad al anotador y/o juez.

También se descartó el uso de teclados y se optó por la opción de toque de pantalla (touch screen). Por la movilidad se requirió de equipos portátiles y estos debían ser sin cableados dadas las características de las competencias.

## **Aporte**

Como se puede apreciar este sistema tiene elementos que pueden ser importantes para la construcción del software que se quiere lograr mediante este trabajo, pues por ejemplo: realizar la entrada de datos en el tiempo en que se ejecuta una partida, no se vuelve dificultoso puesto que posee un menú lo suficientemente sencillo para que la mano del usuario pueda desenvolverse de forma rápida. El sistema en su conjunto es fácil de manipular; no obstante se utilizan en él varios dispositivos y opciones como toque de pantalla, que hacen que la eficiencia del software sea indiscutible. Estas



# Capítulo 1

---

tecnologías son poco accesibles debido a su costo, lo que constituye un inconveniente del sistema, por eso es necesario descartar esta parte.

## **ECP Basket. Sistema Informático para el Control Estadístico y Evaluación del Rendimiento de los Jugadores de Baloncesto**

Consiste en un sistema informático que, basado en la tecnología cliente/servidor, brinda a través de Servicios Web la facilidad de captar los indicadores estadísticos de los partidos de baloncesto, permitiendo a cada colectivo técnico en tiempo real, conocer los rendimientos de sus jugadores. Posibilita así el beneficio en el proceso de toma de decisiones y su incidencia en el resultado final de los partidos.

Esta herramienta informática ofrece numerosas facilidades para obtener reportes en los que reflejan el resumen estadístico al finalizar cada período del partido, así como en cualquier instante del juego, como por ejemplo durante un tiempo muerto computable. El sistema es capaz de brindar sugerencias, en tiempo real, como modificaciones al quinteto presente en la cancha de juego, a partir de los índices de evaluación de los jugadores.

El software ECP Basket para la evaluación del rendimiento competitivo de los jugadores del baloncesto élite cubano, puede ser utilizado en dos vertientes fundamentales: empleado por los estadísticos para el control de los diferentes indicadores de juego y su vaciado posterior en las estadísticas resúmenes de los partidos de Baloncesto, o utilizado por las direcciones técnicas de los conjuntos para la toma de decisiones constantes durante los partidos, basados en los índices de rendimiento defensivos, ofensivos y globales, así como su interpretación en correspondencia a la escala de evaluación cualitativa que presenta la metodología integradora.

Esta herramienta fue desarrollada asumiendo el modelo cliente/servidor, dada la necesidad de la veracidad de los datos, y se escogió la solución que ofrece Visual Estudio, utilizando como lenguaje de programación C# y como servidor de base de datos SQL Server 2000.

La carga visual es adecuada y el lenguaje de las opciones que se ha utilizado es de fácil comprensión para el usuario, teniendo en cuenta que tanto los estadísticos como los integrantes de los colectivos técnicos son especialistas de la materia. (Roberto Espinosa Oliva, 2007)

# Capítulo 1

---

## **Aporte**

Este software es capaz de recoger las estadísticas colectivas para cuando finalice el partido poder llegar a conclusiones de los resultados teniendo como base los datos recogidos durante el mismo. También brinda la posibilidad de tomar decisiones tácticas en tiempo real en que se está realizando el juego, por ejemplo: en el tiempo de descanso, pues es capaz de recoger estadísticas individuales y basándose en los resultados se pueden hacer cambios de jugadores o indicar jugadas necesarias. Aunque no es objetivo de nuestro trabajo recoger datos individuales, sí lo es obtenerlos de forma colectiva. En este caso el lenguaje de las opciones que se ha utilizado es de fácil comprensión para el usuario. Esto es un aspecto importante que hay que tener en cuenta a la hora de realizar la aplicación que se quiere con el presente trabajo, pues es fundamental que el usuario pueda entender fácilmente el manejo de la misma.

## **Sistema para la Gestión de Eventos Competitivos y Anotación de los Juegos en el Béisbol**

Con esta aplicación es posible gestionar todo cuanto tiene relación con el béisbol, de manera que no es necesario que el usuario, en este caso los coordinadores del torneo, tengan que realizar manualmente toda una serie de movimientos y operaciones para conformar los equipos y grupos.

Este sistema posee una interfaz visual usable, funcional y accesible, que logra informatizar el proceso de gestión de los eventos competitivos. Además lleva a cabo un registro amplio y detallado de cada acción del partido y facilita la labor de los informáticos y anotadores y de la Comisión Nacional en el instante de realizar las anotaciones y la gestión de los eventos.

Para el desarrollo de esta aplicación se utilizaron como herramientas desarrolladoras el Visual Studio 2005 para su implementación y el SQL Server 2000 como Sistema de Gestión de Bases de Datos. Para la implementación de este sistema se utilizó la tecnología orientada a objetos. Como lenguajes de programación se emplearon C# y SQL. (José Leandro Cabrera, 2007)

## **Aporte**

Este sistema recoge cada acción del partido haciendo con ellas un registro abarcador. Además el lenguaje que ha utilizado es C#, por lo que es necesario estudiar la sintaxis

con la que está estructurado el software pues es el mismo lenguaje que se ha seleccionado para la realización de la aplicación que se quiere lograr mediante el presente trabajo, que más adelante se explica. No obstante, el software estudiado no está relacionado con el deporte balonmano, lo que hace que difiera en varios aspectos al que se desea realizar, es por ello que no aporta elementos en gran medida.

## 1. 4 Estudio de metodologías

Se entiende por metodología de desarrollo una colección de documentación formal referente a los procesos, las políticas y los procedimientos que intervienen en el desarrollo del software.

La finalidad de una metodología de desarrollo es garantizar la eficacia (p.ej. cumplir los requisitos iniciales) y la eficiencia (p.ej. minimizar las pérdidas de tiempo) en el proceso de generación de software. (2009)

### Metodologías pesadas o tradicionales

Las metodologías pesadas o tradicionales permiten la construcción de proyectos grandes y con un amplio equipo de desarrollo, basándose fundamentalmente en la documentación.

Entre las metodologías tradicionales se puede citar:

#### 1.4.1 Proceso Unificado de Desarrollo de Software (RUP)

Es una metodología de desarrollo pesada para la ingeniería de software que va más allá del análisis y diseño orientado a objetos para proporcionar una familia de técnicas que soportan el ciclo completo de desarrollo del software, además de que unifica completamente a un equipo de desarrollo, optimiza la productividad de cada uno de los miembros del mismo y ayuda a los líderes de proyecto a incrementar su experiencia en el desarrollo de los mismos.

RUP se caracteriza por ser **iterativo e incremental** donde cada fase se desarrolla en iteraciones, de forma tal que se pueda dividir en pequeños proyectos mejorando su comprensión y desarrollo. También está **dirigido por los casos de uso** que no son más que el instrumento para validar la arquitectura del software y extraer los casos de prueba. Esta metodología se **centra en la arquitectura**, aquí los modelos son

# Capítulo 1

---

proyecciones del análisis y el diseño constituye la arquitectura del producto a desarrollar. (Sánchez, 2004)

Como RUP es un proceso, en su modelación define como sus principales elementos:

- Trabajadores (quién)
- Artefactos (qué)
- Actividades (cómo)
- Flujo de actividades (cuándo)

Un proyecto que se desarrolle utilizando la metodología RUP está dividido en 4 fases, 6 flujos de ingeniería y 3 de apoyo.

## Fases

- Inicio
- Elaboración
- Construcción
- Transición

## Flujos de ingeniería

- Modelo de negocio
- Requisitos
- Análisis y diseño
- Implementación
- Prueba
- Despliegue

## Flujos de apoyo

- Gestión de cambios y configuraciones
- Gestión de proyectos
- Ambiente o entorno

## 1.4.2 Plataforma de Soluciones de Microsoft (MSF)

Esta es una metodología flexible e interrelacionada con una serie de conceptos, modelos y prácticas de uso, que controlan la planificación, el desarrollo y la gestión de proyectos tecnológicos. MSF se centra en los modelos de proceso y de equipo dejando en un segundo plano las elecciones tecnológicas. (Sánchez, 2004)

MSF tiene las siguientes características:

- **Adaptable:** es parecido a un compás, usado en cualquier parte como un mapa, del cual su uso es limitado a un específico lugar.
- **Escalable:** puede organizar equipos tan pequeños entre 3 o 4 personas, así como también, proyectos que requieren 50 personas a más.
- **Flexible:** es utilizada en el ambiente de desarrollo de cualquier cliente.
- **Tecnología Agnóstica:** porque puede ser usada para desarrollar soluciones basadas sobre cualquier tecnología.

MSF se compone de varios modelos encargados de planificar las diferentes partes implicadas en el desarrollo de un proyecto: Modelo de Arquitectura del Proyecto, Modelo de Equipo, Modelo de Proceso, Modelo de Gestión del Riesgo, Modelo de Diseño de Proceso y finalmente el Modelo de Aplicación.

## Metodologías ágiles o livianas

Las metodologías ágiles o livianas son utilizadas para la construcción de proyectos pequeños y de un reducido equipo de desarrollo, las mismas se basan en la programación y la sencillez.

## 1.4.3 SCRUM

Desarrollada por Ken Schwaber, Jeff Sutherland y Mike Beedle. Define un marco para la gestión de proyectos, que se ha utilizado con éxito durante los últimos 10 años. Está especialmente indicada para proyectos con un rápido cambio de requisitos. Sus principales características se pueden resumir en dos. El desarrollo de software se realiza mediante iteraciones, denominadas sprints, con una duración de 30 días. El resultado de cada sprint es un incremento ejecutable que se muestra al cliente. La segunda característica importante son las reuniones a lo largo proyecto, entre ellas destaca la reunión diaria de 15 minutos del equipo de desarrollo para coordinación e integración. (José H. Canós, 2008)

## 1.4.4 Programación Extrema (XP)

Es una de las metodologías de desarrollo de software más exitosas en la actualidad utilizadas para proyectos de corto plazo, corto equipo y cuyo plazo de entrega era ayer. La metodología consiste en una programación rápida o extrema, cuya particularidad es tener como parte del equipo, al usuario final, pues es uno de los requisitos para llegar al éxito del proyecto. (Sánchez, 2004)

Los principios originales de la programación extrema son: simplicidad, comunicación, retroalimentación (feedback) y coraje:

- **Simplicidad**

Es la base de la programación extrema. Se simplifica el diseño para agilizar el desarrollo y facilitar el mantenimiento. Un diseño complejo del código junto a sucesivas modificaciones por parte de diferentes desarrolladores hace que la complejidad aumente exponencialmente.

- **Comunicación**

Se realiza de diferentes formas. Para los programadores el código comunica mejor cuanto más simple sea. Si el código es complejo hay que esforzarse para hacerlo inteligible.

- **Retroalimentación (feedback)**

Al estar el cliente integrado en el proyecto, su opinión sobre el estado del proyecto se conoce en tiempo real. Al realizarse ciclos muy cortos tras los cuales se muestran resultados, se minimiza el tener que rehacer partes que no cumplen con los requisitos y ayuda a los programadores a centrarse en lo que es más importante.

- **Coraje o valentía**

Los puntos anteriores parecen tener sentido común, entonces, ¿por qué coraje? Para los gerentes la programación en parejas puede ser difícil de aceptar, porque les parece como si la productividad se fuese a reducir a la mitad ya que solo la mitad de los programadores está escribiendo código. Hay que ser valiente para confiar en que

# Capítulo 1

---

la programación por parejas beneficia la calidad del código sin repercutir negativamente en la productividad.

## **¿Qué es lo que propone XP?**

- Empieza en pequeño y añade funcionalidad con retroalimentación continua.
- El manejo del cambio se convierte en parte sustantiva del proceso.
- El costo del cambio no depende de la fase o etapa.
- No introduce funcionalidades antes que sean necesarias.
- El cliente o el usuario se convierte en miembro del equipo.

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

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

## **Selección de la Metodología**

Después de haber estudiado con detenimiento las metodologías de desarrollo de software anteriores, se llega a la conclusión de que todas en gran medida presentan cualidades sobresalientes. No obstante se considera que RUP no es la metodología más apropiada para la realización del software que se quiere obtener mediante el presente trabajo, pues se utiliza generalmente para proyectos grandes y de larga envergadura y se centra más en la documentación y el burocratismo.

La metodología MSF tampoco es la más conveniente para la realización del software porque se centra en los modelos de proceso y de equipo dejando en un segundo plano las elecciones tecnológicas. SCRUM, a pesar de ser una metodología ágil, fue desechada en esta investigación porque se considera que XP es dentro de las metodologías ágiles la que más éxitos ha obtenido en la construcción de proyectos.

En la actualidad ya existen libros asociados a cada una de las metodologías ágiles existentes y también abundante información en Internet, pero es XP la metodología que resalta por contar con la mayor cantidad de información disponible y es con diferencia, la más popular. La inventiva XP es ágil, consiste en una programación rápida o extrema. Con su utilización como metodología para el desarrollo del presente trabajo se obtendría un resultado más similar a lo que desea el cliente ya que el mismo

estará presente en el desarrollo del sistema. A esto se agrega que brinda la posibilidad de adaptar el software a los cambios de requisitos en cualquier punto de vida del proyecto. Asimismo XP propone que la comunicación y la satisfacción del cliente es lo primordial y es de esta manera que no se profundiza tanto en la documentación pues sugiere que es más importante definir los requerimientos.

## **1.5 Estudio de herramientas**

Para llevar a cabo el desarrollo de un software es necesario apoyarse en varias herramientas que posibiliten su creación, así como también viabilicen el trabajo. Se utilizan herramientas para modelación, implementación y para trabajar con Bases de Datos.

### **1.5.1 Herramientas de modelado**

UML: Lenguaje que permite especificar, visualizar y construir los artefactos de los sistemas de software. (Larman, 2004)

#### **Rational Rose**

Es una de las más poderosas herramientas de modelado visual para el análisis y diseño de sistemas basados en objetos. Se utiliza para modelar un sistema antes de proceder a construirlo. Cubre todo el ciclo de vida de un proyecto. (S.A, 2007)

Características:

- Permite especificar, analizar, diseñar el sistema antes de codificarlo.
- Mantiene la consistencia de los modelos del sistema software.
- Chequeo de la sintaxis UML.
- Ingeniería inversa (crear modelo a partir código).
- Esta herramienta contiene 4 vistas:
  1. Vista de Caso de Uso
  2. Vista Lógica
  3. Vista de Componente
  4. Vista de Despliegue

#### **Visual Paradigm**



# Capítulo 1

---

Visual Paradigm para UML es una herramienta profesional para el modelado que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. El software de modelado UML ayuda a una más rápida construcción de aplicaciones de calidad, mejores y a un menor coste. Permite dibujar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación.

Características:

- Entorno de creación de diagramas para UML.
- Diseño centrado en casos de uso y enfocado al negocio que generan un software de mayor calidad.
- Uso de un lenguaje estándar común a todo el equipo de desarrollo que facilita la comunicación.
- Modelo y código que permanecen sincronizados en todo el ciclo de desarrollo.
- Disponibilidad en múltiples plataformas.
- Es profesional: da la posibilidad de crear un gran conjunto de artefactos de las distintas fases del desarrollo del software, entre los que se encuentran los siguientes:
  - Diagramas de Casos de Uso
  - Diagramas de Clases
  - Diagramas de Secuencia
  - Diagramas de Estado
  - Diagramas de Componentes
  - Diagramas de Despliegue
  - Diagramas de Objetos
  - Diagramas de Interacción
- Es amigable.
- Contiene facilidades para redactar especificaciones de casos de uso del sistema.
- Sincronización entre diagramas de entidad, relación y diagramas de clases.
- Generación de código / ingeniería inversa.
- Interoperabilidad con otras aplicaciones.
- Integración con distintos Ambientes de Desarrollo Integrados (IDE).
- Facilidad de instalación y actualización.

- Compatibilidad entre las diferentes versiones del mismo.

## **Lenguaje Unificado de Modelado (UML)**

Es el lenguaje de modelado de sistemas de software más conocido en la actualidad; es el estándar internacional aprobado por el Grupo de Gestión de Objetos (OMG), consorcio creado en 1989 responsable de la creación, desarrollo y revisión de especificaciones para la industria del software.

UML cuenta con varios tipos de diagramas, los cuales muestran diferentes aspectos de lo que se quiere representar. (Gracia, 2003)

Los diagramas de estructura enfatizan en los elementos que deben existir en el sistema modelado:

- Diagrama de clases
- Diagrama de componentes
- Diagrama de objetos
- Diagrama de estructura compuesta (UML 2.0)
- Diagrama de despliegue
- Diagrama de paquetes
- Diagrama de casos de uso

Los diagramas de comportamiento enfatizan en lo que debe suceder en el sistema modelado:

- Diagrama de actividades
- Diagrama de estados
- Diagramas de Interacción son un subtipo de diagramas de comportamiento, que enfatiza sobre el flujo de control y de datos entre los elementos del sistema modelado:
  - Diagrama de secuencia
  - Diagrama de comunicación, que es una versión simplificada del Diagrama de colaboración (UML 1.x)
  - Diagrama de tiempos (UML 2.0)
  - Diagrama global de interacciones o Diagrama de vista de interacción (UML 2.0)

## **Selección de la herramienta de modelado**

Luego de haber estudiado las características de estas dos herramientas de modelado se llega a la conclusión final que brindan un servicio magnífico ya que ayudan en gran medida a la calidad del producto final y por esta razón es que son utilizadas a gran escala y se toma como herramienta de modelado para el desarrollo del presente trabajo a Visual Paradigm por ser una herramienta que soporta el ciclo de vida completo de un proyecto, es multiplataforma, es compatible con todas las versiones que se han realizado de él, da la posibilidad de generar todo tipo de diagramas, es capaz de lograr una conexión entre el diseño del software y el código de forma que al realizarse algún cambio en cualquiera de estas partes, la otra recibe una actualización del cambio realizado y su utilización es permitida en la UCI pues su licencia fue comprada por la universidad por tiempo indefinido, no siendo así con Rational Rose.

## **1. 5. 2 Herramientas de desarrollo**

### **Sistemas Gestores de Base de Datos (SGBD)**

#### **Postgre SQL**

Es un Sistema de Gestión de Bases de Datos objeto-relacional (ORDBMS) basado en el proyecto POSTGRES, de la Universidad de Berkeley. Es libre, el proyecto está dirigido por la comunidad de desarrolladores y usuarios. (Pecos, 2007)

Principales características:

- Implementación del estándar SQL92/SQL99.
- 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), cadenas de bits, etc. También permite la creación de tipos propios.
- Incorpora una estructura de datos arreglos.
- Incorpora funciones de diversa índole: manejo de fechas, geométricas, orientadas a operaciones con redes, etc.
- Permite la declaración de funciones propias, así como la definición de disparadores.
- Soporta el uso de índices, reglas y vistas.

# Capítulo 1

---

- Incluye herencia entre tablas (aunque no entre objetos, ya que no existen), por lo que a este gestor de bases de datos se le incluye entre los gestores objeto-relacionales.
- Permite la gestión de diferentes usuarios, como también los permisos asignados a cada uno de ellos.
- Es idóneo para los sistemas con gran cantidad de peticiones de consultas a las Bases de Datos.

## MySQL

MySQL es un sistema de gestión de bases de datos relacional, licenciado bajo la Licencia Pública General de GNU. Su diseño multihilo le permite soportar una gran carga de forma muy eficiente. MySQL fue creada por la empresa sueca MySQL AB, que mantiene el copiar y escribir del código fuente del servidor SQL, así como también de la marca.

Este gestor de bases de datos es, probablemente, el gestor más usado en el mundo del software libre, debido a su gran rapidez y facilidad de uso. Esta gran aceptación es debida, en parte, a que existen infinidad de librerías y otras herramientas que permiten su uso a través de gran cantidad de lenguajes de programación, además de su fácil instalación y configuración. (Pecos, 2007)

Características de MySQL:

- Aprovecha la potencia de sistemas multiprocesador, gracias a su implementación multihilo.
- Soporta gran cantidad de tipos de datos para las columnas.
- Gran portabilidad entre sistemas.
- Soporta hasta 32 índices por tabla.
- Gestión de usuarios y contraseñas, manteniendo un muy buen nivel de seguridad en los datos.

MySQL carece de:

- Disparadores y Procedimientos: Se tiene pensado incluir el uso de procedimientos almacenados en la base de datos, pero no el de disparadores,

# Capítulo 1

---

ya que estos reducen de forma significativa el rendimiento de la base de datos, incluso en aquellas consultas que no los activan.

- Integridad referencial: aunque admite la declaración de claves ajenas en la creación de tablas, internamente no las trata de forma diferente al resto de los campos.

## **Lenguaje de Consulta Estructurado (SQL):**

Es un lenguaje declarativo de acceso a bases de datos relacionales que permite especificar diversos tipos de operaciones en éstas. Una de sus características es el manejo del álgebra y el cálculo relacional permitiendo efectuar consultas con el fin de recuperar de una forma sencilla información de interés de una base de datos, así como también hacer cambios sobre ella. (Gallego, 2007)

Existen dos tipos de comandos SQL:

- Los DDL que permiten crear y definir nuevas bases de datos, campos e índices.
- Los DML que permiten generar consultas para ordenar, filtrar y extraer datos de la base de datos.

Características generales:

- El SQL es un lenguaje de acceso a bases de datos que explota la flexibilidad y potencia de los sistemas relacionales permitiendo gran variedad de operaciones sobre los mismos.
- Es un lenguaje declarativo de "alto nivel" o "de no procedimiento", que gracias a su fuerte base teórica y su orientación al manejo de conjuntos de registros y no a registros individuales, permite una alta productividad en codificación y la orientación a objetos. De esta forma una sola sentencia puede equivaler a uno o más programas que es utilizado en un lenguaje de bajo nivel orientado a registro.

Principales ventajas:

- Independencia de los fabricantes.
- Portabilidad a cualquier tipo de plataforma.
- Está estandarizado.

# Capítulo 1

---

- Basado en el modelo relacional.
- Lenguaje de alto nivel.
- Utilización en Lenguaje de programación.
- Múltiples vistas de los datos.
- Definición dinámica de datos.
- Arquitectura cliente/servidor.

## **Archivos de datos XML**

El lenguaje XML (Lenguaje de marcado extensible) es un método flexible para crear formatos de información comunes. Una vez creado el formato de la información, tanto el formato como los datos se pueden compartir en Web o con otros programas y bases de datos. (Guzmán, 2005)

## **Aplicaciones y ventajas de XML**

Algunas de las aplicaciones de XML son:

- Ofrecer mecanismos más versátiles de mostrar datos. Actualmente, bajo el nombre de Modelo de Objetos del Documento (DOM) se está desarrollando una Interfaz de Programación de Aplicaciones (API) que sea soportada por todos los procesadores de XML y HTML. La idea detrás de esta API es que sea posible representar, a través de JavaScripts o JavaApplets, documentos XML en los navegadores Web, pero de una forma más sofisticada que los documentos HTML, ya que XML no solo proporciona una sintaxis, sino también una semántica.
- Buscadores inteligentes. Debido a que la información en los documentos XML está etiquetada por su significado de forma precisa, es posible localizarla de forma mucho más clara que en documentos HTML. Con Definición de Tipo de Documento (DTD) estandarizados para distintas aplicaciones se podrían programar buscadores Web que recuperen información sobre un producto de cualquier Sitio Web (Web Site) en el mundo sabiendo que todos tendrán el mismo formato de datos gracias al DTD, aunque no tengan necesariamente la misma representación gráfica gracias al XML/XSL.

# Capítulo 1

---

- Intercambio de información entre sistemas heterogéneos. El fundamento es el mismo que para los buscadores inteligentes. Debido a que el DTD proporciona un formato estándar para representar la información de un tema específico, puede usarse para simplificar el intercambio de información entre distintas fuentes.

Algunas de las ventajas de XML son:

- Los autores y proveedores pueden diseñar sus propios tipos de documentos usando XML, en vez de limitarse a HTML. Los tipos de documentos pueden ser explícitamente “hechos a la medida de una audiencia”, por lo que las difíciles manipulaciones que se deben hacer con HTML para conseguir efectos especiales, serán cosa del pasado: autores y diseñadores serán libres de inventar sus propias etiquetas.
- La información contenida puede ser más rica y fácil de usar, porque las habilidades hipertextuales de XML son mayores que las de HTML.
- XML puede dar más y mejores facilidades para la representación en los visualizadores.
- Elimina muchas de las complejidades de Lenguaje de Marcado Generalizado (SGML), en favor de la flexibilidad del modelo, con lo que la escritura de programas para manejar XML será más sencilla que haciendo el mismo trabajo en SGML.
- La información será más accesible y reutilizable, porque la flexibilidad de las etiquetas de XML pueden utilizarse sin tener que amoldarse a reglas específicas de un fabricante, como es el caso de HTML.
- Los archivos XML válidos son válidos también en SGML, luego pueden utilizarse también fuera de la Web, en un entorno SGML, una vez la especificación sea estable y el software SGML la adopte. (Guzmán, 2005)

## **Selección del SGBD**

Después de analizar las características de MySQL y PostgreSQL y llegar a la conclusión de que los dos son eficientes y utilizados por las ventajas que brindan, se toma como posible servidor de base de datos a PostgreSQL por ser un potente gestor de Base de Datos, además por las facilidades de uso que posee permitiendo crear una

amplia funcionalidad a través de su sistema de activación de disparadores, del cual carece MySQL. Es ideal para los sistemas con numerosas peticiones de consultas a las bases de datos y sobre todo PostgreSQL es software libre. No obstante para interactuar con la aplicación se designa a los archivos XML, pues la información contenida será más accesible, reutilizable y fácil de manipular. También, al ser el software que se quiere lograr con este trabajo una aplicación de escritorio, el cliente decidió que se utilizará, por ahora, para el manejo de los datos XML porque de esta manera el software podrá utilizarse en varios equipos sin necesidad de instalar ningún programa, lo que permite que el rendimiento del equipo sea mayor.

## 1.6 Estudio de lenguajes de programación

### C

C es un lenguaje de programación de propósito general que ofrece economía sintáctica, control de flujo y estructuras sencillas y un buen conjunto de operadores. No es un lenguaje de muy alto nivel y más bien es pequeño, llano y no está especializado en ningún tipo de aplicación. Esto lo hace un lenguaje potente, con un campo de aplicación ilimitado y sobre todo, se aprende rápidamente. En poco tiempo, un programador puede utilizar la totalidad del lenguaje. (Parra, 2007)

Ventajas:

- Lenguaje eficiente puesto que es posible utilizar sus características de bajo nivel para realizar implementaciones óptimas.
- Es el lenguaje más portado en existencia, habiendo compiladores para casi todos los sistemas conocidos.
- Proporciona facilidades para realizar programas modulares y/o utilizar código o bibliotecas existentes.

Inconvenientes:

El mayor problema que presenta el lenguaje C frente a los lenguajes de tipo de dato dinámico es la gran diferencia en velocidad de desarrollo: es mucho más lento programar en C. La razón consiste en que el compilador de C se limita a traducir código sin apenas añadir nada. La gestión de la memoria es un ejemplo clásico: en C



# Capítulo 1

---

el programador ha de reservar y liberar la memoria explícitamente. En otros lenguajes (como BASIC, Matlab o C#) la memoria es gestionada de forma transparente para el programador. Esto alivia la carga de trabajo humano y en muchas ocasiones previene errores.

El mantenimiento también es más difícil y costoso que con lenguajes de más alto nivel. El código en C se presta a sentencias cortas y enrevesadas de difícil interpretación. Aunque el lenguaje admite código escrito de forma fácilmente legible, si no se siguen normas en el equipo de programación algunos programadores pueden acabar escribiendo código difícil de leer. Esto complica la revisión y el mantenimiento.

C no dispone de sistemas de control automáticos y la seguridad depende casi exclusivamente de la experiencia del programador. La mayor parte de los problemas de seguridad en los sistemas informáticos actuales deriva de haber sido realizados en C. El fallo de seguridad clásico consiste en que algunas entradas de información al programa no se comprueban en longitud. Si un atacante introduce datos lo bastante grandes, puede provocar la sobreescritura de código en la pila del programa e incluso llegar a forzar la ejecución de código pernicioso. Los lenguajes de tipo dinámico cuentan con controles de gestión de memoria y de entrada de datos automáticos.

## **C++**

C++ es un lenguaje de programación diseñado a mediados de los años 80 del pasado siglo por Bjarne Stroustrup. La intención de su creación fue extender el exitoso lenguaje de programación C con mecanismos que permitan la manipulación de objetos. En ese sentido, desde el punto de vista de los lenguajes orientados a objetos, el C++ es un lenguaje híbrido.

El nombre C++ fue propuesto por Rick Mascitti en el año 1983, cuando fue utilizado por primera vez fuera de un laboratorio científico. Antes se había usado el nombre "C con clases". La expresión "C++" es también conocida como un incremento de su lenguaje de origen C.

Posteriormente se añadieron facilidades de programación genérica, que se sumó a los otros dos paradigmas que ya estaban admitidos (programación estructurada y la

# Capítulo 1

---

programación orientada a objetos). Por esto se suele decir que el C++ es un lenguaje de programación multiparadigma.

Una particularidad del C++ es la posibilidad de redefinir los operadores (sobrecarga de operadores), y de poder crear nuevos tipos que se comporten como tipos fundamentales. (Parra, 2007)

## **C #**

C# es un lenguaje de programación orientado a objetos, desarrollado y estandarizado por Microsoft como parte de su plataforma .NET, que después fue aprobado como un estándar por ECMA.

Su sintaxis básica deriva de C/C++ y utiliza el modelo de objetos de la plataforma.NET el cual es similar al de Java aunque incluye mejoras derivadas de otros lenguajes (entre ellos Delphi).

La creación del nombre del lenguaje, C#, proviene de dibujar dos signos positivos encima de los dos signos positivos de "C++", queriendo dar una imagen de salto evolutivo del mismo modo que ocurrió con el paso de C a C++.

C#, como parte de la plataforma.NET, está normalizado por ECMA desde diciembre de 2001. El 7 de noviembre de 2005 salió la versión 2.0 del lenguaje que incluía mejoras tales como tipos genéricos, métodos anónimos, iteradores, tipos parciales y tipos anulables. El 19 de noviembre de 2007 salió la versión 3.0 de C# destacando entre las mejoras los tipos implícitos, tipos anónimos y Consulta Integrada en el Lenguaje (LINQ). (Parra, 2007)

Metas del diseño del lenguaje:

El estándar ECMA lista las siguientes metas en el diseño para C#:

- Lenguaje de programación orientado a objetos: simple, moderno y de propósito general.
- Inclusión de principios de ingeniería de software tales como: revisión estricta de los tipos de datos, revisión de límites de vectores, detección de intentos de usar variables no inicializadas y recolección de basura automática.

# Capítulo 1

---

- Capacidad para desarrollar componentes de software que se puedan usar en ambientes distribuidos.
- Portabilidad del código fuente.
- Fácil migración del programador al nuevo lenguaje, especialmente para programadores familiarizados con C y C++.
- Soporte para internacionalización.
- Adecuación para escribir aplicaciones de cualquier tamaño: desde las más grandes y sofisticadas como sistemas operativos hasta las más pequeñas funciones.
- Aplicaciones económicas en cuanto a memoria y procesado.

## Ventajas frente a C y C++

- Compila a código intermedio (CIL) independiente del lenguaje en que haya sido escrita la aplicación e independiente de la máquina donde vaya a ejecutarse.
- Recolección de basura automática.
- Eliminación del uso de punteros, en C# no se necesitan.
- Capacidades de reflexión.
- No hay que preocuparse por archivos de cabecera ".h".
- No importa el orden en que hayan sido definidas las clases ni las funciones.
- No hay necesidad de declarar funciones y clases antes de definir las.
- No existen las dependencias circulares.
- Soporta definición de clases dentro de otras.
- No existen funciones, ni variables globales, todo pertenece a una clase.
- Todos los valores son inicializados antes de ser usados (automáticamente se inicializan al valor estandarizado, o manualmente se pueden inicializar desde constructores estáticos).
- No se pueden utilizar valores no booleanos (enteros, coma flotante etc) para condicionales. Es mucho más limpio y menos propenso a errores.

## Compiladores:

En la actualidad existen los siguientes compiladores para el lenguaje C#:

- Microsoft.NET plataforma SDK incluye un compilador de C#, pero no un IDE.

# Capítulo 1

---

- Microsoft Visual Studio .NET, IDE por excelencia de este lenguaje, versión 2002, 2003, 2005, 2008 y 2010(beta).
- SharpDevelop, es un IDE libre para C# bajo licencia LGPL, muy similar a Microsoft Visual C#.
- Mono, es una implementación GPL de todo el entorno .NET desarrollado por Novell. Como parte de esta implementación se incluye un compilador de C#.
- Delphi 2006, de Borland Software Corporation.
- DotGNU Portable.NET, de la Fundación de Software Libre.

## Selección del lenguaje

En la implementación del sistema propuesto se utiliza la tecnología orientada a objetos. Como lenguaje de programación se emplea C#. Esta selección se hizo teniendo en cuenta las características que lo distinguen de los demás lenguajes estudiados anteriormente como son: que se torna mucho más lento programar en C que programar en C#; el mantenimiento de C es más difícil y costoso que con lenguajes de más alto nivel como C#, esto se debe a que el código en que está estructurado C se refleja a través de sentencias cortas y de difícil interpretación. Frente a C++ el lenguaje C# tiene las ventajas: que se ha eliminado en este último la utilización de punteros, no es necesario preocuparse por el orden en que se hayan definido las clases ni las funciones, no existen las dependencias circulares; además, que soporta la definición de clases dentro de otras. Todas estas ventajas se produjeron, por tanto el lenguaje C#, es considerado un paso evolutivo del C++.

## 1.7 Estudio de Entornos de Desarrollo Integrado (IDEs)

### SharpDevelop

Con SharpDevelop se dispone de un IDE con el que se crean y desarrollan proyectos en C#, VB.NET y Boo, un lenguaje de programación orientado a objetos cuya sintaxis es similar a Python. Es usado típicamente por aquellos programadores de los citados lenguajes, que no desean o no pueden usar el entorno de desarrollo de Microsoft, el Visual Studio. (Sarabina, 2008)

# Capítulo 1

---

Características de SharpDevelop:

- Diseñador de formularios para los tres lenguajes mencionados anteriormente.
- Soporte de los Plataformas de .Net 1.1 y 2.0.
- Depurador de código integrado.
- Soporte para la edición de XML.
- Auto inserción de código.
- Interfaz en múltiples idiomas.
- Soporte de plantillas de código.
- Capacidad para escribir código en C#, ASP.NET, ADO.NET, XML y HTML.
- Formateo del texto en diferentes colores según el lenguaje de programación usado.

## **Visual Studio 2005**

Visual Studio 2005 se empezó a comercializar a través de Internet a partir del 4 de octubre de 2005 y llegó a los comercios a finales del mes de octubre en inglés. En castellano no salió hasta el 4 de febrero de 2006. Microsoft eliminó .NET, pero eso no indica que se alejara de la plataforma de la cual se incluyó la versión 2.0.

Se incluye un diseñador de implantación, que permite que el diseño de la aplicación sea validado antes de su implantación. También se incluye un entorno para publicación web y pruebas de carga para comprobar el rendimiento de los programas bajo varias condiciones de carga.

Soporta varios lenguajes de programación tales como Visual C++, Visual C#, Visual J#, ASP y Visual Basic, aunque actualmente se han desarrollado las extensiones necesarias para muchos otros. (TRUSTe, 2010)

Visual Studio 2005 proporciona una amplia gama de herramientas que ofrecen multitud de ventajas para desarrolladores individuales y equipos de desarrollo de software:

- Mayor productividad y obtención más rápida de resultados.
- Creación de soluciones dinámicas basadas en Windows, la web, dispositivos móviles y Office.

# Capítulo 1

---

- Comunicación y colaboración más eficaz en sus equipos de software.
- Garantía de calidad rápida y continua en todo el proceso de desarrollo.

## Selección del IDE

Después de analizar las características de los IDEs, se considera a Microsoft Visual Studio como el IDE indicado para la realización del sistema, pues provee un conjunto de herramientas de desarrollo diseñadas para ayudar a los desarrolladores de software a crear soluciones innovadoras; además que soporta el C# como lenguaje de programación, frente a SharpDevelop, tiene mayor productividad y obtiene resultados de forma más rápida; posee una comunicación más eficaz en sus equipos de software; presenta garantía de calidad durante todo el proceso de desarrollo; es mayormente utilizado y es debido a esto que los conocimientos que se asumen sobre su funcionamiento son mayores que los de SharpDevelop. Para compilar se escoge Mono pues es un proyecto de implementación de la Plataforma .NET de Microsoft utilizando código libre y basado en las especificaciones definidas en la Asociación Europea de Fabricantes de Ordenadores (ECMA). Mono permite que el producto que se quiere lograr con este trabajo pueda llegar a ser multiplataforma y además contiene una serie de elementos útiles para la construcción de un software:

- Un Lenguaje Común de Infraestructura (CLI), máquina virtual que contiene un cargador de clases, justo a tiempo de compilación, recolección de basura y un tiempo de ejecución.
- Una biblioteca de clases que puede trabajar con cualquier lenguaje que trabaja en el Lenguaje Común en tiempo de ejecución (CLR). Mono y .NET compatibles con las bibliotecas de clases.
- Un compilador para el lenguaje C #.

## 1.8 Conclusiones

Con la realización de este capítulo se hizo un análisis de las aplicaciones existentes a nivel internacional, nacional y a nivel UCI. También se concibió un resumen de los principales conceptos asociados al problema, lo cual permite un mayor entendimiento del mismo; se realizó un estudio de algunas de las metodologías y herramientas de desarrollo existentes que podrían servir para la realización del trabajo que se propone;

# Capítulo 1

---

se explica además cuáles fueron las herramientas que se seleccionaron y el por qué de la selección, teniendo en cuenta sus características y ventajas.

Atendiendo el estudio realizado anteriormente, se llega a la conclusión que con todos estos conocimientos incorporados están creadas las condiciones para comenzar a desarrollar una aplicación escritorio y no web, debido a las condiciones de los lugares donde se realizan los eventos de Balonmano, que no tienen conexiones de red. Además para el desarrollo de la aplicación se va a utilizar XP como metodología, Visual Paradigm y Visual Studio como herramientas de modelado y desarrollo respectivamente, XML para el acceso a los datos de la aplicación. Asimismo C# y SQL son los lenguajes de programación que se seleccionaron para la realización del trabajo.

### **Capítulo 2: Desarrollo de la solución**

#### **2.1 Introducción**

El presente capítulo se refiere a las tres primeras fases definidas por XP: Exploración, Planificación e Iteraciones. En él se abordarán temas relacionados con el funcionamiento del sistema. Se describe el flujo actual del proceso de recogida de datos en los juegos de balonmano a nivel nacional, se determinan los objetos que van a ser informatizados y se confecciona una propuesta del sistema.

Se muestran las historias de usuarios escritas por el cliente y se definen las iteraciones por los programadores, constituyéndose además el Plan de Entrega.

#### **2.2 Descripción de las acciones vinculadas al campo de acción**

Actualmente el proceso de recogida y almacenamiento de las estadísticas en los juegos de balonmano a nivel nacional es muy engorroso, pues el mismo se realiza de forma manual. Esto hace que los datos estén sumamente expuestos a perderse o deteriorarse en breve tiempo, pues no se cuenta con un sistema confiable. Además, realizar la manipulación y análisis de los mismos se vuelve difícil y poco eficiente debido a la forma en que se recogen, pues carecen de un mecanismo que logren una mayor organización.

##### **2.2.1 Flujo actual de eventos**

El procedimiento para la recogida de datos durante el juego y el completamiento del modelo de registro final se conciben de la siguiente manera:

Para comenzar a recoger estadísticas en un juego de balonmano primero se hace el llenado del registro primario que consiste en recoger acciones tácticas que se producen en el balonmano a partir del accionar de cada equipo en su juego ofensivo y defensivo.

Luego se tabula toda la información recogida en el modelo primario para hacer unos análisis de la misma en cuatro pasos:

1. Como primer paso se cuantifica en las variables de comportamiento táctico el accionar de cada equipo.



## Capítulo 2

---

2. Como segundo paso se calcula el resultado de cada una de las acciones tácticas.
3. En el tercer paso se calculan estos resultados de manera total por equipo y el resultado del accionar de ambos equipos en el tiempo.
4. Finalmente se calcula el accionar de ambos equipos en el partido.

Por último se realiza el modelo de registro final, en él se recoge el resultado final del análisis táctico en el juego de un equipo y su contrario, el cual será utilizado por los especialistas para:

- Conocer la forma en que desarrolló el juego táctico en su equipo durante una competencia.
- Conocer la forma en que desarrollaron el accionar táctico los contrarios.
- Definir nuevos planteamientos tácticos y estratégicos.
- Planificar a partir de los resultados el entrenamiento relacionado con el tiempo del mismo.
- Modificar estructuras de planificación del entrenamiento referente a su duración.
- Caracterizar el comportamiento táctico, estratégico y físico del juego por categorías.

La forma en que se realiza todo este proceso es sumamente vulnerable, pues los pasos se hacen de forma manual lo que trae consigo que se cometan muchos más errores que si se hacen con el apoyo de un sistema informatizado. Con ese sistema se tendrían, además de la posibilidad de entrar datos con más exactitud y rapidez, las fórmulas que están predeterminadas para realizar los diferentes cálculos que son necesarios hacer para obtener algunos resultados durante la actividad de recogida y almacenamiento de estadísticas en los juegos de balonmano.

### **2.2.2 Objetos de automatización**

Las acciones que se realizan para llevar a cabo el proceso de recogida y almacenamiento de estadísticas en los juegos de balonmano deben ser informatizadas, puesto que es engorroso el manejo de muchos datos de forma manual y trae consigo disímiles errores.

## Capítulo 2

---

Se informatizarán actividades como crear un campeonato e insertar equipos que permitirá registrar los equipos participantes en el certamen. También mediante la opción “iniciar juego”, será posible seleccionar los equipos de un campeonato, así como precisar la fecha en que se cumplirá el juego e identificar el mismo mediante un número de juego. Al crearlo, el anotador podrá añadir al ritmo del juego las estadísticas predeterminadas, de forma rápida y sencilla. Se informatizarán las opciones de consultar datos, donde será posible ver referencias de los equipos siguiendo diferentes criterios.

Para darle solución a lo anteriormente expuesto se desarrollará una aplicación de escritorio.

### **2.2.3 Propuesta del Sistema**

La realización del presente trabajo se proyecta hacia la implementación de un sistema que permita llevar el control de un conjunto de datos que se recogerán durante los juegos de balonmano con el objetivo de ser analizados y permitan tomar decisiones.

Se dispondrá de la opción “crear campeonato”, con los datos distintivos de cada uno, incluyendo el número de equipos que lo integren. La opción “insertar equipos”, dará la posibilidad de insertar equipos asociándolos al campeonato correspondiente. Brindará la opción de “iniciar juego”, la que permitirá seleccionar los equipos que protagonizarán el partido e indicar la fecha en que se realizará el mismo, así como asignarle un número de juego. Luego de haber creado el juego se mostrarán tablas mediante las cuales el anotador podrá recoger durante un partido los datos predeterminados en ellas de forma rápida y sencilla teniendo el apoyo del ratón y del teclado. Brindará además opciones como:

- Consultar Datos de un Juego la que le posibilitará al usuario ver las estadísticas del mismo, especificando para ello el campeonato y el número del juego. Esta opción ofrecerá además la posibilidad de ver estos datos en el primer o segundo tiempo del juego.
- Consultar Datos Generales de un Juego, servirá al usuario para ver los datos finales de un juego. Esto es posible luego de haber seleccionado un campeonato y el número de juego.
- Consultar Datos de un Equipo en un Campeonato permitirá consultar los datos de un equipo frente a otros o a todos los equipos en un campeonato. Para ello

## Capítulo 2

---

se debe haber seleccionado el equipo del cual se verán los datos, el campeonato y los equipos pertenecientes al mismo.

- Consultar Datos de un Equipo en Campeonatos, dará la posibilidad de acceder a los resultados de un equipo en campeonatos donde haya participado, para ello debe haber seleccionado el equipo del cual quiera ver los resultados y los campeonatos que se deseen.

Todas estas opciones permitirán llevar un control más organizado durante el proceso de recogida de estadísticas en los juegos de balonmano, además de proporcionarle al especialista la posibilidad de tomar decisiones a partir de los datos consultados.

### 2.3 Desarrollo de la solución

#### Fase I: Exploración

En esta fase se definen las historias de usuario (HU), que son escritas por el cliente a grandes rasgos en un lenguaje no técnico, describiendo algo de lo que el sistema debe realizar. Al mismo tiempo el equipo de desarrollo se familiariza con las herramientas, tecnologías y prácticas que se utilizarán en el proyecto. La duración de la fase de exploración para la realización del presente trabajo será de un par de semanas, ya que no es un proyecto de gran envergadura y se cuenta con experiencia en el manejo de las herramientas seleccionadas.

#### Fase II: Planificación y Entrega

Esta fase es en la que el cliente, los gerentes y el grupo de desarrolladores deciden el orden en que deberán implementarse las historias de usuario estableciendo la prioridad de cada una de ellas y asociadas a éstas, las entregas.

Típicamente esta fase consiste en una o varias reuniones grupales de planificación. El resultado es un Plan de Entregas.

En esta etapa el cliente establece la prioridad de cada historia de usuario y correspondientemente, los programadores realizan una estimación del esfuerzo necesario de cada una de ellas. Se toman acuerdos sobre el contenido de la primera entrega y se determina un cronograma en conjunto con el cliente. Una entrega debe obtenerse en no más de tres meses.

#### Fase III: Iteraciones

## Capítulo 2

---

En esta fase se incluyen varias iteraciones sobre el sistema antes de ser entregado. El Plan de Entrega está compuesto de iteraciones de no más de tres semanas. En la primera iteración se puede intentar realizar una arquitectura del sistema que pueda ser utilizada durante todo el proyecto. Esto se logra seleccionando las HU que influyan en la creación de esta arquitectura; no obstante no siempre es posible ya que es el cliente quien decide qué HU se implementarán en cada iteración. Al final de la última iteración el sistema estará listo para entrar en producción.

Todo el trabajo de la iteración es expresado en tareas de programación, en donde el cliente tiene mucha participación. Cada una de estas tareas son asignadas a un programador como responsable, pero llevadas a cabo por parejas de estos.

### 2.4 Personas relacionadas con el sistema

Las personas relacionadas con el sistema son aquellas que de alguna forma van a tener contacto con la aplicación, incluyendo a los que van a mantener el sistema actualizado y en correcto funcionamiento.

**Tabla 1 Personas relacionadas con el sistema**

<b>Personas relacionadas con el sistema</b>	<b>Descripción</b>
Usuario	Es la persona que interactúa con el sistema sin haberse creado una cuenta y lo hace sin privilegio alguno. Tiene la posibilidad de acceder a los distintos contenidos y servicios que estén disponibles para relacionarse con el sistema.

#### 2.4.1 Historias de usuario (HU)

Las historias de usuario son utilizadas por la metodología XP para especificar los requisitos del software. Las mismas se caracterizan por su simplicidad pues son escritas por el propio cliente describiendo brevemente y sin terminologías técnicas las características que el software debe tener.

## Capítulo 2

---

No es incorrecto si no se identifican todas las historias de usuario en un principio, o no están descritas detalladamente pues al comienzo de cada iteración estarán registrados los cambios en las historias de usuario y según eso se planificará la siguiente iteración.

Las historias de usuario son descompuestas en tareas de programación y asignadas a los programadores para ser implementadas durante una iteración.

Tabla 2 HU Gestionar Campeonato

<b>Historia de Usuario</b>	
<b>Número:</b> 1	<b>Nombre:</b> Gestionar Campeonato
<b>Usuario:</b> Usuario	
<b>Modificación de la Historia de Usuario:</b>	<b>Iteración Asignada:</b> 1
<b>Prioridad en Negocio:</b> Alta (Alta / Media / Baja)	<b>Puntos Estimados:</b> 0.8
<b>Riegos en Desarrollo:</b> Alto (Alta / Media / Baja)	<b>Puntos Reales:</b> 0.6
<b>Descripción:</b> El usuario inserta el nombre del campeonato, sexo, fecha en que se enmarca el mismo y luego lo crea. Además, puede modificar cualquier atributo.	
<b>Observaciones:</b>	

Tabla 3 HU Gestionar Equipo

<b>Historia de Usuario</b>	
<b>Número:</b> 2	<b>Nombre:</b> Gestionar Equipo
<b>Usuario:</b> Usuario	
<b>Modificación de la Historia de Usuario:</b>	<b>Iteración Asignada:</b> 1
<b>Prioridad en Negocio:</b> Alta (Alta / Media / Baja)	<b>Puntos Estimados:</b> 0.6
<b>Riegos en Desarrollo:</b> Alto (Alta / Media / Baja)	<b>Puntos Reales:</b> 0.6
<b>Descripción:</b> El usuario inserta el nombre del equipo y lo asocia al campeonato al que corresponde, luego lo inserta. Además puede modificar cualquier atributo y adicionar un equipo que haya participado en otro campeonato.	
<b>Observaciones:</b>	

## Capítulo 2

---

Tabla 4 HU Registrar Juego

<b>Historia de Usuario</b>	
<b>Número:</b> 3	<b>Nombre:</b> Registrar Juego
<b>Usuario:</b> Usuario	
<b>Modificación de la Historia de Usuario:</b>	<b>Iteración Asignada:</b> 1
<b>Prioridad en Negocio:</b> Alta (Alta / Media / Baja)	<b>Puntos Estimados:</b> 0.6
<b>Riegos en Desarrollo:</b> Alto (Alta / Media / Baja)	<b>Puntos Reales:</b> 0.6
<b>Descripción:</b> El usuario selecciona el campeonato, los equipos que se van a enfrentar pertenecientes a ese campeonato, además la fecha en que se celebrará el juego y le asigna un identificador al mismo; luego lo crea.	
<b>Observaciones:</b>	

Tabla 5 HU Crear Juego

<b>Historia de Usuario</b>	
<b>Número:</b> 4	<b>Nombre:</b> Crear Juego
<b>Usuario:</b> Usuario	
<b>Modificación de la Historia de Usuario:</b>	<b>Iteración Asignada:</b> 1
<b>Prioridad en Negocio:</b> Alta (Alta / Media / Baja)	<b>Puntos Estimados:</b> 1.0
<b>Riegos en Desarrollo:</b> Alto (Alta / Media / Baja)	<b>Puntos Reales:</b> 0.8
<b>Descripción:</b> Cuando el usuario crea un juego tiene la posibilidad de introducir en tablas pertenecientes a cada equipo datos que están predeterminados en las mismas y además puede llevar el control del cronómetro que será utilizado en el juego.	
<b>Observaciones:</b> Para realizar esta actividad el usuario puede auxiliarse del ratón o el teclado. El cronómetro podrá ser visualizado por el usuario.	

Tabla 6 HU Datos de un Juego

<b>Historia de Usuario</b>	
<b>Número:</b> 5	<b>Nombre:</b> Datos de un Juego
<b>Usuario:</b> Usuario	
<b>Modificación de la Historia de Usuario:</b>	<b>Iteración Asignada:</b> 2
<b>Prioridad en Negocio:</b> Media (Alta / Media / Baja)	<b>Puntos Estimados:</b> 0.8
<b>Riegos en Desarrollo:</b> Medio (Alta / Media / Baja)	<b>Puntos Reales:</b> 0.8
<b>Descripción:</b> El usuario puede consultar los datos de un juego dado el campeonato y el número de juego.	

## Capítulo 2

**Observaciones:**  
Se podrán visualizar los datos mediante una tabla.

**Tabla 7 HU Datos de un Equipo en un Campeonato**

<b>Historia de Usuario</b>	
<b>Número:</b> 6	<b>Nombre:</b> Datos de un Equipo en un Campeonato
<b>Usuario:</b> Usuario	
<b>Modificación de la Historia de Usuario:</b>	<b>Iteración Asignada:</b> 2
<b>Prioridad en Negocio:</b> Media (Alta / Media / Baja)	<b>Puntos Estimados:</b> 0.8
<b>Riegos en Desarrollo:</b> Medio (Alta / Media / Baja)	<b>Puntos Reales:</b> 0.8
<b>Descripción:</b> El usuario puede consultar los datos de un equipo frente a otros equipos o frente a todos los equipos de un campeonato.	
<b>Observaciones:</b> Se podrán visualizar los datos mediante una tabla.	

**Tabla 8 HU Datos de un Equipo en Campeonatos**

<b>Historia de Usuario</b>	
<b>Número:</b> 7	<b>Nombre:</b> Datos de un Equipo en Campeonatos
<b>Usuario:</b> Usuario	
<b>Modificación de la Historia de Usuario:</b>	<b>Iteración Asignada:</b> 2
<b>Prioridad en Negocio:</b> Baja (Alta / Media / Baja)	<b>Puntos Estimados:</b> 0.6
<b>Riegos en Desarrollo:</b> Bajo (Alta / Media / Baja)	<b>Puntos Reales:</b> 0.8
<b>Descripción:</b> El usuario puede consultar los datos de un equipo en otros campeonatos o en todos en los que haya participado.	
<b>Observaciones:</b> Se podrán visualizar los datos mediante una tabla.	

**Tabla 9 HU Datos Generales de un Juego**

<b>Historia de Usuario</b>	
<b>Número:</b> 8	<b>Nombre:</b> Datos Generales de un Juego
<b>Usuario:</b> Usuario	
<b>Modificación de la Historia de Usuario:</b>	<b>Iteración Asignada:</b> 2
<b>Prioridad en Negocio:</b> Baja (Alta / Media / Baja)	<b>Puntos Estimados:</b> 0.6

## Capítulo 2

---

<b>Riegos en Desarrollo:</b> Bajo (Alta / Media / Baja)	<b>Puntos Reales:</b> 0.8
<b>Descripción:</b> El usuario puede consultar los datos generales de un juego al finalizar el mismo.	
<b>Observaciones:</b> Se podrán visualizar los datos mediante una tabla.	

### 2.4.2 Iteraciones

#### Iteración 1

Esta primera iteración tiene como objetivo implementar las historias de usuario 1, 2, 3 y 4 las cuales son: Gestionar Campeonato, Gestionar Equipo, Registrar Juego y Crear Juego. Estas historias de usuario son de vital importancia para el cliente pues constituyen la estructura básica del sistema. Al culminar esta iteración se obtendrá la primera entrega del sistema al cliente con el propósito de recibir retroalimentación del mismo.

#### Iteración 2

En esta iteración está centrada la implementación de las historias de usuario que tienen prioridad media para el cliente; estas son la número 5 y 6: Datos de un Juego y Datos de un Equipo en un Campeonato respectivamente. Al concluir esta iteración se contará con las funcionalidades descritas en esta HU que serán mostradas al cliente con el objetivo de realizar cambios, si es necesario, y recoger sugerencias con respecto al resultado mostrado.

#### Iteración 3

La tercera iteración tiene como objetivo la implementación de las restantes historias de usuario que tienen prioridad baja; estas son la 7 y 8 que tienen por nombre: Datos de un Equipo en Campeonatos y Datos Generales de un Juego. El resultado de esta iteración será la aplicación lista para ponerla a prueba a partir de ese momento y de esa manera evaluar el desempeño de la misma.

### 2.4.3 Estimación de esfuerzo por historia de usuario

Tabla 9 Plan de estimación de esfuerzo por HU

Historias de usuario	Tiempo estimado	Iteración	Tiempo
----------------------	-----------------	-----------	--------



## Capítulo 2

	(semana ideal de trabajo)	asignada	real
<ul style="list-style-type: none"> <li>• Gestionar Campeonato</li> <li>• Gestionar Equipo</li> <li>• Registrar Juego</li> <li>• Crear Juego</li> </ul>	3	1	2.6
<ul style="list-style-type: none"> <li>• Datos de un Juego</li> <li>• Datos de un Equipo en un Campeonato</li> </ul>	2	2	1.6
<ul style="list-style-type: none"> <li>• Datos de un Equipo en Campeonatos</li> <li>• Datos Generales de un Juego</li> </ul>	2	3	1.3

### 2.5 Plan de entrega

Cuando el cliente termina de elaborar las historias de usuario, se comienza la realización del plan de entrega. Este se hace con el objetivo de que los programadores obtengan un estimado de una historia de usuario según su nivel de detalles y se pueda determinar el período de tiempo que puede tardar la implementación de la misma.

El plan de entrega permite la obtención de una clasificación teniendo en cuenta el riesgo que se corre a la hora de implementar las historias de usuario.

Estos datos se muestran en campos que permanecen vacíos en la historia de usuario. El responsable de llenar estos datos es únicamente el programador una vez que haya hecho el análisis requerido de los mismos.

**Tabla 10 Plan de entrega**

Historias de usuario	Iteración 1	Iteración 2	Iteración 3
----------------------	-------------	-------------	-------------

## Capítulo 2

---

	(2da semana de marzo)	(1ra semana de abril)	(1ra semana de mayo)
<ul style="list-style-type: none"><li>• Gestionar Campeonato</li><li>• Gestionar Equipo</li><li>• Registrar Juego</li><li>• Crear Juego</li></ul>	1.0	Terminado	Terminado
<ul style="list-style-type: none"><li>• Datos de un Juego</li><li>• Datos de un Equipo en un Campeonato</li></ul>	0.2	1.0	Terminado
<ul style="list-style-type: none"><li>• Datos de un Equipo en Campeonatos</li><li>• Datos Generales de un Juego</li></ul>	0.1	0.8	Terminado

### 2.6 Conclusiones

Conociendo cómo se lleva a cabo actualmente el proceso de recogida y almacenamiento de datos en los juegos de balonmano a nivel nacional, se hizo una propuesta del sistema; se definieron las personas que están relacionadas con el mismo y se desarrolló la descripción de las historias de usuarios. Luego el cliente indicó la prioridad de cada una, definiendo así el orden en el que serán implementadas. Se decidió que estas historias de usuario serán implementadas en tres iteraciones.

### Capítulo 3: Implementación y pruebas de la solución propuesta

#### 3.1 Introducción

En la metodología XP la implementación de un software debe realizarse de forma iterativa e incremental, de manera que al final de cada iteración se obtenga un producto funcional que debe ser probado y mostrado al cliente, permitiendo de esta manera una constante retroalimentación entre el equipo de desarrollo y el cliente, propiciando que los primeros puedan ampliar su visión del producto basándose en la opinión del segundo. En el presente capítulo se detallan las tres iteraciones definidas para llevar a cabo la construcción del sistema, exponiéndose las tareas generadas por cada historia de usuario y las pruebas realizadas al sistema.

#### 3.2 Diseño del sistema

Para el diseño de las aplicaciones, la metodología XP propone técnicas como las tarjetas Contenido, Responsabilidad y Colaboración (CRC). No obstante el uso de diagramas de clases utilizando notación UML puede aplicarse siempre y cuando influyan en el mejoramiento de la comunicación, no sea un peso su mantenimiento, no sean extensos y se enfoquen en la información importante.

Las tarjetas CRC permiten apartarse al método de trabajo basado en procedimientos y centrarse más en una metodología basada en objetos. Estas tarjetas hacen posible que el equipo entero contribuya al diseño.

Una tarjeta CRC se usa para representar un objeto. El nombre de la clase se escribe en la parte superior de la tarjeta como un título. Las responsabilidades de esta clase son escritas en el lado inferior izquierdo y la colaboración con otras clases se coloca en la parte inferior derecha de cada responsabilidad.

**En la siguiente tabla se ilustra cómo estará conformada una clase mediante una tarjeta CRC:**

Tabla 11 Ejemplo de clase

Nombre de la clase	
Responsabilidades	Colaboradores

## Capítulo 3

---

En la aplicación que se quiere lograr mediante el presente trabajo existen varios objetos, por lo que la misma está constituida por las siguientes clases:

- Controladora
- Campeonato
- Juego
- Equipo
- Acción
- Cronómetro

**Tabla 12 CRC Controladora**

<b>Controladora</b>	
Contiene una lista de campeonatos, la que integra los mismos.	Campeonato

**Tabla 13 CRC Campeonato**

<b>Campeonato</b>	
Define todos los atributos de un campeonato.	Controladora, Juegos, Equipo

**Tabla 14 CRC Juegos**

<b>Juegos</b>	
Define todos los atributos de un juego.	Campeonato, Acciones, Cronómetro, Equipo

## Capítulo 3

---

**Tabla 15 CRC Equipo**

Equipo	
Define todos los atributos de un equipo.	Campeonato

**Tabla 16 CRC Acciones**

Acciones	
Define todos los atributos de las acciones realizadas por los equipos.	Juegos

**Tabla 17 CRC Cronómetro**

Cronómetro	
Contiene todas las funcionalidades necesarias para controlar el tiempo en cada uno de los juegos.	Juegos

### 3.2.1 Arquitectura y Patrones

#### Arquitectura:

Arquitectura es la organización fundamental de un sistema (Naranjo, 2005).

Para la realización de la aplicación que se expone a través del presente trabajo se utilizó la Arquitectura basada en capas a la cual se acude cuando se desea manejar la complejidad de un sistema descomponiendo cada uno de sus elementos en capas, de manera que quedaría como: Presentación, Negocio y Acceso a Datos. Cada capa debe ocuparse de un nivel del problema, y debe tener poca cohesión con las demás. Este es uno de los estilos más importantes en la arquitectura pues tiene consecuencias mínimas, como por ejemplo el cambio en una capa, altera muy poco

## Capítulo 3

---

las otras capas; además al cambiar la implementación de una capa, tiene muy bajos efectos en el resto de la aplicación. Esta es la razón por la cual es extremadamente importante su uso en el sistema a implementar.

### Patrones de diseño:

Cada patrón describe un problema que ocurre una y otra vez en nuestro entorno, para describir después el núcleo de la solución a ese problema de tal manera, que esa solución pueda ser usada más de un millón de veces sin hacerlo ni siquiera dos veces de la misma forma. (Christopher)

En el desarrollo de la aplicación se utilizaron diferentes patrones de diseño para la asignación de responsabilidades (GRASP), los mismos se exponen a continuación:

Tabla 18 Patrón Experto

<b>Nombre</b>	Experto
<b>Solución</b>	Asignar una responsabilidad al experto en información: la clase que cuenta con la información necesaria para cumplir la responsabilidad.
<b>Beneficio</b>	Conserva el encapsulamiento, permite la baja cohesión y el alto acoplamiento.

Tabla 19 Patrón Creador

<b>Nombre</b>	Creador
<b>Solución</b>	Asignarle a la clase <b>B</b> la responsabilidad de crear una instancia de clase <b>A</b> en uno de los siguientes casos:  <b>B</b> agrega los objetos <b>A</b> .  <b>B</b> contiene los objetos <b>A</b> .  <b>B</b> registra las instancias de los objetos <b>A</b> .  <b>B</b> utiliza específicamente los objetos <b>A</b> .  <b>B</b> tiene los datos de inicialización que serán transmitidos a <b>A</b> cuando este objeto sea creado (así que <b>B</b> es un Experto respecto a la creación de <b>A</b> ).  <b>B</b> es un creador de los objetos <b>A</b> . Si existe más de una opción, prefiera la clase <b>B</b> que agregue o contenga la clase <b>A</b> .

## Capítulo 3

---

<b>Beneficio</b>	Permite el bajo acoplamiento.
------------------	-------------------------------

**Tabla 20 Patrón Bajo acoplamiento**

<b>Nombre</b>	Bajo acoplamiento
<b>Solución</b>	Asignar una responsabilidad para mantener bajo acoplamiento.
<b>Beneficio</b>	Brinda soporte a una dependencia escasa y a un aumento de la reutilización.

**Tabla 21 Patrón Alta cohesión**

<b>Nombre</b>	Alta cohesión
<b>Solución</b>	Asignar una responsabilidad de modo que la cohesión siga siendo alta.
<b>Beneficio</b>	Mantiene la complejidad dentro de límites manejables.

**Tabla 22 Patrón Controlador**

<b>Nombre</b>	Controlador
<b>Solución</b>	Asignar la responsabilidad de controlar el flujo de eventos del sistema, a clases específicas. Esto facilita la centralización de actividades (validaciones, seguridad, etc.). El controlador no realiza estas actividades, las delega en otras clases con las que mantiene un modelo de alta cohesión.
<b>Beneficio</b>	Mantiene la complejidad dentro de límites manejables.

### **Aplicación de los patrones de diseño**

La utilización de patrón de diseño Experto se ve evidenciado en la clase Acción, la misma tiene la responsabilidad de realizar una labor específica. Esta clase contiene métodos como ASL (Aprovechamiento de la situación de lanzamiento), Efect\_goles (Efectividad en goles) y TACC (Tiempo total cronometrado en las acciones) pues es la

## Capítulo 3

---

que cuenta con todos los elementos necesarios para realizar dichas tareas. La aplicación del patrón Creador se identifica al haberle asignado la responsabilidad a la clase Juego de crear objetos de las clases Equipo y Acción, ya que la primera contiene a las últimas. La presencia del patrón Bajo Acoplamiento se manifiesta al no utilizar la herencia pues uno de los principales síntomas de un mal diseño y alto acoplamiento es una herencia muy profunda. Para garantizar una Alta Cohesión se tuvo en cuenta que cada elemento del diseño realizara una labor única dentro del sistema, no desempeñada por el resto de los elementos. La idea es que cada clase tenga asociadas las responsabilidades que le corresponden de acuerdo a su comportamiento, un ejemplo de esto está vigente en la clase Acción, la cual realiza actividades únicas como ASL, Efect\_goles y TACC. La utilización del patrón Controlador se refleja mediante la clase Controladora que es la que asigna la responsabilidad de controlar el flujo de eventos del sistema a las demás clases ya que el controlador no realiza esta actividad pues la delega en otras clases con las que mantiene un modelo de alta cohesión.

### 3.3 Fase de Producción

Las historias de usuarios se irán implementando durante el transcurso de la iteración a la cual pertenecen. Es por ello que al principio de éstas, se realiza una revisión del plan de iteraciones y se transforma en caso de ser necesario. En este plan se desglosan las historias de usuarios en tareas de desarrollo, que serán realizadas por los programadores del equipo de desarrollo, asignando a cada una de ellas una o varias personas que serán las responsables de su implementación. Estas tareas son para el uso estricto de los programadores, por lo que pueden ser escritas en lenguaje técnico y no necesariamente fácil de entender por el cliente.

#### Historias de usuario divididas en tareas

Tabla 23 Historias de usuarios divididas en tareas

Historia de Usuario	Tareas
Gestionar Campeonato	1 Crear un formulario para el campeonato 2 Permitir introducir datos en el



## Capítulo 3

---

	<p style="text-align: center;">formulario</p> <p>3 Permitir modificar datos en el formulario</p>
Gestionar Equipo	<p>1 Crear un formulario para el equipo</p> <p>2 Permitir insertar el equipo al campeonato al cual pertenece</p> <p>3 Permitir modificar datos en el formulario creado</p>
Registrar Juego	<p>1 Permitir seleccionar el campeonato</p> <p>2 Permitir seleccionar los equipos participantes en el juego</p>
Crear Juego	<p>1 Mostrar tablas con las acciones que puede realizar cada equipo durante el juego.</p> <p>2 Mostrar el cronómetro para controlar el tiempo durante el juego</p>
Datos de un Juego	<p>1 Crear un formulario</p> <p>2 Permitir seleccionar campeonato y el número de juego</p>
Datos de un Equipo en un Campeonato	<p>1 Crear un formulario</p> <p>2 Permitir seleccionar el equipo del que desea obtener los datos</p> <p>3 Permitir seleccionar el campeonato del cual desea obtener los datos</p> <p>4 Permitir seleccionar los equipos contrarios para obtener las estadísticas de los juegos efectuados</p>
Datos de un equipo en Campeonatos	<p>1 Crear un formulario</p> <p>2 Permitir seleccionar el equipo del cual se desea obtener los datos</p> <p>3 Permitir seleccionar los campeonatos</p>

## Capítulo 3

Datos Generales de un Juego	<ol style="list-style-type: none"><li>1 Crear un formulario</li><li>2 Permitir seleccionar un campeonato</li><li>3 Permitir seleccionar el número de juego del cual se desea ver los datos</li></ol>
-----------------------------	--

Atendiendo la planificación realizada anteriormente, se llevarán a cabo tres iteraciones para el desarrollo del sistema, permitiendo que al final de cada una se logre un producto con todas las características propuestas por el cliente. A continuación se detallan cada una de las iteraciones.

### 3.3.1 Iteración 1

Esta Iteración tendrá como objetivo desarrollar las historias de usuarios que son de máxima importancia para el funcionamiento de la aplicación. Las mismas fueron clasificadas por el cliente con prioridad alta, pues recogen las principales funcionalidades del sistema.

#### 3.3.1.1 Historias de usuario abordadas en la primera iteración

Tabla 24 Historias de usuario abordadas en la primera iteración

Historias de usuario	Estimación	Real
Gestionar Campeonato	0.8	0.6
Gestionar Equipo	0.6	0.6
Registrar Juego	0.6	0.6
Crear Juego	1.0	0.8

#### 3.3.1.2 Tareas de las historias de usuario abordadas en la iteración

Tabla 25 Gestionar Campeonato

Tarea	
Número de tarea: 1	Número de HU: 1

## Capítulo 3

---

<b>Nombre de la tarea:</b> Crear un formulario para el campeonato	
<b>Tipo de tarea:</b> Implementación	
<b>Fecha de inicio:</b>	<b>Fecha de fin:</b>
<b>Programador responsable:</b> Heimer Lambert, María de los A. Obregón	
<b>Descripción:</b> Mediante el formulario que es creado el usuario tiene la posibilidad de ver los datos que debe introducir.	

Tabla 26 Gestionar Campeonato

<b>Tarea</b>	
<b>Número de tarea:</b> 2	<b>Número de HU:</b> 1
<b>Nombre de la tarea:</b> Permitir introducir datos en el formulario	
<b>Tipo de tarea:</b> Implementación	
<b>Fecha de inicio:</b>	<b>Fecha de fin:</b>
<b>Programador responsable:</b> Heimer Lambert, María de los A. Obregón	
<b>Descripción:</b> El sistema brinda la posibilidad al usuario de introducir los datos del campeonato.	

Tabla 27 Gestionar Campeonato

<b>Tarea</b>	
<b>Número de tarea:</b> 3	<b>Número de HU:</b> 1
<b>Nombre de la tarea:</b> Permitir modificar datos en la forma creada	
<b>Tipo de tarea:</b> Implementación	
<b>Fecha de inicio:</b>	<b>Fecha de fin:</b>
<b>Programador responsable:</b>	

## Capítulo 3

---

Heimer Lambert, María de los A. Obregón
<b>Descripción:</b> El sistema brinda la posibilidad al usuario de realizar cambios.

**Tabla 28 Gestionar Equipo**

<b>Tarea</b>	
<b>Número de tarea:</b> 1	<b>Número de HU:</b> 2
<b>Nombre de la tarea:</b> Crear un formulario para el equipo	
<b>Tipo de tarea:</b> Implementación	
<b>Fecha de inicio:</b>	<b>Fecha de fin:</b>
<b>Programador responsable:</b> Heimer Lambert, María de los A. Obregón	
<b>Descripción:</b> Mediante el formulario que es creado el usuario tiene la posibilidad de ver los datos que debe seleccionar e introducir.	

**Tabla 29 Gestionar Equipo**

<b>Tarea</b>	
<b>Número de tarea:</b> 2	<b>Número de HU:</b> 2
<b>Nombre de la tarea:</b> Permitir seleccionar el campeonato al cual pertenece	
<b>Tipo de tarea:</b> Implementación	
<b>Fecha de inicio:</b>	<b>Fecha de fin:</b>
<b>Programador responsable:</b> Heimer Lambert, María de los A. Obregón	
<b>Descripción:</b> El sistema brinda la posibilidad al usuario de seleccionar el campeonato al que pertenece el equipo e introducir el nombre del mismo.	

**Tabla 30 Gestionar Equipo**

<b>Tarea</b>
--------------

## Capítulo 3

---

<b>Número de tarea:</b> 3	<b>Número de HU:</b> 2
<b>Nombre de la tarea:</b> Permitir modificar datos en el formulario creado	
<b>Tipo de tarea:</b> Implementación	
<b>Fecha de inicio:</b>	<b>Fecha de fin:</b>
<b>Programador responsable:</b> Heimer Lambert, María de los A. Obregón	
<b>Descripción:</b> El sistema brinda la posibilidad al usuario de realizar cambios.	

Tabla 31 Registrar Juego

<b>Tarea</b>	
<b>Número de tarea:</b> 1	<b>Número de HU:</b> 3
<b>Nombre de la tarea:</b> Permitir seleccionar el campeonato	
<b>Tipo de tarea:</b> Implementación	
<b>Fecha de inicio:</b>	<b>Fecha de fin:</b>
<b>Programador responsable:</b> Heimer Lambert, María de los A. Obregón	
<b>Descripción:</b> El sistema brinda la posibilidad al usuario de seleccionar el campeonato en el cual se va a desarrollar un juego.	

Tabla 32 Registrar Juego

<b>Tarea</b>	
<b>Número de tarea:</b> 2	<b>Número de HU:</b> 3
<b>Nombre de la tarea:</b> Permitir seleccionar los equipos participantes en el juego	
<b>Tipo de tarea:</b> Implementación	
<b>Fecha de inicio:</b>	<b>Fecha de fin:</b>

## Capítulo 3

---

<b>Programador responsable:</b> Heimer Lambert, María de los A. Obregón
<b>Descripción:</b> El sistema brinda la posibilidad al usuario de seleccionar los equipos participantes en un juego.

Tabla 33 Crear Juego

<b>Tarea</b>	
<b>Número de tarea:</b> 1	<b>Número de HU:</b> 4
<b>Nombre de la tarea:</b> Mostrar tablas con las acciones que puede realizar cada equipo durante el juego	
<b>Tipo de tarea:</b> Implementación	
<b>Fecha de inicio:</b>	<b>Fecha de fin:</b>
<b>Programador responsable:</b> Heimer Lambert, María de los A. Obregón	
<b>Descripción:</b> El sistema muestra dos tablas para los dos equipos participantes en un juego. Las mismas contienen las acciones que pueden realizar cada equipo respectivamente.	

Tabla 34 Crear Juego

<b>Tarea</b>	
<b>Número de tarea:</b> 2	<b>Número de HU:</b> 4
<b>Nombre de la tarea:</b> Permitir seleccionar campeonato y el número de juego	
<b>Tipo de tarea:</b> Implementación	
<b>Fecha de inicio:</b>	<b>Fecha de fin:</b>
<b>Programador responsable:</b> Heimer Lambert, María de los A. Obregón	
<b>Descripción:</b> El sistema muestra el cronómetro que permite al usuario controlar el tiempo durante un juego.	

## Capítulo 3

---

### 3.3.2 Iteración 2

Esta iteración está centrada en desarrollar las historias de usuario, las que el cliente consideró con prioridad media pues no son de vital importancia para el funcionamiento de la aplicación y en su mayoría no poseen un nivel de riesgo muy elevado para el programador.

#### 3.3.2.1 Historias de usuario abordadas en la segunda iteración

Tabla 35 Historias de usuario abordadas en la segunda iteración

Historias de usuario	Estimación	Real
Datos de un Juego	0.8	0.8
Datos de un Equipo en un Campeonato	0.8	0.8

#### 3.3.2.2 Tareas de las historias de usuario abordadas en la iteración

Tabla 36 Datos de un Juego

<b>Tarea</b>	
<b>Número de tarea:</b> 1	<b>Número de HU:</b> 5
<b>Nombre de la tarea:</b> Crear un formulario	
<b>Tipo de tarea:</b> Implementación	
<b>Fecha de inicio:</b>	<b>Fecha de fin:</b>
<b>Programador responsable:</b> Heimer Lambert, María de los A. Obregón	
<b>Descripción:</b> Mediante el formulario que es creado el usuario tiene la posibilidad de ver los datos que debe introducir Permitir seleccionar campeonato y el número de juego	

Tabla 37 Datos de un Juego

## Capítulo 3

---

<b>Tarea</b>	
<b>Número de tarea:</b> 2	<b>Número de HU:</b> 5
<b>Nombre de la tarea:</b> Permitir seleccionar campeonato y el número de juego	
<b>Tipo de tarea:</b> Implementación	
<b>Fecha de inicio:</b>	<b>Fecha de fin:</b>
<b>Programador responsable:</b> Heimer Lambert, María de los A. Obregón	
<b>Descripción:</b> El sistema da la posibilidad al usuario de seleccionar el campeonato y el número del juego del cual quiere ver los datos.	

Tabla 38 Datos de un Equipo en un Campeonato

<b>Tarea</b>	
<b>Número de tarea:</b> 1	<b>Número de HU:</b> 6
<b>Nombre de la tarea:</b> Crear un formulario	
<b>Tipo de tarea:</b> Implementación	
<b>Fecha de inicio:</b>	<b>Fecha de fin:</b>
<b>Programador responsable:</b> Heimer Lambert, María de los A. Obregón	
<b>Descripción:</b> Mediante el formulario que es creado el usuario tiene la posibilidad de ver los datos que debe introducir.	

Tabla 39 Datos de un Equipo en un Campeonato

<b>Tarea</b>	
<b>Número de tarea:</b> 2	<b>Número de HU:</b> 6
<b>Nombre de la tarea:</b> Permitir seleccionar el equipo del que desea obtener los datos	



## Capítulo 3

---

<b>Tipo de tarea:</b> Implementación	
<b>Fecha de inicio:</b>	<b>Fecha de fin:</b>
<b>Programador responsable:</b> Heimer Lambert, María de los A. Obregón	
<b>Descripción:</b> El sistema da la posibilidad al usuario de seleccionar el equipo del que desea obtener los datos.	

**Tabla 40 Datos de un Equipo en un Campeonato**

<b>Tarea</b>	
<b>Número de tarea:</b> 3	<b>Número de HU:</b> 6
<b>Nombre de la tarea:</b> Permitir seleccionar el campeonato del cual desea obtener los datos	
<b>Tipo de tarea:</b> Implementación	
<b>Fecha de inicio:</b>	<b>Fecha de fin:</b>
<b>Programador responsable:</b> Heimer Lambert, María de los A. Obregón	
<b>Descripción:</b> El sistema da la posibilidad al usuario de seleccionar el campeonato del cual desea obtener los datos.	

**Tabla 41 Datos de un Equipo en un Campeonato**

<b>Tarea</b>	
<b>Número de tarea:</b> 4	<b>Número de HU:</b> 6
<b>Nombre de la tarea:</b> Permitir seleccionar los equipos contrarios para obtener las estadísticas de los juegos efectuados	
<b>Tipo de tarea:</b> Implementación	
<b>Fecha de inicio:</b>	<b>Fecha de fin:</b>
<b>Programador responsable:</b>	

## Capítulo 3

---

Heimer Lambert, María de los A. Obregón

**Descripción:**

El sistema da la posibilidad al usuario de seleccionar los equipos contrarios.

### 3.3.3 Iteración 3

El objetivo de esta iteración es desarrollar las historias de usuario definidas por el cliente con prioridad baja pues no influyen en el funcionamiento de la aplicación por lo que los programadores consideraron con un nivel de riesgo bajo.

#### 3.3.3.1 Historias de usuario abordadas en la tercera iteración

Tabla 42 Historias de usuario abordadas en la tercera iteración

Historias de usuario	Estimación	Real
Datos de un equipo en Campeonatos	0.6	0.8
Datos Generales de un Juego	0.6	0.8

#### 3.3.3.2 Tareas de las historias de usuario abordadas en la iteración

Tabla 43 Datos de un Equipo en Campeonatos

<b>Tarea</b>	
<b>Número de tarea:</b> 1	<b>Número de HU:</b> 7
<b>Nombre de la tarea:</b> Crear un formulario	
<b>Tipo de tarea:</b> Implementación	
<b>Fecha de inicio:</b>	<b>Fecha de fin:</b>
<b>Programador responsable:</b> Heimer Lambert, María de los A. Obregón	
<b>Descripción:</b> Mediante el formulario que es creada el usuario tiene la posibilidad de ver los datos que	

## Capítulo 3

---

debe introducir.

**Tabla 44 Datos de un Equipo en Campeonatos**

<b>Tarea</b>	
<b>Número de tarea:</b> 2	<b>Número de HU:</b> 7
<b>Nombre de la tarea:</b> Permitir seleccionar el equipo del cual se desea obtener los datos	
<b>Tipo de tarea:</b> Implementación	
<b>Fecha de inicio:</b>	<b>Fecha de fin:</b>
<b>Programador responsable:</b> Heimer Lambert, María de los A. Obregón	
<b>Descripción:</b> El sistema da la posibilidad al usuario de seleccionar el equipo del cual se desea obtener los datos.	

**Tabla 45 Datos de un Equipo en Campeonatos**

<b>Tarea</b>	
<b>Número de tarea:</b> 3	<b>Número de HU:</b> 7
<b>Nombre de la tarea:</b> Permitir seleccionar los campeonatos	
<b>Tipo de tarea:</b> Implementación	
<b>Fecha de inicio:</b>	<b>Fecha de fin:</b>
<b>Programador responsable:</b> Heimer Lambert, María de los A. Obregón	
<b>Descripción:</b> El sistema da la posibilidad al usuario de seleccionar los campeonatos.	

**Tabla 46 Datos Generales de un Juego**

<b>Tarea</b>
--------------

## Capítulo 3

---

<b>Número de tarea:</b> 1	<b>Número de HU:</b> 8
<b>Nombre de la tarea:</b> Crear un formulario	
<b>Tipo de tarea:</b> Implementación	
<b>Fecha de inicio:</b>	<b>Fecha de fin:</b>
<b>Programador responsable:</b> Heimer Lambert, María de los A. Obregón	
<b>Descripción:</b> Mediante el formulario que es creado, el usuario tiene la posibilidad de ver los datos que debe introducir.	

**Tabla 47 Datos Generales de un Juego**

<b>Tarea</b>	
<b>Número de tarea:</b> 2	<b>Número de HU:</b> 8
<b>Nombre de la tarea:</b> Permitir seleccionar un campeonato	
<b>Tipo de tarea:</b> Implementación	
<b>Fecha de inicio:</b>	<b>Fecha de fin:</b>
<b>Programador responsable:</b> Heimer Lambert, María de los A. Obregón	
<b>Descripción:</b> El sistema da la posibilidad al usuario de seleccionar el campeonato en el cual se celebró el juego del cual se desea ver los datos.	

**Tabla 48 Datos Generales de un Juego**

<b>Tarea</b>	
<b>Número de tarea:</b> 3	<b>Número de HU:</b> 8
<b>Nombre de la tarea:</b> Permitir seleccionar el número de juego del cual se desea ver los datos	
<b>Tipo de tarea:</b> Implementación	

## Capítulo 3

---

<b>Fecha de inicio:</b>	<b>Fecha de fin:</b>
<b>Programador responsable:</b> Heimer Lambert, María de los A. Obregón	
<b>Descripción:</b> El sistema da la posibilidad al usuario de seleccionar el número del juego del cual se desea ver los resultados.	

### 3.4 Pruebas

Uno de los pilares de la metodología 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. También permite aumentar la seguridad de evitar efectos colaterales no deseados a la hora de realizar modificaciones y refactorizaciones.

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 diseñadas por el cliente final. (J. J. Gutiérrez, 2009)

#### 3.4.1 Pruebas de aceptación

Las pruebas de aceptación son más importantes que las pruebas unitarias dado que significan la satisfacción del cliente con el producto desarrollado y el final de una iteración y el comienzo de la siguiente. (J. J. Gutiérrez, 2009). Las historias de usuarios seleccionadas serán traducidas a pruebas de aceptación. Una historia de usuario puede tener todas las pruebas de aceptación que necesite para asegurar su correcto funcionamiento. El objetivo final de estas pruebas es garantizar que los requerimientos han sido cumplidos y que el sistema es aceptable. Una historia de usuario no se considera completa hasta que no ha pasado por sus pruebas de aceptación.

A continuación se muestran las pruebas de aceptación propuestas a realizarse por iteración.

##### 3.4.1.1 Iteración 1

## Capítulo 3

Tabla 49 Caso de Prueba de Aceptación HU1P1

Caso de Prueba de Aceptación	
Código: HU1P1	Historia de Usuario: 1
Nombre: Gestionar Campeonato	
Descripción: Prueba para la funcionalidad de insertar los datos de un campeonato.	
Condiciones de Ejecución: El usuario debe insertar los datos que son necesarios para registrar un campeonato.	
Entradas/Pasos de Ejecución: Se intenta insertar los datos que la aplicación solicita mediante el usuario.	
Resultados Esperados: Los datos son insertados sin generar errores.	
Evaluación de la Prueba: Prueba satisfactoria	

Tabla 50 Caso de Prueba de Aceptación HU1P2

Caso de Prueba de Aceptación	
Código: HU1P2	Historia de Usuario: 1
Nombre: Gestionar Campeonato	
Descripción: Prueba para la funcionalidad de modificar los datos de un campeonato.	
Condiciones de Ejecución: Debe existir al menos un campeonato registrado.	
Entradas/Pasos de Ejecución: Se intenta modificar los datos del campeonato que el usuario seleccione.	
Resultados Esperados: Los datos son modificados sin generar errores.	
Evaluación de la Prueba: Prueba satisfactoria	

Tabla 51 Caso de Prueba de Aceptación HU2P1

Caso de Prueba de Aceptación	
Código: HU2P1	Historia de Usuario: 2

## Capítulo 3

---

<b>Nombre:</b> Gestionar Equipo
<b>Descripción:</b> Prueba para la funcionalidad de insertar los datos de un equipo.
<b>Condiciones de Ejecución:</b> El usuario debe insertar los datos que son necesarios para registrar un equipo.
<b>Entradas/Pasos de Ejecución:</b> Se intenta insertar los datos que la aplicación solicita mediante el usuario.
<b>Resultados Esperados:</b> Los datos son insertados sin generar errores.
<b>Evaluación de la Prueba:</b> Prueba satisfactoria

**Tabla 52 Caso de Prueba de Aceptación HU2P2**

<b>Caso de Prueba de Aceptación</b>	
<b>Código:</b> HU2P2	<b>Historia de Usuario:</b> 2
<b>Nombre:</b> Gestionar Equipo	
<b>Descripción:</b> Prueba para la funcionalidad de modificar los datos de un equipo.	
<b>Condiciones de Ejecución:</b> Debe existir al menos un equipo registrado.	
<b>Entradas/Pasos de Ejecución:</b> Se intenta modificar los datos del equipo que el usuario seleccione.	
<b>Resultados Esperados:</b> Los datos son modificados sin generar errores.	
<b>Evaluación de la Prueba:</b> Prueba satisfactoria	

**Tabla 53 Caso de Prueba de Aceptación HU3P1**

<b>Caso de Prueba de Aceptación</b>	
<b>Código:</b> HU3P1	<b>Historia de Usuario:</b> 3
<b>Nombre:</b> Registrar Juego	
<b>Descripción:</b> Prueba para la funcionalidad de insertar los datos de un juego.	
<b>Condiciones de Ejecución:</b>	

## Capítulo 3

Se deben haber registrado al menos un campeonato y dos equipos. El usuario debe insertar los datos que son necesarios para registrar un juego.
<b>Entradas/Pasos de Ejecución:</b> Se intenta insertar los datos que la aplicación solicita mediante el usuario.
<b>Resultados Esperados:</b> Los datos son registrados sin generar errores.
<b>Evaluación de la Prueba:</b> Prueba satisfactoria

**Tabla 54 Caso de Prueba de Aceptación HU4P1**

Caso de Prueba de Aceptación	
<b>Código:</b> HU4P1	<b>Historia de Usuario:</b> 4
<b>Nombre:</b> Crear Juego	
<b>Descripción:</b> Prueba para la funcionalidad de recoger los datos durante un juego.	
<b>Condiciones de Ejecución:</b> Se debe haber registrado el juego. Se debe haber creado el juego. El usuario debe registrar las jugadas que se ejecuten durante el juego.	
<b>Entradas/Pasos de Ejecución:</b> Se intenta registrar las jugadas que se ejecutan durante el juego mediante el usuario.	
<b>Resultados Esperados:</b> Los datos son registrados sin generar errores.	
<b>Evaluación de la Prueba:</b> Prueba satisfactoria	

### 3.4.2.1 Iteración 2

**Tabla 55 Caso de Prueba de Aceptación HU5P1**

Caso de Prueba de Aceptación	
<b>Código:</b> HU5P1	<b>Historia de Usuario:</b> 5
<b>Nombre:</b> Datos de un Juego	



## Capítulo 3

<b>Descripción:</b> Prueba para la funcionalidad de consultar las estadísticas de un juego.
<b>Condiciones de Ejecución:</b> Se deben haber registrado al menos dos equipos. Se debe haber registrado al menos un campeonato. Se debe haber realizado al menos un juego. El usuario debe seleccionar un campeonato y el número del juego que desea consultar.
<b>Entradas/Pasos de Ejecución:</b> Se intenta consultar las estadísticas del juego mediante el usuario.
<b>Resultados Esperados:</b> Los datos son consultados sin generar errores.
<b>Evaluación de la Prueba:</b> Prueba satisfactoria

Tabla 56 Caso de Prueba de Aceptación HU6P1

Caso de Prueba de Aceptación	
<b>Código:</b> HU6P1	<b>Historia de Usuario:</b> 6
<b>Nombre:</b> Datos de un Equipo en un Campeonato	
<b>Descripción:</b> Prueba para la funcionalidad de consultar las estadísticas de un equipo frente a otros equipos en un campeonato.	
<b>Condiciones de Ejecución:</b> Se deben haber registrado al menos dos equipos. Se debe haber registrado al menos un campeonato. Se debe haber realizado al menos un juego. El usuario debe seleccionar un campeonato, el equipo del cual desea ver las estadísticas y sus equipos contrarios.	
<b>Entradas/Pasos de Ejecución:</b> Se intenta consultar las estadísticas del equipo frente a sus contrarios en un campeonato mediante el usuario.	
<b>Resultados Esperados:</b> Los datos son consultados sin generar errores.	
<b>Evaluación de la Prueba:</b>	

## Capítulo 3

---

Prueba satisfactoria

### 3.4.3.1 Iteración 3

Tabla 57 Caso de Prueba de Aceptación HU7P1

Caso de Prueba de Aceptación	
<b>Código:</b> HU7P1	<b>Historia de Usuario:</b> 7
<b>Nombre:</b> Datos de un Equipo en Campeonatos	
<b>Descripción:</b> Prueba para la funcionalidad de consultar las estadísticas de un equipo en campeonatos donde haya participado.	
<b>Condiciones de Ejecución:</b> Se deben haber registrado al menos dos equipos. Se debe haber registrado al menos un campeonato. Se debe haber realizado al menos un juego. El usuario debe seleccionar el equipo del cual desea ver las estadísticas y los campeonatos en los que desea ver el comportamiento del equipo seleccionado.	
<b>Entradas/Pasos de Ejecución:</b> Se intenta consultar las estadísticas del equipo en campeonatos mediante el usuario.	
<b>Resultados Esperados:</b> Los datos son consultados sin generar errores.	
<b>Evaluación de la Prueba:</b> Prueba satisfactoria	

Tabla 58 Caso de Prueba de Aceptación HU8P1

Caso de Prueba de Aceptación	
<b>Código:</b> HU8P1	<b>Historia de Usuario:</b> 8
<b>Nombre:</b> Datos Generales de un Juego	
<b>Descripción:</b> Prueba para la funcionalidad de consultar las estadísticas generales de un juego al terminar el mismo.	
<b>Condiciones de Ejecución:</b> Se deben haber registrado al menos dos equipos.	

## Capítulo 3

---

<p>Se debe haber registrado al menos un campeonato.</p> <p>Se debe haber realizado al menos un juego.</p> <p>El usuario debe seleccionar el campeonato y número de juego del cual desea ver los resultados.</p>
<p><b>Entradas/Pasos de Ejecución:</b></p> <p>Se intenta consultar las estadísticas de un juego mediante el usuario.</p>
<p><b>Resultados Esperados:</b></p> <p>Los datos son consultados sin generar errores.</p>
<p><b>Evaluación de la Prueba:</b></p> <p>Prueba satisfactoria</p>

### 3.5 Conclusiones

En este capítulo se dio una explicación de cómo se realiza el proceso de implementación del sistema, detallándose de esta manera las tareas que fueron generadas por las historias de usuarios durante las tres iteraciones, en las que se compartió la implementación del sistema, para darle a las mismas una solución organizada y eficiente. Además se propusieron las pruebas de aceptación que brindarán al cliente conformidad y seguridad ante las funcionalidades del sistema.

# Conclusiones Generales

---

## Conclusiones

Vivimos en una sociedad regida por las nuevas tecnologías, donde la informática desempeña un papel fundamental en todos los ámbitos. El gran avance de las mismas, así como el nivel de aceptación en todo el mundo, ha permitido la creación de software que apoyen formidablemente los sectores sociales, destacándose el deporte por ser éste una atracción global.

Mediante esta investigación se realizó un estudio del proceso de recogida y almacenamiento de estadísticas durante los juegos de Balonmano que lleva a cabo la Comisión Nacional de este deporte. De esta manera, se demostró la necesidad de construir un sistema informático capaz de agilizar este proceso, además de proporcionar más durabilidad de los datos que se almacenen.

Se revisó y estudió la existencia de sistemas similares con el objetivo de obtener elementos que aportaran para la construcción del software deseado.

Para el desarrollo del presente trabajo se utilizó la metodología XP, que permitió realizar una documentación del mismo a través de todo el ciclo de desarrollo del software. Esto facilitará el estudio del sistema por parte de otros desarrolladores para futuras mejoras. Además mediante el estudio que se le realizó a diferentes herramientas y lenguajes de desarrollo, se seleccionaron los más adecuados para la construcción del tipo de aplicación requerida.

Se llevó a cabo la captura de requisitos del sistema, así como su análisis y diseño según como lo plantea la metodología seleccionada para la construcción del software y luego implementar el mismo en dependencia de las funcionalidades que debe tener.

Con este trabajo se presenta una aplicación de escritorio que permite registrar eventos que se realicen en el país, para recoger las estadísticas de los juegos que se produzcan durante los mismos. El objetivo es ganar en organización, eficiencia y comodidad en la manipulación de los datos que se recojan y llevar a cabo la planificación en los juegos de balonmano.

## Recomendaciones

---

### Recomendaciones

Como consecuencia del proceso de investigación, surgen recomendaciones a tener en cuenta para futuras mejoras del sistema.

A continuación se exponen las mismas:

- Agregar nuevas funcionalidades al sistema de acuerdo con las expectativas del cliente y que permitan hacer un análisis aún más profundo del desempeño colectivo de los equipos en los juegos de balonmano, como:
  - Poder consultar el comportamiento general de todos los equipos en un campeonato determinado.
  - Consultar las estadísticas de los equipos en un juego a intervalos de diez minutos.
  
- El estudio de la documentación y del sistema para el desarrollo de aplicaciones similares por parte de otros desarrolladores.
  
- La utilización de la aplicación en los Juegos Deportivos, que se celebran todos los años en la UCI, con el objetivo de elevar la calidad del deporte balonmano en la universidad.

## Referencias Bibliográficas

---

### Referencias Bibliográficas

**Anoceto, Magda Mesa. 2006.** *Asesorías estadística en la investigación aplicada al deporte.* 2006.

**Christopher, Alexander.** [En línea]

**Gallego, Óscar Marbán. 2007.** *SQL.* [En línea] 2007.  
<http://pegaso.ls.fi.upm.es/BD/Documentacion/06-SQL.pdf> .

**Gonzalez, Valentý. 2007.** *Sistema de Anotación de Dominó.* [En línea] 2007.  
[http://www.creativaconsultores.com/prop\\_tecnica\\_creativa\\_domino.pdf](http://www.creativaconsultores.com/prop_tecnica_creativa_domino.pdf).

**Gracia, Joaquín. 2003.** *IngenieroSoftware.* [En línea] 2003.  
<http://www.ingenierosoftware.com/analisisydiseno/uml.php>.

**Guzmán, Clara López. 2005.** *Modelo para el Desarrollo de Bibliotecas Digitales Especializadas.* [En línea] 2005.  
[http://www.bibliodgsca.unam.mx/tesis/tes7cllg/sec\\_26.htm](http://www.bibliodgsca.unam.mx/tesis/tes7cllg/sec_26.htm) .

**INDER. 1961.** *Resolución 567.* 1961.

—. **1961.** *Resolución 567.* 1961.

**2009.** *Introducción a la Ingeniería de Software. EVA.* [En línea] 2009.  
<http://eva.uci.cu/mod/resource/view.php?id=11402>.

**J. J. Gutiérrez, M. J. Escalona, M. Mejías, J. Torres. 2009.** *PRUEBAS DEL SISTEMA EN PROGRAMACIÓN.* [En línea] 2009.  
[http://www.lsi.us.es/~javierj/investigacion\\_ficheros/PSISEXTREMA.pdf](http://www.lsi.us.es/~javierj/investigacion_ficheros/PSISEXTREMA.pdf).

**José H. Canós, Patricio Letelier y Ma. Carmen Penadés. 2008.** *Metodologías Agiles en el Desarrollo de Software.* [En línea] 2008.  
<http://www.willydev.net/descargas/prev/TodoAgil.pdf>.

**José Leandro Cabrera, Yosvany Rodríguez Batista. 2007.** *Sistema para la Gestión de Eventos Competitivos y Anotación de los Juegos en el Beisbol.* 2007.

**Larman, Craig. 2004.** *UML y Patrones.* La Habana : s.n., 2004.

## Referencias Bibliográficas

---

- Naranjo, Mauricio. 2005.** *Fundamentos de Definición de Arquitectura de Software.* [En línea] 2005. [http://www.google.com.cu/url?sa=t&source=web&ct=res&cd=5&ved=0CCsQFjAE&url=http%3A%2F%2Fwww.acis.org.co%2Ffileadmin%2FBase\\_de\\_Conocimiento%2FXV\\_Salon\\_de\\_Informatica%2FLucasianLabs.ACIS.MNARANJO.DEF-ARQ.PARTE1.ver2.0.0.ppt&rct=j&q=Concepto+de+arquitectura](http://www.google.com.cu/url?sa=t&source=web&ct=res&cd=5&ved=0CCsQFjAE&url=http%3A%2F%2Fwww.acis.org.co%2Ffileadmin%2FBase_de_Conocimiento%2FXV_Salon_de_Informatica%2FLucasianLabs.ACIS.MNARANJO.DEF-ARQ.PARTE1.ver2.0.0.ppt&rct=j&q=Concepto+de+arquitectura).
- Omar A. Pérez Jacinto, Justo Che Soler. 2008.** *Nociones de Estadísticas Aplicadas a la Investigación Pedagógica.* 2008.
- Parra, Adrián Bulnes. 2007.** *URRIELLU.net.* [En línea] 2007. <http://urriellu.net/es/articles-software/csharp-advantages.html> .
- Pecos, Daniel. 2007.** *PostgreSQL vs. MySQL.* [En línea] 2007. [http://www.netpecos.org/docs/mysql\\_postgres/x15.html](http://www.netpecos.org/docs/mysql_postgres/x15.html) .
- Roberto Espinosa Oliva, Edmundo Claudio Pérez. 2007.** *Sistema Informático para el Control Estadístico y Evaluación del Rendimiento de los Jugadores de Baloncesto.* [En línea] 2007. 5. [http://biblioteca.reduc.edu.cu/biblioteca.virtual/cgi/CD-ROM/otros/UCIENCIA%202007%20\(E\)/ponencias/tcfd/PDF/BASKET.UNI.MTZ-3233173248/BASKET.UNI.MTZ.pdf](http://biblioteca.reduc.edu.cu/biblioteca.virtual/cgi/CD-ROM/otros/UCIENCIA%202007%20(E)/ponencias/tcfd/PDF/BASKET.UNI.MTZ-3233173248/BASKET.UNI.MTZ.pdf) .
- , **2007.** *Sistema Informático para el Control Estadístico y Evaluación del Rendimiento de los Jugadores de Baloncesto.* [En línea] 2007. 5. [http://biblioteca.reduc.edu.cu/biblioteca.virtual/cgi/CD-ROM/otros/UCIENCIA%202007%20\(E\)/ponencias/tcfd/PDF/BASKET.UNI.MTZ-3233173248/BASKET.UNI.MTZ.pdf](http://biblioteca.reduc.edu.cu/biblioteca.virtual/cgi/CD-ROM/otros/UCIENCIA%202007%20(E)/ponencias/tcfd/PDF/BASKET.UNI.MTZ-3233173248/BASKET.UNI.MTZ.pdf).
- S.A, Grupo Soluciones Innova. 2007.** *Rational Unified Process.* [En línea] 2007. <http://www.rational.com.ar/herramientas/rup.html> .
- Sánchez, María A. Mendoza. 2004.** *Metodologías de Desarrollo de Software.* [En línea] 2004. <http://www.willydev.net/Descargas/cualmetodologia.pdf>.
- Sarabina, Emilio. 2008.** *Informática, programación, tren real y modelismo ferroviario.* [En línea] 2008. <http://tecnotrenes.blogspot.com/2008/04/sharpdevelop-una-alternativa-visual.html> .

## Referencias Bibliográficas

---

**TRUSTe, Programa de privacidad. 2010.** *Visual Studio 2005.* [En línea] 2010.  
<http://www.microsoft.com/emea/msdn/visualstudio/eses/getthetrials/>.



## Glosario de Términos

**IDE:** Un entorno de desarrollo integrado o IDE (acrónimo en inglés de Integrated Development Environment), es un programa informático compuesto por un conjunto de herramientas de programación. Puede dedicarse en exclusiva a un sólo lenguaje de programación o bien, poder utilizarse para varios.

**Redes (MAC, IP):** La MAC es un número único asignado a cada tarjeta de red; en cuanto identifica dispositivos de red, es también conocida como la dirección física. El protocolo IP, tiene como misión precisamente traducir la dirección IP de una máquina a la dirección física del adaptador de red.

**GNU:** Es una licencia pública general de GNU o más conocida por su nombre en inglés General Public License. Está orientada principalmente a proteger la libre distribución, modificación y uso de software. Su propósito es declarar que el software cubierto por esta licencia es software libre y protegerlo de intentos de apropiación que restrinjan esas libertades a los usuarios.

**LGPL:** La Licencia Pública General Reducida pretende garantizar su libertad de compartir y modificar el software "libre", esto es para asegurar que el software es libre para todos sus usuarios.

**HTML:** Siglas de HyperText Markup Language (Lenguaje de Marcado de Hipertexto), es el lenguaje de marcado predominante para la elaboración de páginas web. Es usado para describir la estructura y el contenido en forma de texto, así como para complementar el texto con objetos tales como imágenes.

**DOM:** El Modelo de Objetos del Documento (DOM) es una interfaz de programación de aplicaciones (API) para documentos HTML y XML. Define la estructura lógica de los documentos y el modo en que se accede y manipula un documento.

**JavaScripts:** Está basado en objetos, utilizado para acceder a objetos en aplicaciones. Principalmente, se utiliza integrado en un navegador web permitiendo el desarrollo de interfaces de usuario mejoradas y páginas web dinámicas.

**JavaApplets:** Pequeño programa basado en internet y escrito en Java.

## Glosario de Términos

---

**SGML:** Son las siglas de Standard Generalized Markup Language o "Lenguaje de Marcado Generalizado". Consiste en un sistema para la organización y etiquetado de documentos.

**DTD:** Definición de Tipo de Documento es una descripción de estructura y sintaxis de un documento XML o SGML. Su función básica es la descripción del formato de datos, para usar un formato común y mantener la consistencia entre todos los documentos que utilicen la misma DTD.

**Sitio Web:** Un sitio web es un conjunto de páginas web, típicamente comunes a un dominio o subdominio de Internet.

**XSL:** Lenguaje extensible de hojas de estilo es una familia de lenguajes basados en el estándar XML que permite describir cómo la información contenida en un documento XML cualquiera debe ser transformada o formateada para su presentación en un medio.

**ECMA:** Es una organización internacional basada en membresías de estándares para la comunicación y la información.