

**SISTEMA DE INFORMACIÓN PARA LA PLANEACIÓN, ORGANIZACIÓN Y
CONTROL DE LOS EVENTOS DEPORTIVOS Y DEL HISTORIAL
DEPORTIVO DE LOS ESTUDIANTES DE LA UNIVERSIDAD INDUSTRIAL
DE SANTANDER**

**DANNY RICARDO BOTELLO CARRILLO
DANIEL ANDRÉS GONZÁLEZ RUIZ**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE CIENCIAS FISICOMECÁNICAS
ESCUELA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA
BUCARAMANGA
2007**

**SISTEMA DE INFORMACIÓN PARA LA PLANEACIÓN,
ORGANIZACIÓN Y CONTROL DE LOS EVENTOS DEPORTIVOS Y DEL
HISTORIAL DEPORTIVO DE LOS ESTUDIANTES DE LA
UNIVERSIDAD INDUSTRIAL DE SANTANDER**

**DANNY RICARDO BOTELLO CARRILLO
DANIEL ANDRÉS GONZÁLEZ RUIZ**

**Proyecto de grado para optar al título de
Ingeniero de Sistemas**

**Director
Ing. JOSÉ CÁRCAMO SEPÚLVEDA**

**Codirector
Prof. CARLOS FUENTES SANDOVAL**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE CIENCIAS FISICOMECÁNICAS
ESCUELA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA
BUCARAMANGA
2007**

AGRADECIMIENTOS

Los autores expresan su agradecimiento a:

Carlos Fuentes Sandoval, Profesor y Codirector del presente proyecto por su colaboración durante el desarrollo del proyecto y por contribuir con sus valiosos conocimientos, aportes y experiencia en la Organización Deportiva para la realización de esta herramienta.

A toda la planta profesoral del Departamento de Educación Física y Deportes, quienes con su experiencia y colaboración ayudaron al buen curso del diseño y desarrollo de la aplicación.

CONTENIDO

	Pág.
Introducción	1
1. Aspectos Generales	2
1.1. Objetivos	2
1.1.1. Objetivo General	2
1.1.2. Objetivos Específicos	2
1.2. Justificación	3
1.2.1. Antecedentes e Identificación del Problema	3
1.3. Viabilidad e Impacto	4
1.4. Organización del Documento	5
2. Marco Teórico	7
2.1. Arquitectura del Software	7
2.1.1. Modelos o Vistas	8
2.1.2. Arquitecturas más Comunes	8
2.2. Soluciones Basadas en Tecnología Internet	13
2.2.1. Internet	13
2.2.2. Intranet	14
2.2.3. Extranet	14
2.2.4. Protocolo TCP/IP	15
2.3. Aplicaciones Web	15
2.3.1. Servidor Web	16
2.3.2. Protocolo http	16
2.3.3. Funcionamiento de una Aplicación Web	16
2.3.4. Tecnologías para el Desarrollo Web	17
2.4. Tecnologías Usadas	21
2.4.1. PHP	21

2.4.2. MySQL	23
2.4.3. Apache Web Server	24
3. Marco Metodológico	26
3.1. Introducción	26
3.2. Ciclos de Vida del Desarrollo Software	27
3.2.1. Lineal	27
3.2.2. DRA	28
3.2.3. Prototipos	29
3.2.4. Espiral	30
3.2.5. Metodología Escogida	31
3.3. UML	32
4. Análisis de Requerimientos y Diseño Global	34
4.1. Recolección de Requerimientos	34
4.2. Especificación del Problema	34
4.2.1. Oportunidad de la Institución	34
4.2.2. Sentencia que Define el Problema	35
4.2.3. Descripción de Usuarios	35
4.2.4. Roles	36
4.3. Identificación de Procesos	36
4.4. Proceso de Inscripción	37
4.4.1. Problema Actual	37
4.4.2. Términos Usados	37
4.4.3. Descripción del Proceso de Inscripción	37
4.4.4. Recomendaciones o Supuestos	38
4.5. Proceso de Organización	38
4.5.1. Problema Actual	38
4.5.2. Términos Usados	39
4.5.3. Documentos a Manejar	39
4.5.4. Descripción del Proceso de Organización	40
4.5.5. Recomendaciones o Supuestos	40
4.6. Proceso de Resultados	41

4.6.1. Problema Actual	41
4.6.2. Términos Usados	41
4.6.3. Documentos a Manejar	42
4.6.4. Descripción del Proceso de Resultados	42
4.6.5. Recomendaciones o Supuestos	43
4.7. Diseño Global de Casos de Uso	43
4.8. Diseño Global del Modelo de Datos	45
5. Diseño Detallado	47
5.1. Primer Miniproyecto	47
5.1.1. Selección y Generación del Evento	47
5.1.2. Inserción de Responsables del Evento	48
5.1.3. Inscripción de Jugadores	48
5.1.4. Generación de Carnés o Carnetización	50
5.2. Segundo Miniproyecto	51
5.2.1. Elaboración del Fixture Equipos	52
5.2.2. Acceso al Sistema por Sesiones	52
5.2.3. Enviar Información por Correo Electrónico	55
5.2.4. Modificación de Delegados y Jugadores	56
5.2.5. Validación de Inscripciones	56
5.2.6. Sección de Contáctenos	56
5.3. Tercer Miniproyecto	58
5.3.1. Elaboración del Fixture Individuales	59
5.3.2. Elaboración de la Programación	60
5.3.3. Eliminación de Usuarios, Equipos y Jugadores	61
5.3.4. Inserción de Marcadores	62
5.3.5. Tabla de Posiciones	64
5.4. Cuarto Miniproyecto	65
5.4.1. Mostrar Fixture, Programación y Resultados en Pantalla	66
5.4.2. Sección de Sanciones	67
5.4.3. Certificado de Participación	68

6. Conclusiones	69
7. Recomendaciones	70
8. Bibliografía	71
9. Anexos	74

LISTA DE FIGURAS

	Pág.
Figura 1. Modelo Cliente-Servidor	10
Figura 2. Arquitectura de Tres Niveles	11
Figura 3. Funcionamiento de una Aplicación Web	17
Figura 4. PHP	22
Figura 5. Ciclo de Vida Lineal	28
Figura 6. Modelo de Desarrollo Rápido de Aplicaciones	28
Figura 7. Ciclo de Vida con Prototipos	29
Figura 8. Ciclo de Vida Espiral	31
Figura 9. Diagrama de Casos de Uso	45
Figura 10. Modelo de Datos de los Deportes de Equipo	46
Figura 11. Modelo de Datos de los Deportes Individuales	46
Figura 12. Diseño Inicial de Acceso al Sistema	48
Figura 13. Crear Evento e Inserción de Responsables del Evento	49
Figura 14. Inscripción de Jugadores a los Deportes del Evento	50
Figura 15. Carnetización	51
Figura 16. Elaboración del Fixture para Deportes de Equipo	53
Figura 17. Crear Contraseñas	54
Figura 18. Modificar Contraseñas	54
Figura 19. Enviar Información por Correo Electrónico	55
Figura 20. Modificación de Delegados y/o Jugadores	57
Figura 21. Validación de Inscripciones	58
Figura 22. Contáctenos	59
Figura 23. Elaboración del Fixture para Deportes Individuales	60
Figura 24. Elaboración de la Programación de los Encuentros	61
Figura 25. Eliminación del Usuario	62
Figura 26. Eliminar Equipo y/o Jugadores	63

Figura 27. Ingreso de Marcadores	64
Figura 28. Partidos Declarados W.O. y Aplazados	65
Figura 29. Tabla de Posiciones	66
Figura 30. Mostrar Documento	66
Figura 31. Sanciones	67
Figura 32. Certificado de Participación	68
Figura 33. Página Principal	74
Figura 34. Top	75
Figura 35. Menú	76
Figura 36. Main	76
Figura 37. Menú Generar	77
Figura 38. Menú Inscripciones y Menú Carrera Atlética	78
Figura 39. Menú Modificar y Menú Eliminar	79
Figura 40. Menú Carnetizar y Menú Notificar	80
Figura 41. Menú Fixture	81
Figura 42. Menú Programación y Menú Resultados	82
Figura 43. Menú Tabla de Posiciones	82

LISTA DE ANEXOS

	Pág.
Anexo A. Manual de Usuario	74
Anexo B. Copias de Seguridad	83
Anexo C. Glosario de Términos de Organización Deportiva	85

RESÚMEN

TÍTULO: SISTEMA DE INFORMACIÓN PARA LA PLANEACIÓN, ORGANIZACIÓN Y CONTROL DE LOS EVENTOS DEPORTIVOS Y DEL HISTORIAL DEPORTIVO DE LOS ESTUDIANTES DE LA UNIVERSIDAD INDUSTRIAL DE SANTANDER^Δ

AUTORES: BOTELLO CARRILLO Danny Ricardo
GONZÁLEZ RUIZ Daniel Andrés^{ΔΔ}

PALABRAS CLAVES: Aplicación Web, Universidad, PHP, Software Libre, Organización Deportiva, Partidos, Fixture, Sistema de Información, Programación.

DESCRIPCIÓN:

El trabajo de grado aquí presentado consiste en el desarrollo de una aplicación web orientada a la automatización de los procesos de planificación, organización y control de los eventos deportivos al interior de la Universidad Industrial de Santander y más específicamente del Departamento de Educación Física y Deportes. La aplicación está diseñada para que funcione al interior del departamento, basada en una arquitectura cliente-servidor e implementada con herramientas de software libre.

El desarrollo de la aplicación se basó en la metodología del ciclo de vida en espiral, el cual a su vez se soporta en la elaboración de prototipos de pequeñas partes del problema, en su evaluación y evolución a partir de la aceptación de los objetivos alcanzados, y para la documentación del sistema se utilizó el Lenguaje de Modelado Unificado (UML), que proporciona unas reglas que nos indican como crear y leer modelos bien formados.

El uso de esta aplicación estará en manos del organizador de eventos deportivos de la universidad, el profesorado del departamento y de los estudiantes que se inscriban en los eventos programados. La herramienta estará en capacidad de dar información concisa de los resultados y demás formalidades del evento en formato PDF y generar datos históricos deportivos de los estudiantes, además está en capacidad de realizar la carnetización de los participantes en los eventos y de expedir el certificado de participación de dichos eventos.

^Δ Proyecto de Grado en la modalidad de investigación

^{ΔΔ} Facultad de Ingenierías Físico-Mecánicas, Escuela de Ingeniería de Sistemas e Informática. Directores:
Ing. José Cárcamo Sepúlveda – Prof. Carlos Fuentes Sandoval.

ABSTRACT

TITLE: INFORMATION SYSTEM FOR PLANNING, ORGANIZATION AND CONTROL OF THE SPORT EVENTS AND THE SPORT RECORD OF THE STUDENTS OF THE INDUSTRIAL UNIVERSITY OF SANTANDER^Δ

AUTHORS: BOTELLO CARRILLO Danny Ricardo
GONZÁLEZ RUIZ Daniel Andrés ^{ΔΔ}

KEY WORDS: Web application, University, PHP, free software, Sport Organization, Fixture, Game, Information system.

DESCRIPTION:

The work of degree presented here consists of the development of an application Web oriented to the automatization of the processes of planning, organization and control of the sport events to the interior of the Industrial University of Santander and more specifically of the Department of Physical education and Sports. The application is designed so that it works to the interior of the department, cradle in an implemented architecture client-servant and with free software tools.

The development of the application was based on the methodology of the service life in spiral, which is supported as well in the elaboration of prototypes of small parts of the problem, in its evaluation and evolution from the acceptance of the reached objectives, and for the documentation of the system the UNIFED MODEL LANGUAGE was used (UML), that provides rules that they indicate to us like creating and reading formed models affluent.

The use of this application will be into the hands of the organizer of sport events of the university, the teaching staff of the department and the students who register in the programmed events. The tool will be in capacity to give to concise information of the results and other formalities of the event in format PDF and of generating historical data sport of the students, it is also in capacity of carrying out the identification card of the participants in the events and of sending the certificate of participation of this events.

^Δ Thesis Project in investigation category

^{ΔΔ} Physical – Mechanical Engineering Department System and informatics Engineering Faculty. Directors:
Ing. José Cárcamo Sepúlveda – Prof. Carlos Fuentes Sandoval.

INTRODUCCIÓN

El avance de las tecnologías de información, implica que usuarios de los sistemas de información alrededor del mundo se capaciten diariamente y se adapten de manera ágil y efectiva a los procesos de cambio que se presentan en el ámbito informático, en especial en las instituciones educativas universitarias y sus organizaciones deportivas. La Universidad Industrial de Santander y más específicamente el Departamento de Educación Física y Deportes no están exentos a lo dicho anteriormente, y por ello se emprende la creación de una aplicación que optimice el proceso actual de organización y control de los eventos deportivos al interior de la Universidad.

El objetivo del documento que se presenta a continuación es ilustrar el proceso de desarrollo de una aplicación orientada al Web y más específicamente, el problema seleccionado para desarrollar un trabajo de grado denominado: SISTEMA DE INFORMACIÓN PARA LA PLANEACIÓN, ORGANIZACIÓN Y CONTROL DE LOS EVENTOS DEPORTIVOS Y DEL HISTORIAL DEPORTIVO DE LOS ESTUDIANTES DE LA UNIVERSIDAD INDUSTRIAL DE SANTANDER¹.

Para poder realizar este trabajo de grado ha sido de vital importancia desarrollar unas actividades previas concernientes a la documentación en las diversas áreas del saber de la Ingeniería de Sistemas; tales como: Bases de Datos, Ingeniería del Software, tecnologías Web, y de la Organización Deportiva.

Para ubicarnos en el contexto del problema nos fue necesario leer una serie de libros y artículos relacionados con los temas nombrados anteriormente, sobretodo lo concerniente a la organización deportiva, también vivimos la experiencia de participar en el proceso de organización de un evento deportivo, buscamos información sobre los procesos orientados al Web y al comenzar la fase de implementación nos fue necesario familiarizarnos con las tecnologías a usar mediante la búsqueda de libros, libros digitales y tutoriales relacionados con ellas.

¹ En adelante se llamará a la UNIVERSIDAD INDUSTRIAL DE SANTANDER como la UIS.

1. ASPECTOS GENERALES

El objetivo de este capítulo es mostrar de forma clara el alcance del trabajo de grado planteado, por ello es necesario describir los fundamentos básicos que nos permitirán cumplir con el objetivo final.

Es en este capítulo donde se marca el inicio de un proceso por fases que nos dará los parámetros, haciendo énfasis en las actividades más importantes, que nos permitieron avanzar hasta cumplir con todos los objetivos propuestos en este proyecto.

Este capítulo se encuentra dividido en tres secciones:

- ✓ Objetivos
- ✓ Justificación
- ✓ Impacto y viabilidad

1.1. OBJETIVOS

1.1.1. Objetivo general

Diseñar e implementar un sistema de información para la planeación, organización y gestión de los eventos deportivos y del historial deportivo de los estudiantes de la Universidad Industrial de Santander; con el fin de automatizar las tareas correspondientes a la estructuración de las actividades competitivas al interior de la Universidad, y de registrar la participación de los estudiantes en la ficha de historial deportivo respectiva.

1.1.2. Objetivos específicos

Elaborar un sistema software que permita:

- ✓ Automatizar el registro de los participantes en las diferentes actividades deportivas-recreativas, individuales y por equipo.
- ✓ Automatizar el proceso de organización del fixture y la programación de cada actividad deportiva-recreativa.

- ✓ Elaborar las plantillas específicas para el control de los diferentes aspectos organizativos de los eventos deportivos-recreativos.
- ✓ Registrar y gestionar el historial deportivo del estudiante durante la permanencia en la Universidad.
- ✓ Certificar el tiempo de dedicación a la actividad deportiva-recreativa y la actitud participativa en las mismas.

1.2. JUSTIFICACIÓN

1.2.1. Antecedentes e identificación del problema.

✓ Ubicación.

La UIS se encuentra ubicada en la carrera 27 con calle 9 en la ciudadela universitaria. Situada en la zona norte de la ciudad de Bucaramanga, Barrio La Universidad, en la comuna 3 San Francisco. En su entorno se encuentran los colegios: Santander e Instituto Técnico Superior "Dámaso Zapata", la universidad UDI y el Estadio Departamental Alfonso López.

El sector está habitado por personas de clase media. Los estudiantes que ingresan a la UIS en su mayoría pertenecen a la clase media, puesto que no sólo recibe alumnos de la ciudad de Bucaramanga y su área Metropolitana, sino de todo el departamento así como también de otros departamentos.

✓ Identificación del problema.

El avance actual de la tecnología de información precisa que las organizaciones deportivas adopten procesos automatizados, que agilicen el flujo de información específica de los eventos deportivos recreativos para su eficiencia administrativa.

El Departamento de Educación Física y Deportes² de la UIS con este proyecto busca mejorar su gestión, basándose en estas nuevas tecnologías de información para lograr una mejor planeación de las actividades deportivas ya que los métodos usados no realizan su función de forma rápida y eficiente.

Actualmente este proceso de organización de los eventos deportivos se desarrolla manualmente y solo se emplean las herramientas básicas de la

² En adelante identificaremos al Departamento de Educación Física y Deportes por la sigla: **DEFD**

informática como Word o Excel para las formalidades³ del evento, lo que implica dificultades en la transmisión de la información, dado que el organizador debe acudir a sus manuscritos y apoyarse en la hoja de cálculo para el control y publicación de resultados; además la falta de información oportuna, puede provocar inconvenientes a la hora de tomar decisiones de cualquier índole organizativo, tales como presentar resultados y recopilar las estáticas de las diferentes actividades.

El Sistema de Información a desarrollar permitirá que los procesos de planeación, organización y gestión de los eventos deportivos recreativos se realicen de manera ágil y eficiente logrando ahorrar tiempo en su ejecución, en el flujo de información y en la recopilación de las estadísticas deportivas⁴. De igual modo, este software permitirá automatizar la inscripción y registro de los participantes, y sistematizar el proceso de generación del fixture⁵ y de la programación de los encuentros, también se generarán todas las planillas utilizadas para el control de los enfrentamientos de cada actividad.

Por otra parte, se pretende influir en la cultura deportiva de la comunidad Universitaria⁶ mediante el registro de las estadísticas deportivas y del historial deportivo estudiantil⁷, datos que motivan a la comunidad universitaria a dejar huella durante su permanencia en la universidad, contribuyendo así a la formación integral y a una cultura de vida sana del futuro egresado.

1.3. VIABILIDAD E IMPACTO

Con relación a este tipo de proyecto existen pocos antecedentes en la UIS, tomando como referente la información disponible en la biblioteca de la universidad, pues no se encontraron referencias de trabajos como este; se ha logrado encontrar algunas pautas para el desarrollo del software en algunos libros y como complemento se cuenta con información, trabajos realizados en otros países y la codirección especializada en el campo de la administración y organización deportiva por parte del DEFD, lo cual nos brinda el soporte necesario para la investigación y creación de la aplicación.

También se dispone de recursos técnicos para el soporte del desarrollo de la solución software dada como son los lenguajes de programación, manejadores de base de datos, computadores, Internet, etc. Y se cuenta con los equipos requeridos, como son los equipos de desarrollo dados por el DEFD, y los servidores externos o propios del Departamento que podrían alojar el sistema de información desarrollado, para colocarlo al servicio de la Universidad.

³ Estas formalidades son: comunicados, resultados, cartas, etc.

⁴ Estadísticas deportivas o record, como: goleador, encestador, etc.

⁵ Terminó en inglés que en español significa cuadro de enfrentamientos.

⁶ La comunidad Universitaria esta compuesta además de estudiantes y profesores, de empleados, visitantes, estudiantes de otras universidades y colegios, y de la comunidad en general.

⁷ Tiempo del estudiante dedicado a las participaciones deportivas recreativas.

Se tiene referencia de una herramienta software utilizada y promocionada por el Instituto de la Juventud, el deporte y la recreación de Bucaramanga – INDERBU, el cual después de ser analizado por el Departamento de Educación Física y Deportes no se adapta a las necesidades actuales del Departamento, una de las razones tienen que ver con la rigidez de la aplicación, pues ésta no permite la creación de eventos con deportes diferentes a los básicos, es decir, fútbol, baloncesto, voleibol, atletismo, etc. Y como es sabido, en la Universidad se practican una gran variedad de deportes. Por tal motivo se inicia este proyecto buscando desarrollar un sistema que se ajuste a las exigencias de la UIS.

Por otro lado se busca promover la cultura deportiva dentro de la Comunidad Universitaria⁸, incentivando la participación de la misma en los diferentes eventos organizados por el DEFD, para lograr una formación integral que será de vital relevancia en el ámbito laboral tanto en nuestro país como en el extranjero, en los cercanos años.

1.4. ORGANIZACIÓN DEL DOCUMENTO

El documento consta de cinco (5) capítulos y tres (3) anexos. En el primer capítulo se muestran los objetivos y fundamentos generales del proyecto de forma resumida.

En el segundo capítulo se desglosa la información teórica considerada necesaria para una buena comprensión del proyecto, dentro de esta información se podrán encontrar conceptos sobre tecnologías Web, arquitecturas de software y de organización deportiva, que pueden colaborar en el desarrollo del presente proyecto.

El tercer capítulo se centra en la selección de un marco metodológico. Se presentan los principales ciclos de vida para el desarrollo del software, de manera que se muestra al lector los aspectos clave por los cuales se seleccionó el modelo de espiral como ciclo de vida a seguir.

En el cuarto capítulo abarcamos los temas de la recolección de requisitos y del diseño global del sistema, también identificamos los procesos involucrados dentro del desarrollo y los pasos que se siguen para su comprensión y puesta en marcha en el software.

Por último, en el quinto capítulo, hablamos del diseño detallado de la solución planteada, nos apoyamos en los diagramas de actividades de UML para dar mayor entendimiento del desarrollo realizado.

⁸ La comunidad Universitaria esta compuesta además de estudiantes y profesores, de empleados, visitantes, estudiantes de otras universidades y colegios, y de la comunidad en general.

En el anexo A describiremos el manual de usuario de la presente aplicación y en el anexo B explicaremos la forma de realizar los backups, o copias de seguridad, de las bases de datos de este sistema y, por último, en el anexo C habrá un glosario de los términos más comunes de la Organización Deportiva.

2. MARCO TEÓRICO

En este capítulo se presenta todo lo relacionado con la descripción de las herramientas que se utilizaron en el desarrollo del software, al igual que la conceptualización de los muchos de los términos que se emplearon en su desarrollo. No es una transcripción al pie de la letra de los fundamentos teóricos de los sistemas de información, ni de las tecnologías Internet, es más bien un abrebocas a lo que el lector debe saber antes de empezar a describir el desarrollo de este trabajo de grado.

Este marco teórico está dividido en cuatro partes:

- ✓ Arquitectura del Software
- ✓ Soluciones Basadas en Tecnología Internet
- ✓ Aplicaciones

2.1. ARQUITECTURA DEL SOFTWARE

En los inicios de la informática, la programación se consideraba un arte, debido a la dificultad que entrañaba para la mayoría de los mortales, pero con el tiempo se han ido desarrollando metodologías y fórmulas o trucos para conseguir nuestros propósitos. Y a todas estas técnicas se les ha dado en llamar Arquitectura Software.

Una Arquitectura Software, también denominada Arquitectura lógica, consiste en un conjunto de patrones y abstracciones coherentes que proporcionan el marco de referencia necesario para guiar la construcción del software para un sistema de información.

La arquitectura software establece los fundamentos para que analistas, diseñadores, programadores, etc. trabajen en una línea común que permita alcanzar los objetivos y necesidades del sistema de información.

Una arquitectura software se selecciona y diseña con base en unos objetivos y restricciones. Los objetivos son aquellos prefijados para el sistema de información, pero no solamente los de tipo funcional, también otros objetivos como la mantenibilidad, auditabilidad, flexibilidad e interacción con otros sistemas de información. Las restricciones son aquellas limitaciones derivadas de las tecnologías disponibles para implementar sistemas de información. Unas arquitecturas son más recomendables de implementar con ciertas tecnologías mientras que otras tecnologías no son aptas para determinadas arquitecturas.

Por ejemplo, no es viable emplear una arquitectura software de tres capas para implementar sistemas en tiempo real.

La arquitectura software define, de manera abstracta, los componentes que llevan a cabo alguna tarea de computación, sus interfaces y la comunicación entre ellos. Toda arquitectura software debe ser implementable en una arquitectura física, que consiste simplemente en determinar qué computadora tendrá asignada cada tarea de computación.

2.1.1. Modelos o vistas.

Toda arquitectura software debe describir diversos aspectos del software. Generalmente, cada uno de estos aspectos se describe de una manera más comprensible si se utilizan distintos modelos o vistas. Es importante destacar que cada uno de ellos constituye una descripción parcial de una misma arquitectura y es deseable que exista cierto solapamiento entre ellos. Esto es así porque todas las vistas deben ser coherentes entre sí, dado que describen la misma cosa.

Cada paradigma de desarrollo exige diferente número y tipo de vistas o modelos para describir una arquitectura. No obstante, existen al menos tres vistas absolutamente fundamentales en cualquier arquitectura:

- ✓ *La visión estática:* describe qué componentes tiene la arquitectura.
- ✓ *La visión funcional:* describe qué hace cada componente.
- ✓ *La visión dinámica:* describe cómo se comportan los componentes a lo largo del tiempo y como interactúan entre sí.

Las vistas o modelos de una arquitectura pueden expresarse mediante uno o varios lenguajes. El más obvio es el lenguaje natural, pero existen otros lenguajes tales como los diagramas de estado, los diagramas de flujo de datos, etc. Estos lenguajes son apropiados únicamente para un modelo o vista. Afortunadamente existe cierto consenso en adoptar UML (Lenguaje de Modelado Unificado) como lenguaje único para todos los modelos o vistas. Sin embargo, un lenguaje generalista corre el peligro de no ser capaz de describir determinadas restricciones de un sistema de información o expresarlas de manera incomprensible.

2.1.2. Arquitecturas más comunes.

Generalmente, no es necesario inventar una nueva arquitectura software para cada sistema de información. Lo habitual es adoptar una arquitectura conocida

en función de sus ventajas e inconvenientes para cada caso en concreto. Así, las arquitecturas más universales son:

✓ **Monolítica.**

La arquitectura monolítica implementa todos los servicios del sistema operativo en el espacio dominado por el modo supervisor del computador, es decir, un conjunto de procedimientos donde cada uno de ellos puede llamar a los demás cuando quiera. Esto conduce a un programa donde se tiene un proceso principal que utiliza una serie de procedimientos para resolver el servicio requerido. Estos procedimientos a su vez pueden utilizar otros procedimientos para realizar determinadas funciones.

Las desventajas de este tipo de arquitectura son evidentes. Es más difícil destacar alguna ventaja sobre otro tipo de arquitecturas. Una ventaja podría ser que la calificación del personal informático para desarrollar y mantener este tipo de entornos no suele ser muy grande. El mantenimiento y posibilidades de crecimiento de este tipo de arquitecturas se hace cada vez más difícil, dando lugar a situaciones insostenibles.

✓ **Cliente-servidor.**

Llamado modelo cliente-servidor es una forma de dividir y especializar programas y equipos de cómputo a fin de que la tarea que cada uno de ellos realizada se efectúe con la mayor eficiencia, y permita simplificarlas.

En esta arquitectura la capacidad de proceso está repartida entre el servidor y los clientes. En la funcionalidad de un programa distribuido se pueden distinguir tres (3) capas o niveles:

1. Manejador de Base de Datos (Nivel de almacenamiento)
2. Procesador de aplicaciones o reglas del negocio (Nivel lógico)
3. Interfase del usuario (Nivel de presentación)

En un comienzo, los mainframes concentraban la funcionalidad de almacenamiento y lógica, y a ellos se conectaban terminales tontas (consisten de un teclado y una pantalla de salida, que puede ser usada para dar entrada y transmitir datos, o desplegar datos desde una computadora remota a la cual se está conectado), posiblemente en sitios remotos.

Figura 1: Modelo Cliente-Servidor



En el modelo cliente-servidor, en cambio, el trabajo se reparte entre dos ordenadores. De acuerdo con la distribución de la lógica de la aplicación hay dos posibilidades: *Cliente liviano (o cliente fino)* si el cliente solo se hace cargo de la presentación; o *Cliente pesado (o cliente grueso)* si el cliente asume también la lógica del negocio.

➤ **Ventajas de la arquitectura cliente-servidor.**

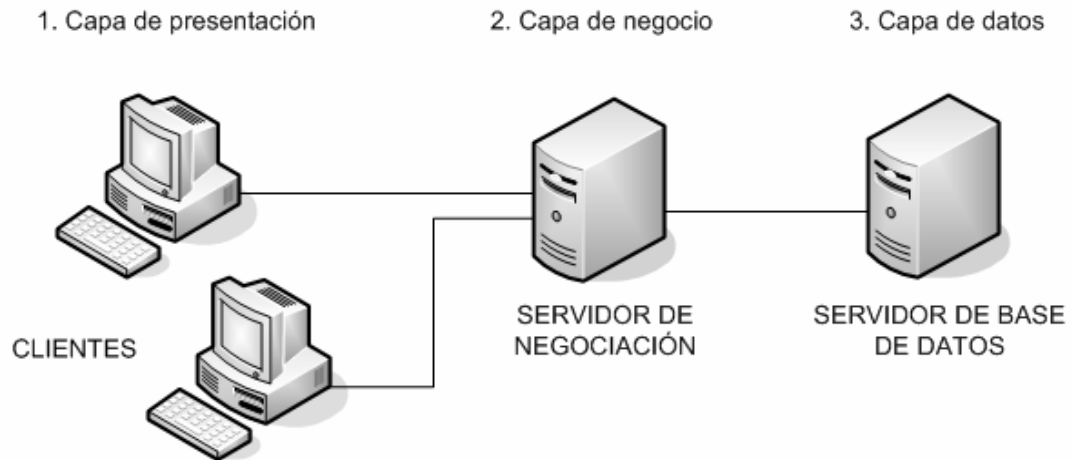
- El servidor no necesita tanta potencia de procesamiento, parte del proceso se reparte con los clientes.
- Se reduce el tráfico de red considerablemente. Idealmente, el cliente se conecta al servidor cuando es estrictamente necesario, obtiene los datos que necesita y cierra la conexión dejando la red libre para otra conexión.

✓ **Arquitectura de tres niveles.**

La programación por capas es un estilo de programación en la que el objetivo primordial es la separación de la lógica de negocios de la lógica de diseño, un ejemplo básico de esto es separar la capa de datos de la capa de presentación al usuario.

La ventaja principal de este estilo, es que el desarrollo se puede llevar a cabo en varios niveles y en caso de algún cambio sólo se ataca al nivel requerido sin tener que revisar entre código mezclado.

Figura 2: Arquitectura de Tres Niveles



Además permite distribuir el trabajo de creación de una aplicación por niveles, de este modo, cada grupo de trabajo está totalmente abstraído del resto de niveles, simplemente es necesario conocer la API (Interfaz de Programación de Aplicaciones) que existe entre niveles.

En el diseño de sistemas informáticos actual se suele usar las arquitecturas multinivel o Programación por capas. En dichas arquitecturas a cada nivel se le confía una misión simple, lo que permite el diseño de arquitecturas escalables (que pueden ampliarse con facilidad en caso de que las necesidades aumenten).

El diseño más en boga actualmente es el diseño en tres niveles (o en tres capas).

➤ **Capas o niveles.**

- *Capa de presentación:* es la que ve el usuario, presenta el sistema al usuario, le comunica la información y captura la información del usuario dando un mínimo de proceso (realiza un filtrado previo para comprobar que no hay errores de formato). Esta capa se comunica únicamente con la capa de negocio.

- *Capa de negocio:* es donde residen los programas que se ejecutan, recibiendo las peticiones del usuario y enviando las respuestas tras el proceso. Se denomina capa de negocio (e incluso de lógica del negocio) pues es aquí donde se establecen todas las reglas que deben cumplirse. Esta capa se comunica con la capa de presentación, para recibir las solicitudes y presentar los resultados, y con la capa de datos, para solicitar al gestor de base de datos para almacenar o recuperar datos de él.
- *Capa de datos:* es donde residen los datos. Está formada por uno o más gestores de bases de datos que realizan todo el almacenamiento de datos, reciben solicitudes de almacenamiento o recuperación de información desde la capa de negocio.

Todas estas capas pueden residir en un único computador (no sería lo normal), si bien lo más usual es que haya una multitud de computadores donde reside la capa de presentación (son los clientes de la arquitectura cliente/servidor). Las capas de negocio y de datos pueden residir en el mismo ordenador, y si el crecimiento de las necesidades lo aconseja se pueden separar en dos o mas computadores. Así, si el tamaño o complejidad de la base de datos aumenta, se puede separar en varios computadores los cuales recibirán las peticiones del ordenador en que resida la capa de negocio.

Si por el contrario fuese la complejidad en la capa de negocio lo que obligase a la separación, esta capa de negocio podría residir en uno o más computadores que realizarían solicitudes a una única base de datos. En sistemas muy complejos se llega a tener una serie de computadores sobre los cuales corre la capa de datos, y otra serie de computadores sobre los cuales corre la base de datos.

En una arquitectura de tres niveles, los términos "capas" y "niveles" no significan lo mismo ni son similares. El término "capa" hace referencia a la forma como una solución es segmentada desde el punto de vista lógico: Presentación/ Lógica de Negocio/ Datos.

En cambio, el término "nivel", corresponde a la forma en que las capas lógicas se encuentran distribuidas de forma física. Por ejemplo:

- Una solución de tres capas (presentación, lógica, datos) que residen en un solo computador (Presentación + lógica + datos). Se dice, que la arquitectura de la solución es de tres capas y *un nivel*.
- Una solución de tres capas (presentación, lógica, datos) que residen en dos computadores (presentación + lógica, lógica + datos). Se dice que la arquitectura de la solución es de tres capas y *dos niveles*.

- Una solución de tres capas (presentación, lógica, datos) que residen en tres computadores (presentación, lógica, datos). La arquitectura que la define es: solución de tres capas y *tres niveles*.

2.2. SOLUCIONES BASADAS EN TECNOLOGIA INTERNET

En la actualidad, la Internet se ha transformado en el medio por excelencia para la difusión de la mayor parte de la información mundial, esto la ha convertido en una herramienta prácticamente prioritaria e imprescindible para el desarrollo de un gran porcentaje de las actividades educativas, mercantiles, financieras y además de otros tipos de actividades que se realizan actualmente, esto es debido a que ha facilitado la comunicación entre las entidades participantes en este proceso.

Por lo tanto se ha hecho necesario el desarrollo de soluciones basadas en tecnologías Internet, que nos permitan avanzar y así estar a la vanguardia en todo lo relacionado con las tecnologías de información.

A continuación definiremos algunos de los conceptos más usados en el mundo de la Internet y la Web, para tener una idea más clara.

2.2.1. Internet.

Es una red mundial de computadores interconectada con un conjunto de protocolos, el más destacado, el TCP/IP y aparece por primera vez en 1960. También se usa este nombre para designar a cualquier red de redes que use las mismas tecnologías que Internet, independientemente de su extensión o de que sea pública o privada. Cuando se dice red de redes se hace referencia a que es una red formada por la interconexión de otras redes menores.

Al contrario de lo que se piensa comúnmente, Internet no es sinónimo de World Wide Web. Ésta es parte de aquella, siendo la World Wide Web uno de los muchos servicios ofertados en la red Internet. La Web es un sistema de información mucho más reciente que emplea Internet como medio de transmisión.

Algunos de los servicios disponibles en Internet aparte de la Web son el acceso remoto a otras máquinas (SSH y telnet), transferencia de archivos (FTP), correo electrónico (SMTP), boletines electrónicos (news o grupos de noticias), conversaciones en línea (IRC y chats), mensajería instantánea, transmisión de archivos (P2P, P2M, Descarga Directa), etc.

2.2.2. Intranet.

Es una red de computadoras de una red LAN⁹ privada, empresarial o educativa que proporciona herramientas de Internet, las cuales tienen como función principal proveer lógica de negocios para aplicaciones de captura, reportes, consultas, etc. con el fin de auxiliar la producción de dichos grupos de trabajo; es también un importante medio de difusión de información interna a nivel de grupo de trabajo.

No necesariamente proporciona Internet a la organización; normalmente, tiene como base el protocolo TCP/IP de Internet y, por ser privada, puede emplear mecanismos de restricción de acceso a nivel de programación como lo son usuarios y contraseñas de acceso o incluso a nivel de hardware como un sistema cortafuegos (Firewall) que pueda restringir el acceso a la red organizacional.

La Intranet fue creada para mayor seguridad para poder compartir archivos, carpetas y recursos. Es una excelente opción de bajo costo para las empresas.

2.2.3. Extranet (Extended Intranet).

Es una red privada virtual (VPN¹⁰) resultante de la interconexión de dos o más intranets que utiliza Internet como medio de transporte de la información entre sus nodos.

Es una red que tiene acceso limitado y que esta disponible únicamente a usuarios específicos, tales como clientes o proveedores.

✓ Aplicaciones.

- Transacciones seguras de actividades entre los sistemas internos de los negocios.
- Permite reducción en costos y ahorro de tiempo para la empresa.
- Facilita rápido acceso a información de socios (clientes y proveedores).
- Proporciona servicios de red y desarrollo organizacional.
- Integración de los sistemas.

⁹ LAN significa: Local Area Network o Red de Área Local.

¹⁰ VPN es: una tecnología de red que permite una extensión de la red local sobre una red pública o no controlada, como por ejemplo Internet.

2.2.4. El Protocolo TCP/IP.

Desde sus comienzos, en la Red coexisten computadoras de muy diverso tipo, por lo que se hizo necesario un protocolo común y único, de forma que todas pudieran entender e interpretar correctamente la información que circula. Este protocolo se denominó TCP/IP.

En realidad son dos acrónimos distintos; TCP son las siglas de "Transmisión Control Protocol", mientras que IP significa "Internet Protocol", y fueron propuestos originariamente por ARPA (Defensa Americana) como estándar de comunicaciones para intercomunicar las pocas redes existentes en aquellos momentos.

En la actualidad el concepto TCP/IP engloba toda una filosofía de operación, basada por supuesto en la conmutación de paquetes. Pero más que de dos protocolos, se trata de todo un conjunto que constituyen la base del funcionamiento de Internet. Concretamente, IP es un estándar que subyace en todas las comunicaciones de la red.

Incluye todas las especificaciones necesarias para hacer inteligible a cualquier máquina la información contenida en cada datagrama (paquete) transmitido. Entre otras, el tamaño normalizado de las cabeceras; remite; códigos de control; de integridad, etc.

Uno de sus elementos más destacados lo constituye un sistema universal y unificado para establecer las "Direcciones" de los ordenadores de la red. A esto se le denomina Dirección IP ("Internet Protocol Address").

Así pues, en principio IP es el encargado de la transmisión de los datos; que sea posible su tránsito de un ordenador a otro. Mientras que TCP es el encargado de juntar los paquetes, pedir los que faltan (en su caso) y finalmente ordenarlos, puesto que, la Red no garantiza la llegada de todos los paquetes ni tampoco que su llegada sea en orden.

2.3. APLICACIONES WEB

Una aplicación Web es aquella que los usuarios usan accediendo a un servidor Web a través de Internet o de una intranet. Las aplicaciones Web son populares debido a la practicidad del navegador Web como cliente ligero. La habilidad para actualizar y mantener aplicaciones Web sin distribuir e instalar software en miles de potenciales clientes es otra razón de su popularidad.

2.3.1. Servidor web.

Un servidor Web es un programa que implementa el protocolo HTTP (Hypertext Transfer Protocol). Este protocolo está diseñado para transferir lo que llamamos hipertextos, páginas Web o páginas HTML (Hypertext Markup Language): textos complejos con enlaces, figuras, formularios, botones y objetos incrustados como animaciones o reproductores de sonidos.

2.3.2. El protocolo http.

El protocolo de transferencia de hipertexto (HTTP) es el protocolo usado en cada transacción de la Web. El hipertexto es el contenido de las páginas Web, y el protocolo de transferencia es el sistema mediante el cual se envían las peticiones de acceder a una página Web, y la respuesta de esa Web, remitiendo la información que se verá en pantalla. También sirve el protocolo para enviar información adicional en ambos sentidos, como formularios con mensajes y otros similares.

HTTP es un protocolo sin estado, es decir, que no guarda ninguna información sobre conexiones anteriores. Al finalizar la transacción todos los datos se pierden. Por esto se popularizaron las cookies, que son pequeños ficheros guardados en el propio ordenador que puede leer un sitio Web al establecer conexión con él, y de esta forma reconocer a un visitante que ya estuvo en ese sitio anteriormente. Gracias a esta identificación, el sitio Web puede almacenar gran número de información sobre cada visitante, ofreciéndole así un mejor servicio.

2.3.3. Funcionamiento de una aplicación web.

La aplicación Web está compuesta de un grupo de páginas Web, ya sean estáticas o dinámicas, las cuales son ejecutadas por los servidores Web dando como respuesta datos estáticos (páginas HTML e imágenes) a los navegadores Web (Clientes Web). Un servidor Web recibe la petición de una página Web, identificada por su URL¹¹. Esta URL identifica un fichero local situado en cualquier lugar del sistema de ficheros del servidor. En caso de ser encontrado el fichero, se carga desde donde se encuentra y se envía al navegador Web a través de la red que se esté usando. Este intercambio de datos se lleva a cabo gracias al protocolo HTTP.

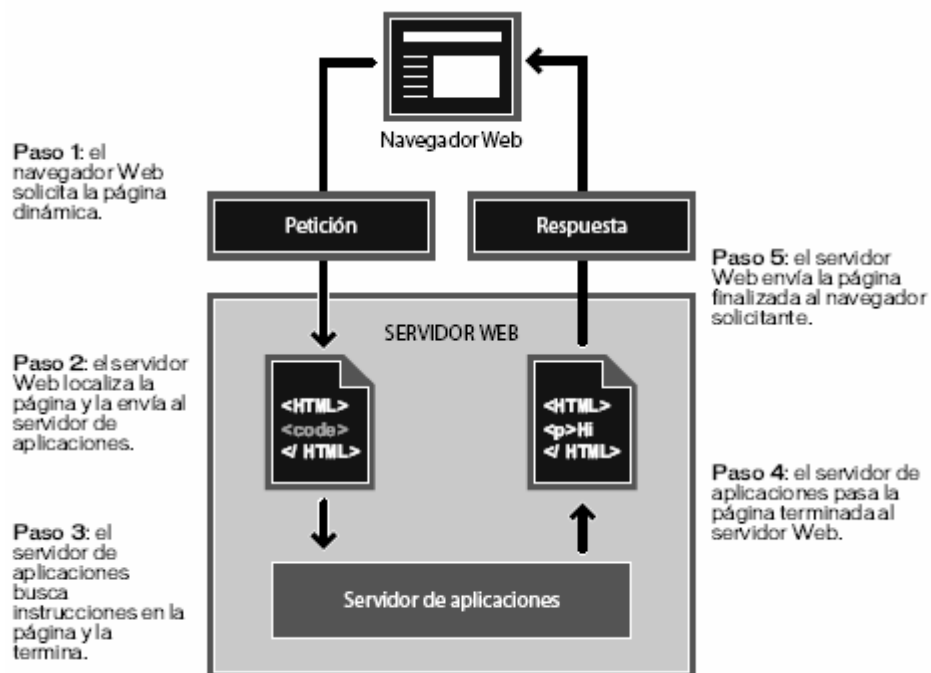
Uno de los avances más importantes en el ámbito de la tecnología Web es la aparición de páginas Web Dinámicas, las cuales son creadas en función de una

¹¹ El URL es la cadena de caracteres con la cual se asigna una dirección única a cada uno de los recursos de información disponibles en la Internet. Existe un URL único para cada página de cada uno de los documentos de la World Wide Web, la cual permite que el navegador la encuentre y la muestre de forma adecuada.

petición del usuario, ya sea de forma directa o indirecta. La forma más utilizada para desarrollar estas páginas es mediante la utilización de CGI¹² (Common Gateway Interface), dicho estándar define cómo el servidor debe ejecutar las páginas locales y enviar las salidas generadas a través de un servidor Web hacia el cliente Web.

Tras todo esto, el usuario no tiene por qué enterarse que las páginas eran dinámicas (en el lado del servidor), por que el CGI consiste básicamente en una extensión del protocolo del servidor Web.

Figura 3: Funcionamiento de una Aplicación Web



2.3.4. Tecnologías para el desarrollo web.

Para desarrollar las aplicaciones Web podemos trabajar de dos formas: Programación del Lado Cliente y Programación del Lado Servidor, ambas muy utilizadas actualmente, veamos en que consisten:

¹² CGI especifica un estándar para transferir datos entre el cliente y el programa. Es un mecanismo de comunicación entre el servidor Web y una aplicación externa.

✓ **Programación en el Lado Cliente.**

La ejecución de las páginas Web desde el lado del cliente se realiza de la siguiente forma:

- 1.** El navegador envía una solicitud.
- 2.** El servidor envía una respuesta que contiene código que el navegador entiende.
- 3.** El navegador interpreta el código enviado por el servidor y realiza una determinada acción.

La principal ventaja es que la ejecución de la aplicación se delega al cliente, con lo cual se evita recargar el servidor de trabajo. Además se hace un mejor aprovechamiento del canal de comunicación, puesto que se minimiza el tráfico de datos entre el cliente y el servidor al evitar continuos pasos de información.

Una desventaja de ésta metodología es que el código que el servidor envía es "sensible" a qué cosas puede o no hacer el navegador, es decir, la ejecución de las páginas dependen de la tecnología, el navegador y la plataforma existente en el lado del cliente.

A continuación se presenta un resumen de los principales lenguajes para la programación de las aplicaciones Web en el lado del cliente:

- **HTML (Hipertext Markup Language)**
Lenguaje que se utiliza para la creación de páginas Web. Consta de un conjunto de elementos, denominados etiquetas o marcas, que se utiliza para incluir textos, dibujos cabeceras, etc.
- **JavaScript**
Lenguaje interpretado incrustado en las páginas Web para permitir ejecutar código en el cliente a través del manejo de objetos y la captura de eventos producidos en el navegador.
- **DHTML**
Extensiones de HTML que cubre el uso de hojas de estilo en cascada (CSS), uso de layers o capas para el posicionamiento y visibilidad de elementos en el navegador, programación con lenguajes de scripts y modelado de objetos de documentos (DOM).
- **VBScript**
Lenguaje interpretado similar a JavaScript proporcionado por Microsoft.

- **Applets de Java**

Pequeña aplicación accesible en un servidor Internet, que se transporta por la red, se instala automáticamente y se ejecuta como parte de un documento Web.

- **ActiveX**

Tecnología de Microsoft que permite ejecutar un programa que siga este estándar en los equipos de los usuarios al ser vinculado como objeto en una página Web. Los programas ActiveX dependen del sistema operativo Windows, por lo tanto no funcionan fuera de este ambiente.

En este proyecto serán utilizados los dos primeros lenguajes para la programación en el lado del cliente.

✓ **Programación en el Lado Servidor.**

La ejecución de las páginas Web desde el lado del servidor se realiza de la siguiente manera:

1. El navegador envía una solicitud.
2. El servidor ejecuta una aplicación que realiza una determinada acción.
3. El servidor envía el resultado de dicha aplicación al cliente.
4. El navegador muestra el resultado recibido del servidor.

Se tiene como gran ventaja que cualquier cosa puede hacerse sin depender del tipo de cliente, ya que la aplicación se ejecuta en un ambiente bajo control.

La desventaja se encuentra en la sobrecarga de trabajo a la que se ve abocado el servidor Web, ya que además de servir páginas, es el responsable de ejecutar las aplicaciones. Como consecuencia los requisitos hardware son mayores a medida que el servidor debe soportar más y más servicios.

Para el desarrollo de aplicaciones del lado del servidor existen tres (3) metodologías a utilizar: El protocolo CGI, La API provista por el servidor Web y un "módulo" del servidor Web, las cuales veremos a continuación.

- **El protocolo CGI (Common Gateway Interface).**

Es un protocolo estándar de comunicación entre un servidor Web y una aplicación, de tal modo que esta última pueda interactuar con Internet.

Un CGI es un programa que se ejecuta en tiempo real en un servidor Web en respuesta a una solicitud de un buscador o navegador. Cuando esto ocurre el servidor Web ejecuta un proceso hijo que recibirá los datos que envía el usuario (en caso de que existan), pone a disposición del mismo algunos datos en forma de variables de ambiente y captura la salida del programa para enviarla como respuesta al navegador.

La desventaja de este tipo de aplicaciones consiste en que el servidor debe realizar un proceso y ejecutar la aplicación en un ciclo que se cumple cada vez que se ejecuta la aplicación CGI lo que consume muchos recursos y en general es costoso en tiempo para el servidor Web.

- **API (Application Programming Interface).**

La API es provista por el propio servidor Web para desarrollar aplicaciones, es decir, que el servidor provee un lenguaje en el cual se pueden desarrollar aplicaciones o programas. Es mucho más eficiente ya que el servidor Web es el encargado de ejecutar las aplicaciones de forma directa sin necesidad de crear un proceso hijo.

Las desventajas son, sin embargo, importantes: Primero, las aplicaciones no son portables ya que solo pueden ejecutarse en un servidor Web determinado. Segundo, un error de programación de una aplicación podría ocasionar que el servidor deje de funcionar, genere un error, pierda memoria u otros problemas; esto ocasiona que este tipo de aplicación no sea confiable.

- **Módulo del servidor web.**

Es la tecnología más reciente para la ejecución de aplicaciones, consiste en anexar a un servidor Web "módulos" que le permitan interpretar un determinado lenguaje. De ésta forma se logra eficiencia ya que el servidor no necesita crear un nuevo proceso por cada aplicación que ejecuta.

Las aplicaciones son portables ya que son desarrolladas en un lenguaje estándar que no depende del servidor Web. Las aplicaciones son confiables ya que si bien puede producir un error en el lenguaje en que están diseñadas, si el módulo es sólido, dichos errores no pueden comprometer al servidor.

A continuación presentamos un resumen de los principales lenguajes con una breve descripción de su uso en programación del lado del servidor.

- **PERL (Practical Extraction and Report Language)**
Lenguaje interpretado creado con el objeto principal de simplificar las tareas de administración de un sistema UNIX. Hoy en día se ha convertido en un lenguaje de propósito general.
- **Python**
Es un lenguaje interpretado que permite escribir programas pequeños, utilizado en desarrollo Web para la creación de CGI.
- **C, C++**
Utilizado para la creación de CGI.
- **PHP (Hypertext Preprocesor)**
PHP es un lenguaje interpretado diseñado para el desarrollo de sitios dinámicos. La distribución más popular de PHP es como módulo para el servidor Apache, aunque puede funcionar como un interprete para ejecutar aplicaciones CGI.
- **ASP (Active Server Pages)**
Tecnología creada por Microsoft destinada a la creación de sitios Web.
- **JSP (Java Server Pages)**
Es un lenguaje interpretado insertado en páginas Web y basadas en Java para el desarrollo de sitios dinámicos.

Para el desarrollo de éste proyecto de grado fue escogido como lenguaje de desarrollo PHP, debido a su alta confiabilidad, fácil aprendizaje, tiempos cortos en el desarrollo y gran eficiencia en aplicaciones Web, por tanto a continuación realizamos una corta introducción a este lenguaje de programación.

2.4. TECNOLOGÍAS USADAS

2.4.1. PHP (Hypertext Preprocesor).

PHP es un lenguaje creado por una gran comunidad de personas. El sistema fue desarrollado originalmente en el año 1994 por Rasmus Lerdorf como un CGI escrito en C que permitía la interpretación de un número limitado de comandos. El sistema fue denominado Personal Home Page Tools y adquirió relativo éxito gracias a que otras personas pidieron a Rasmus que les permitiese utilizar sus programas en sus propias páginas. Dada la aceptación del primer PHP y de manera adicional, su creador diseñó un sistema para procesar formularios al que le atribuyó el nombre de FI (Form Interpreter) y el

conjunto de estas dos herramientas, sería la primera versión compacta del lenguaje: PHP/FI¹³.

La siguiente gran contribución al lenguaje se realizó a mediados de 1997 cuando se volvió a programar el analizador sintáctico, se incluyeron nuevas funcionalidades como el soporte a nuevos protocolos de Internet y el soporte a la gran mayoría de las bases de datos comerciales. Todas estas mejoras sentaron las bases de PHP versión 3. PHP, en su versión 4, utiliza el motor Zend, desarrollado con mayor meditación para cubrir las necesidades actuales y solucionar algunos inconvenientes de la anterior versión. Algunas mejoras de esta versión son su rapidez gracias a que primero se compila y luego se ejecuta, mientras que antes se ejecutaba mientras se interpretaba el código-, su mayor independencia del servidor web -creando versiones de PHP nativas para más plataformas- y un API más elaborado y con más funciones.

Es un lenguaje embebido en páginas HTML que se ejecutan en el servidor. Productos similares y propietarios son Active Server Pages (ASP) de Microsoft, ColdFusion de Allaire y Java Server Pages (JSP) de Sun.

PHP es fácil de aprender comparado con otros mecanismos para obtener la misma funcionalidad. A diferencia de JSP o CGI basados en C, PHP no requiere un conocimiento exhaustivo del lenguaje de programación. A diferencia de Perl, PHP tiene una sintaxis muy fácil de comprender y a diferencia de ASP, no requiere conocer más de un lenguaje de programación o de la instalación de módulos externos o comerciales para realizar tareas más complicadas no previstas en el lenguaje más usado (Visual Basic Script).

Figura 4: PHP



La mayoría de las funciones más útiles están predefinidas en PHP:

¹³ Informe de Miguel Ángel Álvarez Director desarrolloweb.com

Acceso a bases de datos: ODBC, Oracle, Postgres, SQL Server, MySQL, Informix, Interbase, SyBase, Msq, dBase entre otros.
Conectividad: http, FTP, COM, YPNIS, SNMP, Sockets, CORBA, LDAP.
Servicios Correo y Noticias: POP, IMAP, SMTP, NNTP.
Manejo de textos y graficos: XML, HTML, PDF, GD, Flash.
Funciones Matemáticas.
POSIX: semáforos, memoria compartida, acceso a ficheros, expresiones regulares, cronometros, etc.
Comercio Electrónico: Cybercash, Verisign.
Formularios.
Encriptación y Comprensión: MD5, Gzip, Bzip2, OpenSSL, etc.

Con PHP no se encuentra limitado a resultados en HTML. Entre las habilidades de PHP se incluyen: creación de imágenes, archivos PDF y películas Flash (usando libswf y Ming) sobre la marcha. También puede presentar otros resultados, como XHTML y archivos XML. PHP puede autogenerar estos archivos y almacenarlos en el sistema de archivos en vez de presentarlos en la pantalla.

2.4.2. MySQL.

MySQL es un sistema de gestión de base de datos relacional, multihilo y multiusuario con más de seis millones de instalaciones¹. MySQL AB desarrolla MySQL como software libre en un esquema de licenciamiento dual. Por un lado lo ofrece bajo la GNU GPL, pero, empresas que quieran incorporarlo en productos privativos pueden comprar a la empresa una licencia que les permita ese uso.

Al contrario de proyectos como el Apache, donde el software es desarrollado por una comunidad pública, y el copyright del código está en poder del autor individual, MySQL es propiedad y está patrocinado por una empresa privada, que posee el copyright de la mayor parte del código. Esto es lo que posibilita el esquema de licenciamiento anteriormente mencionado. Además de la venta de licencias privativas, la compañía ofrece soporte y servicios. Para sus operaciones contratan trabajadores alrededor del mundo que colaboran vía Internet. MySQL AB fue fundado por David Axmark, Allan Larsson, y Michael Widenius.

✓ Historia del proyecto.

SQL (Lenguaje de Consulta Estructurado) fue comercializado por primera vez en 1981 por IBM, el cual fue presentado a ANSI y desde ese entonces ha sido considerado como un estándar para las bases de datos relacionales. Desde 1986, el estándar SQL ha aparecido en diferentes versiones como por ejemplo: SQL:92, SQL:99, SQL:2003. MySQL es una idea originaria de la empresa

opensource MySQL AB establecida inicialmente en Suecia en 1995 y cuyos fundadores son David Axmark, Allan Larsson, y Michael "Monty" Widenius. El objetivo que persigue esta empresa consiste en que MySQL cumpla el estándar SQL, pero sin sacrificar velocidad, fiabilidad o usabilidad.

Michael Widenius en la década de los 90 trató de usar mSQL para conectar las tablas usando rutinas de bajo nivel ISAM, sin embargo, mSQL no era rápido y flexible para sus necesidades. Esto lo conllevó a crear una API SQL denominada MySQL para bases de datos muy similar a la de mSQL pero más portable. La procedencia del nombre de MySQL no es clara.

2.4.3. Apache web server.

El servidor HTTP Apache es un software (libre) servidor HTTP de código abierto para plataformas Unix (BSD, GNU/Linux, etc.), Windows, Macintosh y otras, que implementa el protocolo HTTP y la noción de sitio virtual. Cuando comenzó su desarrollo en 1995 se basó inicialmente en código del popular NCSA HTTPd 1.3, pero más tarde fue reescrito por completo. Su nombre se debe a que originalmente Apache consistía solamente en un conjunto de parches a aplicar al servidor de NCSA. Era, en inglés, a patchy server (un servidor "parcheado").

El servidor Apache se desarrolla dentro del proyecto HTTP Server (httpd) de la Apache Software Foundation. Apache presenta entre otras características mensajes de error altamente configurables, bases de datos de autenticación y negociado de contenido, pero fue criticado por la falta de una interfaz gráfica que ayude en su configuración.

Apache tiene amplia aceptación en la red: en el 2005, Apache es el servidor HTTP más usado, siendo el servidor HTTP del 70% de los sitios Web en el mundo y creciendo aún su cuota de mercado (estadísticas históricas y de uso diario proporcionadas por Netcraft).

La arquitectura del servidor Apache es muy modular. El servidor consta de una sección core y mucha de la funcionalidad que podría considerarse básica para un servidor Web es provista por módulos. Algunos de estos son:

- mod_ssl - Comunicaciones Seguras vía TLS.
- mod_rewrite - reescritura de direcciones servidas (generalmente utilizado para transformar páginas dinámicas como PHP en páginas estáticas HTML para así engañar a los navegantes o a los motores de búsqueda en cuanto a como fueron desarrolladas estas páginas).
- mod_dav - Soporte del protocolo WebDAV (RFC 2518).
- mod_auth_ldap - Permite autenticar usuarios contra un servidor LDAP.

El servidor de base puede ser extendido con la inclusión de módulos externos entre los cuales se encuentran:

- mod_perl - Páginas dinámicas en Perl.
- mod_php - Páginas dinámicas en PHP.
- mod_python - Páginas dinámicas en Python.
- mod_security - Filtrado a nivel de aplicación, para seguridad.

3. MARCO METODOLÓGICO

Todo proyecto de ingeniería tiene unos fines ligados a la obtención de un producto, proceso o servicio que es necesario generar a través de diversas actividades. Algunas de estas actividades pueden agruparse en fases porque globalmente contribuyen a obtener un producto intermedio, necesario para continuar hacia el producto final y facilitar la gestión del proyecto.

Al conjunto de las fases empleadas se le denomina "ciclo de vida".

Este capítulo se compone de algunas de las diferentes metodologías utilizadas para el desarrollo del software:

- ✓ Lineal
- ✓ DRA (Desarrollo Rápido de Aplicaciones)
- ✓ Prototipos
- ✓ Espiral

3.1. INTRODUCCIÓN

En un sentido amplio se puede entender un proceso como aquel conjunto de actividades que, tomadas en su totalidad, producen un resultado de valor para un cliente. Tomando como centro nuestra área de interés un proceso software deberá especificar:

- ✓ La secuencia de actividades a realizar por el equipo de desarrollo.
- ✓ Productos que deben crearse (qué productos y en qué momento).
- ✓ Asignación de tareas a cada miembro del equipo y al equipo como un todo.
- ✓ Criterios de control.

"La función principal de un modelo de ciclo de vida es establecer el orden en el que se especifica, se realizan los prototipos. Establece los criterios que se utilizan para determinar el paso de una actividad a otra."¹⁴

Sin embargo, la forma de agrupar las actividades, los objetivos de cada fase, los tipos de productos intermedios que se generan, etc. pueden ser muy

¹⁴ McConnell, Steve desarrollo y gestión de proyectos informáticos, Primera edición McGraw-Hill, España
Pág. 146

diferentes dependiendo del tipo de producto o proceso a generar y de las tecnologías empleadas.

La complejidad de las relaciones entre las distintas actividades crece exponencialmente con el tamaño, con lo que rápidamente se haría inabordable si no fuera por la vieja táctica de "divide y vencerás". De esta forma la división de los proyectos en fases sucesivas es un primer paso para la reducción de su complejidad, tratándose de escoger las partes de manera que sus relaciones entre sí sean lo más simples posibles.

Las principales diferencias entre distintos modelos de ciclo de vida están en:

- El **alcance** del ciclo dependiendo de hasta dónde llegue el proyecto correspondiente. Un proyecto puede comprender un simple estudio de viabilidad del desarrollo de un producto, o su desarrollo completo o, llevando la cosa al extremo, toda la historia del producto con su desarrollo, fabricación, y modificaciones posteriores hasta su retirada del mercado.
- Las **características** (contenidos) de las fases en que dividen el ciclo. Esto puede depender del propio tema al que se refiere el proyecto (no son lo mismo las tareas que deben realizarse para proyectar un avión que un puente), o de la organización (interés de reflejar en la división en fases aspectos de la división interna o externa del trabajo).
- La **estructura** de la sucesión de las fases que puede ser lineal, con prototipos, o en espiral. Veámoslo con más detalle:

La selección equivocada de un modelo de ciclo de vida puede ocasionar omisión de tareas o una secuenciación inapropiada de las mismas, lo cual va en contra de la planificación y eficiencia del proyecto. Una buena elección por el contrario, es una garantía de que el esfuerzo se utiliza eficientemente.

3.2. CICLOS DE VIDA DEL DESARROLLO SOFTWARE

3.2.1. Lineal.

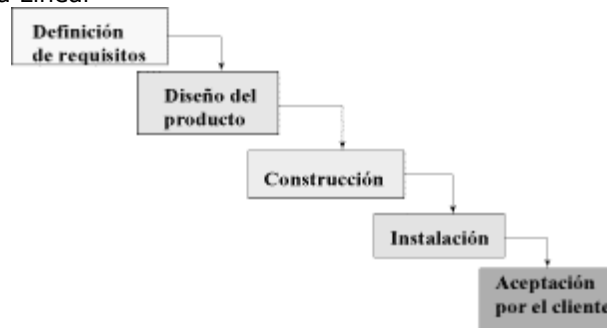
Es el más utilizado, siempre que es posible, precisamente por ser el más sencillo. Consiste en descomponer la actividad global del proyecto en fases que se suceden de manera lineal, es decir, cada una se realiza una sola vez, cada una se realiza tras la anterior y antes que la siguiente.

Con un ciclo lineal es fácil dividir las tareas entre equipos sucesivos, y prever los tiempos (sumando los de cada fase).

Requiere que la actividad del proyecto pueda descomponerse de manera que una fase no necesite resultados de las siguientes (realimentación), aunque pueden admitirse ciertos supuestos de realimentación correctiva.

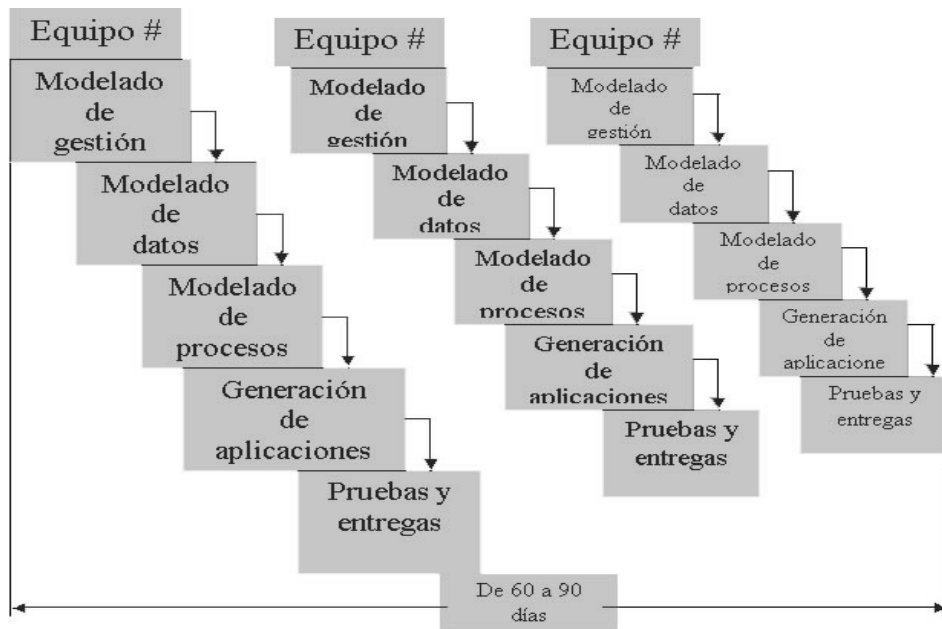
Desde el punto de vista de la gestión (para decisiones de planificación), requiere también que se sepa bien de antemano lo que va a ocurrir en cada fase antes de empezarla.

Figura 5: Ciclo de Vida Lineal



3.2.2. DRA (Desarrollo Rápido de Aplicaciones).

Figura 6: Modelo Desarrollo Rápido de Aplicaciones (DRA)



El modelo DRA es una adaptación a alta velocidad del modelo en cascada en el que se logra el desarrollo rápido utilizando un enfoque de construcción basado en componentes reutilizables y utilizando técnicas de cuarta generación en lugar de software con lenguajes de programación de tercera generación.

En proyectos donde se comprenden bien los requisitos, se limita correctamente el ámbito del proyecto, y el sistema se puede dividir en módulos, el proceso DRA permite al equipo de desarrollo crear un sistema completamente funcional dentro de periodos cortos de tiempo (60-90 días).

3.2.3. Prototipos.

A menudo ocurre en desarrollos de productos con innovaciones importantes, o cuando se prevé la utilización de tecnologías nuevas o poco probadas, que las incertidumbres sobre los resultados realmente alcanzables, o las ignorancias sobre el comportamiento de las tecnologías, impiden iniciar un proyecto lineal con especificaciones cerradas.

Si no se conoce exactamente cómo desarrollar un determinado producto o cuáles son las especificaciones de forma precisa, suele recurrirse a definir especificaciones iniciales para hacer un prototipo, o sea, un producto parcial (no hace falta que contenga funciones que se consideren triviales o suficientemente probadas) y provisional (no se va a fabricar realmente para clientes, por lo que tiene menos restricciones de coste y/o prestaciones). Este tipo de procedimiento es muy utilizado en desarrollo avanzado.

La experiencia del desarrollo del prototipo y su evaluación deben permitir la definición de las especificaciones más completas y seguras para el producto definitivo. A diferencia del modelo lineal, puede decirse que el ciclo de vida con prototipos repite las fases de definición, diseño y construcción dos veces: para el prototipo y para el producto real.

Figura 7: Ciclo de Vida con Prototipos



3.2.4. Espiral.

El ciclo de vida en espiral puede considerarse como una generalización del anterior para los casos en que no basta con una sola evaluación de un prototipo para asegurar la desaparición de incertidumbres y/o ignorancias. El propio producto a lo largo de su desarrollo puede así considerarse como una sucesión de prototipos o miniproyectos que progresan hasta llegar a alcanzar el estado deseado. En cada ciclo (espirales) las especificaciones del producto se van resolviendo paulatinamente.

El cliente evalúa el trabajo de ingeniería y sugiere modificaciones. Sobre la base de los comentarios del cliente se produce la siguiente fase de planificación y de análisis de riesgo. En cada bucle alrededor de la espiral, la culminación del análisis de riesgo resulta en una decisión de "seguir o no seguir". Cada prototipo o miniproyecto se centra en uno o más riesgos importantes hasta que todos estos estén controlados.

El concepto "Riesgo" se define ampliamente en este contexto, y puede referirse a requerimientos y arquitecturas poco comprensibles, problemas de ejecución importantes, problemas con la tecnología subyacente, y demás.

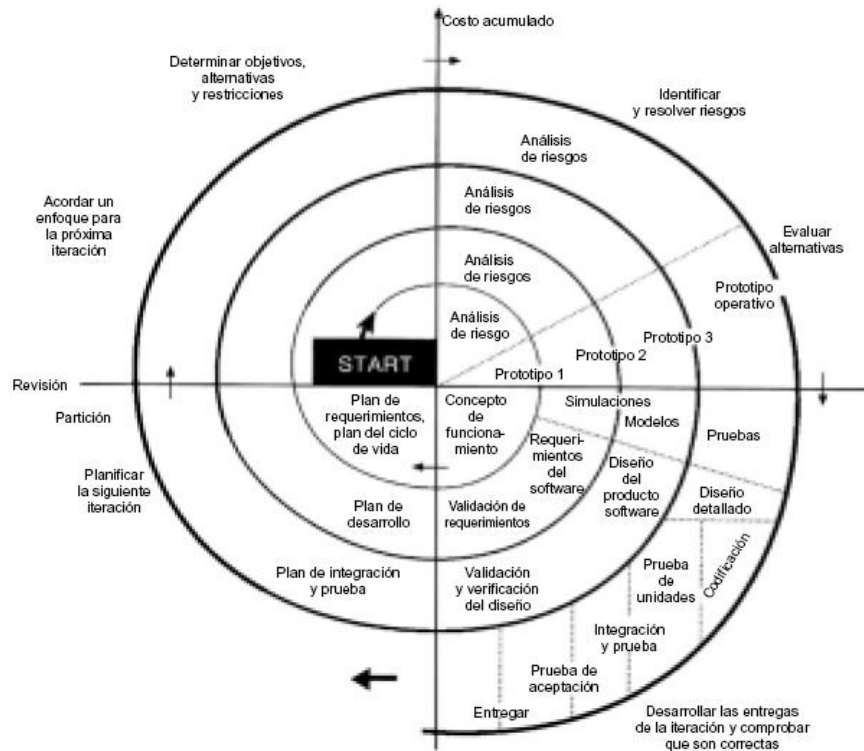
El esquema del ciclo de vida para estos casos puede representarse por un bucle en espiral, donde los cuadrantes son, habitualmente, fases de **especificación, diseño, realización y evaluación** (o conceptos y términos análogos). En cada vuelta el producto gana en "madurez" (aproximación al final deseado) hasta que en una vuelta la evaluación lo apruebe y el bucle pueda abandonarse.

El modelo espiral captura algunos principios básicos:

- Decidir qué problema se quiere resolver antes de viajar a resolverlo.
- Examinar tus múltiples alternativas de acción y elegir una de las más convenientes.
- Evaluar qué tienes hecho y qué tienes que haber aprendido después de hacer algo.
- No ser tan ingenuo para pensar que el sistema que estás construyendo será "EL" sistema que el cliente necesita.
- Conocer (comprender) los niveles de riesgo, que tendrás que tolerar.

El modelo en espiral es conocido cariñosamente como "El rollo de canela en rama".

Figura 8: Ciclo de Vida en Espiral



El paradigma del modelo en espiral para la ingeniería de software es actualmente el enfoque más realista para el desarrollo de software y de sistemas a gran escala. Utiliza un enfoque evolutivo para la ingeniería de software, permitiendo al desarrollador y al cliente entender y reaccionar a los riesgos en cada nivel evolutivo. Utiliza la creación de prototipos como un mecanismo de reducción de riesgo, pero, lo que es más importante permite a quien lo desarrolla aplicar el enfoque de creación de prototipos en cualquier etapa de la evolución de prototipos.

3.2.5. Metodología Escogida.

Al considerar estas posibles metodologías utilizadas en el desarrollo de proyectos informáticos y conociendo las respectivas ventajas y desventajas de sus aplicaciones, nos permite concluir que para la elaboración de este proyecto usaremos la metodología del **Ciclo de Vida en Espiral**.

Las razones por las cuales nos inclinamos por esta metodología se expresan a continuación:

- Al iniciar con este proyecto no había una definición exacta de los requisitos del sistema, se contaba con un conjunto de ideas o especificaciones de lo que se quería hiciera el sistema a realizar, esto abría la posibilidad de hacer modificaciones posteriores una vez iniciado el desarrollo.
- Nos damos cuenta que en el ámbito de la Administración y Organización Deportiva nuestro conocimiento es escaso; nuestro cliente es quien posee los conceptos claros y la experiencia para guiarnos en la construcción del sistema.
- Recurrir al desarrollo de un primer prototipo o miniproyecto, el cual ayudara a especificar los requerimientos iniciales del sistema, así como identificar una cierta cantidad de requisitos pasados por alto inicialmente que se ajusten a las necesidades identificadas, mediante la retroalimentación útil de este prototipo, con los cuales cumplir a cabalidad los objetivos del proyecto.
- Otro factor importante e influyente en esta elección, fue la urgencia de contar con un software adecuado que agilizara los procesos llevados a cabo por el Departamento de Educación Física y Deportes, en el ámbito de la organización de los eventos deportivos al interior de la UIS.

3.3. UML

A mediados de los noventa existían muchos métodos de análisis y diseño orientado a objetos lo que suponía que los mismos conceptos tenían distinta notación según el método de que se tratara. Ante ésta situación de confusión, en 1994 Booch, Rumbaugh y Jacobson decidieron unificar sus métodos dando lugar a *UML*. Esta unificación fue promovida por el OMG de tal manera que *UML* se convirtió en la notación estándar para la descripción de métodos software. Según su definición, *UML* es un lenguaje para visualizar, especificar, construir y documentar los artefactos de un sistema que involucra una gran cantidad de software, desde una perspectiva orientada a objetos.

UML es un lenguaje: proporciona un vocabulario y unas reglas que se centran en la representación conceptual y física de un sistema, y que indican cómo crear y leer modelos bien formados. Sin embargo, no dice qué modelos crear ni cuándo se deberían crear, ésta es la tarea del proceso de desarrollo de software.

UML es un lenguaje para visualizar: es un lenguaje gráfico que mezcla gráficos y texto, pero es algo más que un simple montón de símbolos. De hecho, detrás de cada símbolo en la notación UML hay una semántica bien definida, de manera que un desarrollador puede escribir un modelo en UML, y otro

desarrollador, o incluso otra herramienta, puede interpretar ese modelo sin ambigüedad.

UML es un lenguaje para especificar: cubre la especificación de todas las decisiones de análisis, diseño e implementación que deben realizarse al desarrollar y desplegar un sistema con gran cantidad de software.

UML es un lenguaje para construir: no es un lenguaje visual, pero sus modelos pueden conectarse de forma directa con una gran variedad de lenguajes de programación. Es posible establecer correspondencias desde un modelo *UML* a un lenguaje de programación como Java o C++, o incluso a tablas en una base de datos relacional o al almacenamiento persistente en una base de datos orientada a objetos. Permite ingeniería directa e inversa.

UML es un lenguaje para documentar: cubre toda la documentación de la arquitectura de un sistema y todos sus detalles. También proporciona un lenguaje para expresar requisitos y pruebas del software. Finalmente, *UML* proporciona un lenguaje para modelar las actividades de planificación de proyectos y gestión de versiones.

4. ANALISIS DE REQUERIMIENTOS Y DISEÑO GLOBAL.

En este capítulo se describen los procesos de análisis de requerimientos y de diseño del caso a tratar en este proyecto de grado, el cual a sido denominado "SISTEMA DE INFORMACIÓN PARA LA PLANEACIÓN, ORGANIZACIÓN Y GESTIÓN DE LAS COMPETENCIAS DEPORTIVAS Y DEL HISTORIAL DEPORTIVO DE LOS ESTUDIANTES DE LA UNIVERSIDAD INDUSTRIAL DE SANTANDER".

4.1. RECOLECCIÓN DE REQUERIMIENTOS.

Teniendo en cuenta que para iniciar el desarrollo de este sistema de información se contaba solo con un esbozo de la situación que se quería automatizar, por ello se realizaron una serie de entrevistas informales con el codirector del presente proyecto quien aportó su conocimiento y experiencia en el ámbito de la Organización de Eventos Deportivos al interior de la Universidad, de tal forma que concretáramos la idea inicial.

En la primera reunión discutimos el ¿Qué?, ¿Por qué? y ¿para que?, de la realización del sistema y en las subsecuentes entrevistas nos centramos en temas específicos y vitales para el proyecto tales como: el ingreso de datos, los usuarios de la aplicación, los sistemas de eliminación, la presentación de la información recopilada y procesada, y demás temas que nos proporcionaran una comprensión muy detallada del sistema desarrollado.

Como resultado de estas entrevistas obtuvimos una visión más amplia y clara de la aplicación, y también nos permitió esbozar, definir y concretar los requerimientos de la aplicación a desarrollar.¹⁵

4.2. ESPECIFICACIÓN DEL PROBLEMA

4.2.1. Oportunidad de la institución.

El presente sistema permitirá al DEFD utilizar la tecnología que posee para realizar una de las actividades más importantes y representativas en su quehacer pedagógico (organizar los eventos deportivos-recreativos), esto supone un acceso a los datos, mediante interfaces gráficas, y la actualización permanente de los datos accedidos lo cual es primordial para certificar el historial deportivo.

¹⁵ Ver la Sección 1.1. dedicada a los Objetivos del proyecto.

4.2.2. Sentencia que define el problema.

El problema de:

Sistema de información para la planeación, organización y control de los eventos deportivos y del historial deportivo de los estudiantes de la Universidad Industrial de Santander.

Afecta a las siguientes personas:

- Departamento de Educación Física y Deportes.
- Coordinador de Eventos
- Profesores adscritos al Departamento de Educación Física y Deportes.
- Alumnos Matriculados, Empleados, Administrativos y Docentes.
- Futuros Egresados.

El impacto asociado es:

Recabar la información referente a los eventos, participantes, mixtures y programación y que dicha información esté accesible, actualizada y veraz; proceso difícil de lograr sino se encuentra informatizado.

La solución adecuada sería:

Informatizar el proceso de organizar eventos deportivos, usando una red local con una base de datos accesible desde los distintos nodos de la red y generar interfaces mediante las cuales acceder a la base de datos en mención.

4.2.3. Descripción de Usuarios.

Para proveer de una forma más eficaz y efectiva productos y servicios que se ajusten a las actuales necesidades de los usuarios, es necesario saber identificar e involucrar a todos los participantes en el proyecto como parte del proceso de modelado de requerimientos.

También es necesario identificar a los usuarios del sistema y asegurarse de que el conjunto de participantes en el proyecto los representa adecuadamente.

Esta sección muestra un perfil de los participantes y de los usuarios involucrados en el proyecto. No describe sus requisitos específicos, en lugar de esto proporciona la justificación de por qué estos requisitos son necesarios.

4.2.4. Roles.

Nombre: Coordinador de Eventos

Descripción: Realiza las actividades propias para planear y gestionar las actividades propias de la Organización Deportiva, los recursos humanos y técnicos, y la aplicación correcta y efectiva de los mismos. También toma las decisiones de cualquier índole organizativa, y además se encarga de la administración de cuentas de usuario del sistema.

Nombre: Profesor

Descripción: Desempeña labores de asistente del coordinador, puede encargarse de gestionar algunas actividades de la Organización Deportiva y ser de apoyo a las decisiones que tome la coordinación, así como supervisar las actividades que se llevarán a cabo.

Nombre: Estudiante

Descripción: Es una de las fuentes de información, pero a su vez es el usuario más restringido pues, aunque posee cuenta en el sistema, solo le está permitido inscribirse, ver resultados y, si tiene alguna queja, contactarse con la coordinación. Además puede consultar información no restringida como fixture, programación de encuentros y tablas de posiciones.

4.3. IDENTIFICACIÓN DE PROCESOS

El primer paso en el ámbito del modelado del problema a resolver es la identificación de los posibles procesos que se presentan al interior de esta situación. Obtener los principales procesos del caso seleccionado es una parte crucial en el modelado pues establecerá los límites del mismo.

A partir de los objetivos identificados y de la necesidad existente en el DEFD en el campo de la organización deportiva, se lograron identificar los siguientes procesos:

- Proceso de Inscripción.
- Proceso de Organización.
- Proceso de Resultados.

4.4. PROCESO DE INSCRIPCIÓN

4.4.1. Problema actual.

El DEFD de la UIS presenta problemas en este ítem debido a la falta de un sistema ágil y eficiente para la recolección de la información de las inscripciones por parte de los estudiantes a los diversos eventos deportivos-recreativos organizados por ese departamento de una manera adecuada. Sin duda la abundante información que se maneja y la gran acogida estudiantil por este tipo de actividades, hace que esta deba ser manipulada de una manera ágil.

Actualmente el manejo que se le da a toda esta información ha generado en varias ocasiones inconvenientes pues al momento de hacer la organización y programación del evento no se cuenta con la información completa o inclusive ya comenzado el evento todavía se está receptando o llamando para que se complete. Por estas razones, se plantea a continuación una posible solución, basándonos en el conocimiento y la experiencia de la persona encargada de este proceso en el DEFD.

4.4.2. Términos usados.

Delegado: Es la persona encargada de representar a un equipo participante en los deportes de equipo. En los deportes individuales el jugador podrá hacer las veces de delegado.

Preinscripción: Es el registro que se hace de una persona o un equipo en alguna actividad deportiva, sea individual o de equipo. Posterior a esta y junto con otro requisito, se hará la inscripción.

Validación: Es el paso previo a la inscripción, esto implica la revisión por parte de la organización del cumplimiento de los requisitos preestablecidos.

Inscripción: Es esencialmente la autorización del equipo o la persona para su participación en el deporte elegido.

4.4.3. Descripción del proceso de inscripción.

Para todos los participantes:

Paso 1: El proceso de preinscripción de los participantes se hará vía Web bajo la supervisión del DEFD en los días establecidos, con anterioridad, por este ente.

Paso 2: Para la preinscripción, los interesados deben acercarse al DEFD y solicitar la contraseña de acceso al sistema, esta dependerá de la cantidad de cupos asignados por el departamento para cada deporte. Para disciplinas de equipo se acercará el delegado del equipo a asentar la preinscripción.

Paso 3: Con la contraseña el delegado o participante ingresará al sistema y completará los datos allí pedidos a cabalidad, los cuales son requeridos por el DEFD para autorizar su participación.

Paso 4: Los integrantes del equipo junto con su delegado y los demás participantes deben dirigirse al DEFD para la toma o identificación de su respectiva fotografía, indispensable para el carné.

Paso 5: A cada delegado y participante se le entregará el carné que lo acredita como participante en el evento, este se le entregará el día de la toma de fotos o en otro día establecido por el DEFD.

4.4.4. Recomendaciones o supuestos.

Información Incompleta: aunque la información solicitada correspondiente a los jugadores, delegados y equipos no entra en muchos detalles es necesario que tales datos sean digitados completamente.

Cupos Incompletos: Dependiendo del deporte existe un límite de jugadores a inscribir, se busca que la planilla que presenta el sistema en esta instancia sea llenada hasta el límite permitido para evitar incurrir en otro proceso posterior a la inscripción, aunque el sistema cuenta con esta opción, lo ideal es concienciar a los participantes de no dejar las cosas para lo último.

4.5. PROCESO DE ORGANIZACIÓN

4.5.1. Problema actual.

Debido la gran cantidad de información que la organización de un evento implica, también el tiempo empleado en la elaboración del fixture y su respectiva programación y además la utilización de grandes cantidades de papel, ya que actualmente este proceso de organización de los eventos deportivos se desarrolla manualmente y solo se emplean las herramientas básicas de la informática como Word o Excel para las formalidades del evento, lo que implica dificultades en la transmisión de la información, dado que el organizador debe acudir a sus manuscritos y apoyarse en la hoja de cálculo para el control y publicación de resultados; además la falta de información

oportuna, ha provocado inconvenientes a la hora de tomar decisiones de cualquier índole organizativo, tales como presentar resultados y recopilar las estáticas de las diferentes actividades.

Como sabemos las actuales tecnologías permiten que los usuarios de un sistema tengan la información a su disposición sin la necesidad de acudir a pesados libros, o complejos archivos alfabéticos como se viene manejando aún en el DEFD, además con la tecnología existente en el Departamento es suficiente para el desarrollo de un sistema que solucione estos problemas.

La posible solución, aquí planteada, permitirá que los procesos de organización y gestión de los eventos deportivos recreativos se realicen de manera ágil y eficiente logrando ahorrar tiempo en su ejecución, en el flujo de información y en la recopilación de las estadísticas deportivas. De igual modo, sistematizar y estandarizar la generación del fixture y la programación de los encuentros, haciéndolo de manera transparente a los participantes y a los organizadores del evento.

4.5.2. Términos usados.

Organización: Es la selección de evento se llevará a cabo y que deportes se jugarán durante el semestre académico, así como el número máximo de participantes en cada actividad, los premios y la promoción del campeonato.

Sistemas de Eliminación: Son los diferentes métodos por medio de los cuales jugarán los equipos o personas para obtener un único ganador. Algunos dependen de la cantidad de jugadores y otros del tipo de deporte.

Fixture: Es el primer paso previo a la programación, aquí se hace la distribución de los equipos o personas y se organizan los enfrentamientos, aunque estén registrados, no se identifican ni equipos ni personas, tampoco donde ni a que hora se enfrentarán.

Programación: Es tomar los datos generados por el fixture y, en este caso, aplicar los nombres de equipos y personas, y además asignarles un escenario, una fecha y una hora al enfrentamiento o partido.

4.5.3. Documentos a manejar.

Fixture: Este documento debe ser generado por el sistema para cada deporte programado por la organización, y deberá contener la siguiente información: nombre del evento y deporte a realizar, el cuadro de enfrentamientos y los equipos o participantes en el deporte, según el sistema de eliminación.

Programación: Este documento también debe generarlo el sistema por cada deporte programado, aquí se incluirá los datos del fixture pero con los nombre de equipos y participantes identificados, y como característica de este documento se mostrarán los datos correspondientes al escenario, la fecha y la hora de los encuentros a disputar.

4.5.4. Descripción del proceso de organización.

Paso 1: El DEFD decidirá el evento a organizar y seleccionará que deportes serán los que se realizarán, así como los sistemas de eliminación para cada deporte, también se encargará de la promoción, difusión, divulgación y de estipular los días de inscripción.

Paso 2: Con los cupos para los deportes de equipo e individuales completos, el DEFD proseguirá a organizar los fixture de los diferentes deportes seleccionando de la manera más adecuada, y basados en las experiencias anteriores, los sistemas de eliminación para cada uno de ellos.

Paso 3: Ya definidos los cuadros de enfrentamiento para cada deporte en particular, lo siguiente es organizar la programación de estos, en este paso por cada deporte se debe darle al encuentro un escenario, una fecha y una hora determinada para su disputa. Este paso debe repetirse por cada fase (clasificatoria, eliminatoria, semifinal y final) avanzada por el evento.

Paso 4: Organizados los fixture y la programación, lo siguiente es imprimir ambos documentos, para enviarlos a los diferentes medios de comunicación de la Universidad.

4.5.5. Recomendaciones o supuestos.

Cantidad de Participantes: Para la selección del sistema de eliminación se debe tener en cuenta la cantidad de participantes inscritos para elegir el sistema adecuado de competencia.

Repositorio para Fixture y Programaciones: Para una mejor organización de los documentos generados por el sistema se recomienda crear un lugar donde guardar específicamente los diferentes archivos .pdf generados.

4.6. PROCESO DE RESULTADOS

4.6.1. Problema actual.

El manejo que actualmente da el DEFD a los resultados se limita a archivar, en carpetas, las planillas de juego correspondiente a cada deporte e incluir el reporte del marcador del encuentro en una hoja de cálculo, como Excel, donde cuentan con una serie de fórmulas para publicación de las tablas de posiciones y de los mismos resultados.

En esas carpetas archivadas quedan guardadas las estadísticas de los jugadores, la única temida en cuenta es la de mayor anotador, cualquiera sea el deporte, las demás quedan archivadas. Por otro lado, en base a las planillas de juego se toman decisiones sobre la(s) sanción(es) para los jugadores con reportes de tarjetas por faltas incurridas y señaladas en la planilla.

Esta posible solución, plantea que cuando se reciban las planillas de juego los resultados del encuentro sean incluidos en el sistema, junto con la participación o no del jugador, sus estadísticas y las amonestaciones a las que se hizo acreedor. El software se encargará de organizar la tabla de posiciones y de registrar las estadísticas del participante y en el historial su respectiva participación en el evento.

Además de esto, el sistema le dará al jugador una posible sanción dependiendo de la amonestación recibida por parte del juez del encuentro, esta sanción será avalada por el coordinador y los profesores a cargo.

4.6.2. Términos usados.

Planilla de Juego: Es un documento en el cual se consigna la información del encuentro: los nombres de quienes se enfrentan, el marcador, quienes jugaron, quienes hicieron puntos, a quienes amonestaron, etc. Este papel se entrega a la coordinación apenas terminado el encuentro y firmado por el juez y los delegados.

Marcador: Es el resultado de la confrontación entre equipos o jugadores. La mayoría de veces se da en forma numérica y determina un ganador, un perdedor o un empate, si el deporte lo permite, y una asignación de puntos que depende del deporte realizado.

Aplazado: Es cuando el encuentro a realizarse debe ser suspendido y jugado en un día diferente al establecido, para esto deben existir causales de fuerza mayor para tomar esta decisión.

W.O.: Es cuando uno de los participantes de un encuentro no se presenta en el lugar determinado a la hora establecida, esta falta acarrea unas sanciones al equipo o persona que incurra en ello.

Amonestación: En la mayoría de los deportes estas amonestaciones están sujetas al color de unas tarjetas que tienen en su haber los jueces del encuentro y a unas faltas establecidas por los entes internacionales del respectivo deporte. Estas pueden corresponder a un primer llamado de atención, una expulsión del juego y a las sanciones dispuestas en relación a dicha falta.

Sanción: Consiste en el castigo impuesto al jugador por la falta cometida, depende de lo agravante que sea dicha falta está establecido el castigo, este puede representar solo la expulsión por un partido o más, la suspensión del campeonato o hasta la suspensión de toda participación en cualquier actividad organizada por el DFED.

4.6.3. Documentos a manejar.

Historial Deportivo Estudiantil: Este documento está conformado por la información de las participaciones del estudiante en los eventos organizados por el DFED de la UIS; este documento se le entregará al estudiante cuando concluya su permanencia en la universidad, e incluirá el evento y deportes en que participó y además el tiempo dedicado a esta actividad deportivo-recreativa.

Tabla de Posiciones: El documento es la compilación de todos los resultados que se presenten en los diferentes encuentros del deporte seleccionado, contiene la sumatoria de partidos ganados, perdidos o empatados, puntos anotados y recibidos, la diferencias entre ellos y los puntos otorgados, según sea el caso del deporte jugado. Estará ordenado de la mayor puntuación a la menor y tendrá en cuenta unos ítems de desempate según sea el caso, para determinar la posición.

4.6.4. Descripción del proceso de resultados.

Paso 1: La coordinación del evento recibirá las planillas de juego correspondientes a los encuentros disputados durante el día. El coordinador o el encargado de ingresar al sistema los marcadores de los enfrentamientos, digitará junto con los marcadores, la participación del jugador, sus estadísticas y amonestaciones; luego se dispondrá a archivar las planillas.

Paso 2: Se evaluarán y se avalarán o modificarán las sanciones impuestas por el sistema por parte del coordinador y el comité asignado para tal efecto.

Paso 3: El sistema se encargará de hacer las anotaciones a la tabla de posiciones de los resultados y de la participación del jugador en su respectivo historial deportivo. Es responsabilidad del coordinador imprimir la tabla de posiciones con los resultados del día.

Paso 4: La tabla de posiciones debe ser publicada en la cartelera del DEFD y, de ser considerado conveniente, enviada mediante correo electrónico a los delegados de los equipos y a los demás jugadores, en los diferentes deportes organizados.

4.6.5. Recomendaciones o supuestos.

Programación Aplazados: Tener en cuenta que al momento de asignarle el estado de aplazado al encuentro, su reprogramación se hace de manera inmediata, lo cual hace que se determine el lugar, la fecha y la hora disponible de igual forma.

Estadísticas Jugador: las estadísticas recabadas sobre los jugadores son los datos básicos que se pueden obtener basados en las planillas de juego porque no se puede delegar a alguien que recolecte esos datos en cada encuentro.

4.7. DISEÑO GLOBAL DE CASOS DE USO

La aplicación Web, para el DEFD, contempla la posibilidad de formular responsabilidades a tres (3) tipos de usuario, previamente identificados (Coordinador, Profesor, Estudiante).

Cada uno será caracterizado como un actor en el diagrama de casos de uso; a continuación se describirán con más detalle cada actor y luego se mostrará su respectivo diagrama.

Actor Coordinador

Representa al usuario responsable de planear y gestionar la organización de los eventos deportivos y de asignar claves a otros usuarios.

Responsabilidades

- Crear y organizar del evento a realizar.
- Crear y asignar las claves a los demás usuarios.
- Tomar decisiones de índole organizativa.
- Imprimir el fixture, la programación y los certificados.
- Validar la inscripción de equipos, delegados y participantes.

Necesidades

- Tener la información actualizada y disponible.
- Administrar toda la organización bajo su responsabilidad.
- Imprimir varios documentos.
- Buscar e imprimir certificados.
- Modificar y eliminar equipos, delegados o participantes que no cumplan con los requisitos.

Actor Profesor

Es el responsable de gestionar algunas actividades de la organización deportiva, asiste a la coordinación en la toma de decisiones. También cumple el papel de supervisor de las actividades programadas.

Responsabilidades

- Validar la inscripción de equipos, delegados y participantes.
- Inserción de los marcadores y estadísticas de los encuentros.
- Imprimir el fixture, la programación y los certificados.

Necesidades

- Tener la información actualizada y disponible.
- Recibir y manipular las planillas de juego.
- Buscar e imprimir varios documentos.

Actor Estudiante

Es el responsable de ingresar la información requerida de los participantes sean equipos o individuales; en el caso de los equipos este actor es el delegado del equipo.

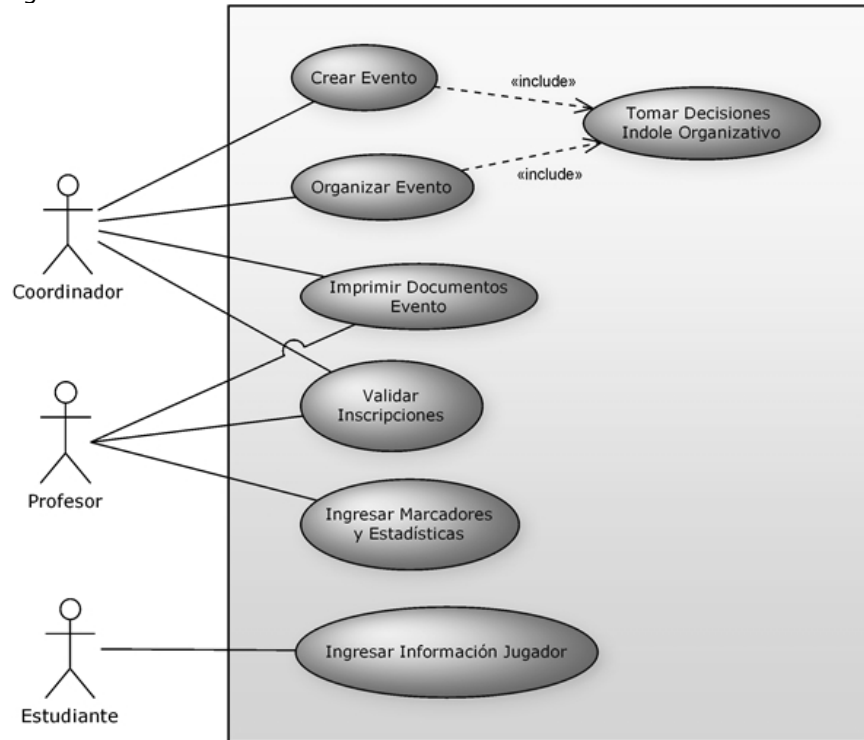
Responsabilidades

- Insertar la información personal del jugador y/o los jugadores.
- Leer los reglamentos del evento y del deporte a participar.

Necesidades

- Tener a su disposición los reglamentos del evento y del deporte en el cual participará.

Figura 9: Diagrama de Casos de Uso



4.8. DISEÑO GLOBAL DEL MODELO DE DATOS

Para comenzar con el desarrollo del proyecto se realizó inicialmente un diseño global del modelo de datos, o base de datos, el cual a medida que se avanzó en el proyecto se fue refinando durante la etapa de diseño detallado hasta llegar finalmente al modelo definitivo.

A continuación presentamos en modelo de datos global planteado, este modelo se encuentra dividido en dos (2) partes:

- El modelo de datos de los deportes de equipo (Figura 10)
- El modelo de datos de los deportes individuales (Figura 11).

Figura 10: Modelo de datos de los Deportes de Equipo.

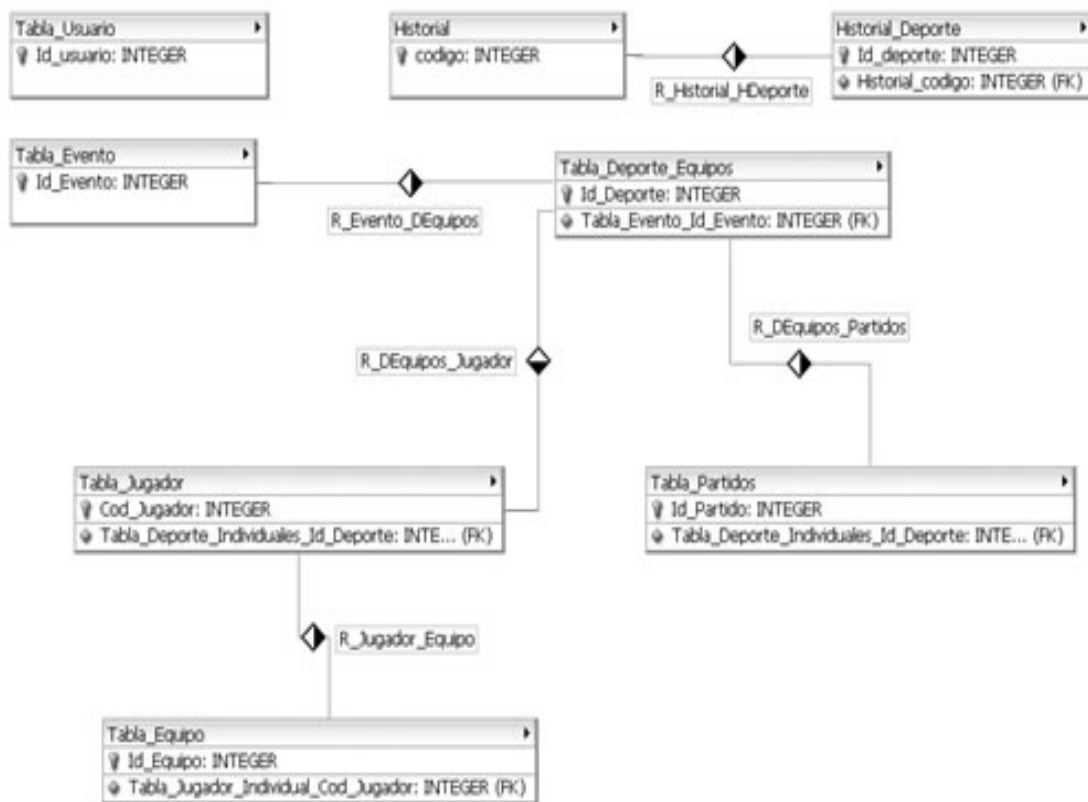
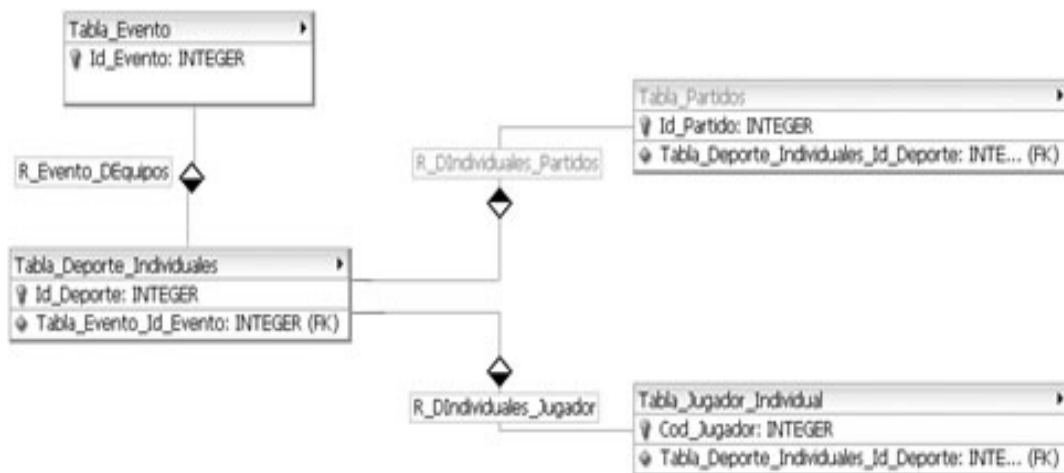


Figura 11: Modelo de datos de los Deportes Individuales.



5. DISEÑO DETALLADO

En este capítulo se describirá como se ha procedido en la realización del sistema planteado. Partiremos desde primer miniproyecto y continuaremos por sus sucesivos, los cuales irán abarcando los procesos identificados a medida que se avanza en su desarrollo. Desarrollamos este trabajo de grado en cuatro (4) miniproyectos.

5.1. PRIMER MINIPROYECTO

En este paso se comenzó con la elaboración de una interfaz básica, más que todo funcional, también en este punto se implementó el diseño del modelo de datos. Los demás desarrollos fueron:

- Selección y generación del evento y sus respectivos deportes.
- Inserción de la información correspondiente a los responsables del evento.
- Inscripción de jugadores en los diferentes deportes generados.
- Generación de carnés de los jugadores registrados.

5.1.1. Selección y generación del evento.

Este primer miniproyecto tuvo como principal objetivo ayudar a la codirección a encontrar e identificar requerimientos pasados por alto al momento de hacer la definición de estos. Por esta razón la interfaz se hizo más funcional que decorativa y además se implementó el modelo de datos para refinarlo en el caso que fuere necesario.

En una de las reuniones sostenidas con la codirección del proyecto se acordó contemplar en el sistema la organización de todos los eventos que se realizan en la UIS, son nueve (9) los cuales son: Olimpiadas, Interroscas, Interfacultades, Intercolonias, Centros de estudio, Carrera Atlética, y Regional, Zonal y Nacional ASCUN¹⁶. Además se determinaron los deportes que se realizan en cada uno de estos eventos, la determinación de estos se basó en los registros históricos existentes en la coordinación de eventos del DEFD¹⁷.

¹⁶ **ASCUN** es la sigla de la Asociación Colombiana de Universidades

¹⁷ **DEFD**, sigla utilizada para identificar al Departamento de Educación Física y Deportes.

Figura 12: Diseño inicial de acceso al sistema.

The image shows a web interface for system access. At the top right, it says "CONSTRUIMOS FUTURO". The main header is "DEPARTAMENTO DE EDUCACIÓN FÍSICA Y DEPORTES". On the right side, there is a logo of the University of Santander (UIS) and the text "Universidad Industrial de Santander". In the center, there are two input fields: "EMAIL USUARIO" and "CONTRASEÑA", followed by an "Entrar" button.

5.1.2. Inserción de los responsables del evento.

Como sabemos toda organización de algún evento de cualquier índole requiere de unos responsables que lideren esta actividad; la organización deportiva no esta exenta de ello, por lo cual en el sistema se contempla la opción de identificar e ingresar en el sistema dichos responsables, en el caso de la UIS serían: Organizador (que es la entidad, la escuela, la división o el departamento que avala la organización), Director y Coordinador del evento, además de los datos de contacto de cada uno de ellos, tales como: el teléfono, el correo electrónico, el teléfono celular, etc.

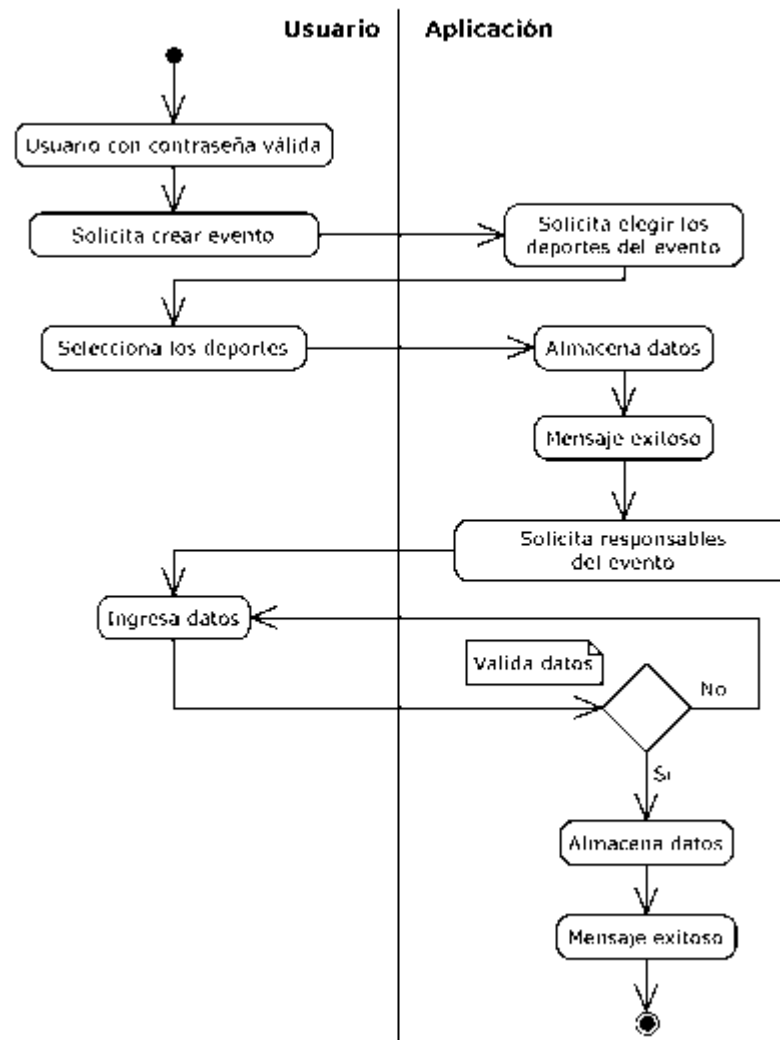
Este paso está ligado a la selección del evento a crear y organizar en el sistema dado que estos responsables pueden variar dependiendo del evento que se seleccione.

5.1.3. Inscripción de jugadores.

En cuanto a las inscripciones a los deportes se ha planteado que este proceso se haga vía Web, aquí tomamos la consideración de no hacer una inscripción directa sino una preinscripción, ya que la participación del estudiante en los deportes está sujeta a la validación del pago de la cuota determinada por la organización para tal efecto. Una vez realizada dicha validación, se autoriza la

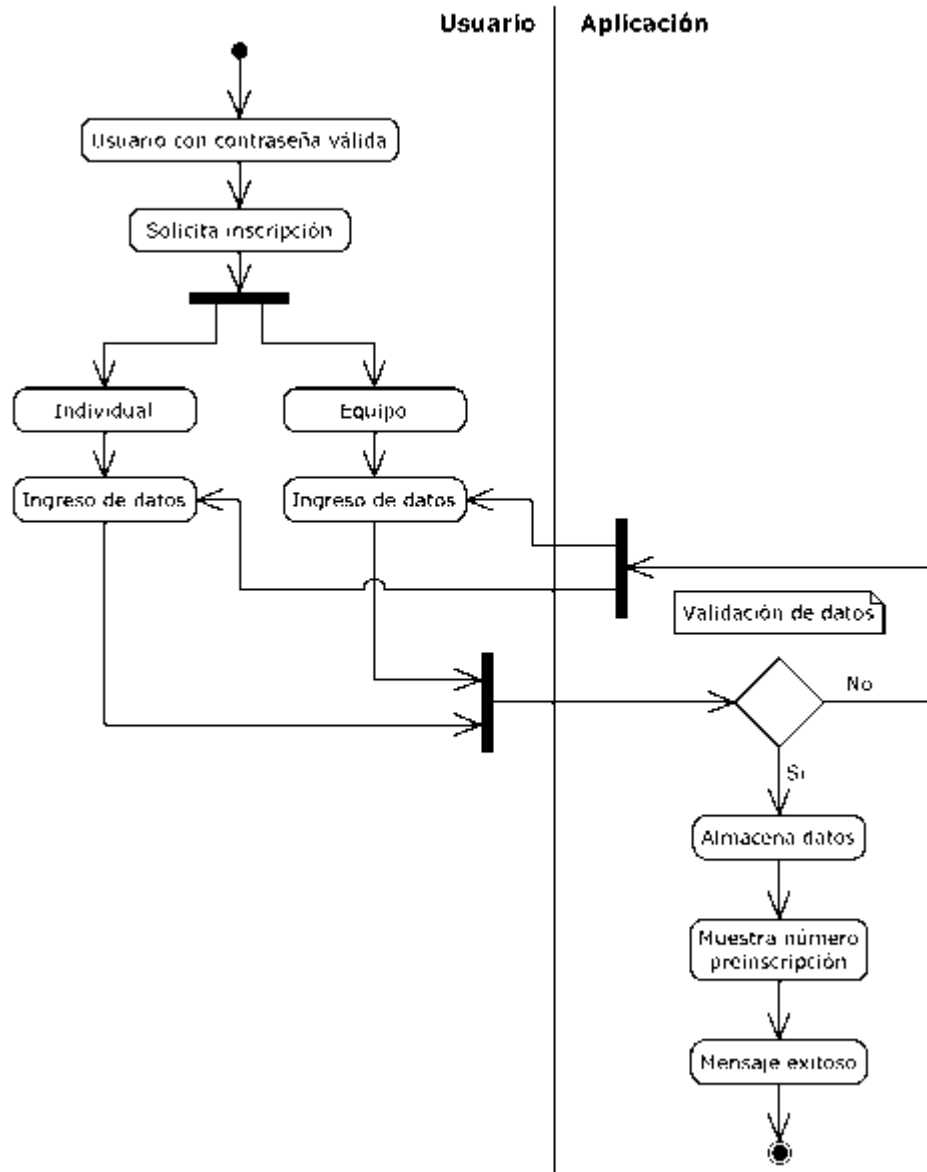
participación y se registra la foto del jugador, y también del delegado en el caso de los equipos.

Figura 13: Crear evento e insertar responsables del evento.



El proceso de inscripción es muy similar entre los deportes de equipo y los deportes individuales; como sabemos el equipo debe ser representado por un delegado, los datos de este delegado son solicitados junto con los datos del equipo, luego se procede mediante una planilla a la inscripción de los jugadores del equipo, el número de jugadores a inscribir está limitado por el deporte escogido. Para los deportes individuales solo se solicitan los datos del jugador, estos incluyen su teléfono y su correo electrónico.

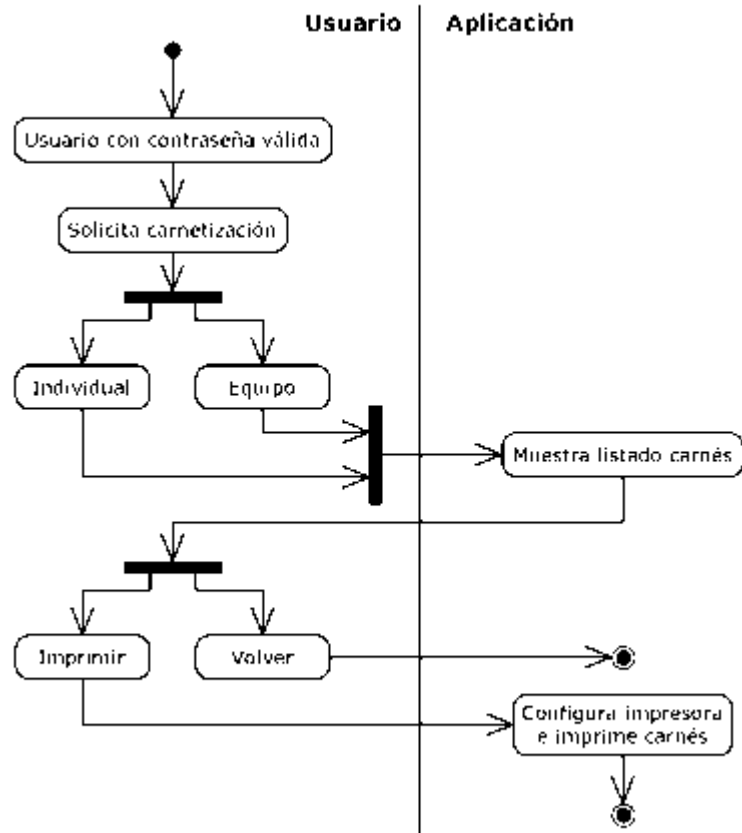
Figura 14: Inscripción de jugadores a los deportes del evento.



5.1.4. Generación de carnés o carnetización.

Una vez hecha la validación y obtenida la foto del jugador se procede a generar el carné correspondiente al jugador, este será el documento de identificación durante su participación en el evento. En el sistema es posible generar el carné por deporte, ya sean deportes individuales, o por equipo, en el caso de los deportes de equipo. Además se da la opción de imprimir los carnés desde la página que los muestra, la visualización e impresión de estos carnés se realiza de dos por fila.

Figura 15: Carnetización.



5.2. SEGUNDO MINIPROYECTO

Durante este miniproyecto se realizaron investigaciones sobre: la generación y validación de contraseñas encriptadas y la forma óptima de enviar información adjunta por correo mediante PHP. Otros desarrollos en este paso fueron:

- Elaboración del fixture para deportes de equipo.
- Acceso al sistema por medio de sesiones y envío de contraseñas por correo electrónico.
- Enviar información referente al evento por correo electrónico a Delegados y Jugadores.
- Modificación de Delegados y Jugadores.
- Validación de las inscripciones.
- Elaboración de la sección de contáctenos.

5.2.1. Elaboración del cuadro de enfrentamiento equipos.

Podríamos decir que este es el corazón de la organización de cualquier evento deportivo, en este punto entran en juego muchas variables que permitirán organizar y programar los diversos deportes del evento; recordemos que existen dos tipos de deportes: los deportes individuales y los deportes de equipo, en este miniproyecto comenzamos por estos últimos.

Para determinar los sistemas de eliminación existente para los deportes de equipo, se investigó en varios libros relacionados con la organización deportiva sobre cuales de estos sistemas se podrían aplicar al software, estos fueron: Eliminación Sencilla, Eliminación por Grupos, Eliminación por Puntos y Eliminación Combinada; para elegir alguno de estos se debe tener en cuenta la cantidad de equipos y el deporte que se irá a organizar.

Otro de los requisitos del sistema implica la impresión para su difusión a los delegados y jugadores involucrados del fixture y la programación de los deportes; en este aspecto nos interesamos en generar estos documentos en formato PDF.

Luego de consultar sobre algún método eficaz y sencillo de manejar optamos por utilizar la clase FPDF¹⁸, la cual es freeware¹⁹ y nos permite generar documentos PDF directamente desde PHP, mediante la utilización de las instrucciones incluidas en esta clase. Su implementación nos permitió alcanzar este requisito lo cual hace más fácil la manera de difundir estos dos documentos a los usuarios.

La elaboración del fixture se realizó de tal manera que el organizador haga el papel de supervisor, pues es el sistema quien, de manera aleatoria; organiza a los participantes, los integrantes de los grupos y hace las llaves de enfrentamientos. Esto se realizó en base a los lineamientos de la generación de fixture investigados y para dar mayor transparencia a este proceso.

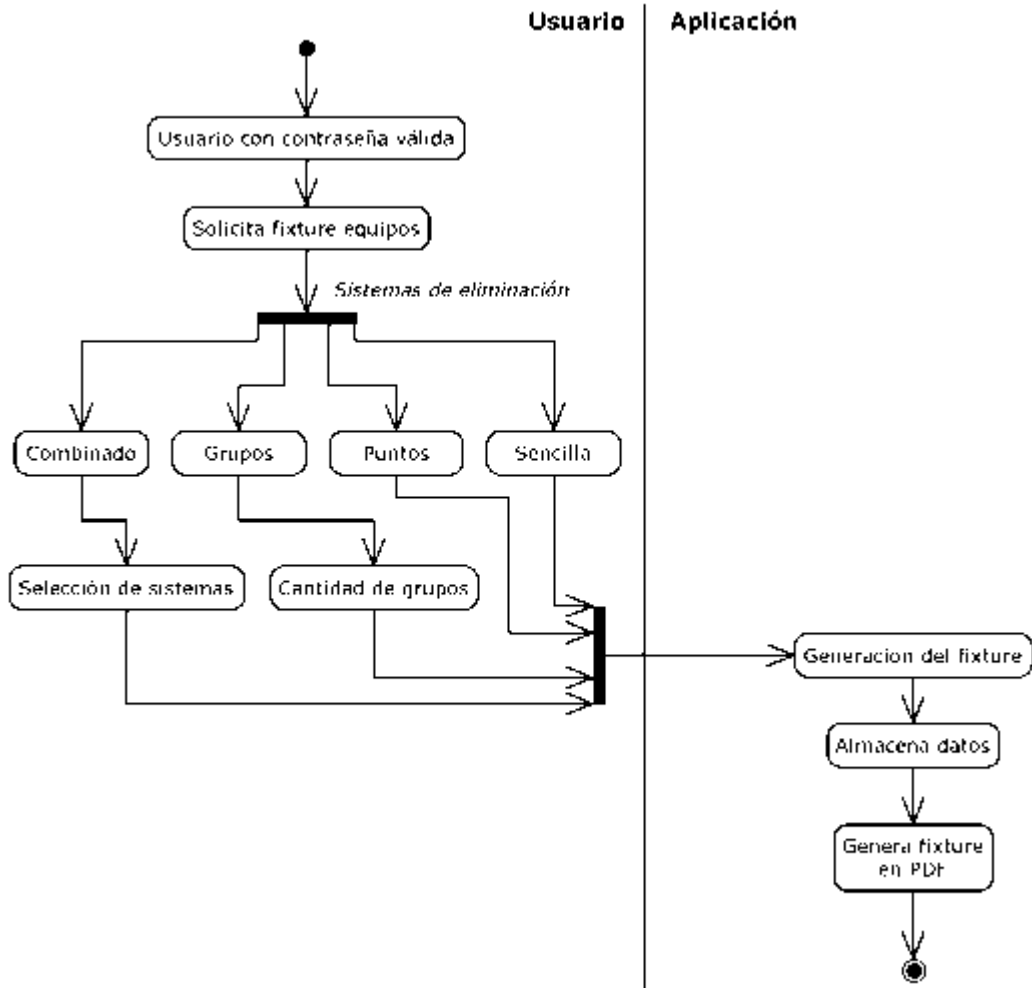
5.2.2. Acceso al sistema por sesiones.

En esta parte se decidió implementar los tipos de usuarios identificados para el sistema, para tal efecto se acordó brindarles acceso al software mediante el uso de contraseñas encriptadas para dar mayor seguridad a la información recopilada y, además, que dicha contraseña fuese enviada por vía Web al correo electrónico de la persona a la cual se le adjudicara.

¹⁸ **FPDF**, del inglés Free PDF que significa: gratis PDF.

¹⁹ **Freeware** es un software de computadora que se distribuye sin cargo.

Figura 16: Elaboración del fixture para deporte de equipo.



Para realizar estas dos actividades se realizó una búsqueda sobre las mejores formas de hacerlas; en el caso de las contraseñas se optó por emplear la generación automática usando solo letras, mayúsculas y minúsculas, y números; también se utilizó un algoritmo de encriptación, en este caso el MD5²⁰, para agregar seguridad a las contraseñas generadas.

Para hacer el envío de estas contraseñas vía Web, se realizó una extensa investigación y el resultado fue optar por la implementación de la clase PHPMailer de sourceforge.net²¹, la cual nos permite de una forma sencilla

²⁰ **MD5**, del inglés Message-Digest Algorithm 5 que significa: Algoritmo de Resumen del Mensaje 5. Es un algoritmo de reducción criptográfico de 128 bits ampliamente usado. Es rápido, fácil y potente, de hecho su gran ventaja es la facilidad y rapidez de uso.

²¹ Mayor información en <http://phpmailer.sourceforge.net/>

realizar tareas complejas como por ejemplo: enviar datos adjuntos y mensajes en formato html.

Figura 17: Crear contraseñas.

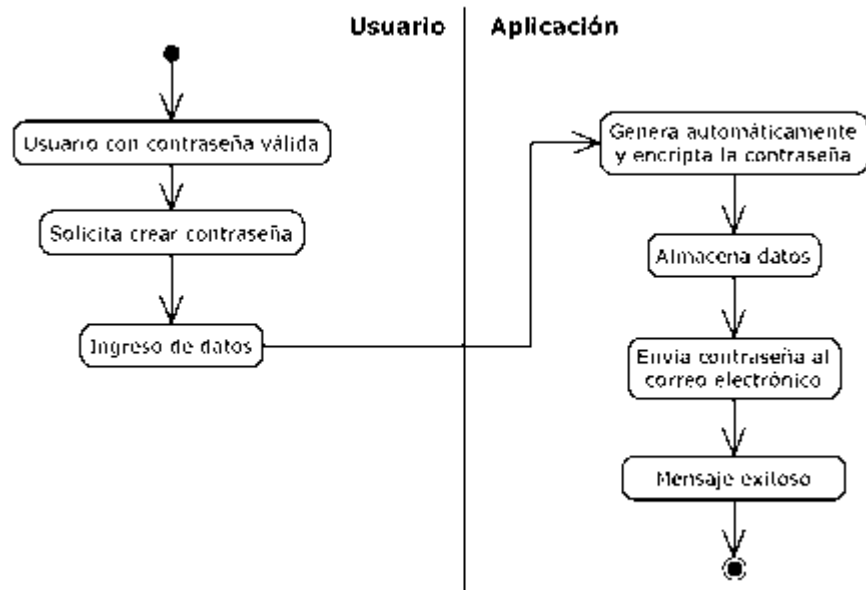
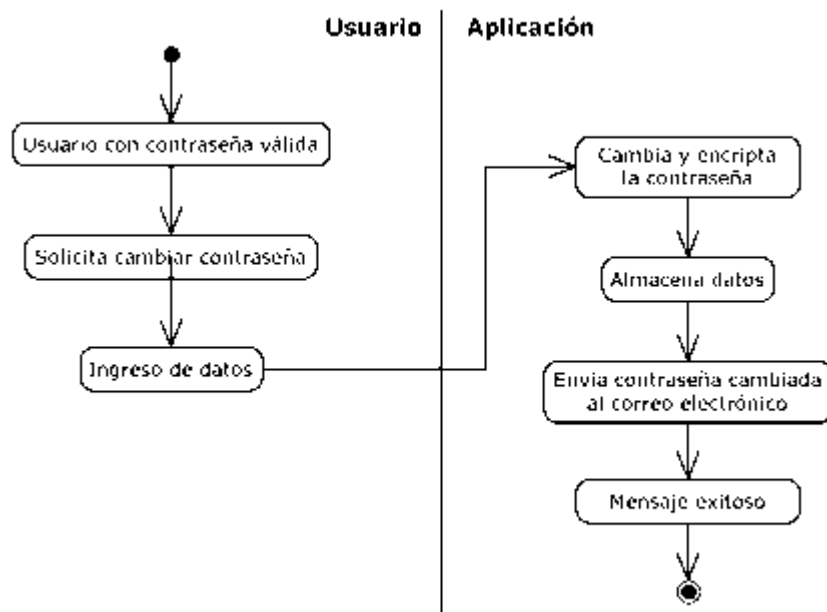


Figura 18: Modificar contraseñas.

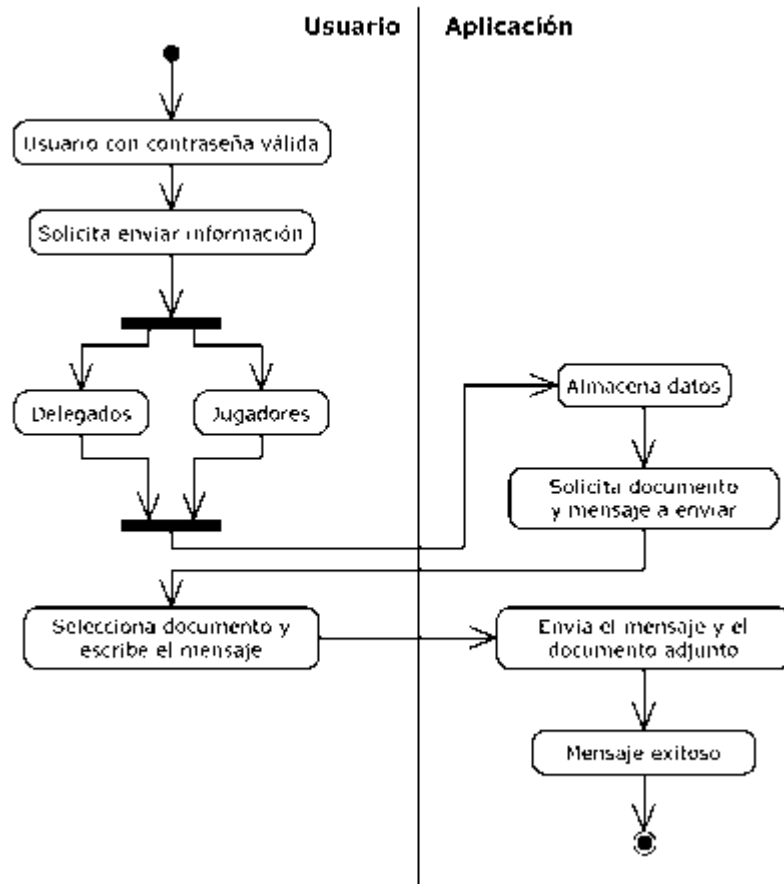


5.2.3. Enviar información por correo electrónico.

Otro de los puntos contemplados en el desarrollo del sistema era poder enviar información relacionada con el evento, que se estuviera desarrollando, a los correos electrónicos de los delegados y de los jugadores; en los cuales se les diera a conocer decisiones de los organizadores respecto a los diversos temas concernientes al ámbito deportivo.

Para ello, usando la clase PHPMailer, y en la búsqueda de hacer más amigable la interfaz y dar mayor comodidad al usuario, se implementó el uso del editor de texto llamado FCKeditor, el cual proporciona muchas de las potentes funcionalidades de editores de texto tradicionales, y no requiere ningún tipo de instalación en el ordenador del usuario. Nos permite mezclar en un documento textos, imágenes, tablas, etc., y mostrar en pantalla una imagen exacta de cómo quedaría impreso dicho documento.

Figura 19: Enviar información por correo electrónico.



5.2.4. Modificación de delegados y jugadores.

Este desarrollo se hizo debido a los posibles errores de digitación en los que puedan incurrir los delegados y/o los jugadores que ingresan la información al momento de la preinscripción. Estas modificaciones solo las puede realizar el usuario de tipo coordinador del sistema. No es permitido cambiar el código, la carrera y el equipo; el resto de datos están sujetos a modificaciones.

5.2.5. Validación de inscripciones.

Debido a un requisito de validación de la participación del estudiante por parte de la coordinación del evento, la inscripción realizada por el participante es tomada en el sistema como una preinscripción a la cual se le asigna un número consecutivo el cual depende del deporte al cual se inscriba y de la cantidad de participantes registrados en el mismo.

Esta validación comprende dos aspectos: el primero es el pago correspondiente a la cuota estipulada por la organización para solventar la participación del equipo o del jugador; y lo segundo es el cumplimiento de los requisitos establecidos por la organización para avalar la participación de los participantes, como por ejemplo, la carrera que representa un equipo preinscrito.

5.2.6. Sección contáctenos.

La sección de contáctenos nace de la necesidad de escuchar las opiniones, sugerencias, quejas o reclamos de los equipos y de los jugadores referente a: la organización del evento, las decisiones tomadas y las sanciones impuestas. Además con esto se resalta la importancia de mantener la comunicación entre los participantes y los directivos, y viceversa.

Para realizar esta sección nuevamente nos apoyamos en la clase PHPMailer y en el editor de texto FCKeditor anteriormente nombrados, los cuales nos brindan herramientas de fácil manejo para la realización de esta parte del proyecto.

Figura 20: Modificación de delegados y/o jugadores.

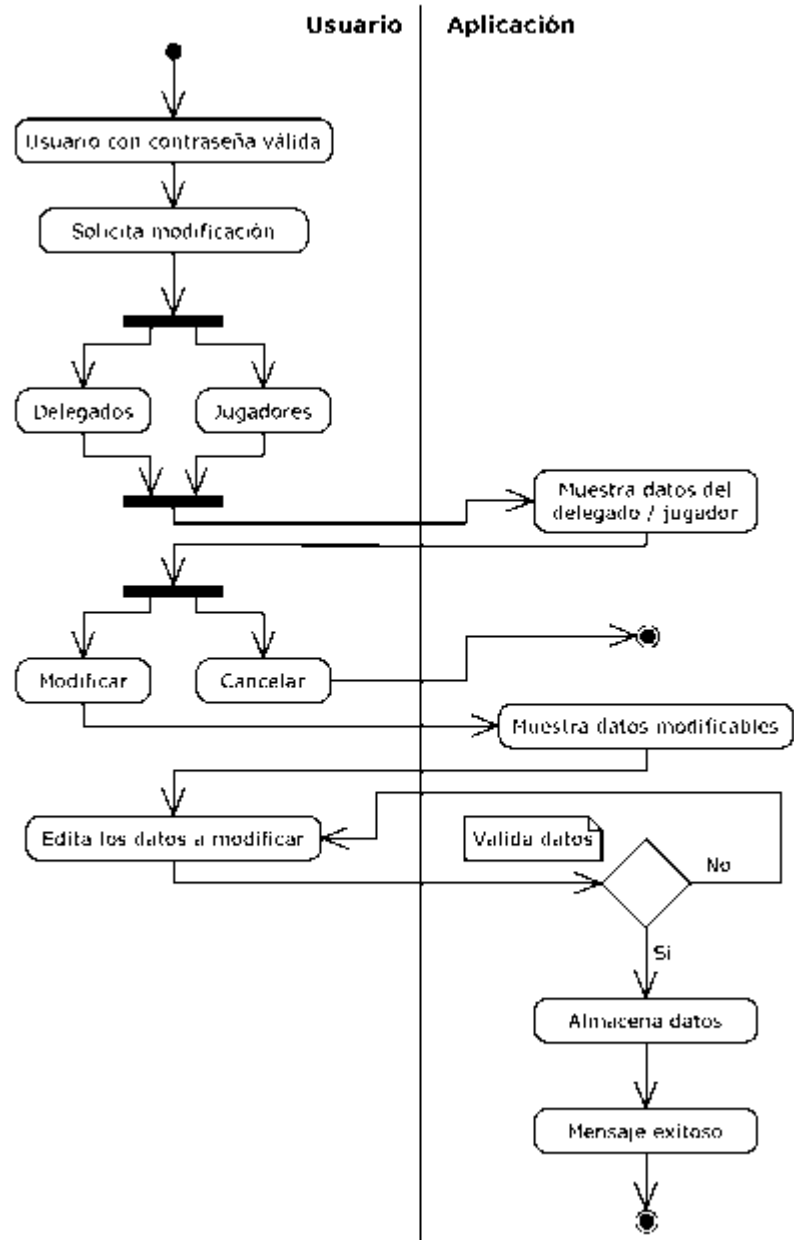
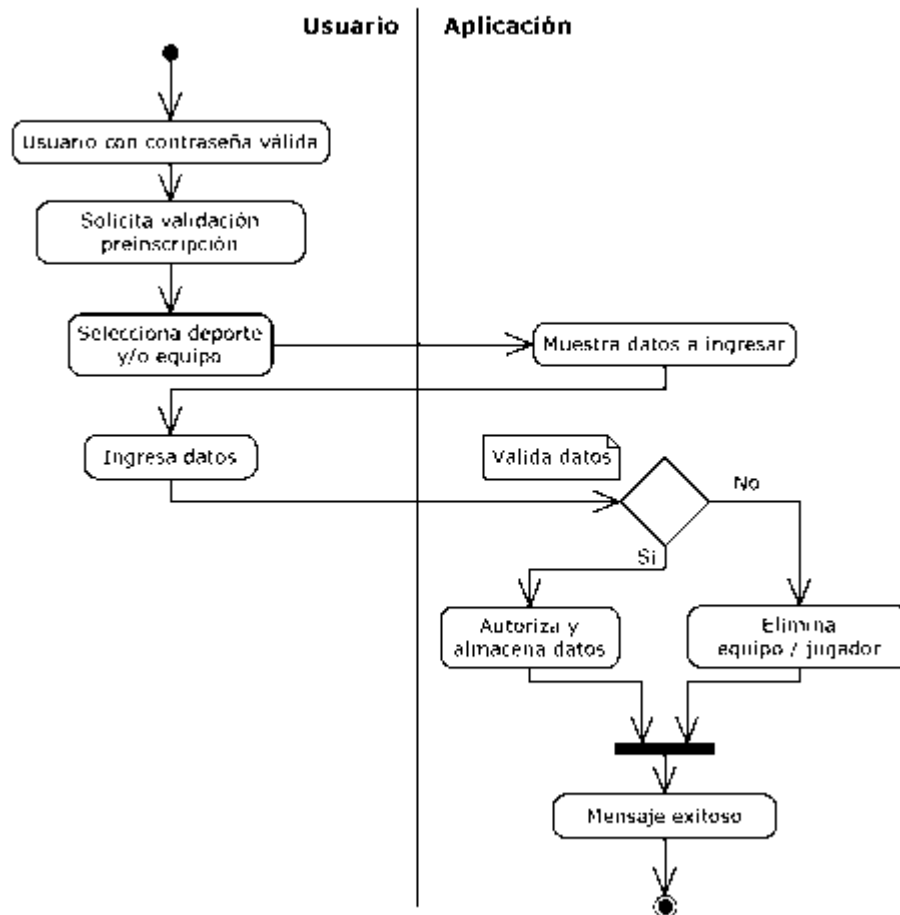


Figura 21: Validación de inscripciones.



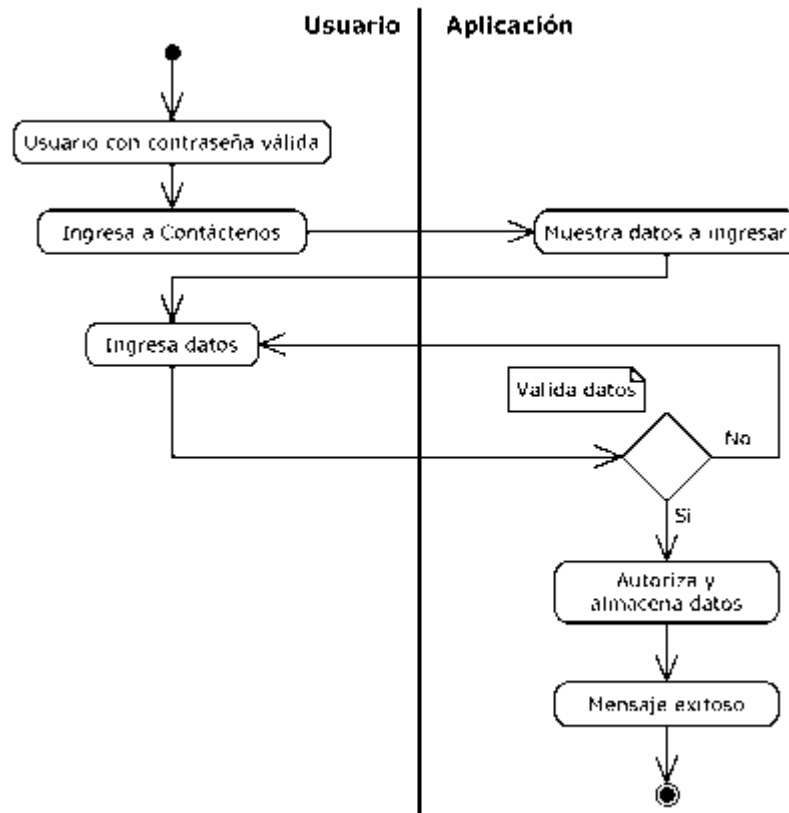
5.3. TERCER MINIPROYECTO.

En este miniproyecto se realizó una investigación sobre la generación de documentos en formato PDF²² mediante PHP, además se determinaron los ítems de desempate de cada deporte para la realización de la tabla de posiciones. Los demás desarrollos fueron:

- Elaboración del fixture para deportes individuales.
- Elaboración de la programación de los encuentros resultantes.
- Eliminación de Usuarios, Equipos y Jugadores.
- Inserción de los marcadores de los encuentros realizados.
- Generación de la tabla de posiciones.

²² **PDF**, del inglés Portable Document Format que significa: Formato de Documento Portátil. Esta sigla hace referencia a un formato de almacenamiento de documentos que especifica toda la información necesaria para la presentación final del documento, determinando todos los detalles de cómo va a quedar, no requiriéndose procesos anteriores de ajuste ni de composición de página.

Figura 22: Contáctenos.



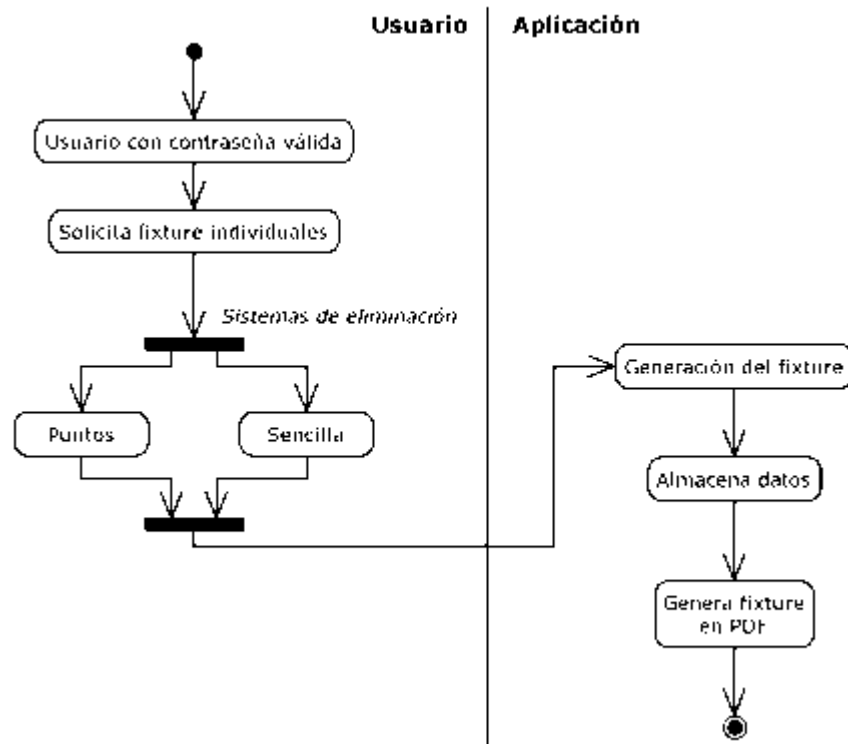
5.3.1. Elaboración del cuadro de enfrentamiento individuales.

La elaboración del fixture es uno de los puntos neurálgicos del sistema pues es la base de la generación de la programación de los encuentros que resulten del fixture, como vemos es un paso realmente importante. En este caso nos centraremos en la elaboración del fixture para los deportes individuales, nuevamente en este punto entran en juego muchas variables que permitirán organizar y programar los diversos deportes del evento.

Como en el caso de los deportes de equipo, para los individuales revisamos de nuevo los libros de consulta para determinar los sistemas de eliminación existente para estos deportes, los sistemas de eliminación encontrados fueron: Eliminación Sencilla y Round Robin o Eliminación por Puntos; como con los deportes de equipo aquí también se debe tener en cuenta la cantidad de equipos y el deporte que se irá a organizar para elegir el sistema de eliminación a emplear.

Nuevamente empleamos la clase FPDF para la generación en formato PDF para la difusión de este documento a los demás usuarios con el objetivo de mantener la información con estos.

Figura 23: Elaboración del fixture para deportes individuales.



Aquí al igual que para los deportes individuales, la distribución de los participantes y de las llaves de enfrentamiento se realizan de manera aleatoria, esto se debe a las razones expuestas en la elaboración del fixture de equipos (ver sección 5.2.1).

5.3.2. Elaboración de la programación.

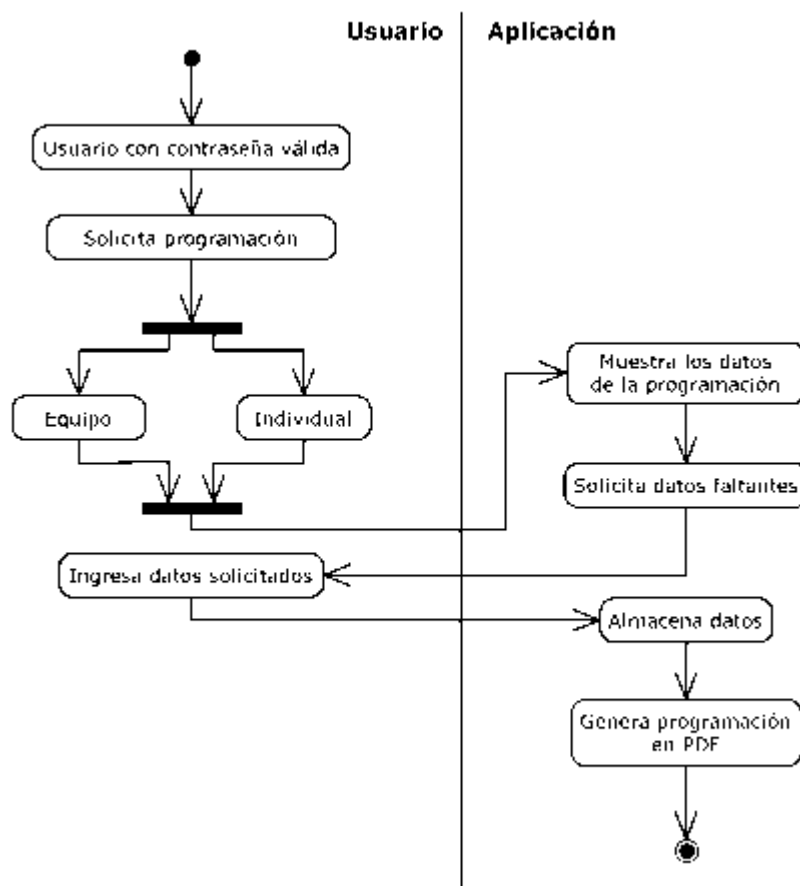
Una vez elaborados los fixture de los deportes de equipo e individuales, el siguiente paso es hacer la programación de los encuentros o partidos que se han generado a raíz del fixture; esta programación asigna a cada partido el lugar, la hora y el día en el cual se llevará a cabo.

Para realizar esta asignación se debe tener en cuenta: los escenarios deportivos existentes por deporte, la determinación de un horario de juego y la

duración del torneo o evento. Estos parámetros son clave sobretodo durante las primeras jornadas o rondas, pues es donde la mayor cantidad de encuentros se realizarán y deben programarse de manera óptima.

Elaborar la programación de los encuentros también requiere que este documento sea generado en formato PDF para su posterior distribución a los delegados de los equipos y a los jugadores de los deportes individuales. Por ello hacemos nuevamente uso de la clase FPDF, la cual ya nombramos, para poder realizar esta generación.

Figura 24: Elaboración de la programación de los encuentros.



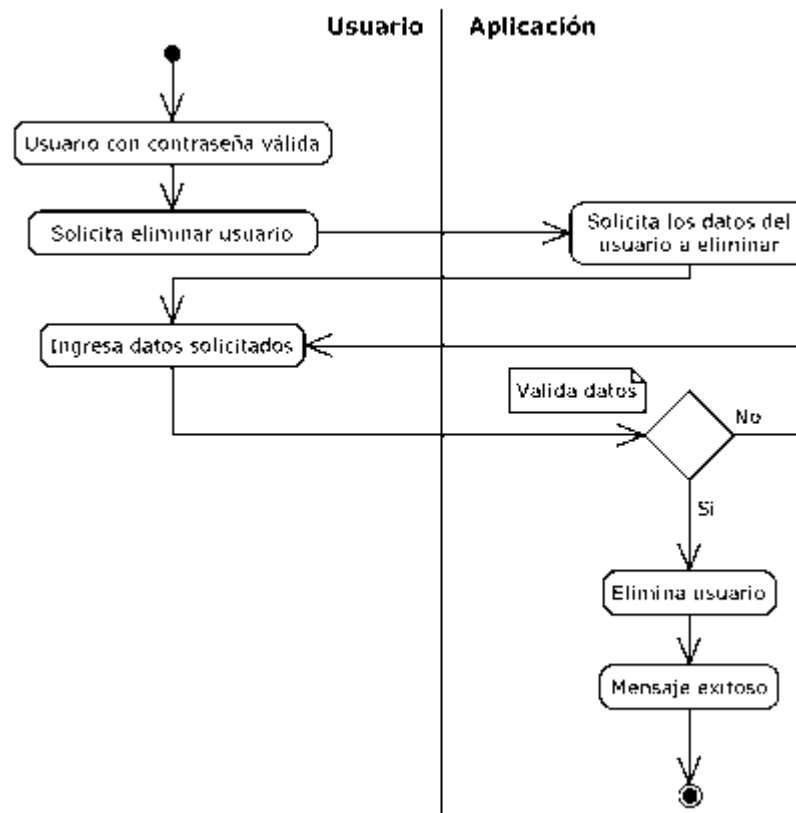
5.3.3. Eliminación de usuarios, equipos y jugadores.

Dado que los equipos y los jugadores registrados están sujetos al resultado de la validación de la coordinación el cual puede ser favorable o desfavorable; en caso de ser desfavorable el registro del equipo o del jugador debe ser

eliminado. Por el lado de los usuarios (coordinador, profesor y estudiante), deben ser eliminados al finalizar el evento por motivos de seguridad.

Dos razones por las cuales los equipos o los jugadores son eliminados del sistema son debido al no pago de la cuota establecida o por no cumplir los requisitos establecidos por la organización.

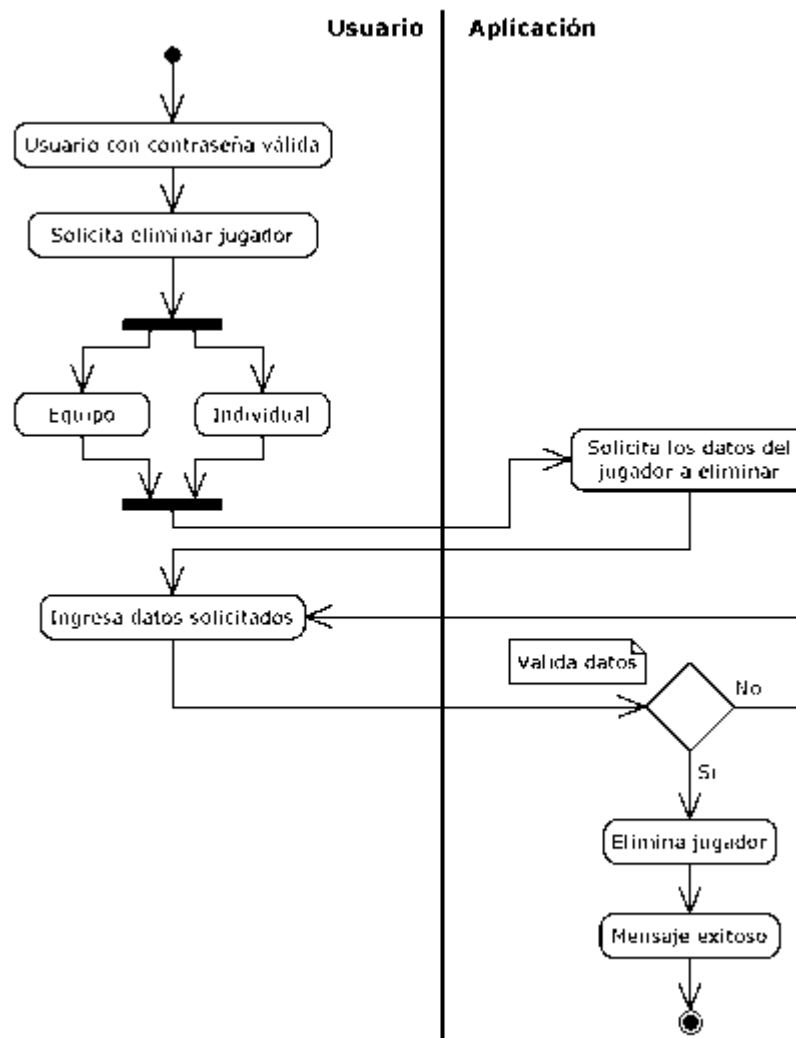
Figura 25: Eliminación del usuario (Coordinador, Profesor o Estudiante).



5.3.4. Inserción de marcadores.

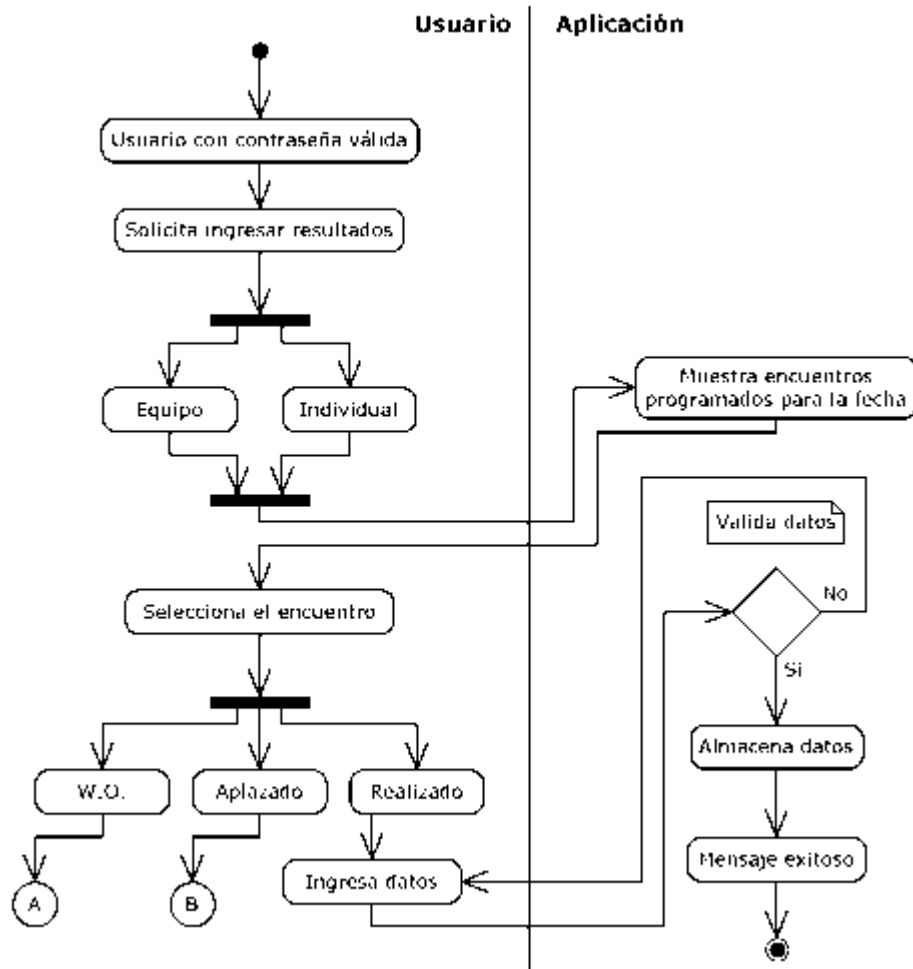
Una vez realizados los encuentros o partidos, la coordinación se dispone a recibir las planillas de juego de los respectivos partidos. Estas planillas contienen los resultados, o marcadores, de los encuentros así como también las estadísticas mínimas y la participación de los jugadores en el partido.

Figura 26: Eliminar equipo y/o jugadores.



Los datos consignados en las planillas de juego deben ser ingresados al sistema para ir generando la tabla de posiciones y para registrar las estadísticas y la participación de los jugadores, esto se realiza para cumplir con el requisito de crear la ficha de historial deportivo del estudiante y tener la suficiente información para generar la respectiva certificación de su actividad deportiva en la universidad. Para ingresar los marcadores se tiene en cuenta la manera en que se puntúa en los diferentes deportes del evento, además también se contemplan las posibilidades de aplazar o declarar w.o. cualquiera de los encuentros.

Figura 27: Ingreso de marcadores (continua en la figura 28).



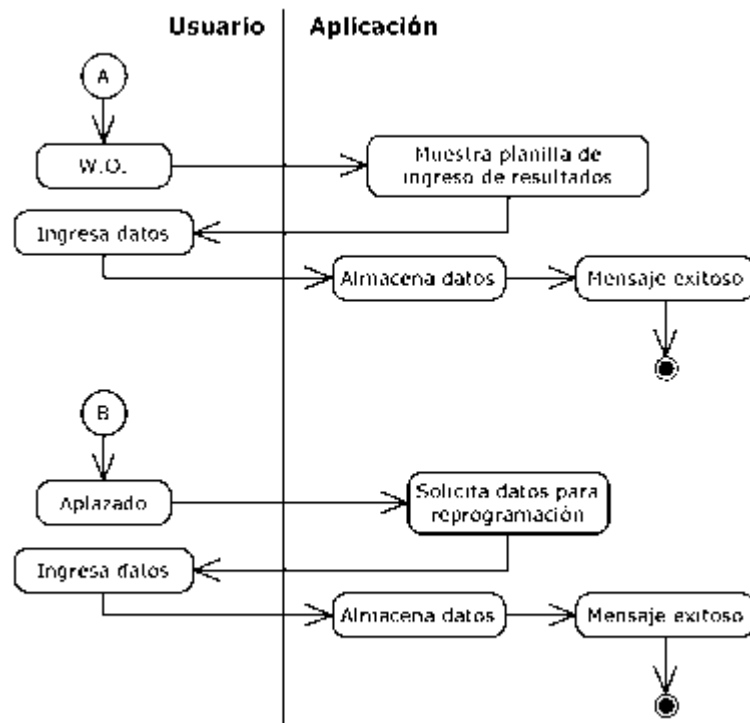
En cuanto a los aplazamientos, el sistema obliga a la reprogramación del lugar, el día y la hora del encuentro; la cual debe darse antes del final de la ronda. En el caso de los w.o., para cada deporte se contemplaron las sanciones y distribución de puntos establecida por la universidad o por las asociaciones internacionales de cada deporte.

5.3.5. Tabla de posiciones.

Para la realización de la tabla de posiciones la información base y necesaria para ordenarla fueron los ítems de desempate establecidos en cada deporte, sin excluir los marcadores y la puntuación dada por los mismos. Esta tabla de posiciones se programó de tal manera que la ordenación se haga automáticamente y para que soportara hasta un triple empate.

Los ítems de desempate para cada deporte pueden variar, esto nos dio la idea de desarrollar una tabla de posiciones por deporte al no poder generalizarla debido a los ítems cambiantes. Algunos de estos ítems son: goles a favor, partido jugado entre los empatados, partidos ganados, cesta promedio, etc.

Figura 28: Partidos declarados W.O. y aplazados.

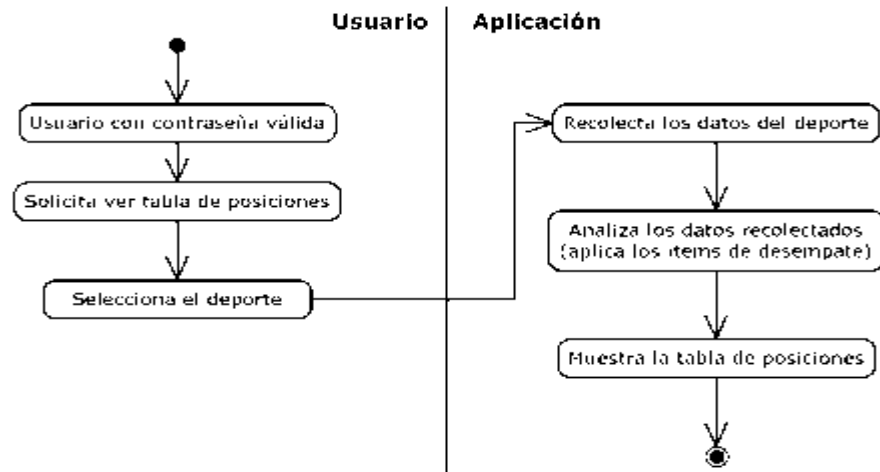


5.4. CUARTO MINIPROYECTO.

En este último paso, los desarrollos realizados fueron:

- Mostrar en pantalla el fixture, la programación y los resultados
- Sección de sanciones.
- Certificado de participación.

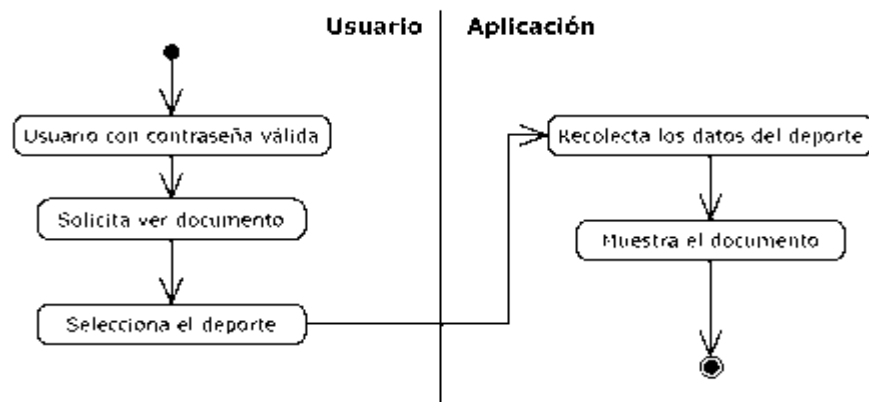
Figura 29: Tabla de posiciones.



5.4.1. **Mostrar cuadro de enfrentamientos²³, programación y resultados en pantalla.**

Durante los desarrollos previos a este miniproyecto, los documentos nombrados son visibles a los organizadores por ello nos era necesario crear dentro del sistema la posibilidad a los jugadores y delegados de observar estos tres documentos (el fixture, la programación y los marcadores). Como se esgrime en el título de este apartado, solo se muestran los resultados en pantalla pues es deber de la coordinación enviar a los delegados y jugadores los documentos generados en formato PDF de cada uno de ellos.

Figura 30: Mostrar documento.

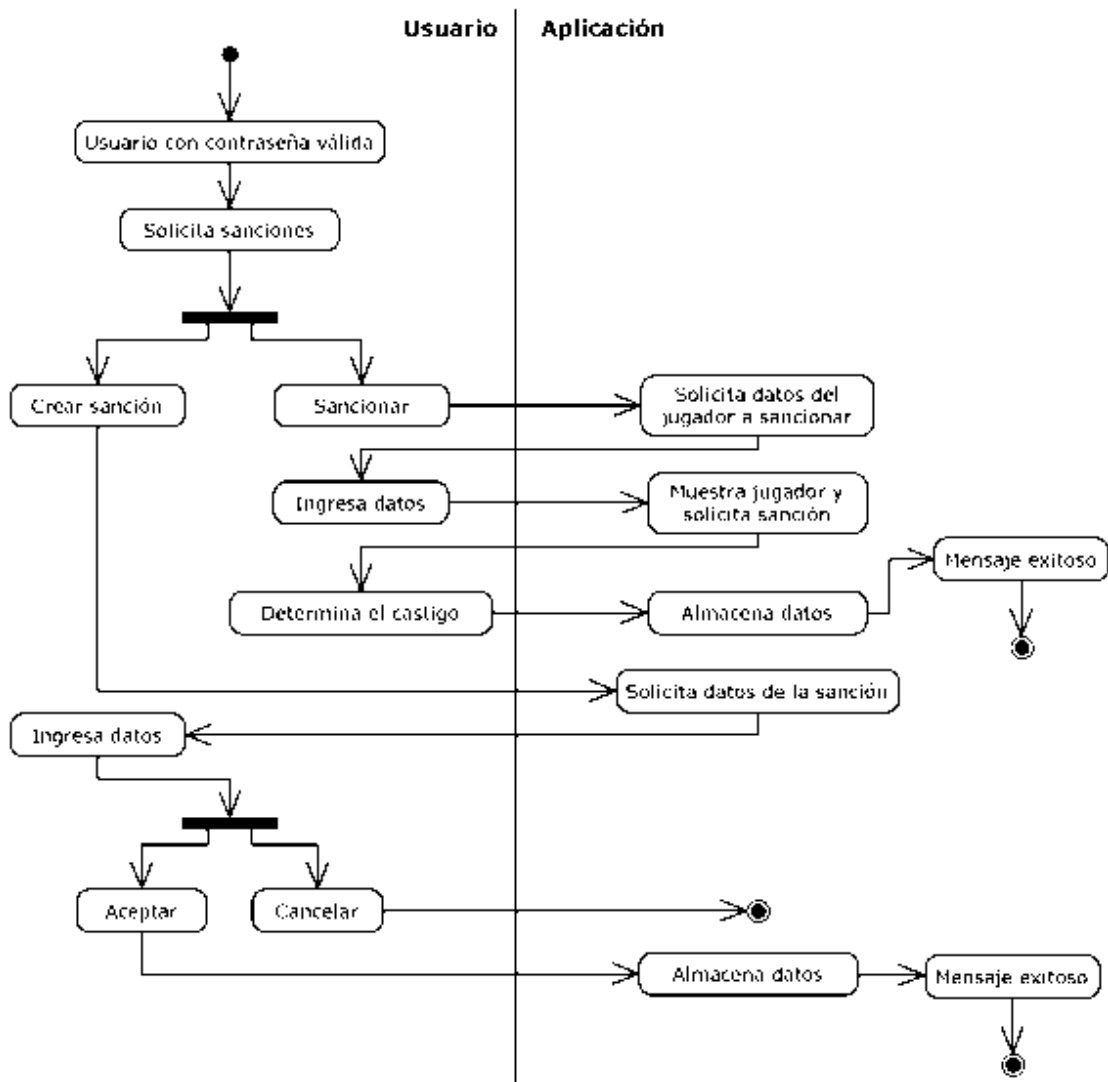


²³ **Cuadro de Enfrentamientos** es el término en español de la palabra Fixture.

5.4.2. Sección de sanciones.

En esta sección se trabaja la asignación de sanciones y castigos a los jugadores con tarjetas (amarillas, rojas, etc.) o amonestaciones; para recolectar estas sanciones recurrimos a las planillas de juego establecidas para cada deporte, pues en ellas están consignadas las faltas por las cuales el jugador se hace acreedor a la amonestación.

Figura 31: Sanciones.



En este punto nos centramos en determinar la sanción o el castigo dado al jugador; esta determinación hace referencia a las fechas de suspensión

6. CONCLUSIONES

- La información es el recurso más valioso de cualquier organización y por ello contar con tecnología y personal indicado para su correcto uso es de vital importancia para una institución como la Universidad Industrial de Santander. Por esta razón se planteó dentro del desarrollo de este proyecto utilizar tecnologías que brindarán una nueva perspectiva de trabajo.
- Las herramientas de software libre que en la actualidad se encuentran en el medio son suficientes para el correcto desarrollo del proyecto desarrollado. La velocidad de consulta en las base de datos, tiempos de acceso a la aplicación, capacidad de almacenamiento, cifrado de contraseñas entre otros aspectos, son puntos a favor que poseen este tipo de tecnologías. Internet proporciona una vía fácil para tener acceso a herramientas de software libre, esto combinado con la investigación y lectura continúa, derivó a la culminación exitosa del proyecto de grado.
- El uso de la metodología de espiral fue acertado, ya que garantizó la obtención de una aplicación que cumpliera con los objetivos planteados al inicio del proyecto, al permitir diseñar y detallar los miniproyectos realizados de acuerdo con las necesidades existentes al interior de la institución y a plantear a través del diseño global una oportunidad de vislumbrar el trabajo a desarrollar posteriormente. De esta manera queda claro que el uso del modelo espiral garantizó para el presente caso de estudio una excelente elección y que el éxito en gran parte dependió de dicha selección.
- El desarrollo de esta herramienta ofrece un soporte más ágil y eficiente a los procesos que el Departamento de Educación Física y Deportes lleva a cabo manualmente durante la organización de los eventos deportivos realizados en la Universidad.
- La aplicación desarrollada, esperamos, sea el comienzo de una serie de trabajos de grado relacionados con el área deportiva de la Universidad, encabezada por Departamento de Educación Física y Deportes.

7. RECOMENDACIONES

- Brindar el ambiente para dar continuidad a más proyectos que complementen el presente trabajo de grado, es el aspecto más importante en esta sección.
- Realizar la selección y montaje de un servidor Web apropiado para el buen funcionamiento de la aplicación desarrollada.
- Continuar ampliando la aplicación desarrollada, con una mejora en el cronometraje de los resultados en las pruebas atléticas para realizar el ingreso de estos resultados de manera automática, en tiempo real.
- Otra mejora a la aplicación sería la captación de una estadística más detalle durante la realización de los encuentros, para acumular un record más amplio de las estadísticas de los jugadores, de tal forma que se incentive más la participación del estudiantado en los eventos.
- Si se quisiera se podría emplear herramientas licenciadas, y no herramientas libres, para rediseñar o replantear el desarrollo de la presente aplicación.
- A largo plazo sería interesante que la aplicación desarrollada contara con el soporte de la División de Servicios de Información – DSI – para poner a disposición institucional el trabajo de grado desarrollado.

8. BIBLIOGRAFÍA

LIBROS - E-BOOKS.

Bakken, Stig Sæther; Aulbach, Alexander; Schmid, Egon; Winstead, Jim; Torben Wilson, Lars; Lerdorf, Rasmus; Zmievski, Andrei; Ahto, Jouni

Manual de PHP (en español)

Editado por: Rafael Martínez (Coordinador), Víctor Fernández Leonardo Boshell
08-07-2002 Copyright © 2002 por el Grupo de documentación de PHP

Cárcamo Sepúlveda, José

Programación en Entorno Web

Santander, Colombia, Ediciones UIS, Mayo 2007.

Erosa Alvarado, Juan Ramón Lic.

Administración en Educación Física y Deporte

México, Editorial Didáctica Moderna, 1996.

Fowler, Martin; Kendall, Scout

UML Gota a gota

Editorial Prentice Hall, Pearson, Addison Wesley Longman

Gaitán, Edgar

Aplicaciones para Internet.

Guía de instalación de PHP, Apache, MySQL y phpMyAdmin.

Guía desarrollada por el SENA

Cursos Virtuales – www.senalvirtual.edu.co

Gil Rubio, Francisco Javier; Tejedor Cerbel, Jorge A.; Yague Panadero, Agustín

Creación de Sitios Web en PHP4

Editorial Osborne-McGrawHill, 2001.

Gómez Flórez, Luis Carlos

Guía para el desarrollo de proyectos de grado

Universidad Industrial de Santander, 1999.

Martín Andrés, Oscar.

Manual Práctico de Organización Deportiva

Madrid, España, Editorial Gymnos, 1996.

Martins Carrizo, Marta Beatriz

Dreamweaver 4 - Fireworks 4 practico Guía de aprendizaje

Editorial Osborne McGraw-Hill

Mediavilla, Gerardo

Comunicación corporativa en el deporte

Editorial Gymnos, Madrid, España, 2001.

Paria Roche, Fernando

La planificación estratégica en las organizaciones deportivas

Editorial Paidotribo, Barcelona, España.

Parks, Janet B.; R. K. Zanger, Beverly

Gestión deportiva

Ediciones Martínez Roca S.A., Barcelona, España

Ratschiller, Tobias y Gerken, Till

Creación de Aplicaciones Web con PHP4

Madrid, España, Pearson Educación, 2001.

SITIOS WEB

www.apache.org: Sitio oficial de The apache software foundation, Apache Documentation.

www.baluart.net/articulo/109/fckeditor-i-parte-su-instalacion.php: sitio Web del editor de texto FCKeditor.

www.desarrolloweb.com: sitio donde se puede acceder a varios manuales sobre diferentes lenguajes de programación, entre ellos PHP.

www.fpdf.org y www.fpdf.org/es/doc : sitio Web y manual de referencia de la clase FPDF usada para generar documentos en formato PDF desde PHP.

www.getec.etsit.upm.es/docencia/gproyectos/planificacion/cvida.htm: sitio Web donde se explican los pasos de la planificación de proyectos en todas las áreas.

www.gestiondeportiva.net: es una revista digital dedicada en exclusiva a contenidos de organización y gestión en el ámbito deportivo que surge dentro la asignatura "Gestión y organización de actividades y eventos deportivos" de la Facultad de Ciencias de la Actividad Física de la Universidad de León, para facilitar a los alumnos de esta facultad el acceso a información relacionada con la gestión deportiva.

www.php.net: sitio oficial de PHP

http://phpmailer.sourceforge.net y

http://phpmailer.sourceforge.net/tutorial.html: sitio Web y manual de referencia de la clase PHPmailer usada para enviar correos electrónicos con datos adjuntos vía Internet.

www.programación.net/php/articulos,

www.tutorial-enlace.net/tutoriales-Php.html y www.tufuncion.com: sitios Web referentes de artículos e información sobre PHP y MySQL.

9. ANEXOS

ANEXO A: MANUAL DE USUARIO.

El siguiente documento comprende una explicación, para los usuarios involucrados, de los pasos a seguir al momento de la creación y organización de alguno de los eventos deportivos realizados en la UIS.

A.1. INGRESO AL SISTEMA

Al ejecutar la página inicial de la aplicación nos aparece un formato de ingreso, este formato le solicita al posible usuario el correo electrónico y la contraseña de acceso al sistema, con estos datos la aplicación puede determinar si quien ingresa esta información es o no usuario del sistema y respectivamente concede o no el acceso a la página principal del usuario.

A.2. PÁGINA PRINCIPAL DEL USUARIO

Figura 33: Página principal



La página principal que se muestra después de acceder al sistema depende del tipo de usuario que ingrese, pues todos los usuarios tienen un nivel de interacción diferente, esta diferencia es notable en el menú de opciones con el cual trabaja cada usuario. A continuación describiremos el aspecto de la página principal y luego describiremos las opciones incluidas en los menús.

A.3. DESCRIPCIÓN DE LA PÁGINA PRINCIPAL

La página principal de encuentra dividida en tres partes o frames, estos son:







A.3.1. Parte Superior o Top

Aquí encontramos el encabezado de la página e incluye el logotipo de la universidad.

Figura 34: Parte superior o Top.



De la parte superior o top se encuentra una barra de ayuda, con los siguientes botones:

-  **Atrás:** Retrocede a la página anterior.
-  **Adelante:** Retorna a la página de donde se retrocedió.
-  **Actualizar:** Refresca o actualiza la página actual.
-  **Contáctenos:** Da la oportunidad al usuario de enviar un correo electrónico a los organizadores.
-  **Inicio:** Vuelve a la página inicial de la aplicación.
-  **Cerrar Sesión:** Cierra la sesión del usuario actual.

A.3.2. Parte Izquierda o Menú

Allí ubicamos el menú de opciones para cualquiera de los tres tipos de usuario.

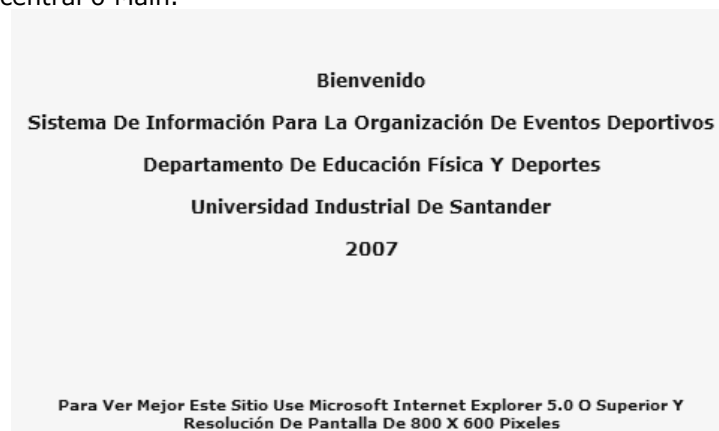
A.3.3. Parte Central o Main

En esta parte se despliegan todas las páginas relacionadas con la opción seleccionada en el menú.

Figura 35: Parte izquierda o Menú.



Figura 36: Parte central o Main.



A.4. DESCRIPCIÓN DEL MENÚ

A.4.1. Menú Generar.

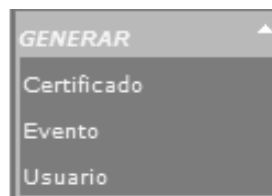
Este menú está disponible para el usuario de tipo coordinador y le permite a este crear, modificar y eliminar nuevos usuarios del sistema. Las opciones que presenta este menú son:

- **Certificado:** seleccionando esta opción accedemos a la generación del certificado de participación o ficha de historial deportivo del estudiante. Es necesario saber el código que identifica al estudiante para generarlo, en este certificado aparecerán todas las participaciones del estudiante en los diferentes eventos realizados durante su estancia en la

universidad. Se mostrará en pantalla y también se generará en formato PDF, si así se selecciona.

- **Evento:** una vez seleccionada esta opción debemos elegir entre los eventos antes determinados u otro, seguido de ello también debemos seleccionar los deportes que se incluirán en este evento; una vez realizado esto veremos un mensaje indicando el registro de los deportes y evento. El paso siguiente es determinar los responsables del evento; es necesario incluir en nombre, teléfono, extensión, celular y correo electrónico del organizador, coordinador y director del evento; una vez validados se procederá al registro o a reingresar los datos en caso de que la validación sea negativa.
- **Usuario:** con esta opción creamos un nuevo usuario del sistema, debemos tener a la mano datos del usuario a crear como: el nombre, el correo electrónico y asignarle el tipo. Al momento de pulsar Enviar, el sistema registra los datos y genera la contraseña respectiva, además muestra un mensaje indicando el registro y el envío de la contraseña al correo del usuario creado.

Figura 37: Menú Generar.



A.4.2. Menú Inscripciones.

Esta opción es acerca de las inscripciones de los jugadores a los deportes seleccionados previamente en la generación del evento. Las opciones dentro de este menú son:

- **Equipo:** las inscripciones de equipos se dividen en dos pasos; en el primero se debe ingresar los datos del delegado y del equipo, estos datos son validados y si están correctos se pasa al segundo paso y si ocurre lo contrario de deben reescribir. En el segundo paso se debe ingresar los datos de los jugadores del equipo, seguido a esto se muestra un mensaje indicando la aceptación del registro y el número de preinscripción dado al equipo.
- **Extemporánea:** Esta opción se realizó debido a que muchas veces los equipos no inscriben la totalidad de la plantilla de inscripción y tiempo después acuden a inscribir a los jugadores restantes, con esta opción es posible hacer esto pero dentro de unos plazos establecidos por la organización.

- **Individual:** las inscripciones a deportes individuales son sencillas, el posible jugador debe ingresar todos los datos solicitados de ser válidos muestra un mensaje indicando su registro y número de preinscripción; de lo contrario debe reescribir los datos.
- **Validaciones:** esto se realiza para confirmar la participación del equipo o jugador en el evento una vez realizado el pago del valor de su participación, se requiere el número de preinscripción y las fotos de los jugadores y delegado para realizarla.

A.4.3. Menú Carrera Atlética.

Este menú depende de la organización o no de la carrera atlética de la universidad y es donde se nos permite asignar el número al participante, registrar su marca y ver los resultados.

- **Asignar Número:** se asigna el número con el cual se identificará al participante durante la carrera.
- **Borrar Participante:** elimina a cualquier participante registrado en la carrera atlética, es necesario el número de la cédula para usar esta opción.
- **Ingresar Resultados:** esta opción permite al usuario ingresar los tiempos o marcas de los participantes de la carrera.
- **Ver Resultados:** la opción permite ver los resultados ingresados de la prueba atlética.

Figura 38: Menú Inscripciones (izq.) y Carrera Atlética (der.).



A.4.4. Menú Modificar.

Este menú nos permite cambiar los datos ingresados erróneamente durante la preinscripción, como también cambiar la contraseña de ingreso al sistema.

- **Cambiar Contraseña:** como su nombre lo indica en esta opción se cambia la contraseña generada por el sistema por una personalizada, es necesario tener el correo electrónico del usuario que cambiará la

contraseña, la contraseña generada y la nueva contraseña. Una vez registrado el cambio el sistema informará sobre el hecho y enviará la nueva contraseña al correo electrónico del interesado.

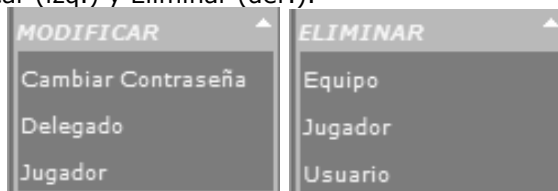
- **Delegado:** esta opción nos permite ver y modificar los datos de los delegados de los diversos equipos en competencia, se necesita el código, equipo y deporte para proceder a la modificación.
- **Jugador:** al igual que la anterior nos permite ver y modificar los datos del jugador, se necesita el código y el deporte para realizar la modificación.

A.4.5. Menú Eliminar.

Esta opción nos permite eliminar los equipos y jugadores que no cumplan con la validación, como también a los usuarios que ya no tienen permiso para acceder a la aplicación.

- **Equipo:** se debe tener a la mano la información del nombre del equipo y del deporte en el cual participa para realizar la eliminación, automáticamente elimina a los jugadores relacionados con el equipo.
- **Jugador:** elimina a cualquier jugador registrado en los deportes individuales, es necesario el código para operar esta opción.
- **Usuario:** en esta opción es necesario el correo electrónico del usuario a eliminar para proceder a borrarlo de la base de datos.

Figura 39: Menú Modificar (izq.) y Eliminar (der.).



A.4.6. Menú Carnetizar.

Al seleccionar esta opción nos permite ingresar a la generación de carnés o carnetización de los jugadores registrados en los diversos deportes del evento.

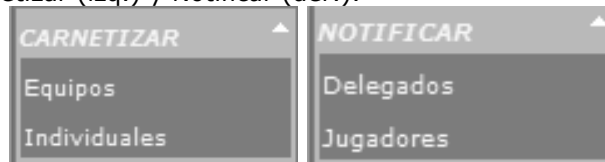
- **Equipos:** en esta opción se imprime en pantalla los carnés de un determinado equipo, es necesario conocer el deporte y el nombre del equipo, los carnés se muestran de dos en dos y está la posibilidad de imprimirlos directamente de la página que los muestra.
- **Individuales:** esta opción imprime en pantalla a todos los jugadores inscritos en determinado deporte individual; también es posible imprimirlos desde la página que los muestra.

A.4.7. Menú Notificar.

Este menú nos permite enviar información referente al evento a los delegados o jugadores registrados. Dependiendo de la opción seleccionada se cargan en el mensaje los correos electrónicos de los delegados o de los jugadores.

- **Delegados:** esta opción carga los correos electrónicos de los delegados de los diversos equipos y muestra un formulario para ingresar el mensaje que se desea enviar y además, si se desea, de archivos adjuntos.
- **Jugadores:** en esta opción se cargan los correos electrónicos de los jugadores inscritos en los deportes individuales y muestra un formulario para el ingreso del mensaje que se desea enviar y además, si se desea, de archivos adjuntos.

Figura 40: Menú Carnetizar (izq.) y Notificar (der.).



A.4.8. Menú Fixture.

El menú fixture es uno de los menús más importantes del sistema, este menú nos permite empezar a organizar cada uno de los deportes del evento, además de ser paso fundamental de la programación.

- **F. Equipos:** con esta opción damos inicio al proceso de organización los deportes de equipo, debemos tener en cuenta que deporte vamos a organizar, el número de equipos registrados y los sistemas de eliminación determinados. Una vez realizado esto el sistema genera en formato PDF del fixture elaborado.
- **Ver F. Equipos:** nos permite ver en pantalla el fixture organizado para el deporte de equipo seleccionado.
- **F. Individuales:** esta opción da por iniciado el proceso de organizar los deportes individuales; en este punto es necesario tener en cuenta que deporte organizaremos, el número de inscritos y que sistema de eliminación se usará. Después de elaborado el fixture el sistema se encargará de generar su respectivo PDF.
- **Ver F. Individuales:** la opción nombrada nos da la posibilidad de ver en pantalla el fixture del deporte individual elaborado.

Figura 41: Menú Fixture.



A.4.9. Menú Programación.

En este menú se asignarán los lugares, las horas y los días en los cuales se llevarán a cabo los encuentros resultantes de la elaboración del fixture, se encuentra dividida en dos partes, deportes de equipo y deportes individuales, como a continuación se describe.

- **P. Equipos:** aquí se programarán los encuentros correspondientes a los deportes de equipo. Esta opción muestra un listado con todos los partidos a programar y da la opción de asignar el lugar, la hora y el día, en el que se realizará; antes de esto se debe seleccionar el deporte que se programará. Después de la asignación se genera el respectivo documento PDF de la programación.
- **Ver P. Equipos:** muestra en pantalla el resultado de la programación.
- **P. Individuales:** aquí se programará los partidos referentes a los deportes individuales. Debemos seleccionar el deporte a programar, luego de esto se muestra un listado con todos los partidos a programar y da la opción de asignar el lugar, la hora y el día, en el que se realizará. Una vez realizada la asignación se genera el respectivo documento PDF de la programación.
- **Ver P. Individuales:** muestra en pantalla el resultado de la programación.

A.4.10. Menú Resultados.

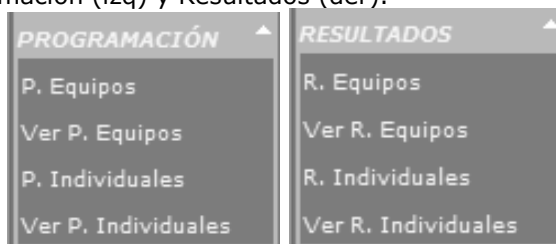
Este menú nos permite ingresar los resultados, o marcadores, de los partidos realizados; también es posible elegir entre un partido realizado, aplazado o declarado W.O. Estos marcadores son recogidos mediante las planillas de juego de los respectivos encuentros y deportes.

- **R. Equipos:** esta opción muestra un listado con todos los partidos programados, da la opción de elegir alguno de ellos y seleccionar su estado (Realizado, Aplazado, W.O.). Si el estado es Realizado se procede a ingresar el marcador y las estadísticas del encuentro; si el estado es Aplazado inmediatamente se debe reprogramar e ingresar el nuevo día,

hora y lugar del encuentro; y si el estado es W.O., se asignan las sanciones especificadas por los reglamentos. Los datos ingresados son validados.

- **Ver R. Equipos:** muestra en pantalla los resultados de los encuentros realizados.
R. Individuales: esta opción nos muestra un listado con todos los partidos programados, nos da la opción de elegir alguno de ellos y seleccionar su estado (Realizado, Aplazado, W.O.). Los pasos son los mismos que se realizan en los deportes de equipo.
- **Ver R. individuales:** muestra en pantalla los resultados de los encuentros realizados.

Figura 42: Menú Programación (izq) y Resultados (der).



A.4.11. Tabla de Posiciones.

Este menú permite ver la tabla de posiciones de un evento y deporte determinado, en ella se puede observar como quedaron las posiciones, los puntajes y las estadísticas de los equipos o jugadores en competencia.

Figura 43: Menú Tabla de Posiciones.



ANEXO B: COPIAS DE SEGURIDAD.

En este anexo se describirá la forma de hacer las copias de seguridad, o backups, de las bases de datos de la aplicación. Para ello, implementamos el uso de una herramienta llamada *MySQL Administrator*, la cual viene contenida en el paquete GUI Tools; esta aplicación de MySQL nos proporciona una forma muy sencilla de realizar los backups y sus restauraciones.

B.1. INSTALACIÓN MYSQL ADMINISTRATOR

Descargar y ejecutar la aplicación, el archivo que contiene esta herramienta tiene por nombre: `mysql-gui-tools-5.0-r12-win32.msi`

Seleccionar la instalación tipo *Custom*, optar por instalar las herramientas comunes y el MySQL Administrador, y continuar.

Con estos pasos se instalará dentro del grupo de Programas MySQL la herramienta MySQL Administrator.

B.2. CONFIGURACIÓN

Ejecutar la herramienta, menú inicio, todos los programas, mysql, mysql administrator. Una vez iniciada nos mostrará una ventana solicitándonos los datos de la conexión.

Hacemos clic en el botón a la derecha de *Stored Connection* y completamos los datos pedidos, estos son:

Connection	:	nombre dado a la conexión.
UserName	:	root
HostName	:	nombre del host o localhost
Port	:	3306

Le damos *Save* (Guardar) y luego *Close* (Cerrar); esto nos retorna a la ventana principal.

NOTA: Esta configuración sólo se realiza una vez, siempre y cuando no exista o sea necesaria otra conexión; en caso de realizar otra conexión se siguen los pasos ya descritos.

B.3. BACKUPS

En Stored Connection, en la ventana principal de la aplicación, seleccionamos el nombre de la conexión realizada en el punto anterior e ingresamos la contraseña del usuario establecido y hacemos clic en OK.

NOTA: Cada vez que ingresemos a la aplicación debemos escribir la contraseña del usuario que se estableció en la conexión.

Nos aparece la ventana de las actividades que podemos realizar, en el menú izquierdo de la pantalla seleccionamos Backup (Copia de seguridad) y luego en la parte inferior derecha oprimimos New Project (Nuevo proyecto).

En Project Name (Nombre del proyecto) damos un nombre a la actividad a realizar.

Buscamos la Sección Schemata, donde se encuentran las bases de datos, y arrastramos de esta sección hacia Backup Content, los nombres de la(s) base(s) de datos a copiar.

Oprimimos Save Project (Guardar proyecto) y luego Execute Backup Now (Ejecutar backup ahora).

Elegimos el nombre con el cual guardar el Backup y donde se guardará, le damos aceptar y con ello se realizará la copia de seguridad de las bases de datos seleccionadas.

B.4. RESTAURAR BACKUP

Para restaurar un backup, en el menú izquierdo de la aplicación seleccionamos Restore (Restaurar) y oprimimos Open Backup File (Abrir archivo backup).

Buscamos el archivo que se quiere cargar y luego hacemos clic en Star Restore (Comenzar restauración) y al final del proceso Close (Cerrar).

Con estos pasos la herramienta carga el backup de la base de datos realizado con anterioridad.

ANEXO C: GLOSARIO DE TÉRMINOS DE LA ORGANIZACIÓN DEPORTIVA.

PLANEACIÓN: Es prever las diferentes situaciones que se pueden presentar durante los desarrollos del evento, reconocidos o posibles de suceder en la institución.

CONTROL: consiste en establecer las normas de registro para cada una de las situaciones previstas en el desarrollo del evento.

EVENTO DEPORTIVO: Suceso de todas las actividades deportivas que conforman el programa

HISTORIAL DEPORTIVO: Registro real de participación en partidos y tiempo de actividad física realizada por cada estudiante durante su permanencia en la Universidad.

PARTIDOS: Los encuentros entre dos equipos o participantes

GESTIÓN DE EVENTOS DEPORTIVOS: La suma de actividades y tareas técnicas, institucionales, de socialización y comercialización que se realizan para lograr la mejor organización del torneo o campeonato.

ACTIVIDADES DEPORTIVAS-RECREATIVAS: Torneos que primordialmente, buscan objetivos educativos mediante la organización de disciplinas reglamentadas oficialmente, pero que se adaptan para alcanzar los objetivos planeados

PLANTILLAS: tablas o planillas que se diseñan para el control de las diferentes acciones o resultados planeados

ASPECTOS ORGANIZATIVOS: distribución y agrupación adecuada de las actividades y tareas necesarias para el desarrollo del evento

TABLAS DE RESULTADOS: Son las planillas donde se anotan los detalles más importantes de un juego y permiten la agilización de la información y la correcta interpretación del resultado de actividades de conjunto o individuales.

RECOPIACIÓN DE LAS ESTADÍSTICAS DEPORTIVAS: Es la compilación y archivo de todos los datos registrados durante el desarrollo de un evento o Historial Deportivo

CONTROL DE ENFRENTAMIENTOS: Es el registro hecho de la realización de un partido o encuentro.

TABLAS DE POSICIONES: Datos acumulados y ordenados de resultados obtenidos por cada equipo o participante que indican la posición o puesto que se ocupa en el torneo.

ORGANIZACIÓN: Es la selección del evento que se llevará a cabo y que deportes se jugarán durante el semestre académico, así como el número máximo de participantes en cada actividad, los premios y la promoción del campeonato.

SISTEMAS DE COMPETENCIA: Son los diferentes métodos por medio de los cuales jugarán los equipos o personas para obtener un único ganador. Algunos dependen de la cantidad de jugadores y otros del tipo de deporte.

FIXTURE: Es el primer paso previo a la programación, aquí se hace la distribución de los equipos o personas y se organizan los enfrentamientos, aunque estén registrados, no se identifican ni equipos ni personas, tampoco donde ni a que hora se enfrentarán.

PROGRAMACIÓN: Es tomar los datos generados por el fixture y, en este caso, aplicar nombres de equipos y personas, y además asignarles un lugar, una fecha y una hora al enfrentamiento o partido.

PARTICIPANTES INSCRITOS: El numero total de participantes que cumplieron los requisitos establecidos por la organización y fueron aprobados

PLANILLA DE JUEGO: Es un documento en el cual se consigna la información del encuentro: los nombres de quienes se enfrentan, el marcador, quienes jugaron, quienes hicieron puntos, a quienes amonestaron, etc. Este papel se entrega a la coordinación apenas terminado el encuentro y firmado por el juez y los delegados.

MARCADOR: Es el resultado de la confrontación entre equipos o jugadores. La mayoría de veces se da en forma numérica y determina un ganador, un perdedor o un empate, si el deporte lo permite, y una asignación de puntos que depende del deporte realizado.

APLAZADO: Es cuando el encuentro a realizarse debe ser suspendido y jugado en un día diferente al establecido, para esto deben existir causales de fuerza mayor para tomar esta decisión.

W.O.: Es cuando uno de los participantes de un encuentro no se presenta en el lugar determinado a la hora establecida, esta falta acarrea unas sanciones al equipo o persona que incurra en ello.

AMONESTACIÓN: En la mayoría de los deportes estas amonestaciones están sujetas al color de unas tarjetas que tienen en su haber los jueces del encuentro y a unas faltas establecidas por los entes internacionales del respectivo deporte. Estas pueden corresponder a un primer llamado de atención, una expulsión del juego y a las sanciones dispuestas en relación a dicha falta.

SANCIÓN: Consiste en el castigo impuesto al jugador por la falta cometida, depende de lo agravante que sea dicha falta está establecido el castigo, este puede representar solo la expulsión por un partido o más, la suspensión del campeonato o hasta la suspensión de toda participación en cualquier actividad organizada por el DFED.