



UNIVERSIDAD TÉCNICA DE AMBATO

**FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E
INDUSTRIAL**

**CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES E
INFORMÁTICOS**

Tema:

“Sistema automatizado para el control de la fiabilidad de la información en la gestión de los campeonatos de fútbol en Liga Deportiva Cantonal Pelileo”

Trabajo de Graduación. Modalidad: TEMI. Trabajo Estructurado de Manera Independiente, presentado previo la obtención del título de Ingeniero en Sistemas Computacionales e Informáticos.

SUBLINEA DE INVESTIGACIÓN: Aplicaciones web y Orientación a Objetos

AUTOR: Moreno Moreno Byron Rodrigo

TUTOR: Ing. Galo López

Ambato – Ecuador

Enero - 2015

APROBACIÓN DEL TUTOR

En mi calidad de Tutor del trabajo de investigación sobre el tema: “**Sistema automatizado para el control de la fiabilidad de la información en la gestión de los campeonatos de fútbol en Liga Deportiva Cantonal Pelileo.**”, del señor Byron Rodrigo Moreno Moreno, estudiante de la Carrera de Ingeniería en Sistemas Computacionales e Informáticos de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, de la Universidad Técnica de Ambato, considero que el informe investigativo reúne los requisitos suficientes para que continúe con los trámites y consiguiente aprobación de conformidad con el Art. 16 del Capítulo II, del Reglamento de Graduación para obtener el título terminal de tercer nivel de la Universidad Técnica de Ambato.

Ambato enero 06, 2015

EL TUTOR

Ing. Galo López

AUTORÍA

Yo Byron Rodrigo Moreno Moreno, soy el responsable de la elaboración, desarrollo e investigación del presente trabajo de investigación titulado: **“Sistema automatizado para el control de la fiabilidad de la información en la gestión de los campeonatos de fútbol en Liga Deportiva Cantonal Pelileo.”**. En tal virtud declaro que el presente trabajo es original, auténtico y personal, el contenido, efectos legales y académicos que se desprenden del trabajo propuesto son de exclusiva responsabilidad del autor.

Ambato enero 06, 2015

EL AUTOR

Byron Rodrigo Moreno Moreno
C.C. 1803980844

APROBACIÓN DE LA COMISIÓN CALIFICADORA

La Comisión Calificadora del presente trabajo conformada por los señores docentes Ing. Renato Urvina e Ing. Patricio Gonzalez, revisó y aprobó el Informe Final del trabajo de graduación titulado **“Sistema automatizado para el control de la fiabilidad de la información en la gestión de los campeonatos de fútbol en Liga Deportiva Cantonal Pelileo.”**, presentado por el señor Byron Rodrigo Moreno Moreno de acuerdo al Art. 18 del Reglamento de Graduación para obtener el título Terminal de tercer nivel de la Universidad Técnica de Ambato.

Ing. Mg. Vicente Morales
PRESIDENTE DEL TRIBUNAL

Ing. Mg. Renato Urvina
DOCENTE CALIFICADOR

Ing. Mg. Patricio Gonzalez
DOCENTE CALIFICADOR

DEDICATORIA

El presente trabajo se lo dedico a mi madre, por su apoyo incondicional durante todas etapas de vida. Por qué me ha mostrado que en el camino hacia la meta se necesita de la dulce fortaleza para aceptar las derrotas y del sutil coraje para derribar los miedos.

A mi familia por siempre brindarme su apoyo, consejos, comprensión y su ayuda en los momentos difíciles.

*“La dicha de la vida consiste en tener siempre algo que hacer, alguien a quien amar y alguna cosa que esperar”. **Thomas Chalmers.***

AGRADECIMIENTO

Quiero extender mi más sincero y eterno agradecimiento a mi madre por darme la vida, por formarme como persona de bien, por inculcarme siempre buenos valores y por demostrarme que lo imposible no existe. A mi familia por su inmensurable apoyo, por brindarme siempre su respaldo de forma incondicional.

A la Facultad de Ingeniería en Sistemas, Electrónica e Industrial de la Universidad Técnica de Ambato por darme la oportunidad de formarme profesionalmente. Así como también a todos y cada uno de mis maestros quienes a lo largo de mi formación académica supieron transmitirme sus conocimientos y valores.

Especial agradecimiento a mi Director de Tesis el Ing. Galo López por sus consejos, ayuda y amistad.

Y a todos quienes de una u otra forma contribuyeron con mi formación académica, personal y profesional mi más sincero y eterno agradecimiento a cada uno de ellos.

*“Es la hora de partir, la dura y fría hora que la noche sujeta a todo horario”. **Pablo Neruda.***

ÍNDICE DE CONTENIDOS

| | |
|--|------|
| Páginas preliminares | |
| Portada | i |
| Aprobación del tutor | ii |
| Autoría | iii |
| Aprobación de la comisión calificadora | iv |
| Dedicatoria | v |
| Agradecimiento | vi |
| Índice de contenidos | vii |
| Índice de Tablas | xi |
| Índice de Figuras | xiii |
| Resumen ejecutivo | xv |
| Summary | xvii |
| Glosario de términos y acrónimos | xix |
| CAPÍTULO I | 3 |
| EL PROBLEMA | 3 |
| 1.1 Tema de investigación | 3 |
| 1.2 Planteamiento del problema | 3 |
| 1.2.1 Contextualización | 3 |
| 1.2.2 Análisis crítico | 5 |
| 1.2.3 Formulación del problema | 7 |
| 1.2.4 Delimitación del problema | 7 |
| 1.3 Justificación | 8 |
| 1.4 Objetivos | 9 |

| | |
|---|----|
| 1.4.1 Objetivo General | 9 |
| 1.4.2 Objetivos Específicos | 9 |
| CAPÍTULO II | 11 |
| MARCO TEÓRICO | 11 |
| 2.1. Marco Teórico | 11 |
| 2.1.1 Sistema automatizado | 12 |
| 2.1.2 Objetivos de la automatización | 13 |
| 2.1.3 Ventajas de los Sistemas Automatizados | 14 |
| 2.1.4 Desventajas de los Sistemas Automatizados | 14 |
| 2.1.5 Control de la fiabilidad de la información | 14 |
| 2.1.6 Medidas generales de protección de la información | 15 |
| 2.1.7 Protección física de la información | 16 |
| 2.1.8 Gestión de campeonatos de fútbol | 16 |
| 2.1.9 La metodología Microsoft Solution FrameWork | 17 |
| CAPÍTULO III | 19 |
| METODOLOGÍA | 19 |
| 3.1. Modalidad de la investigación | 19 |
| 3.2 Recolección de la información | 19 |
| 3.3 Procesamiento de análisis de datos | 20 |
| 3.4 Desarrollo del proyecto | 20 |
| CAPÍTULO IV | 22 |
| DESARROLLO DE LA PROPUESTA | 22 |
| 4 Metodología | 22 |
| 4.1 Visión | 23 |

| | | |
|--------|--|-----|
| 4.1.1 | Elaboración del Documento de Alcance y Estrategia | 23 |
| 4.1.2 | Formación del equipo de trabajo y distribución de competencias | 23 |
| 4.1.3 | Elaboración del plan de trabajo | 25 |
| 4.1.4 | Matriz de riesgos y plan de contingencia | 26 |
| 4.2 | Planificación | 27 |
| 4.2.1 | Planificación y diseño de la arquitectura | 27 |
| 4.2.2 | Construcción del plan de laboratorio | 28 |
| 4.2.3 | Estimación de costos | 29 |
| 4.3 | Desarrollo | 29 |
| 4.3.1 | Diseño de los diagramas de casos de uso | 29 |
| 4.3.2 | Diseño de los diagramas de secuencias | 53 |
| 4.3.3 | Diseño de los diagramas de flujo de datos | 57 |
| 4.3.4 | Diseño de los diagramas de estados | 58 |
| 4.3.5 | Diseño de la base de datos | 62 |
| 4.3.6 | Creación del modelo lógico de la base de datos | 63 |
| 4.3.7 | Diseño del modelo físico de la base de datos | 68 |
| 4.3.8 | Diseño de las interfaces de usuario | 71 |
| 4.3.9 | Creación de la capa de negocios en LightSwitch. | 76 |
| 4.3.10 | Desarrollo del sistema (Pantallas, Formas y el Menú) | 98 |
| 4.3.11 | Creación de reportes e informes que tendrá el sistema | 111 |
| 4.3.12 | Ingreso y documentación de los parámetros del sistema | 112 |
| 4.4 | Estabilización | 115 |
| 4.4.1 | Configuración del servidor web IIS en un ámbito de pruebas | 115 |
| 4.4.2 | Realización de las pruebas BETA | 118 |

| | |
|--|-----|
| 4.4.3 Configuración del servidor web IIS en ambiente de producción | 119 |
| 4.4.4 Gestionar las incidencias que presente el sistema | 120 |
| 4.4.5 Elaboración del plan de despliegue o implementación | 121 |
| 4.5 Despliegue o Implementación | 122 |
| 4.5.1 Liberación del sistema en producción | 122 |
| 4.5.2 Elaboración de guías y manuales del sistema | 122 |
| 4.5.3 Creación de un registro de mejoras y sugerencias | 122 |
| 4.5.4 Entrega del proyecto final | 123 |
| 4.6. Discusión y resultados | 123 |
| CAPÍTULO V | 125 |
| CONCLUSIONES Y RECOMENDACIONES | 125 |
| 5.1. Conclusiones | 125 |
| 5.2. Recomendaciones | 126 |
| BIBLIOGRAFÍA | 127 |
| Anexo 1: Especificación de requerimientos según estándar IEEE 830 | 131 |
| Anexo 2: Resumen de entrevista. | 149 |
| Anexo 3: Manual de usuario | 151 |
| Anexo 4: Manual técnico | 174 |
| Anexo 5: Acta de configuración. | 181 |

Índice de Tablas

| | |
|---|----|
| Tabla 1 Equipo de Trabajo | 24 |
| Tabla 2 Matriz de riesgos y plan contingencia | 27 |
| Tabla 3 Plan de laboratorio | 28 |
| Tabla 4 Caso de uso Administrar Ingresos al Sistema | 31 |
| Tabla 5 Caso de uso Administrar roles | 31 |
| Tabla 6 Caso de uso Administrar usuarios | 32 |
| Tabla 7 Caso de uso Administrar formularios del sistema. | 33 |
| Tabla 8 Caso de uso Gestionar Parámetros del sistema. | 33 |
| Tabla 9 Caso de uso Gestionar Categorías | 34 |
| Tabla 10 Caso de uso Gestionar Empleados | 35 |
| Tabla 11 Caso de uso Gestionar Comisiones | 35 |
| Tabla 12 Caso de uso Gestionar Directivos de Liga | 36 |
| Tabla 13 Caso de uso Gestionar Información Institucional | 37 |
| Tabla 14 Caso de uso Gestionar Estadios | 37 |
| Tabla 15 Caso de uso Gestionar Clubes | 39 |
| Tabla 16 Caso de uso Asignar jugadores propios | 40 |
| Tabla 17 Caso de uso Administrar Pases de Jugadores | 41 |
| Tabla 18 Caso de uso Gestionar Árbitros y Vocales | 42 |
| Tabla 19 Caso de uso Gestionar Campeonatos, Fases, Fechas y Grupos | 43 |
| Tabla 20 Caso de uso Asignar Equipos a un campeonato | 43 |
| Tabla 21 Caso de uso Asignar Jugadores a Equipos que están en un campeonato | 44 |
| Tabla 22 Caso de uso Gestionar Partidos | 45 |
| Tabla 23 Caso de uso Gestionar Sanciones y Multas | 46 |
| Tabla 24 Caso de uso Gestionar Directivos de Clubes | 46 |
| Tabla 25 Caso de uso Gestionar Equipos | 48 |
| Tabla 26 Caso de uso Gestionar Jugadores | 49 |
| Tabla 27 Caso de uso Gestionar Calendarios | 49 |
| Tabla 28 Caso de uso Gestionar Reportes | 50 |

| | |
|--|-----|
| Tabla 29 Caso de uso Carnetizar Jugadores y Generar Carnets | 51 |
| Tabla 30 Caso de uso Login al sistema | 52 |
| Tabla 31 Caso de uso consultar y generar reportes | 52 |
| Tabla 32 Módulos de la base de datos | 64 |
| Tabla 33 Parámetros del sistema | 113 |
| Tabla 34 Parámetros para tarjetas | 113 |
| Tabla 35 Tipos de sanción | 114 |
| Tabla 36 Listado de los estados con sus respectivos procesos | 115 |
| Tabla 37 Pruebas beta | 119 |
| Tabla 38 Incidencias del sistema | 120 |
| Tabla 39 Plan de despliegue e implementación | 121 |
| Tabla 40 Registro de mejoras y sugerencias | 123 |
| Tabla 41 Aspectos anteriores y posteriores a la implementación del sistema | 124 |

Índice de Figuras

| | |
|--|----|
| Fig.1 Árbol de problema | 5 |
| Fig.2 Modelo de la metodología MSF | 18 |
| Fig.3 Versiones de la metodología MSF | 18 |
| Fig.4 Ciclo de la metodología MSF | 22 |
| Fig.5 Plan de trabajo | 26 |
| Fig.6 Diagrama de caso de uso del rol Administrador | 30 |
| Fig.7 Diagrama de caso de uso rol Secretaría | 38 |
| Fig.8 Diagrama de caso de uso del rol Auxiliar de Secretaría | 47 |
| Fig. 9 Diagrama de caso de uso del rol Invitado | 51 |
| Fig. 10 Diagrama de secuencia Login Usuario | 53 |
| Fig. 11 Diagrama de secuencia Mantenimiento de equipo | 53 |
| Fig. 12 Diagrama de secuencia Mantenimiento de jugador | 54 |
| Fig. 13 Diagrama de secuencia Mantenimiento de calendario | 55 |
| Fig. 14 Diagrama de secuencia Mantenimiento de partido | 56 |
| Fig. 15 Diagrama de flujo de datos nivel | 57 |
| Fig. 16 Diagrama de estado crear equipo | 58 |
| Fig. 17 Diagrama de estado registro de jugador | 58 |
| Fig. 18 Diagrama de estado crear calendario | 59 |
| Fig. 19 Diagrama de estado nueva multa | 59 |
| Fig.20 Diagrama de estado nueva sanción | 60 |
| Fig. 21 Diagrama de estado creación de empleado | 60 |
| Fig. 22 Diagrama de estado creación de directivo | 61 |
| Fig. 23 Diagrama de estado creación de vocal | 61 |
| Fig.24 Modelo Lógico del Módulo General | 65 |
| Fig. 25 Modelo Lógico del Módulo Liga | 66 |
| Fig. 26 Modelo Lógico del Módulo de Seguridad | 67 |
| Fig. 27 Modelo Lógico del Módulo de Recursos Humanos | 67 |
| Fig. 28 Modelo Físico del Módulo General | 68 |
| Fig. 29 Modelo Físico del Módulo Liga | 69 |

| | |
|---|-----|
| Fig. 30 Modelo Físico del Módulo de Seguridad | 70 |
| Fig. 31 Modelo Físico del Módulo de Recursos Humanos | 70 |
| Fig. 32 Interfaz de Inicio de Sesión | 71 |
| Fig. 33 Interfaz de pantallas de consultas y manipulación de datos | 72 |
| Fig. 34 Interfaz área de trabajo o de manipulación de datos | 73 |
| Fig. 35 Interfaz de pantallas de consultas y manipulación de datos con filtros en cascada | 75 |
| Fig. 36 Interfaz de pantallas de reportes | 76 |
| Fig. 37 Capa de negocios en LightSwitch | 77 |
| Fig. 38 Listado de tablas en la capa de negocios en LightSwitch | 78 |
| Fig. 39 Página de inicio de sesión o página de login del sistema | 99 |
| Fig. 40 Página principal del sistema o página maestra | 100 |
| Fig. 41 Pantalla de inicio del sistema o accesos rápidos a procesos | 103 |
| Fig. 42 Pantalla de manipulación de datos simple | 104 |
| Fig. 43 Pantalla de manipulación de datos con filtros | 107 |
| Fig. 44 Pantallas con pestañas | 108 |
| Fig. 45 Pantalla de Ingreso o Modificación de datos | 109 |
| Fig. 46 Pantallas de visualización y generación de reportes | 111 |
| Fig. 47 Instalación de Internet Information Services | 116 |
| Fig. 48 Pantalla de Administración de Internet Information Service | 116 |
| Fig. 49 Convertir en una aplicación web el aplicativo | 117 |
| Fig. 50 Configuración del servidor web IIS | 119 |

RESUMEN EJECUTIVO

La Liga Deportiva Cantonal Pelileo (LDCP) y el comité de fútbol perteneciente a la misma, son los entes que se encargan año tras año de organizar el torneo de fútbol, con el propósito de integrar a la comunidad pelileña, promover la práctica del deporte, mejorar e incentivar el desarrollo físico y mental del ser humano.

La demanda de equipos y jugadores que desean participar en el torneo de fútbol va en incremento año tras año, esto genera algunos problemas como retraso en el inicio de los campeonatos de fútbol, la información de las cartas pase de los jugadores no es fiable. Todo esto acompañado de insuficientes recursos tecnológicos genera que tanto jugadores y equipos no cumplan a cabalidad con los requisitos y exigencias que establece la LDCP.

La presente investigación propone el desarrollo de un sistema automatizado para el control de la fiabilidad de la información en la gestión de los campeonatos de fútbol en la LDCP, el mismo que permitirá administrar, gestionar y controlar de mejor manera la información de los campeonatos de fútbol, así como también agilizar los procesos y mejorar el servicio a la ciudadanía.

Mediante técnicas de recolección de información se ha realizado el levantamiento de requerimientos, los cuales posteriormente han sido transformados en diagramas informáticos para facilitar la interpretación de los procesos. La etapa del desarrollo de la propuesta sigue la metodología Microsoft Solution Framework (MSF), la cual está enfocada a dirigir proyectos o soluciones de innovación.

En base a diagramas de Lenguaje Unificado de Modelado (UML), Diagramas de flujo de datos (DFD) y Diagramas entidad relación, conjuntamente con la metodología (MSF) se plantea el diseño del sistema así como los requisitos necesarios para su implementación y funcionamiento. De la misma forma se han seleccionado las herramientas para el desarrollo del mismo.

Una vez implementada la aplicación se ha hecho un cuadro comparativo, que permite visualizar los resultados y diferencias entre un estado anterior y posterior a la implementación del sistema.

SUMARY

The “Liga Deportiva Cantonal Pelileo (LDPC)” and football committee of it are the entities that are responsible year after year to organize the football championship, in order to integrate the “pelileña” community, promote the practice of sport, enhance and encourage physical and mental development of human beings.

Demand of teams and players who want to participate in the football championship continues to increase year after year. This creates some problems such as delay in starting football championships, the information about "cartas pase" of players are unreliable. All this accompanied with insufficient technological resources generate that players and teams do not comply fully with the conditions and requirements established by the LDPC.

This research proposes the development of an automated system for checking the reliability of information in the management of football championships in the LDPC, the same permitted to administer, manage and control better the information of the football championships, well as streamline the processes and improve service to the public.

Using techniques of data collection was performed requirements gathering which have subsequently been transformed into computer diagrams to facilitate the processes interpretation. The stage of development of the proposal follows the methodology Microsoft Solution Framework (MSF), which is focused on managing projects or innovative solutions.

Based on diagrams of Unified Modeling Language (UML), data flow diagrams (DFD) and entity relationship diagrams, together with the methodology (MSF) arises system design and the requirements for their implementation and operation. Just as the tools have been selected for development to it.

Once implemented the application has made a comparative table, that displays the results and differences between a status before and after of implementation system.

GLOSARIO DE TÉRMINOS Y ACRÓNIMOS

FRAMEWORK: Es una estructura de soporte definida, sobre la cual un proyecto de software puede ser desarrollada.

GUI: (Graphical User Interface o Interfaz Gráfica de Usuario), es un conjunto de formas y métodos los cuales posibilitan la interacción de un sistema informático con los usuarios, esto a través de gráficas e imágenes.

IEEE: (Institute of Electrical and Electronics Engineers o Instituto de Ingeniería Eléctrica y Electrónica), es una organización mundial de técnicos e ingenieros la cual se dedica a la estandarización de procesos.

LDCP: (Liga Deportiva Cantonal Pelileo), ente organizador de torneos deportivos en la ciudad de Pelileo.

MSF: (Microsoft Solution FrameWork) metodología ágil enfocada a dirigir proyectos o soluciones de innovación.

UML: (Unified Modeling Language o Lenguaje Unificado de Modelado), es un lenguaje gráfico el cual sirve para visualizar, construir y documentar un sistema.

INTRODUCCIÓN

El presente trabajo de tesis cuyo tema es, “**Sistema automatizado para el control de la fiabilidad de la información en la gestión de los campeonatos de fútbol en Liga Deportiva Cantonal Pelileo**“, el cual consta de cinco capítulos los cuales se detallan a continuación.

Capítulo I. “El problema”, aquí se define el problema que está aconteciendo en el contexto de la realidad, para lo cual delimitamos sus alcances, justificamos debidamente el problema y planteamos los objetivos, los cuales serán la guía de todo el proyecto.

Capítulo II. “Marco Teórico”, en este capítulo consta el fundamento teórico el cual ayuda a comprender de una forma clara el problema, para posteriormente plantear una propuesta de solución al mismo.

Capítulo III. “Metodología”, se describe la metodología y las modalidades de investigación a seguir para conseguir el resultado inicialmente propuesto, así como también la forma de cómo se recolectará y se procesará la información, además de una descripción global de cómo se desarrolla el proyecto.

Capítulo IV. “Desarrollo de la Propuesta”, en este capítulo se describe el desarrollo de la propuesta de solución, definiendo los requisitos necesarios que se han obtenido del proceso de levantamiento de información y requisitos, documentando el proceso de desarrollo del software a través de casos de uso, diagramas de estado y flujo de datos, diseño de la base de datos, diseño de la interfaz gráfica de usuario, estabilización de la propuesta de solución, así como también el despliegue o implementación de la solución final.

Capítulo V. “Conclusiones y recomendaciones”, se establecen las conclusiones a las que ha llegado el investigador luego del desarrollo del proyecto, de la misma forma se hacen las recomendaciones que el investigador a considerado necesarias.

Finalmente se incluye las referencias citadas en este documento, en los anexos se incluye los instrumentos utilizados para la recolección de la información así como también se incluyen los manuales correspondientes del presente proyecto.

CAPÍTULO I

EL PROBLEMA

1.1 Tema de investigación

Sistema automatizado para el control de la fiabilidad de la información en la gestión de los campeonatos de fútbol en Liga Deportiva Cantonal Pelileo.

1.2 Planteamiento del problema

1.2.1 Contextualización

La Federación Ecuatoriana de Fútbol (F.E.F.) máximo organismo regulador y controlador de los campeonatos de fútbol a nivel nacional, es la única entidad que cuenta con un sistema informático para la gestión, control y manejo de los campeonatos de fútbol. El Sistema Integrado Federación Ecuatoriana de Fútbol SIFEF, fue desarrollado por el departamento de sistemas de federación. Este sistema está disponible pero solo para Asociaciones Deportivas Provinciales, más no para ligas deportivas cantonales, parroquiales y barriales. Este sistema es genérico y no se ajustaría al 100% a las necesidades que tienen las ligas deportivas.

En la en nuestro país gran parte de ligas deportivas, ya sean estas cantonales, parroquiales o barriales no cuentan con un sistema informático que permita controlar la fiabilidad de la información que presentan los equipos de sus jugadores y gestionar de una forma eficiente

los campeonatos de fútbol, esto es lo que muchos de los dirigentes deportivos han manifestado en las diferentes entrevistas y encuestas que se les ha realizado, es decir no pueden controlar de forma manual que se cumplan a cabalidad las leyes y reglamentos que rigen para estas instituciones deportivas. Además no cuentan con una red de datos, todo esto se genera debido al poco presupuesto con el que cuentan estas instituciones y por otra parte también por la falta de autogestión de dirigentes y directivos de las ligas deportivas del país.

Dentro de la provincia de Tungurahua el organismo que cuenta con sistema informático para la administración y de cierta forma para la gestión de las diferentes disciplinas deportivas es la Federación Deportiva de Tungurahua. Pero la F. D. T. no maneja campeonatos de fútbol en la medida en la que se manejan en una liga deportiva cantonal o parroquial por lo que no tienen mayores complicaciones con este tema. Además la F. D. T. tiene acceso al SIFEF por ser un organismo agremiado y que forma parte de la F. E. F.

Algunas de las ligas deportivas provinciales, cantonales y parroquiales, como la Asociación Provincial de Ligas Deportivas Parroquiales y Barriales de Tungurahua, la Liga Deportiva Parroquial de Ambatillo, cuentan con pequeños sistemas informáticos, con los cuales pueden carnetizar a los jugadores y realizar algunas cosas sencillas. Pero estos sistemas no tienen la capacidad de gestionar, administrar y controlar grandes volúmenes de información de que se generan en los grandes campeonatos de fútbol, como los que se generan en una liga deportiva cantonal.

La Liga Deportiva Cantonal Pelileo al momento al momento de realizar la presente investigación no cuenta con ningún sistema informático o automatizado que le ayude a controlar la fiabilidad de la información que se maneja y se genera producto de los campeonatos de fútbol en sus diferentes categorías. Así como también a gestionar la información de los campeonatos. Lo cual conlleva problemas que la institución viene atravesando desde ya hace algunos años atrás, y los problemas van en aumento a medida que pasa el tiempo y los campeonatos de fútbol en sus diferentes categorías cada vez tiene

un mayor número de participantes. Algunos de estos problemas son: Jugadores que han sido sancionados en campeonatos anteriores juegan sin haber terminado de cumplir su sanción. Los jugadores se cambian la fotografía del carnet. La información de las cartas pase de los jugadores no está disponible en el momento oportuno. No existen medidas de seguridad necesarias para el control de la información proporcionada por los clubes de sus jugadores. Jugadores que son inscritos en otro club diferente del club dueños de sus derechos deportivos sin la previa autorización del mismo. La institución cuenta con equipos de cómputo y acceso a internet con una conexión telefónica (ADSL).

1.2.2 Análisis crítico

Árbol del problema

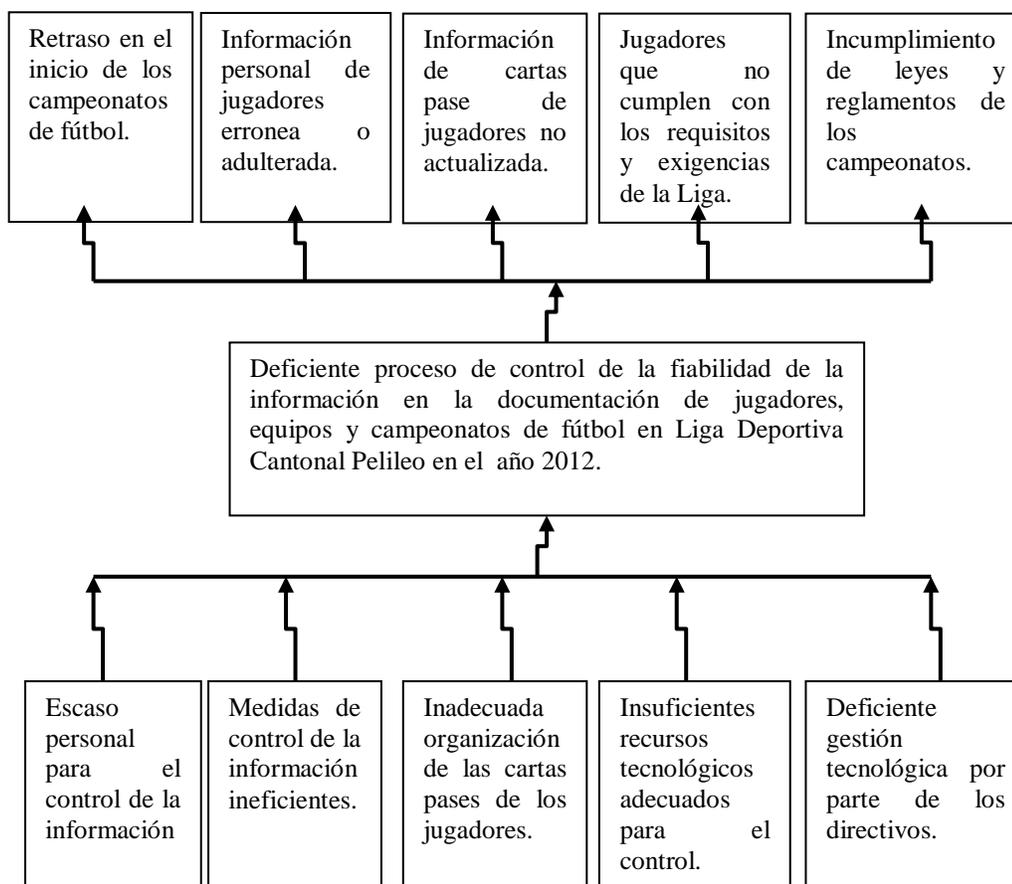


Fig.1: Árbol de problema
Elaborado por: El Investigador.

El escaso personal con el que cuenta la Liga Deportiva Cantonal Pelileo para realizar el control de la información tanto de jugadores, equipos, cartas pases de jugadores y campeonatos de fútbol, trae como consecuencias en varios casos la manipulación de la información personal de los jugadores, es decir: para la inscripción de un equipo en un campeonato de fútbol, la institución pide una copia de la cédula de cada uno de los integrantes de un equipo, entonces algunos equipos por inscribir a jugadores que no cumplen con los reglamentos estipulados para ese campeonato, manipulan la información de las copias de las cédulas, modificando el lugar, la fecha de nacimiento o también la foto del jugador, la información de las copias de las cédulas no concuerdan con la información del original de la cédula del jugador.

Las medidas o filtros que se aplican para el proceso de control de la información de los datos de los jugadores son ineficientes. Ya que en primer lugar se lo hace de forma manual y no existe el suficiente personal para validar la información proporcionada por los equipos de sus jugadores. Por otro lado como se piden solo las copias de las cédulas de los jugadores, estas pueden ser manipuladas. El personal encargado de la verificación de la información no se da abasto para revisar detenidamente la misma, eso debido a la gran cantidad de información que tiene que revisar, también provoca lentitud en los procesos y retrasos de inicios de los campeonatos de fútbol. Una vez emitido los carnets de los jugadores algunos equipos cambian las fotografías originales con las de otros jugadores, esto según versiones de los mismos funcionarios de la LDGP.

Otro de los problemas que viene atravesando Liga Deportiva Cantonal Pelileo es una inadecuada organización de las cartas pases y de las fichas de los jugadores, esto genera que la información de las cartas pase de los jugadores no este actualizada y disponible en el momento oportuno, ya que jugadores que pertenecen a un club juegan en otro u otros clubes sin la debida autorización del club dueño de sus derechos deportivos. Esto debido a la inadecuada organización de las fichas de los jugadores y debido a que esto se lo hace de forma manual.

Los insuficientes recursos tecnológicos adecuados para el control de la fiabilidad de la información y la gestión de los campeonatos de fútbol genera que los jugadores y equipos no cumplan a cabalidad con los requisitos y exigencias de Liga Deportiva Cantonal Pelileo, esto también genera malestar en los otros equipos que si cumplen con los requisitos y exigencias de la institución deportiva.

Una deficiente gestión a nivel tecnológico por parte de los directivos para solucionar estos problemas ha generado el incumplimiento de leyes y reglamentos que rigen dentro de la institución para la organización de los campeonatos de fútbol tanto jugadores como clubes han incumplido las mismas, la falta de interés por parte de algunos directivos ha impedido que se automatice todo el proceso que tiene que ver la administración de los campeonatos de fútbol, estas fueron las manifestaciones de algunos directivos de clubes y otras personas entendidas en la materia.

1.2.3 Formulación del problema

¿De qué forma incide un sistema automatizado en el control de la fiabilidad de la información y la gestión de los campeonatos de fútbol en Liga Deportiva Cantonal Pelileo?

1.2.4 Delimitación del problema

El presente proyecto se realizará en Liga Deportiva Cantonal Pelileo, ubicada en la ciudad de Pelileo en la Av. 22 de Julio y Celiano Monge. Este proyecto está enmarcado dentro del área académica de Software, línea de investigación Desarrollo de Software, sublínea Aplicaciones web y Orientación a Objetos.

La realización del proyecto está planificada para un período de seis meses a partir de la fecha de aprobación del perfil.

Área Académica: Software.

Línea de Investigación: Desarrollo de Software.

Sublínea: Aplicaciones web y Orientación a Objetos.

1.3 Justificación

Los problemas que vienen atravesando gran parte de las Ligas Deportivas ya sean estas Cantonales, Parroquiales o Barriales en la provincia del Tungurahua y el impacto que tiene dentro del ámbito deportivo especialmente del fútbol, han motivado su investigación, con el fin de determinar los factores que causan estos problemas y a través del uso y la aplicación de nuevas tecnologías, solucionar o disminuir estos problemas.

El desarrollo del presente proyecto es de gran importancia ya que agilizará y controlará de una manera eficiente y automatizada los procesos de control de la fiabilidad de información de jugadores, equipos y campeonatos, así como también se llevará una mejor gestión de la información de los campeonatos de fútbol, brindará información veraz, fiable y oportuna en tiempo real a los usuarios, permitiendo que los directivos tengan información de forma oportuna para una mejor toma de decisiones. Y finalmente a través del uso de nuevas tecnologías se tendrá una excelente organización de los datos y de la información que se maneja de los campeonatos de fútbol.

Al desarrollarse el sistema para automatizar el proceso de control de la fiabilidad de la información, se utilizará otro tipo de lenguaje de consulta como es LINQ para evitar la inyección de SQL y mejorar la seguridad y la velocidad de las consultas a la base de datos. Se creará un módulo para la gestión automática de las cartas pases de los jugadores. También se podría realizar un sistema de carnetización automática, se aplicará auditorías a las partes críticas del sistema, y se gestionará todo el sistema mediante roles y usuarios. Todo esto brindaría un alto grado de confianza y seguridad de la información que se maneja en la organización.

El desarrollo y la implementación del presente proyecto de investigación beneficiaría a todos los que forman parte, ya sea de forma directa o indirecta de Liga Deportiva Cantonal Pelileo y están relacionados con los campeonatos de fútbol es decir a futbolistas, equipos, directivos de los equipos, directivos de la institución y empleados administrativos de la misma, ayudaría también a mejorar la efectividad operativa, reduciría costo, aumentaría la satisfacción de sus usuarios. Contribuiría con el medio ambiente en la reducción del uso del papel.

El impacto del desarrollo de este proyecto será para los equipos y jugadores que no cumplen con los reglamentos establecidos por la institución, los cuales de encontrarse con algún tipo de anomalías en su documentación podrían ser sancionados por la institución.

El presente proyecto es factible de realizar ya que se puede obtener la información necesaria para el tema de desarrollo, se cuenta con el apoyo y la colaboración de todo el personal de la institución beneficiaria los cuales brindaran toda la información del problema, también se cuenta con el apoyo personal técnico para el asesoramiento del presente trabajo, esto conlleva a que el proyecto se desarrolle sin ningún problema.

1.4 Objetivos

1.4.1 Objetivo General

Implantar un sistema automatizado para el control de la fiabilidad de la información en la gestión de los campeonatos de fútbol en Liga Deportiva Cantonal Pelileo.

1.4.2 Objetivos Específicos

- Analizar el proceso de control de la fiabilidad de la información de jugadores, equipos y campeonatos de fútbol en Liga Deportiva Cantonal Pelileo.

- Establecer los requerimientos y alcances del sistema para el control de la fiabilidad de la información en la gestión de los campeonatos de fútbol en Liga Deportiva Cantonal Pelileo.
- Diseñar una base de datos con seguridades y restricciones de acceso para almacenar la información.
- Desarrollar el sistema de control de la fiabilidad de la información utilizando la Metodología Ágil MSF (Microsoft Solution Framework).

CAPÍTULO II

MARCO TEÓRICO

2.1. Marco Teórico

Los sistemas son un conjunto de partes integradas, las cuales cumplen una función específica y la unión o integración de cada una de estas partes permite alcanzar un objetivo específico.

Las características que reúne un sistema son:

1. Conjunto de partes.-Se denomina así, puesto que se considera que un sistema tiene más de un elemento, del cual están compuestos.
2. Partes integradas.- Existe una relación lógica entre las partes que constituyen un sistema, los sistemas para el control de la fiabilidad de la información consisten en procedimientos integrados para administrar, gestionar y controlar que el flujo de la información que se genera, para que esta tenga consistencia.
3. Propósito común para alcanzar determinado objetivo.- Todo sistema existe para alcanzar uno o más objetivos, y sus partes integrantes deben ajustarse entre sí para lograr el objetivo global del sistema. Un sistema existirá siempre y cuando exista un objetivo que este tenga que cumplir.

Los resultados que genere un sistema deben ser mayores que la suma de sus partes que lo componen, el efecto que produce entre la interrelación de las partes, es un efecto multiplicador denominado Sinergia, en cambio si las partes no están correctamente interrelacionadas puede entrar en un estado de descomposición y desintegración llamado entropía, si las partes de un sistema se interrelacionan correctamente el sistema está cumpliendo con su objetivo para el cual fue creado o desarrollado y esto tiene un gran impacto a la sociedad ya que el sistema estaría ayudando a resolver los problemas.

Los sistemas pueden estar compuestos por módulos, también llamados subsistemas, los cuales pueden realizar un determinado proceso o tarea, esto permite tener un mejor control de los procesos.

Un sistema está compuesto por varios componentes los cuales permiten su funcionamiento, entre estos tenemos. Las entradas, es todo lo que ingresa al sistema para que este pueda funcionar, un ejemplo de esto puede ser los parámetros que tiene un sistema. Las operaciones son todos los procesos que realiza el sistema ya sea de forma automatizada o con la interacción de las personas que los manejan. Las salidas son el resultado de las operaciones de los procesos que realiza o la visualización de la información que en el sistema se encuentra. La retroalimentación es uno de los ingredientes fundamentales dentro de cualquier proceso de control, esta ofrece las decisiones que ajustan al sistemas [1].

2.1.1 Sistema automatizado

La automatización de un sistema es transferir tareas de producción, control, administración, gestión, etc. Realizadas habitualmente de forma manual por las personas a un conjunto de elementos tecnológicos que realicen este trabajo de forma automática siguiendo un conjunto de instrucciones lógicas [2].

Un sistema automatizado está formado o consta de dos partes principales:

- Parte de mando
- Parte operativa

La parte de mando suele ser autónoma programable (Tecnología programada), aquí se programan a través de algoritmos, las tareas que realizará el sistema para el cumplimiento del objetivo para el cual fue creado.

La parte operativa es aquella que actúa directamente sobre la máquina, está compuesto por varios elementos que hacen que la máquina se mueva y realice la operación o la acción deseada. En el caso de la automatización de un sistema informático la parte que realiza este trabajo viene a ser el sistema operativo el cual gestiona y controla todos los recursos y procesos.

2.1.2 Objetivos de la automatización

- Mejorar la productividad de la empresa, reduciendo los costes de producción y mejorando la calidad de la misma.
- Incrementar la seguridad y las condiciones de trabajo del personal.
- Realizar las operaciones imposibles de controlar manualmente.
- Simplificar los procesos para los trabajadores.
- Integrar y gestionar de forma eficiente los recursos tecnológicos de la empresa [3].

Los Sistemas Automatizados son sistemas de información hechos por el hombre. Aunque hay diferentes tipos de sistemas automatizados, todos tienden a tener componentes en común (hardware, software, personas, datos y procedimientos).

2.1.3 Ventajas de los Sistemas Automatizados

La automatización de un proceso de cualquier índole frente al control manual del mismo proceso, brinda ciertas ventajas y beneficios de orden económico, social, tecnológico, pudiéndose resaltar las siguientes:

- Asegura una mejora en la calidad del trabajo del operador y en el desarrollo del proceso, esto dependerá de la eficiencia del sistema implementado.
- Reducción de costos, puesto que se racionaliza el trabajo, se reduce tiempo y dinero dedicado al mantenimiento.
- Reducción de tiempos en el procesamiento de la información.
- Flexibilidad para adaptarse a nuevos productos.
- Incremento de la seguridad y disponibilidad de los datos de los datos.

2.1.4 Desventajas de los Sistemas Automatizados

Algunas de las desventajas que pueden generar los sistemas automatizados son muy pocas en referencia a las ventajas que estos tienen:

- Incremento en la dependencia del mantenimiento y reparación.
- Decremento en la flexibilidad del sistema [4].

2.1.5 Control de la fiabilidad de la información

Uno de los principales problemas con los cuales las organizaciones se encuentran hoy en día es el control de la información en todos los ciclos de vida que esta tiene. Ya sea que esta información sea generada de forma manual o automatizada, el control de la información consiste en varias instancias que van desde la validación de la veracidad de información,

pasando por el control de acceso a la misma por parte de los entes consumidores de esta información [5].

El control de la información no solo implica tener restringido el acceso a la misma, sino que también implica validar esa información, es decir que la información con la que contamos sea autentica y no haya sido alterada.

2.1.6 Medidas generales de protección de la información

Dentro de la seguridad de la información el principal objetivo es la protección de los datos en sí, tratando de evitar la pérdida y la modificación no autorizada de los mismos [6].

Los motivos por los cuales nace la implementación de normas o de medidas generales para la protección de la información, corresponden a la seguridad de la información así también al propio interés de la institución o de las personas que majan los datos, debido a que si esta información se pierde o es adultera puede causar daños ya sean estos materiales, económicos, sociales entre otros, los cuales afectarían directamente a la persona o a la institución dueña de esa información.

De esta manera, las medidas de seguridad en los niveles Básico y Medio contemplan las siguientes acciones:

1. Descripción de las rutinas de control de datos de los programas que se utilicen para ingresarlos en las bases.
2. Inclusión de acciones a tomar en el caso de errores detectados en los datos o en la documentación.
3. Copias de respaldo de la información.

4. Designación de un responsable de adopción y control de cumplimiento de medidas.
5. Identificación del personal que accede a la información.
6. Procedimientos de autenticación de usuario y control de acceso.
7. Adoptar medidas de prevención a efectos de impedir amenazas por la intromisión de software malicioso que pudiese afectar archivos con datos.
8. Adoptar procedimientos que garanticen una adecuada gestión de los soportes que contengan datos personales.
9. Realización de auditorías que tengan como objeto el control del cumplimiento de las medidas y de los principios de finalidad, integridad, veracidad de los datos personales [7].

2.1.7 Protección física de la información

Las personas encargadas del manejo de la información deben adaptar y promover la implementación de medidas de seguridad para que los documentos o dispositivos que contengan información (papeles, archivos, recursos compartidos de red, cd, un sistema informático o cualquier otro tipo) [8].

2.1.8 Gestión de campeonatos de fútbol

La oferta del deporte aficionado y semiprofesional suele ser muy amplia en Liga Deportiva Cantonal Pelileo, y cada vez va en aumento y con esto la gestión de los diferentes campeonatos de fútbol se complica por el gran volumen de datos que ahí se maneja.

Al organizar una competición de fútbol hay que diferenciarse de las demás ligas o instituciones que organizar estas competiciones mediante una propuesta de calidad y originalidad, que vaya más allá de una simple oferta de un torneo de juego, un balón y el control de un calendario y una tabla de posiciones. Gestionar un torneo de este tipo que genere ingresos para la institución y satisfacción en los participantes implica ocuparse muchos aspectos derivados de cada partido, de cada jugador o usuario y de los diferentes servicios paralelos que se pueda ofertar [9].

Para conseguir competitividad en cada torneo es necesario que se garantice la transparencia y la fiabilidad de la información que se maneja, es decir que dicha información sea verídica, que se evite la manipulación de los datos ya que esto crea un habiente de desconfianza por parte de los participantes en los campeonatos de fútbol. Un buen control de la fiabilidad de la información generada en los campeonatos de fútbol, así como también una buena gestión, administración y organización de los mismos, ayudados por las tecnologías de la información que actualmente existen, permite generar un alto grado de satisfacción y confiabilidad en la institución que organiza los campeonatos de fútbol.

2.1.9 La metodología Microsoft Solution Framework

La metodología Microsoft Solution Framework (MSF) es una herramienta que nos ayuda a llevar a cabo planes y acciones creados y definidos en los procesos administrativos. Ha sido diseñada para que las organizaciones obtengan soluciones tecnológicas de alta calidad, puntuales y según el presupuesto establecido [16].

Esta metodología se centra en la administración y gestión de los proyectos para lograr el impacto deseado. Se enfoca en dirigir proyectos o soluciones innovadoras.

Las características con las que cuenta esta metodología son:

- Adaptable

- Escalable
- Flexible

La metodología MSF consta de 5 fases:

- Visión
- Planificación
- Desarrollo
- Estabilización
- Despliegue o Implementación

Modelo de la metodología MSF

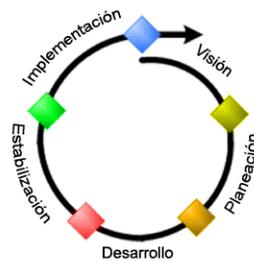


Fig.2: Modelo de la metodología MSF

Fuente: La Bitácora de AudieMan(Internet; 23-03-2011; 15-11-2013; 14:50h).

Con esta metodología se puede llevar a cabo un ciclo de forma interactiva, de manera que cuando liberemos una solución podemos iniciar nuevamente con la metodología para darle mucho más funcionalidad, en nuestro caso al software [10].

Versiones de la metodología MSF.

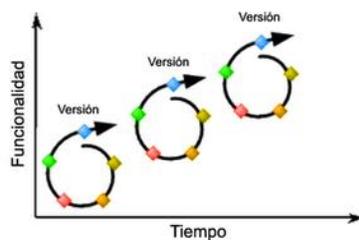


Fig.3: Versiones de la metodología MSF

Fuente: La Bitácora de AudieMan(Internet; 23-03-2011; 15-11-2013; 14:50h).

CAPÍTULO III

METODOLOGÍA

3.1. Modalidad de la investigación

La presente investigación es un proyecto de desarrollo, ya que se va a implantar un sistema para el control de la fiabilidad de la información.

Las modalidades que se utilizarán para el desarrollo de la presente investigación son:

Investigación de campo.- Porque el investigador acudirá al lugar de los hechos, como también se aplicarán entrevistas y la observación en el lugar mismo donde ocurren los hechos, es decir, Liga Deportiva Cantonal Pelileo.

Investigación Documental.- El investigador se sustenta en libros, revistas, periódicos, tesis y el internet como fuentes de información para recolectar y comparar esa información con el problema a investigarse.

Se llegará al nivel de investigación explicativa.

3.2 Recolección de la información

Se recolectará información para el levantamiento de requerimientos para la propuesta.

Se recolectará información a través de libros e internet.

3.3 Procesamiento de análisis de datos

Para la recolección, procesamiento y análisis de la información se aplicará los siguientes procedimientos:

1. Elaboración y validación del instrumento de relación de datos.
2. Organización de la información.
3. Elaboración y validación del documento Alcance y Estrategia
4. Análisis cualitativo y cuantitativo de la información.
5. Informe de resultados.

3.4 Desarrollo del proyecto

- 1 Visión.
 - 1.1 Elaboración del Documento de Alcance y Estrategia.
 - 1.2 Formación del Equipo de Trabajo y distribución de competencias.
 - 1.3 Elaboración del plan de trabajo.
 - 1.4 Matriz de riesgos y plan de contingencia.
- 2 Planificación.
 - 2.1 Planificación y diseño de la arquitectura.
 - 2.2 Plan de laboratorio.
 - 2.3 Estimación de costos.
- 3 Desarrollo.
 - 3.1 Diseño de los diagramas de casos de uso.
 - 3.2 Diseño de los diagramas de secuencias.
 - 3.3 Diseño de los diagramas de flujo de datos.

- 3.4 Diseño de los diagramas de estados.
- 3.5 Diseño de la base de datos.
- 3.6 Creación del modelo lógico de la base de datos.
- 3.7 Diseño del modelo físico de la base de datos.
- 3.8 Diseño de las interfaces de usuario.
- 3.9 Creación de la capa de negocio en LightSwitch.
- 3.10 Desarrollo del sistema (Pantallas, Formas y Menú).
- 3.11 Creación de reportes e informes que tendrá el sistema.
- 3.12 Ingreso y documentación de los parámetros del sistema.
- 4 Estabilización.
 - 4.1 Configuración del servidor web IIS en un ámbito de pruebas.
 - 4.2 Realización de las pruebas BETA.
 - 4.3 Configuración del servidor web IIS en ambiente de producción.
 - 4.4 Gestionar las incidencias que presente el sistema.
 - 4.5 Elaboración del plan de despliegue o implementación.
- 5 Despliegue o Implementación.
 - 5.1 Liberación del sistema en producción.
 - 5.2 Elaboración de guías y manuales del sistema.
 - 5.3 Creación de un registro de mejoras y sugerencias.
 - 5.4 Entrega del proyecto final.

CAPÍTULO IV

DESARROLLO DE LA PROPUESTA

4 Metodología

Para el desarrollo de un proyecto, es necesario utilizar algún tipo de metodología, que nos permita llevar todos los procesos de una forma ágil y ordenada. Para el presente proyecto utilizaremos la metodología ágil MSF (Microsoft Solution Framework) esta metodología fue desarrollada por Microsoft y toma las ventajas de los modelos en cascada y espiral, esto con el fin de solucionar los problemas que presentan estas metodologías dando como resultado una metodología ágil para el desarrollo de proyecto.

El modelo del proceso de MSF en el cual se incluye al soporte como la última fase del ciclo que en este trabajo de investigación no se lo considera, forma un triángulo en el que todas las fases están conectadas entre sí [11].

Ciclo de la metodología MSF.



Fig.4: Ciclo de la metodología MSF

Fuente: Felipe Lozano Leon(slideshare) (Internet; 11-07-2012; 17-11-2013; 19:00h).

4.1 Visión

En esta fase se debe realizar un estudio de lo que pretendemos que en el futuro haga nuestra aplicación o nuestro proyecto, para ello debemos realizar un documento de estrategia y alcance donde deben quedar pactadas las necesidades de funcionalidad y servicio con las cuales debe contar la solución. Debemos crear los equipos de trabajo junto con el plan de trabajo, para asegurar el éxito del proyecto es importante tener en cuenta el análisis de riesgos y plan de contingencia [12].

4.1.1 Elaboración del Documento de Alcance y Estrategia

Para la elaboración del documento de alcance, es decir del levantamiento de la información y la especificación de los requerimientos, se utilizara el estándar de la IEEE 830 [13].

Especificación de requerimientos según el estándar IEEE 830.

El desarrollo de este proyecto cumplirá con cada uno de los requerimientos y especificaciones establecidas en el documento, todo lo que no conste en el documento de levantamiento y especificación de requerimiento no será realizado o implementado en este proyecto el documento de requerimientos lo podemos encontrar en el **Anexo 1**.

4.1.2 Formación del equipo de trabajo y distribución de competencias

El equipo de trabajo para el desarrollo de este proyecto estará compuesto únicamente por el investigador y el tutor que será la persona encarga de la supervisión del desarrollo del proyecto de investigación, y las personas externas al proyecto serán la secretaria de LDCP [10].

Por lo tanto la distribución de trabajo, competencias y las responsabilidades será de la siguiente manera:

Investigador: Sera el responsable del desarrollo del proyecto en todas sus etapas, la realización de las pruebas correspondientes, hasta la implementación final y definitiva del mismo.

Tutor: Sera la persona encargada de la supervisión del desarrollo del proyecto así como también verificara el cumplimiento de todo lo planteado y establecido en el presente proyecto.

Secretaria: La secretaria de LDCP será quien colabore con las pruebas del sistema, quien reportara al investigador los errores y vulnerabilidades que presente el sistema una vez que sea liberado la versión beta así como también cuando se libere la versión de pruebas en ambiente real, hasta la liberación de la versión final en ambiente de producción.

Formación de Equipo de Trabajo.

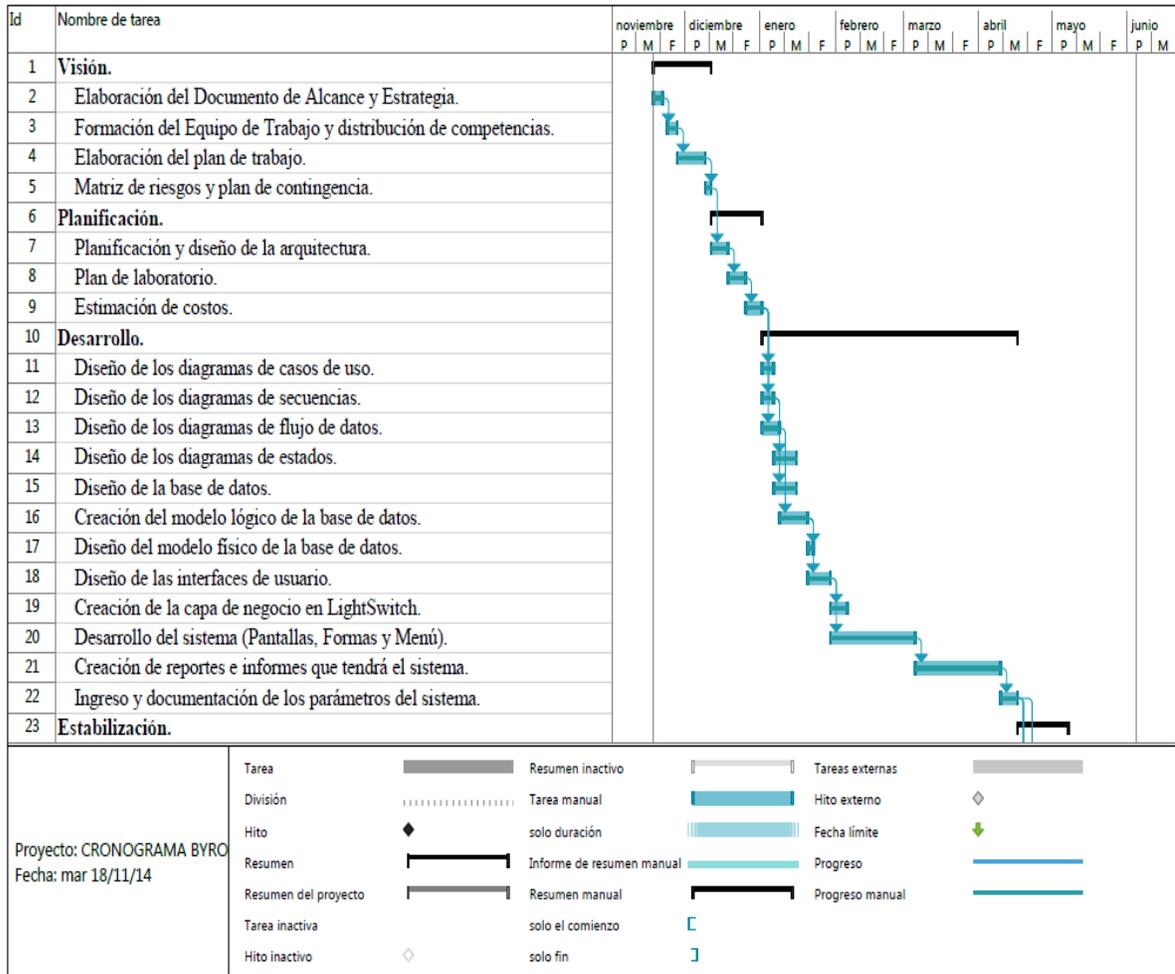
| EQUIPO DE TRABAJO | RESPONSABILIDAD |
|-------------------|--|
| Investigador | El desarrollo del proyecto en todas sus etapas, realización de las pruebas, corrección de errores y la implementación final. |
| Tutor | Supervisar el desarrollo del proyecto y el cumplimiento de los objetivos. |
| Secretaria | Realización de pruebas, detección de errores y vulnerabilidades del sistema. |

Tabla 1: Equipo de Trabajo
Elaborador por: El Investigador.

4.1.3 Elaboración del plan de trabajo

El plan de trabajo consta de las siguientes actividades, y el responsable directo de cada una de las mismas es el investigador.

Plan de trabajo



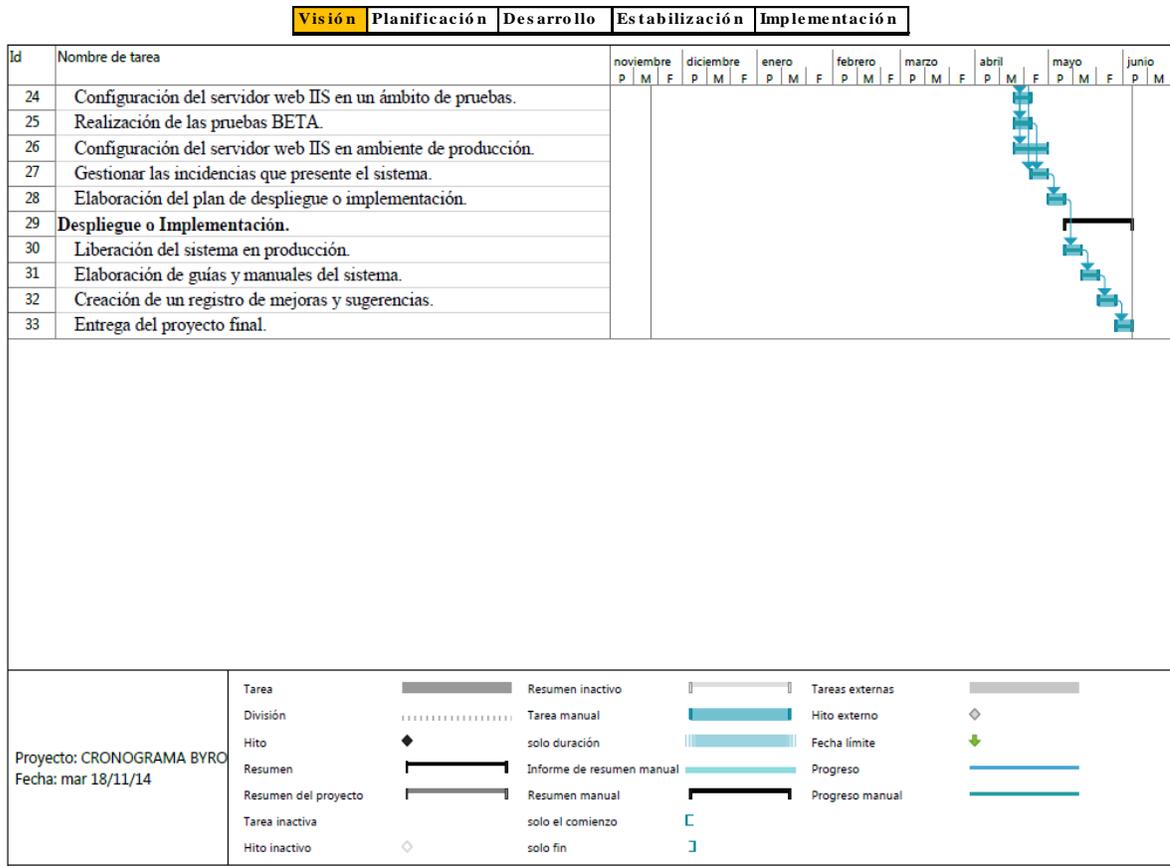


Fig.5: Plan de trabajo
Elaborado por: El Investigador.

4.1.4 Matriz de riesgos y plan de contingencia

Los riesgos detectados para el desarrollo del presente proyecto se detallan a continuación así como también las contingencias para cada uno de los riesgos.

Matriz de riesgos y plan contingencia.

| N. | Riesgo | Probabilidad /Ocurrencia | Consecuencia | Contingencia |
|----|---|--------------------------|---|--|
| 1 | Fallas en la planificación del tiempo de desarrollo del proyecto. | Baja | Retraso en la entrega del proyecto final. | Solicitar una prórroga para la entrega del proyecto. |
| 2 | Problemas o fallos | Baja | Perdida de | Configurar un equipo de respaldo |

| | | | | |
|---|---|------|--|---|
| | con el equipo de desarrollo. | | información y retraso en el desarrollo del proyecto. | en el cual se puede seguir trabajando en el proyecto. Respaldar los archivos fuentes del proyecto. |
| 3 | Problemas o fallos con el equipo servidor del aplicativo. | Baja | Perdida de datos. | Configurar un equipo el cual pueda hacer de segundo servidor. Respaldar de forma constante la base de datos. |
| 4 | Errores de diseño en la aplicación. | Baja | Inconsistencia en los procesos. | Revisar el documento de requerimientos y corregir los errores en base al documento. |

Tabla 2: Matriz de riesgos y plan contingencia
Elaborado por: El Investigador.

4.2 Planificación

Esta es la fase en la cual debemos definir de forma clara, precisa y concisa, como va a estar estructurada nuestra aplicación o solución final, para lograr esto debemos establecer la planificación y el diseño de la arquitectura. Se especificara el software que se utilizara en el proyecto.

Aquí es donde se realiza la mayor parte de la planificación del proyecto ya que se preparan las especificaciones funcionales del software, se realiza el diseño de la solución, una estimación de costos y los cronogramas de los entregables de las diferentes versiones del sistema, hasta llegar a la entrega final del proyecto.

4.2.1 Planificación y diseño de la arquitectura

Dentro de la planificación de la arquitectura de la aplicación, esta funcionará solamente en sistemas operativos windows en cualquiera de sus versiones desde XP o Server 2003 en adelante, funcionará con el servidor Web Internet Information Service (IIS) configurado para aplicaciones web generada por LightSwitch, las cuales unen varias tecnologías actuales como son: HTML5, SilverLight, C#, Linq entre otras.

El diseño de la arquitectura de la aplicación será web, a la cual se podrá acceder desde cualquier navegador web dentro de la intranet de la Liga Deportiva Cantonal Pelileo, si se desea acceder desde el internet se lo podrá hacer siempre y cuando se cuente con un dominio público.

Cabe señalar que la aplicación también podrá ser utilizada en un ambiente de escritorio, de así ser el caso. La aplicación utiliza una base de datos SQL Server 2008 la versión express la cual es gratuita. Y utiliza el Framework 4.0 que también es gratuito, la aplicación necesita de este Framework o de una versión superior, no funciona con una versión anterior.

4.2.2 Construcción del plan de laboratorio

El plan de laboratorio permite determinar los diversos escenarios a ser considerados, los criterios de validez y el control de las incidencias de los mismos.

Plan de laboratorio.

| Fase(s) | Responsable(s) | Contenido(s) |
|---|--|---|
| Servicios Web y provisión de contenidos de red. | <ul style="list-style-type: none"> El Investigador. | Instalación y configuración de IIS 7.0. Configuración del sistema en ambiente web. |
| Servicios básicos de red. | <ul style="list-style-type: none"> Liga Deportiva Cantonal Pelileo. El Investigador. | Verificación de los servicios de red con los que cuenta LDCCP. |
| Pruebas del sistema. | <ul style="list-style-type: none"> Liga Deportiva Cantonal Pelileo. El Investigador. | Pruebas con datos reales tanto de LDCCP así como también con datos de otras ligas según lo considere el investigador. |

Tabla 3: Plan de laboratorio
Elaborador por: El Investigador.

4.2.3 Estimación de costos

La estimación de costos para el presente proyecto el investigador no lo ha considerado necesario ya que todo el proyecto es costado con auto finamiento por parte del investigador. Y todo el software que se utiliza tanto para el desarrollo del proyecto como para el funcionamiento del sistema propuesto en el presente proyecto es software en versiones libres o también denominadas versiones Express las cuales no tienen costos de licenciamiento y se las puede utilizar sin ningún problema.

4.3 Desarrollo

4.3.1 Diseño de los diagramas de casos de uso

Los diagramas de casos de uso permiten representar las diferentes acciones que puede realizar un determinado usuario en el sistema, se ha identificado cuatro tipos de roles que en este caso cada rol tendría un usuario, por lo que también podríamos decir que el sistema tendrá cuatro tipos de usuarios con diferentes roles cada uno. [15].

- ✓ Administrador del sistema
- ✓ Secretaría
- ✓ Auxiliar de Secretaría
- ✓ Invitado

Diagrama de caso de uso del rol Administrador

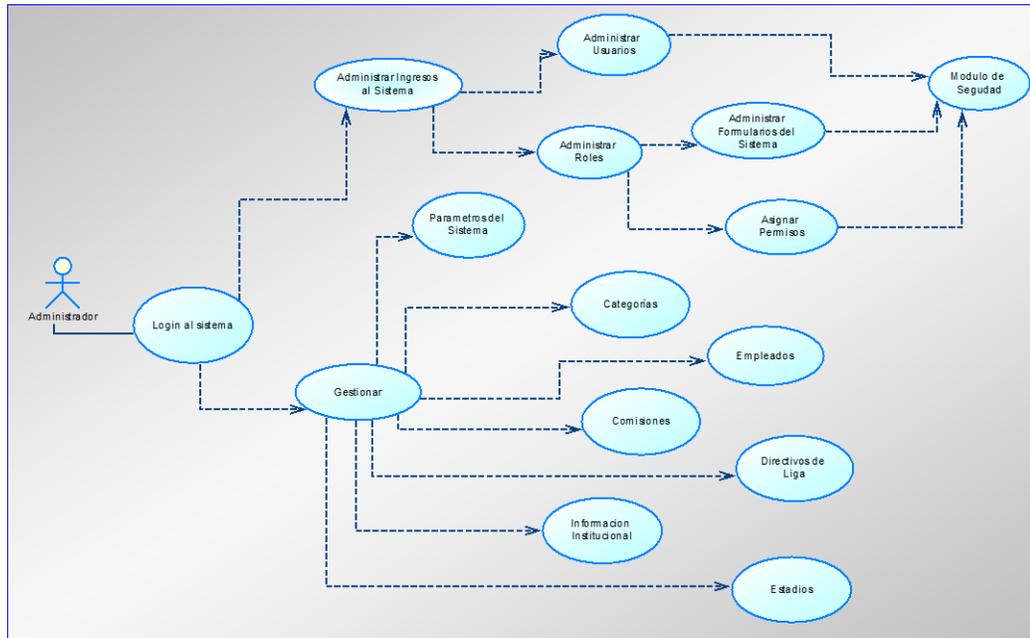


Fig.6: Diagrama de caso de uso del rol Administrador
Elaborador por: El Investigador.

Especificaciones de los casos de uso del rol Administrador

A continuación se especifica y se detalla los diferentes casos de uso correspondientes al rol Administrador, esto con la finalidad de tener una idea más clara y una visión más específica de los mismos.

Especificación del Caso de uso Administrar Ingresos al Sistema.

| | |
|------------------------|--|
| CASO DE USO #01 | Administrar Ingresos al Sistema. |
| ACTORES | Administrador. |
| OBJETIVO | Gestionar quienes pueden acceder al sistema. |
| PRECONDICIONES | ✓ Entrar en el sistema como Administrador. |
| POSCONDICIONES | ➤ El sistema tiene pantallas en las cuales le permite administrar quienes pueden ingresar al sistema y que pueden hacer en él. |
| | Actividades 1. Crear roles, usuario y asignar permisos. |

| | | |
|--|---|---|
| FLUJO DE EVENTO | del Actor | |
| | Respuesta del Sistema | 2. El sistema comprueba la validez de los datos ingresado. |
| MANEJO DE SITUACIONES EXCEPCIONALES | ✓ Si los datos ingresados son incorrectos el sistema mostrará un error. | |

Tabla 4: Caso de uso Administrar Ingresos al Sistema
Elaborador por: El Investigador.

Especificación del Caso de uso Administrar roles.

| | | |
|--|--|---|
| CASO DE USO #02 | Administrar roles. | |
| ACTORES | Administrador. | |
| OBJETIVO | Crear, Modificar y Eliminar roles en el sistema. | |
| PRECONDICIONES | ✓ Tener acceso al módulo de seguridad y a pantalla de administración de roles (Funciones) y permisos. | |
| POSCONDICIONES | <ul style="list-style-type: none"> ➤ El nuevo registro del rol quedara almacenado en la BD. ➤ El rol podrá ser modificado o eliminado. | |
| FLUJO DE EVENTO | Actividades del Actor | 1. Elegir la opción Seguridad => Roles => Funciones 3. Digitar información requerida ya sea para ingresar, modificar o borrar. 5. Pulsar el botón guardar |
| | Respuesta del Sistema | 2. El sistema despliega el formulario correspondiente para la administración de roles. 4. En la pantalla se muestra un (*) indicando que hay datos que deben ser guardados. 6 Los datos son guardados en el sistema. |
| MANEJO DE SITUACIONES EXCEPCIONALES | <ul style="list-style-type: none"> ✓ Si se crea un nuevo rol y este ya existe en la BD, el sistema mostrará un mensaje indicando que ese rol ya existe. ✓ Si se omite información requerida, se mostrará un mensaje indicando los campos faltantes y estos son pintados de color rojo. | |

Tabla 5: Caso de uso Administrar roles
Elaborador por: El Investigador.

Especificación del Caso de uso Administrar usuarios.

| | | |
|--|--|---|
| CASO DE USO #03 | Administrar usuarios. | |
| ACTORES | Administrador. | |
| OBJETIVO | Crear, Modificar y Eliminar usuarios en el sistema. | |
| PRECONDICIONES | ✓ Tener acceso al módulo de seguridad y a pantalla de administración de usuarios. | |
| POSCONDICIONES | <ul style="list-style-type: none"> ➤ El nuevo registro del usuario quedara almacenado en la BD. ➤ El usuario podrá ser modificado o eliminado. | |
| FLUJO DE EVENTO | Actividades del Actor | <ol style="list-style-type: none"> 1. Elegir la opción Seguridad => Usuarios 3. Digitar información requerida ya sea para ingresar, modificar o borrar. 5. Pulsar el botón guardar |
| | Respuesta del Sistema | <ol style="list-style-type: none"> 2. El sistema despliega el formulario correspondiente para la administración de usuarios. 4. En la pantalla se muestra un (*) indicando que hay datos que deben ser guardados. 6 Los datos son guardados en el sistema. |
| MANEJO DE SITUACIONES EXCEPCIONALES | <ul style="list-style-type: none"> ✓ Si se crea un nuevo usuario y este ya existe en la BD, el sistema mostrará un mensaje indicando que ese rol ya existe. ✓ Si se omite información requerida, se mostrará un mensaje indicando los campos faltantes y estos son pintados de color rojo. | |

Tabla 6: Caso de uso Administrar usuarios
Elaborador por: El Investigador.

Especificación del Caso de uso Administrar formularios del sistema.

| | | |
|------------------------|--|--|
| CASO DE USO #04 | Administrar formularios del sistema. | |
| ACTORES | Administrador. | |
| OBJETIVO | Dar acceso a los formularios del sistema por roles. | |
| PRECONDICIONES | ✓ Tener acceso al módulo de seguridad y a pantalla de administración roles y permisos. | |
| POSCONDICIONES | <ul style="list-style-type: none"> ➤ Los nuevos permisos quedaran almacenados en la BD. ➤ Estos podrán ser quitados a los roles. | |
| FLUJO DE EVENTO | Actividades del Actor | <ol style="list-style-type: none"> 1. Elegir la opción Seguridad => Roles => Permisos 3. Seleccionar el rol a cual se desea asignar permisos. 4. Seleccionar los permisos que se desea asignar. 6. Pulsar el botón guardar |
| | | 2. El sistema despliega el formulario correspondiente |

| | | |
|--|-----------------------|--|
| | Respuesta del Sistema | para la administración de formularios. 5. En la pantalla se muestra un (*) indicando que hay datos que deben ser guardados. 7 Los datos son guardados en el sistema. |
| MANEJO DE SITUACIONES EXCEPCIONALES | ✓ | Si el permiso ya está asignado se muestra un mensaje indicando que el rol ya cuenta con ese permiso. |

Tabla 7: Caso de uso Administrar formularios del sistema.
Elaborador por: El Investigador

Especificación del Caso de uso Gestionar Parámetros del sistema.

| | | |
|--|--|---|
| CASO DE USO #05 | Gestionar Parámetros del sistema. | |
| ACTORES | Administrador. | |
| OBJETIVO | Definir los parámetros necesarios del sistema. | |
| PRECONDICIONES | ✓ Tener acceso al módulo de Parámetros y a las pantallas de parámetros generales y específicos. | |
| POSCONDICIONES | ➤ Los parámetros quedarán almacenados en la BD. ➤ Estos podrán ser modificados según corresponda. | |
| FLUJO DE EVENTO | Actividades del Actor | 1. Elegir la opción Parámetros => Parámetros Generales o Específicos. 3. Ingresar la información requerida. 5. Pulsar el botón guardar |
| | Respuesta del Sistema | 2. El sistema despliega un formulario con pestañas para cada tipo de parámetro. 4. En la pantalla se muestra un (*) indicando que hay datos que deben ser guardados. 6 Los datos son guardados en el sistema. |
| MANEJO DE SITUACIONES EXCEPCIONALES | ✓ Si se crea un parámetro y este ya existe en la BD, el sistema mostrará un mensaje indicando que ya existe. ✓ Si se omite información requerida, se mostrará un mensaje indicando los campos faltantes y estos son pintados de color rojo. | |

Tabla 8: Caso de uso Gestionar Parámetros del sistema.
Elaborador por: El Investigador

Especificación del Caso de uso Gestionar Categorías.

| | | |
|--|---|--|
| CASO DE USO #06 | Gestionar Categorías. | |
| ACTORES | Administrador. | |
| OBJETIVO | Incluir en el sistema los datos de las categorías a las cuales pertenecerán los equipos y los campeonatos. | |
| PRECONDICIONES | ✓ Tener acceso al módulo de Empresa y a la pantalla Categorías. | |
| POSCONDICIONES | <ul style="list-style-type: none"> ➤ Las categorías quedarán almacenadas en la BD. ➤ Se podrán modificar algunos datos de las categorías. | |
| FLUJO DE EVENTO | Actividades del Actor | <ol style="list-style-type: none"> 1. Elegir la opción Empresa => Categorías. 3. Ingresar la información requerida. 5. Pulsar el botón guardar |
| | Respuesta del Sistema | <ol style="list-style-type: none"> 2. El sistema despliega un formulario para agregar, modificar o eliminar las categorías. 4. En la pantalla se muestra un (*) indicando que hay datos que deben ser guardados. 6 Los datos son guardados en el sistema. |
| MANEJO DE SITUACIONES EXCEPCIONALES | <ul style="list-style-type: none"> ✓ Si se crea una categoría y está ya existe en la BD, el sistema mostrará un mensaje indicando que ya existe. ✓ Si se omite información requerida, se mostrará un mensaje indicando los campos faltantes y estos son pintados de color rojo. | |

Tabla 9: Caso de uso Gestionar Categorías
Elaborador por: El Investigador.

Especificación del Caso de uso Gestionar Empleados.

| | | |
|------------------------|---|--|
| CASO DE USO #07 | Gestionar Empleados. | |
| ACTORES | Administrador. | |
| OBJETIVO | Incluir en el sistema datos de los empleados. | |
| PRECONDICIONES | ✓ Tener acceso al módulo de Capital Humano y a la pantalla Empleados. | |
| POSCONDICIONES | <ul style="list-style-type: none"> ➤ Los empleados quedarán almacenadas en la BD. ➤ Se podrán modificar algunos datos de los empleados. | |
| FLUJO DE EVENTO | Actividades del Actor | <ol style="list-style-type: none"> 1. Elegir la opción Capital Humano => Empleados. 3. Ingresar la información requerida. 5. Pulsar el botón guardar |
| | Respuesta del Sistema | <ol style="list-style-type: none"> 2. El sistema despliega un formulario para agregar, modificar o eliminar empleados así como también tiene una sesión para gestionar datos telefónicos. |

| | | |
|--|---|---|
| | | <p>4. En la pantalla se muestra un (*) indicando que hay datos que deben ser guardados.</p> <p>6 Los datos son guardados en el sistema.</p> |
| MANEJO DE SITUACIONES EXCEPCIONALES | <ul style="list-style-type: none"> ✓ Si se crea un empleado y este ya existe en la BD, el sistema mostrará un mensaje indicando que ya existe. ✓ Si se omite información requerida, se mostrará un mensaje indicando los campos faltantes y estos son pintados de color rojo. | |

Tabla 10: Caso de uso Gestionar Empleados
Elaborador por: El Investigador.

Especificación del Caso de uso Gestionar Comisiones.

| | | |
|--|--|---|
| CASO DE USO #08 | Gestionar Comisiones. | |
| ACTORES | Administrador. | |
| OBJETIVO | Incluir en el sistema los datos de las comisiones las cuales se les deberá asignar miembros. | |
| PRECONDICIONES | ✓ Tener acceso al módulo de Empresa y a la pantalla Comisiones. | |
| POSCONDICIONES | <ul style="list-style-type: none"> ➤ Las comisiones quedarán almacenadas en la BD. ➤ Se podrán modificar algunos datos de las comisiones. | |
| FLUJO DE EVENTO | Actividades del Actor | <p>1. Elegir la opción Empresa => Comisiones.</p> <p>3. Ingresar la información requerida.</p> <p>5. Pulsar el botón guardar</p> |
| | Respuesta del Sistema | <p>2. El sistema despliega un formulario para agregar, modificar o eliminar las comisiones.</p> <p>4. En la pantalla se muestra un (*) indicando que hay datos que deben ser guardados.</p> <p>6 Los datos son guardados en el sistema.</p> |
| MANEJO DE SITUACIONES EXCEPCIONALES | <ul style="list-style-type: none"> ✓ Si se crea una comisión y está ya existe en la BD, el sistema mostrará un mensaje indicando que ya existe. ✓ Si se omite información requerida, se mostrará un mensaje indicando los campos faltantes y estos son pintados de color rojo. | |

Tabla 11: Caso de uso Gestionar Comisiones
Elaborador por: El Investigador.

Especificación del Caso de uso Gestionar Directivos de Liga LDGP.

| | | |
|--|---|--|
| CASO DE USO #9 | Gestionar Directivos de Liga LDGP. | |
| ACTORES | Administrador. | |
| OBJETIVO | Incluir en el sistema información de los directivos de la Liga así como también de los que forman parte de las comisiones. | |
| PRECONDICIONES | ✓ Tener acceso al módulo de Empresa y a la pantalla Directivos Liga. | |
| POSCONDICIONES | <ul style="list-style-type: none"> ➤ Los directivos de liga LDP quedarán almacenadas en la BD. ➤ Se podrán modificar algunos datos de los directivos. | |
| FLUJO DE EVENTO | Actividades del Actor | <ol style="list-style-type: none"> 1. Elegir la opción Empresa => Directivos Liga. 3. Ingresar la información requerida. 5. Pulsar el botón guardar |
| | Respuesta del Sistema | <ol style="list-style-type: none"> 2. El sistema despliega un formulario con pestañas para agregar, modificar o eliminar los directivos de liga y los que forman parte de las comisiones. 4. En la pantalla se muestra un (*) indicando que hay datos que deben ser guardados. 6 Los datos son guardados en el sistema. |
| MANEJO DE SITUACIONES EXCEPCIONALES | <ul style="list-style-type: none"> ✓ Si se crea un directivo o se asigna a una comisión y este ya existe en la BD, el sistema mostrará un mensaje indicando que ya existe. ✓ Si se omite información requerida, se mostrará un mensaje indicando los campos faltantes y estos son pintados de color rojo. | |

Tabla 12: Caso de uso Gestionar Directivos de Liga
Elaborador por: El Investigador.

Especificación del Caso de uso Gestionar Información Institucional.

| | | |
|------------------------|--|--|
| CASO DE USO #10 | Gestionar Información Institucional. | |
| ACTORES | Administrador. | |
| OBJETIVO | Permitir incluir la información institucional en el sistema. | |
| PRECONDICIONES | ✓ Tener acceso al módulo Empresa y a la pantalla Empresa y Oficinas | |
| POSCONDICIONES | <ul style="list-style-type: none"> ➤ La información quedará almacenada en la BD. ➤ Esta información podrá ser modificada en cualquier momento. ➤ La información de la empresa, las oficinas o los departamentos no podrá ser eliminada. | |
| | Actividades | <ol style="list-style-type: none"> 1. Elegir la opción Empresa => Empresa y Oficinas. 3. Ingresar la información requerida. |

| | | |
|--|--|---|
| FLUJO DE EVENTO | del Actor | 5. Pulsar el botón guardar |
| | Respuesta del Sistema | 2. El sistema despliega un formulario con pestañas para ingresar o modificar la información necesaria. 4. En la pantalla se muestra un (*) indicando que hay datos que deben ser guardados. 6 Los datos son guardados en el sistema. |
| MANEJO DE SITUACIONES EXCEPCIONALES | <ul style="list-style-type: none"> ✓ Si se crean datos que ya existen en la BD, el sistema mostrará un mensaje indicando que ya existe. ✓ Si se omite información requerida, se mostrará un mensaje indicando los campos faltantes y estos son pintados de color rojo. | |

Tabla 13: Caso de uso Gestionar Información Institucional
Elaborador por: El Investigador.

Especificación del Caso de uso Gestionar Estadios.

| | | |
|--|--|---|
| CASO DE USO #11 | Gestionar Estadios. | |
| ACTORES | Administrador. | |
| OBJETIVO | Incluir en el sistema información de los estadios en los cuales se juegan los diferentes campeonatos en sus diversas categorías. | |
| PRECONDICIONES | <ul style="list-style-type: none"> ✓ Tener acceso al módulo de Empresa y a la pantalla Geografía. ✓ Tener ingresado en el sistema la información relacionada al estadio como, país, provincia, ciudad y parroquia. | |
| POSCONDICIONES | <ul style="list-style-type: none"> ➤ La información de los estadios quedará almacenada en la BD. ➤ Esta información podrá ser modificada en cualquier momento. | |
| FLUJO DE EVENTO | Actividades del Actor | 1. Elegir la opción Empresa => Geografía. 3. Ingresar la información requerida. 5. Pulsar el botón guardar |
| | Respuesta del Sistema | 2. El sistema despliega un formulario con pestañas para ingresar o modificar la información de los estadios. 4. En la pantalla se muestra un (*) indicando que hay datos que deben ser guardados. 6 Los datos son guardados en el sistema. |
| MANEJO DE SITUACIONES EXCEPCIONALES | <ul style="list-style-type: none"> ✓ Si se crea un estadio que ya existen en la BD, el sistema mostrará un mensaje indicando que ya existe. ✓ Si se omite información requerida, se mostrará un mensaje indicando los campos faltantes y estos son pintados de color rojo. | |

Tabla 14: Caso de uso Gestionar Estadios
Elaborador por: El Investigador.

Diagrama de caso de uso del rol Secretaría

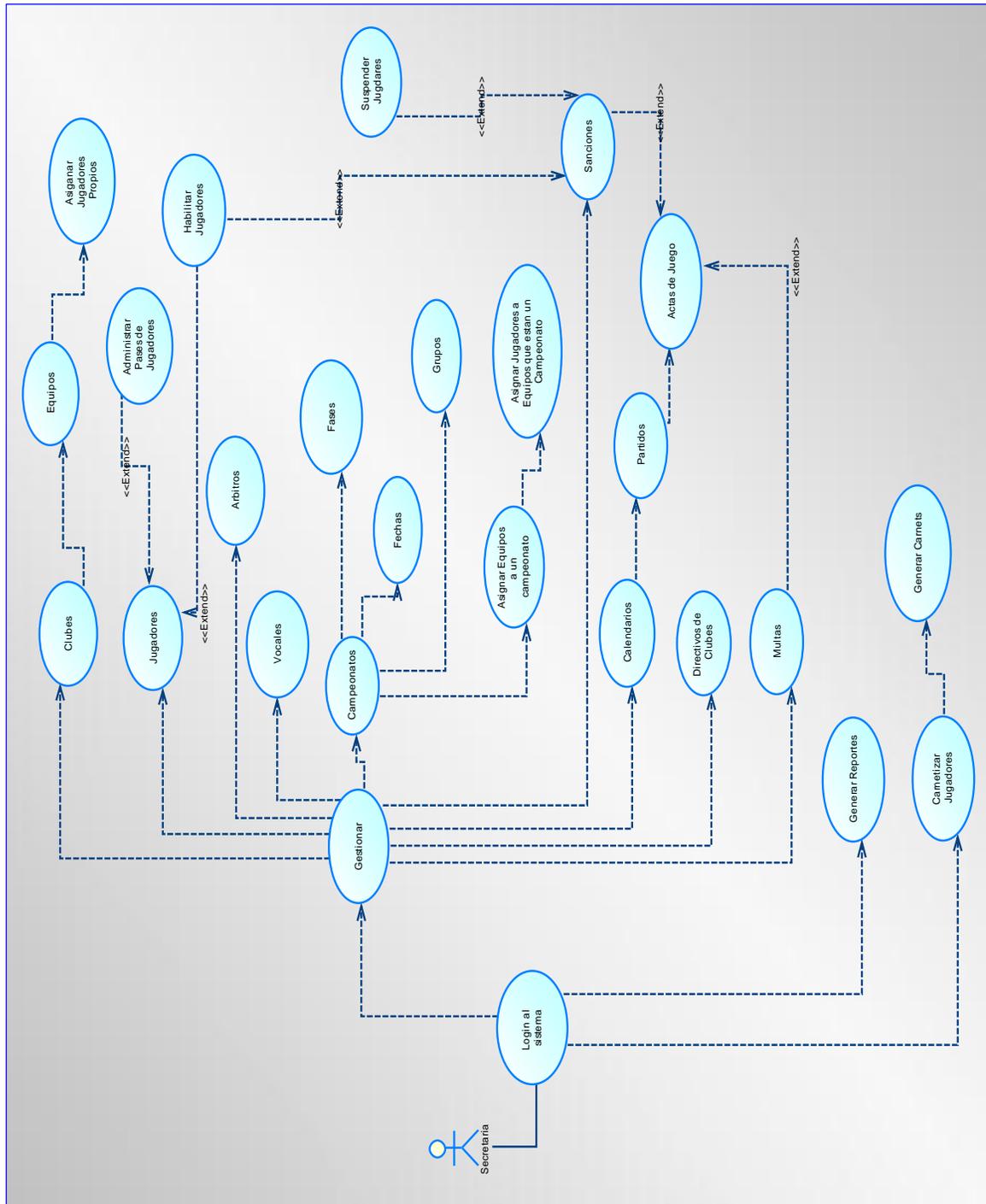


Fig.7: Diagrama de caso de uso rol Secretaría
Elaborador por: El Investigador.

Especificaciones de los casos de uso del rol Secretaría

A continuación se especifica y se detalla los diferentes casos de uso correspondientes al rol Secretaría, con la finalidad de tener una idea más clara y una visión más específica de los mismos.

Especificación del Caso de uso Gestionar Clubes.

| | | |
|--|---|---|
| CASO DE USO #12 | Gestionar Clubes. | |
| ACTORES | Secretaría. | |
| OBJETIVO | Incluir en el sistema los datos de los clubes los cuales tendrá equipos en las diferentes categorías. | |
| PRECONDICIONES | <ul style="list-style-type: none"> ✓ Tener acceso al módulo Clubes y a la pantalla Clubes. ✓ Tener ingresado en el sistema la información de las parroquias. | |
| POSCONDICIONES | <ul style="list-style-type: none"> ➤ La información de los clubes quedará almacenada en la BD. ➤ Esta información podrá ser modificada en cualquier momento. | |
| FLUJO DE EVENTO | Actividades del Actor | <ol style="list-style-type: none"> 1. Elegir la opción Clubes => Clubes. 3. Presionar el botón Ingresar, Editar o Eliminar según la acción que se desee realizar. 4. Ingresar la información requerida. 6. Pulsar el botón guardar |
| | Respuesta del Sistema | <ol style="list-style-type: none"> 2. El sistema despliega un formulario en el cual se podrá ingresar la información del club. 5. En la pantalla se muestra un (*) indicando que hay datos que deben ser guardados. 7 Los datos son guardados en el sistema. |
| MANEJO DE SITUACIONES EXCEPCIONALES | <ul style="list-style-type: none"> ✓ Si se crea un club que ya existen en la BD, el sistema mostrará un mensaje indicando que ya existe. ✓ Si se omite información requerida, se mostrará un mensaje indicando los campos faltantes y estos son pintados de color rojo. | |

Tabla 15: Caso de uso Gestionar Clubes
Elaborador por: El Investigador.

Especificación del Caso de uso Asignar jugadores propios.

| | | |
|--|--|--|
| CASO DE USO #13 | Asignar jugadores propios. | |
| ACTORES | Secretaria. | |
| OBJETIVO | Incluir en el sistema los datos de los jugadores propios con los que cuenta un club. | |
| PRECONDICIONES | <ul style="list-style-type: none"> ✓ Tener acceso al módulo Jugadores & Prestamos y a la pantalla Jugadores Propios. ✓ Tener ingresado en el sistema la información de equipos y jugadores. | |
| POSCONDICIONES | <ul style="list-style-type: none"> ➤ La información de los jugadores propios de un club quedará almacenada en la BD. ➤ Esta información podrá ser modificada en cualquier momento. | |
| FLUJO DE EVENTO | Actividades del Actor | <ol style="list-style-type: none"> 1. Elegir la opción Jugadores & Prestamos => Jugadores Propios. 3. Seleccionar la categoría y el equipo. 4. Ingresar la información requerida. 6. Pulsar el botón guardar |
| | Respuesta del Sistema | <ol style="list-style-type: none"> 2. El sistema despliega un formulario en cual se podrá ingresar la información de los jugadores propios de un club. 5. En la pantalla se muestra un (*) indicando que hay datos que deben ser guardados. 7. Los datos son guardados en el sistema. |
| MANEJO DE SITUACIONES EXCEPCIONALES | <ul style="list-style-type: none"> ✓ Si se asigna un jugador propio que está registrado en la BD, el sistema mostrará un mensaje indicando que ya existe. ✓ Si se omite información requerida, se mostrará un mensaje indicando los campos faltantes y estos son pintados de color rojo. | |

Tabla 16: Caso de uso Asignar jugadores propios
Elaborador por: El Investigador.

Especificación del Caso de uso Administrar Pases de Jugadores.

| | |
|------------------------|---|
| CASO DE USO #14 | Administrar Pases de Jugadores. |
| ACTORES | Secretaría. |
| OBJETIVO | Incluir en el sistema los datos de los pases de los jugadores y generar un historial de los equipos en los cuales milito. |
| PRECONDICIONES | ✓ Tener acceso al módulo Jugadores & Prestamos y a las pantallas Jugadores Propios, Prestamos Jugadores y Regresar Prestamos. |

| | |
|--|--|
| | ✓ Tener ingresado en el sistema la información de equipos y jugadores. |
| POSCONDICIONES | <ul style="list-style-type: none"> ➤ La información de los préstamos y pases de los jugadores quedará almacenada en la BD. ➤ Esta información podrá ser modificada en cualquier momento. |
| FLUJO DE EVENTO | <p>Actividades del Actor</p> <ol style="list-style-type: none"> 1. Elegir la opción Jugadores & Prestamos => Jugadores Propios, Detalles de los préstamos. 3. Seleccionar la categoría y el equipo. 4. Ingresar la información requerida. 6. Pulsar el botón guardar <p>Seguir los mismos pasos incluyendo estos.</p> <ol style="list-style-type: none"> 1.1 Elegir la opción Jugadores & Prestamos => Prestamos Jugadores. 1.3 Seleccionar el jugador. 1.2 Elegir la opción Jugadores & Prestamos => Regresar Prestamos. 1.3 Pulsar el botón Regresar Prestamos. |
| | <p>Respuesta del Sistema</p> <ol style="list-style-type: none"> 2. El sistema despliega un formulario en cual se podrá ingresar la información de los préstamos del jugador. 5. En la pantalla se muestra un (*) indicando que hay datos que deben ser guardados. 7. Los datos son guardados en el sistema. 1.2. El sistema despliega un formulario en cual se podrá ver la información de los préstamos. 1.4. Los datos son guardados en el sistema. |
| MANEJO DE SITUACIONES EXCEPCIONALES | <ul style="list-style-type: none"> ✓ Si en cualquiera de los procesos se genera un error el sistema mostrar un mensaje indicando cual fue el error generado. ✓ Si se omite información requerida, se mostrará un mensaje indicando los campos faltantes y estos son pintados de color rojo. |

Tabla 17: Caso de uso Administrar Pases de Jugadores
Elaborador por: El Investigador.

Especificación del Caso de uso Gestionar Árbitros y Vocales.

| | |
|------------------------|--|
| CASO DE USO #15 | Gestionar Árbitros y Vocales. |
| ACTORES | Secretaria. |
| OBJETIVO | Incluir en el sistema información de los Árbitros y de los vocales los cuales serán actores de un partido. |

| | | |
|--|---|--|
| PRECONDICIONES | ✓ Tener acceso al módulo de Empresa y a la pantalla Vocales y al módulo Otros a la pantalla Árbitros. | |
| POSCONDICIONES | ➤ La información de los Árbitros y Vocales quedará almacenada en la BD. | |
| FLUJO DE EVENTO | Actividades del Actor | <ol style="list-style-type: none"> 1. Elegir la opción Empresa => Vocales o 1. Elegir la opción Otros => Árbitros. 3. Ingresar la información requerida. 5. Pulsar el botón guardar |
| | Respuesta del Sistema | <ol style="list-style-type: none"> 2. El sistema despliega un formulario para ingresar o modificar los datos. 4. En la pantalla se muestra un (*) indicando que hay datos que deben ser guardados. 6 Los datos son guardados en el sistema. |
| MANEJO DE SITUACIONES EXCEPCIONALES | <ul style="list-style-type: none"> ✓ Si se crea un registro que ya existen en la BD, el sistema mostrará un mensaje indicando que ya existe. ✓ Si se omite información requerida, se mostrará un mensaje indicando los campos faltantes y estos son pintados de color rojo. | |

Tabla 18: Caso de uso Gestionar Árbitros y Vocales
Elaborador por: El Investigador.

Especificación del Caso de uso Gestionar Campeonatos, Fases, Fechas y Grupos.

| | | |
|------------------------|--|---|
| CASO DE USO #16 | Gestionar Campeonatos, Fases, Fechas y Grupos. | |
| ACTORES | Secretaría. | |
| OBJETIVO | Incluir en el sistema información de los diferentes campeonatos de fútbol conjuntamente con las fases, fechas y grupos. | |
| PRECONDICIONES | ✓ Tener acceso al módulo Campeonatos y a la pantalla Campeonatos. | |
| POSCONDICIONES | <ul style="list-style-type: none"> ➤ La información de los campeonatos sus fases, fechas y grupos quedará almacenada en la BD. ➤ Esta información podrá ser modificada en cualquier momento. | |
| FLUJO DE EVENTO | Actividades del Actor | <ol style="list-style-type: none"> 1. Elegir la opción Campeonatos => Campeonatos. 3. Ingresar la información requerida. 5. Pulsar el botón guardar. |
| | Respuesta | 2. El sistema despliega un formulario con 4 pestañas (Listado Campeonatos, Listado Fases, Listado Fechas, Listado Grupos) en las cuales se podrá ingresar la información correspondiente. |

| | | |
|--|--|--|
| | del Sistema | <p>4. En la pantalla se muestra un (*) indicando que hay datos que deben ser guardados.</p> <p>6. Los datos son guardados en el sistema.</p> |
| MANEJO DE SITUACIONES EXCEPCIONALES | <p>✓ Si se crea un registro que ya existen en la BD, el sistema mostrará un mensaje indicando que ya existe.</p> <p>✓ Si se omite información requerida, se mostrará un mensaje indicando los campos faltantes y estos son pintados de color rojo.</p> | |

Tabla 19: Caso de uso Gestionar Campeonatos, Fases, Fechas y Grupos
Elaborador por: El Investigador.

Especificación del Caso de uso Asignar Equipos a un campeonato.

| | | |
|--|--|---|
| CASO DE USO #17 | Asignar Equipos a un campeonato. | |
| ACTORES | Secretaria. | |
| OBJETIVO | Asignar equipos a un campeonato de fútbol ya sea en un solo grupo o en diferentes grupos. | |
| PRECONDICIONES | ✓ Tener acceso al módulo de Campeonatos y a la pantalla Equipos – Campeonatos. | |
| POSCONDICIONES | ➤ La información de los equipos que pertenecen a un campeonato quedará almacenada en la BD. | |
| FLUJO DE EVENTO | Actividades del Actor | <p>1. Elegir la opción Campeonatos => Equipos – Campeonatos.</p> <p>3. Seleccionar la Categoría => Campeonato => Grupo.</p> <p>4. Ingresar la información requerida.</p> <p>6. Pulsar el botón guardar.</p> <p>1. Parte dos sigue los pasos 1, 2, 3, el siguiente (4) y 7.</p> <p>4. Pulsar el botón Agregar Equipo.</p> |
| | Respuesta del Sistema | <p>2. El sistema despliega un formulario para ingresar o modificar los datos.</p> <p>5. En la pantalla se muestra un (*) indicando que hay datos que deben ser guardados.</p> <p>7 Los datos son guardados en el sistema.</p> |
| MANEJO DE SITUACIONES EXCEPCIONALES | <p>✓ Si se agrega a un equipo que este registrado en ese campeonato, el sistema mostrará un mensaje indicando que ya existe.</p> <p>✓ Si se omite información requerida, se mostrará un mensaje indicando los campos faltantes y estos son pintados de color rojo.</p> | |

Tabla 20: Caso de uso Asignar Equipos a un campeonato
Elaborador por: El Investigador.

Especificación del Caso de uso Asignar Jugadores a Equipos que están en un campeonato.

| | | |
|--|---|--|
| CASO DE USO #18 | Asignar Jugadores a Equipos que están en un campeonato. | |
| ACTORES | Secretaría. | |
| OBJETIVO | Asignar jugadores a los equipos que ya están inscritos en un campeonato. | |
| PRECONDICIONES | ✓ Tener acceso al módulo de Campeonatos y a la pantalla Jugadores – Campeonatos. | |
| POSCONDICIONES | ➤ La información de los Jugadores que tiene cada equipo en un determinado campeonato quedará almacenada en la BD. | |
| FLUJO DE EVENTO | Actividades del Actor | <ol style="list-style-type: none"> 1. Elegir la opción Campeonatos => Jugadores – Campeonatos. 3. Seleccionar la Categoría => Campeonato => Equipo. 4. Ingresar la información requerida. 6. Pulsar el botón guardar. |
| | Respuesta del Sistema | <ol style="list-style-type: none"> 1. Parte dos sigue los pasos 1, 2, 3, el siguiente (4) y 7. 4. Pulsar el botón Agregar Jugadores. 2. El sistema despliega un formulario para ingresar o modificar los datos. 5. En la pantalla se muestra un (*) indicando que hay datos que deben ser guardados. 7 Los datos son guardados en el sistema. |
| MANEJO DE SITUACIONES EXCEPCIONALES | <ul style="list-style-type: none"> ✓ Si se agrega a un jugador que ya esté jugando en otro equipo, el sistema mostrará un mensaje indicando que ya existe. ✓ Si se omite información requerida, se mostrará un mensaje indicando los campos faltantes y estos son pintados de color rojo. | |

Tabla 21: Caso de uso Asignar Jugadores a Equipos que están en un campeonato
Elaborador por: El Investigador.

Especificación del Caso de uso Gestionar Partidos.

| | |
|------------------------|---|
| CASO DE USO #19 | Gestionar Partidos. |
| ACTORES | Secretaria. |
| OBJETIVO | Incluir en el sistema la información de los partidos que tiene un campeonato de fútbol, así como también de los detalles de un partido. |
| PRECONDICIONES | ✓ Tener acceso al módulo de Campeonatos y a la pantalla Partidos. |
| POSCONDICIONES | ➤ La información de los partidos que tiene un campeonato quedará |

| | | |
|--|---|---|
| | almacenada en la BD. | |
| FLUJO DE EVENTO | Actividades del Actor | <ol style="list-style-type: none"> 1. Elegir la opción Campeonatos => Partidos. 3. Seleccionar la Categoría => Campeonato => Fase=>fecha. 4. Ingresar la información requerida. 7. Pulsar el botón guardar. |
| | Respuesta del Sistema | <ol style="list-style-type: none"> 2. El sistema despliega un formulario para ingresar o modificar los datos en el cual también incluye el detalle de los partidos. 5. En la pantalla se muestra un (*) indicando que hay datos que deben ser guardados. 6. Dependiendo de la información que se ingrese de las actas de juego es decir de los partidos y sus detalles se genera automáticamente las sanciones o multas según corresponda y se suspenderán jugadores según la parametrización del sistema. 8 Los datos son guardados en el sistema. |
| MANEJO DE SITUACIONES EXCEPCIONALES | <ul style="list-style-type: none"> ✓ Si se crea un registro que ya existen en la BD, el sistema mostrará un mensaje indicando que ya existe. ✓ Si se omite información requerida, se mostrará un mensaje indicando los campos faltantes y estos son pintados de color rojo. | |

Tabla 22: Caso de uso Gestionar Partidos
Elaborador por: El Investigador.

Especificación del Caso de uso Gestionar Sanciones y Multas.

| | | |
|------------------------|--|--|
| CASO DE USO #20 | Gestionar Sanciones y Multas. | |
| ACTORES | Secretaría. | |
| OBJETIVO | Gestionar las sanciones y multas que genera un partido de fútbol. | |
| PRECONDICIONES | ✓ Tener acceso al módulo de Multas & Sanciones y a las pantallas Multas Generadas, Sanciones Generadas y Levantar Sanciones. | |
| POSCONDICIONES | ➤ La información de las multas y sanciones quedará almacenada en la BD. | |
| FLUJO DE EVENTO | Actividades del Actor | <ol style="list-style-type: none"> 1. Elegir la opción Multas & Sanciones => Opción correspondiente. 3. Consultar información requerida o 3. Procesar las sanciones, esto habilitar a los jugadores sancionados. |
| | Respuesta | <ol style="list-style-type: none"> 2. El sistema despliega un formulario para cada |

| | | |
|--|---|---|
| | del Sistema | opción, este formulario es de consulta. |
| MANEJO DE SITUACIONES EXCEPCIONALES | ✓ Si al procesar las sanciones se genera algún error, el sistema mostrará un mensaje con el error generado. | |

Tabla 23: Caso de uso Gestionar Sanciones y Multas
Elaborador por: El Investigador.

Especificación del Caso de uso Gestionar Directivos de Clubes.

| | | |
|--|--|---|
| CASO DE USO #21 | Gestionar Directivos de Clubes. | |
| ACTORES | Secretaria. | |
| OBJETIVO | Incluir en el sistema información de los directivos de los clubes. | |
| PRECONDICIONES | ✓ Tener acceso al módulo de Clubes y a la pantalla Directivos. | |
| POSCONDICIONES | ➤ La información de los directivos de los clubes quedará almacenadas en la BD. ➤ Se podrán modificar algunos datos de los directivos. | |
| FLUJO DE EVENTO | Actividades del Actor | 1. Elegir la opción Clubes => Directivos. 3. Ingresar la información requerida. 5. Pulsar el botón guardar. |
| | Respuesta del Sistema | 2. El sistema despliega un formulario para ingresar o modificar los datos. 4. En la pantalla se muestra un (*) indicando que hay datos que deben ser guardados. 6 Los datos son guardados en el sistema. |
| MANEJO DE SITUACIONES EXCEPCIONALES | ✓ Si se crea un directivo que ya está registrado en la BD, el sistema mostrará un mensaje indicando que ya existe. ✓ Si se omite información requerida, se mostrará un mensaje indicando los campos faltantes y estos son pintados de color rojo. | |

Tabla 24: Caso de uso Gestionar Directivos de Clubes
Elaborador por: El Investigador.

Diagrama de caso de uso del rol Auxiliar de Secretaría

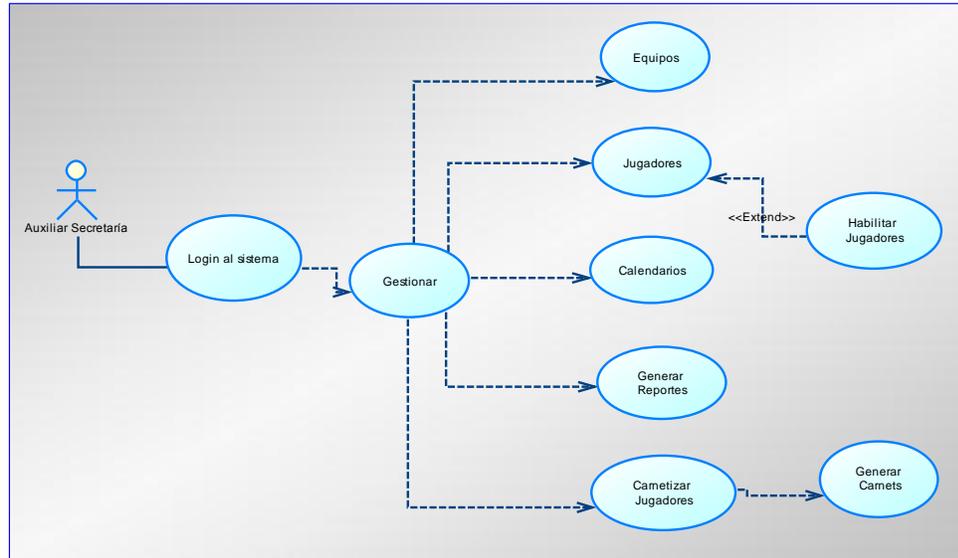


Fig.8: Diagrama de caso de uso del rol Auxiliar de Secretaría
Elaborador por: El Investigador.

Especificaciones de los casos de uso del rol Auxiliar de Secretaría y de los casos de uso que también corresponden al rol Secretaría.

A continuación se especifica y se detalla los diferentes casos de uso correspondientes al rol Auxiliar de Secretaría así como también a los que corresponde al rol Secretaría, es decir los casos de uso que comparten estos dos roles, con la finalidad de tener una idea más clara y una visión más específica de los mismos.

Especificación del Caso de uso Gestionar Equipos.

| | |
|------------------------|---|
| CASO DE USO #22 | Gestionar Equipos. |
| ACTORES | Secretaría, Auxiliar Secretaría. |
| OBJETIVO | Incluir en el sistema información de los equipos a los cuales se les asignara en un campeonato y jugadores. |
| PRECONDICIONES | ✓ Tener acceso al módulo Clubes y a la pantalla Equipos. |

| | | |
|--|---|--|
| | ✓ Tener ingresado en el sistema la información del Club al que pertenece el equipo. | |
| POSCONDICIONES | ➤ La información de los equipos quedará almacenada en la BD. ➤ Esta información podrá ser modificada en cualquier momento. | |
| FLUJO DE EVENTO | Actividades del Actor | 1. Elegir la opción Clubes => Equipos. 3. Presionar el botón Ingresar, Editar o Eliminar según la acción que se desee realizar. 5. Ingresar la información requerida. 7. Pulsar el botón guardar |
| | Respuesta del Sistema | 2. El sistema despliega un formulario en cual se podrá ingresar la información del equipo. 4. El sistema mostrará una pantalla tipo popup con los campos a ser llenados o modificados, cuando se pulsa el botón Nuevo o Editar. 6. En la pantalla se muestra un (*) indicando que hay datos que deben ser guardados. 8 Los datos son guardados en el sistema. |
| MANEJO DE SITUACIONES EXCEPCIONALES | ✓ Si se crea un equipo que ya existen en la BD, el sistema mostrará un mensaje indicando que ya existe. ✓ Si se omite información requerida, se mostrará un mensaje indicando los campos faltantes y estos son pintados de color rojo. | |

Tabla 25: Caso de uso Gestionar Equipos
Elaborador por: El Investigador.

Especificación del Caso de uso Gestionar Jugadores.

| | | |
|------------------------|---|---|
| CASO DE USO #23 | Gestionar Jugadores. | |
| ACTORES | Secretaria, Auxiliar Secretaría. | |
| OBJETIVO | Incluir en el sistema los datos de los jugadores los cuales podrán pertenecer a un club. | |
| PRECONDICIONES | ✓ Tener acceso al módulo Clubes y a la pantalla Jugadores. | |
| POSCONDICIONES | ➤ La información de los jugadores quedará almacenada en la BD. ➤ Esta información podrá ser modificada en cualquier momento. | |
| FLUJO DE EVENTO | Actividades del Actor | 1. Elegir la opción Clubes => Jugadores. 3. Presionar el botón Ingresar, Editar o Eliminar según la acción que se desee realizar. 5. Ingresar la información requerida. 7. Pulsar el botón guardar |
| | | 2. El sistema despliega un formulario en cual se podrá |

| | | |
|--|--|--|
| | Respuesta del Sistema | <p>ingresar la información del jugador.</p> <p>4. El sistema mostrará una pantalla tipo popup con los campos a ser llenados o modificados, cuando se pulsa el botón Nuevo o Editar.</p> <p>6. En la pantalla se muestra un (*) indicando que hay datos que deben ser guardados.</p> <p>8 Los datos son guardados en el sistema.</p> |
| MANEJO DE SITUACIONES EXCEPCIONALES | <ul style="list-style-type: none"> ✓ Si se crea un jugador que ya existen en la BD, el sistema mostrará un mensaje indicando que ya existe. ✓ Si se omite información requerida, se mostrará un mensaje indicando los campos faltantes y estos son pintados de color rojo. | |

Tabla 26: Caso de uso Gestionar Jugadores
Elaborador por: El Investigador.

Especificación del Caso de uso Gestionar Calendarios.

| | | |
|--|---|--|
| CASO DE USO #24 | Gestionar Calendarios. | |
| ACTORES | Secretaría, Auxiliar Secretaría. | |
| OBJETIVO | Incluir en el sistema la información de los calendarios que tiene un campeonato de fútbol. | |
| PRECONDICIONES | ✓ Tener acceso al módulo de Campeonatos y a la pantalla Calendarios. | |
| POSCONDICIONES | ➤ La información de los calendarios que tiene un campeonato quedará almacenada en la BD. | |
| FLUJO DE EVENTO | Actividades del Actor | <p>1. Elegir la opción Campeonatos => Calendarios.</p> <p>3. Seleccionar la Categoría => Campeonato => Fase.</p> <p>4. Ingresar la información requerida.</p> <p>6. Pulsar el botón guardar.</p> |
| | Respuesta del Sistema | <p>2. El sistema despliega un formulario para ingresar o modificar los datos.</p> <p>5. En la pantalla se muestra un (*) indicando que hay datos que deben ser guardados.</p> <p>7 Los datos son guardados en el sistema.</p> |
| MANEJO DE SITUACIONES EXCEPCIONALES | <ul style="list-style-type: none"> ✓ Si se agrega una fecha que existe en esa fase y para ese grupo, el sistema mostrará un mensaje indicando que ya existe. ✓ Si se omite información requerida, se mostrará un mensaje indicando los campos faltantes y estos son pintados de color rojo. | |

Tabla 27: Caso de uso Gestionar Calendarios
Elaborador por: El Investigador.

Especificación del Caso de uso Gestionar Reportes.

| | | |
|--|--|--|
| CASO DE USO #25 | Gestionar Reportes. | |
| ACTORES | Secretaria, Auxiliar Secretaría. | |
| OBJETIVO | Gestionar el listado de los reportes con los que cuenta el sistema. | |
| PRECONDICIONES | ✓ Tener acceso al módulo de Empresa y a la pantalla Listado Reportes. | |
| POSCONDICIONES | ➤ La información del listado de los reportes con los que cuenta el sistema quedará almacenadas en la BD. | |
| FLUJO DE EVENTO | Actividades del Actor | <ol style="list-style-type: none"> 1. Elegir la opción Empresa => Listado Reportes. 3. Ingresar la información requerida. 5. Pulsar el botón guardar |
| | Respuesta del Sistema | <ol style="list-style-type: none"> 2. El sistema despliega un formulario para ingresar o modificar los datos. 4. En la pantalla se muestra un (*) indicando que hay datos que deben ser guardados. 6 Los datos son guardados en el sistema. |
| MANEJO DE SITUACIONES EXCEPCIONALES | ✓ Si se omite información requerida, se mostrará un mensaje indicando los campos faltantes y estos son pintados de color rojo. | |

Tabla 28: Caso de uso Gestionar Reportes
Elaborador por: El Investigador.

Especificación del Caso de uso Carnetizar Jugadores y Generar Carnets.

| | | |
|------------------------|--|--|
| CASO DE USO #26 | Carnetizar Jugadores y Generar Carnets | |
| ACTORES | Secretaría, Auxiliar Secretaría. | |
| OBJETIVO | Gestionar la carnetización de jugadores. | |
| PRECONDICIONES | <ul style="list-style-type: none"> ✓ Tener acceso a los módulos de Campeonatos y Clubes y a las pantallas Campeonato, Jugadores – Campeonato y Jugadores. ✓ Habilitar a los jugadores y asignarlos a un equipo en un campeonato. | |
| POSCONDICIONES | ➤ Se genera un reporte con los carnets de los jugadores que estén habilitados. | |
| FLUJO DE EVENTO | Actividades del Actor | <ol style="list-style-type: none"> 1. Elegir la opción Correspondiente. 3. Pulsar el botón Carnetizar. |
| | Respuesta del Sistema | <ol style="list-style-type: none"> 2. El sistema despliega un formulario. 4. El sistema despliega un formulario con el reporte el cual puede ser guardado o puede imprimirse |

| | | |
|--|---|---|
| | | directamente. |
| MANEJO DE SITUACIONES EXCEPCIONALES | ✓ | Si se genera algún error en el envío de los parámetros para la generación del reporte, el sistema indicara esta situación con un mensaje. |

Tabla 29: Caso de uso Carnetizar Jugadores y Generar Carnets
Elaborador por: El Investigador.

Diagrama de caso de uso del rol Invitado

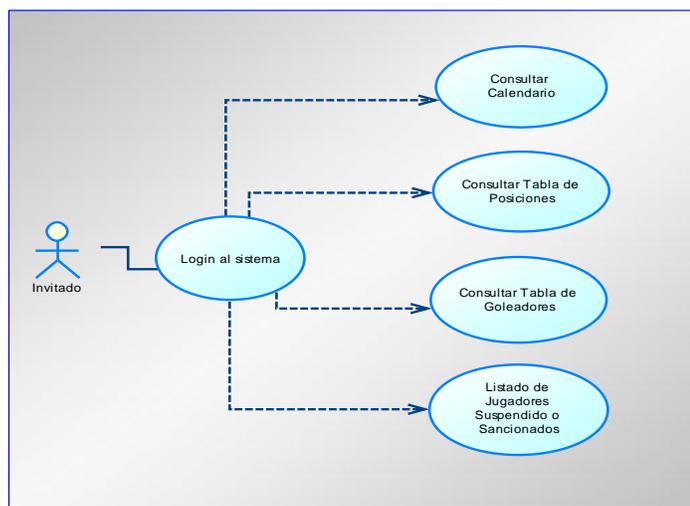


Fig. 9: Diagrama de caso de uso del rol Invitado
Elaborado por: El Investigador.

Especificaciones de los casos de uso de los roles Invitado, Auxiliar de Secretaría, Secretaría y Administrador del sistema.

A continuación se especifica y se detalla los diferentes casos de uso correspondientes a todos los roles, estos casos de uso son generales para todos los roles.

Especificación del Caso de uso Login al sistema.

| | |
|------------------------|--|
| CASO DE USO #27 | Login al sistema. |
| ACTORES | Administrador , Secretaría, Auxiliar de Secretaría, Invitado. |

| | | |
|--|---|---|
| OBJETIVO | Permitir el ingreso al sistema y visualizar el menú según el rol que corresponda a ese usuario. | |
| PRECONDICIONES | ✓ El usuario debe estar registrado y debe tener asignado un rol. | |
| POSCONDICIONES | ➤ El usuario ingresa a la pantalla principal con un menú de opciones de acuerdo a su rol. | |
| FLUJO DE EVENTO | Actividades del Actor | <ol style="list-style-type: none"> 1. Digitar nombre de usuario y contraseña. 2. Pulsar el botón inicio de sesión o dar un enter. |
| | Respuesta del Sistema | <ol style="list-style-type: none"> 3. El sistema comprueba la validez del usuario y contraseña. 4. Muestra el menú del sistema de acuerdo con el rol del usuario. |
| MANEJO DE SITUACIONES EXCEPCIONALES | ✓ Si el nombre del usuario o la contraseña son incorrectos el sistema mostrará un mensaje indicando que los datos ingresados son incorrectos. | |

Tabla 30: Caso de uso Login al sistema
Elaborador por: El Investigador.

Especificación del Caso de uso Consultar Calendarios, Tabla de posiciones, goleadores y jugadores sancionados o suspendidos.

| | | |
|--|---|--|
| CASO DE USO #28 | Consultar Calendarios, Tabla de posiciones, goleadores y jugadores sancionados o suspendidos. | |
| ACTORES | Secretaría, Administrador, Invitado, Auxiliar Secretaría | |
| OBJETIVO | Permitir consultar y generar reportes de Calendarios, Tabla de posiciones, goleadores y jugadores sancionados o suspendidos. | |
| PRECONDICIONES | ✓ Tener acceso a los módulos de Consultas. | |
| POSCONDICIONES | ➤ Se genera un reporte según el caso que corresponda. | |
| FLUJO DE EVENTO | Actividades del Actor | <ol style="list-style-type: none"> 1. Elegir la opción Correspondiente. 3. Pulsar el botón Tabla o Generar Reporte según sea el caso. |
| | Respuesta del Sistema | <ol style="list-style-type: none"> 2. El sistema despliega un formulario. 4. El sistema despliega un formulario con el reporte el cual puede ser guardado o puede imprimirse directamente. |
| MANEJO DE SITUACIONES EXCEPCIONALES | ✓ Si se genera algún error en el envío de los parámetros para la generación del reporte, el sistema indicara esta situación con un mensaje. | |

Tabla 31: Caso de uso consultar y generar reportes
Elaborador por: El Investigador.

4.3.2 Diseño de los diagramas de secuencias

Los diagramas de secuencia permiten entender de mejor manera la interacción del usuario con la interfaz y la interfaz con la base de datos. Se han diseñado algunos diagramas de secuencia para mostrar la interacción.

Diagrama se secuencia Login de Usuarios.

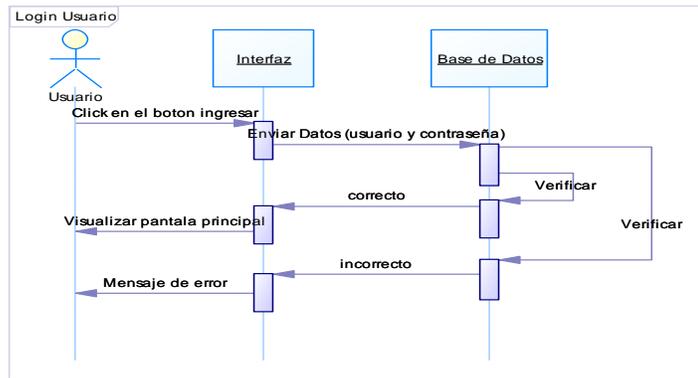


Fig. 10: Diagrama de secuencia Login Usuario
Elaborado por: El Investigador.

Diagrama se secuencia Mantenimiento de equipo.

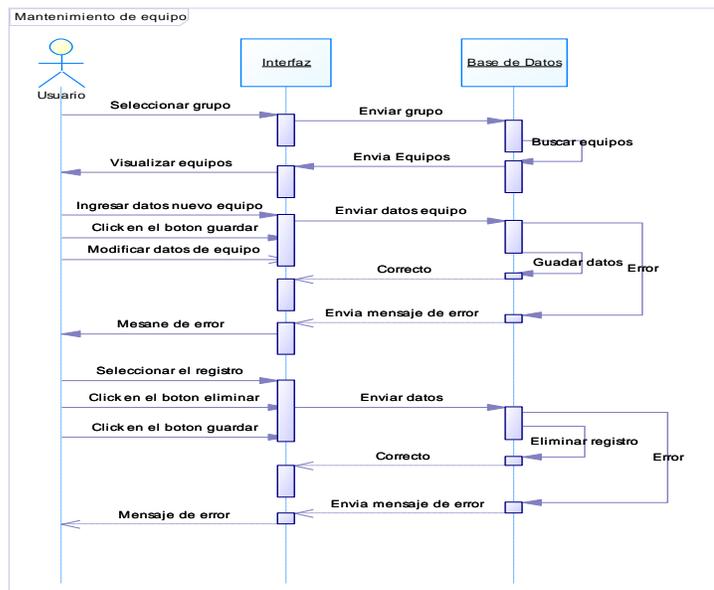


Fig. 11: Diagrama de secuencia Mantenimiento de equipo
Elaborado por: El Investigador.

Diagrama de secuencia Mantenimiento de jugador.

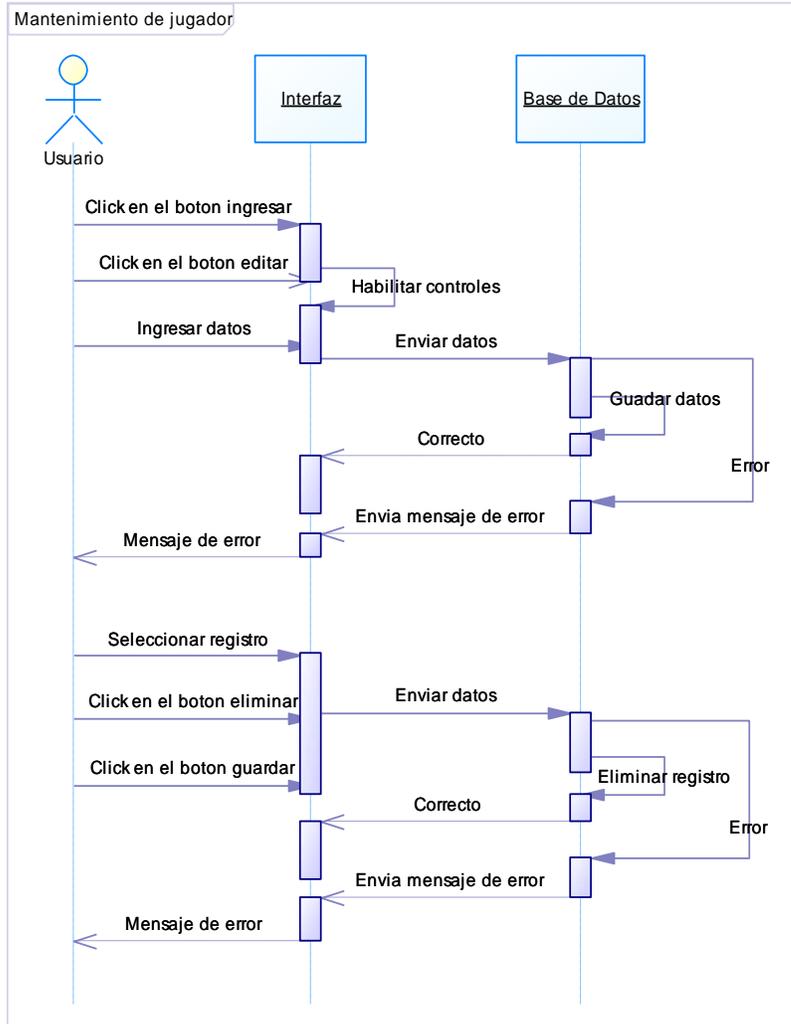


Fig. 12: Diagrama de secuencia Mantenimiento de jugador
Elaborado por: El Investigador.

Diagrama de secuencia Mantenimiento de calendario.

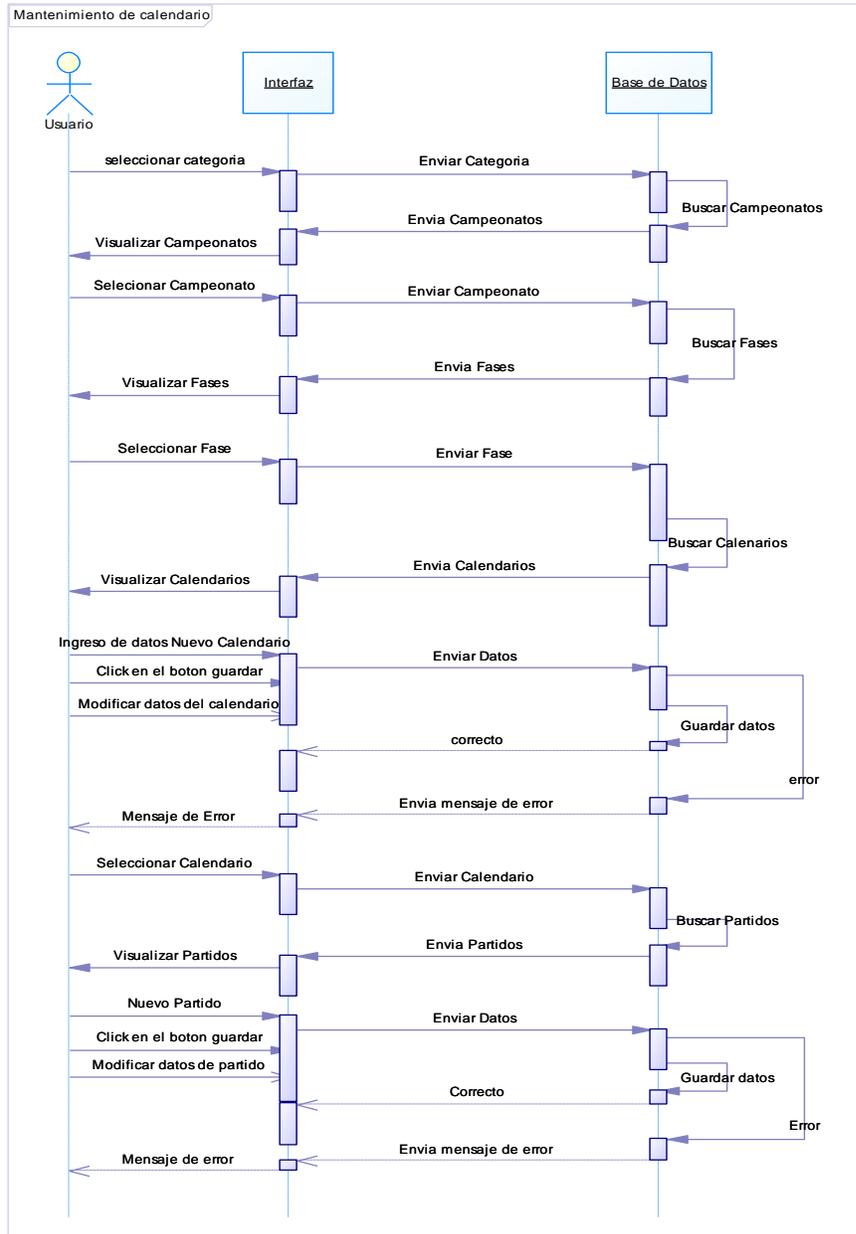


Fig. 13: Diagrama de secuencia Mantenimiento de calendario
Elaborado por: El Investigador.

Diagrama de secuencia Mantenimiento de partido.

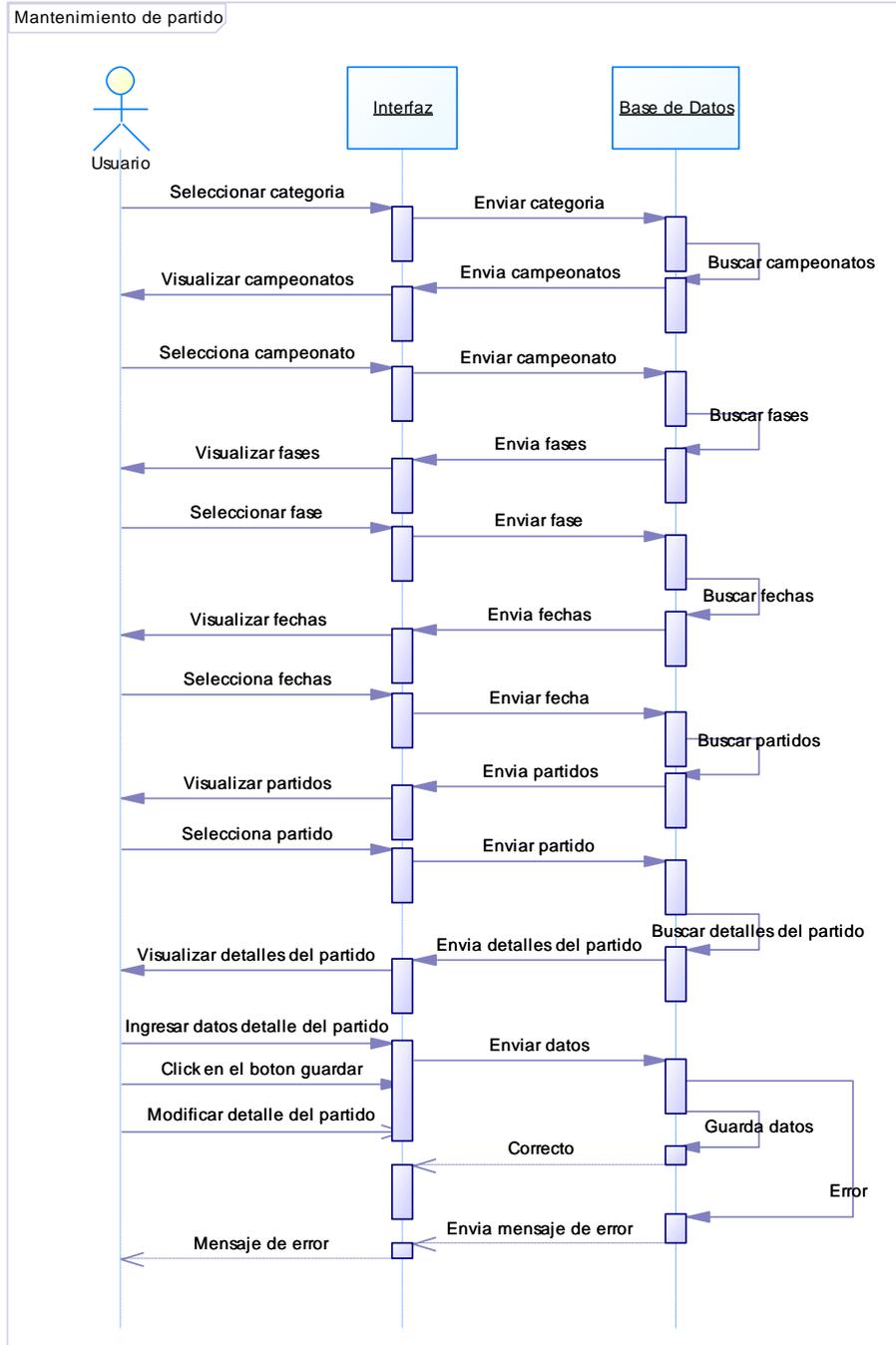


Fig. 14: Diagrama de secuencia Mantenimiento de partido
Elaborado por: El Investigador.

4.3.3 Diseño de los diagramas de flujo de datos

Para el presente proyecto solo se ha considerado necesario hacer el diseño de flujo de datos de nivel 0. Ya que se utilizó casos de uso para detallar más a profundidad los procesos. A continuación de determinan las principales entradas y salidas que tendrá el sistema.

Diagrama de flujo de datos nivel 0.

Nivel 0 (contextos)

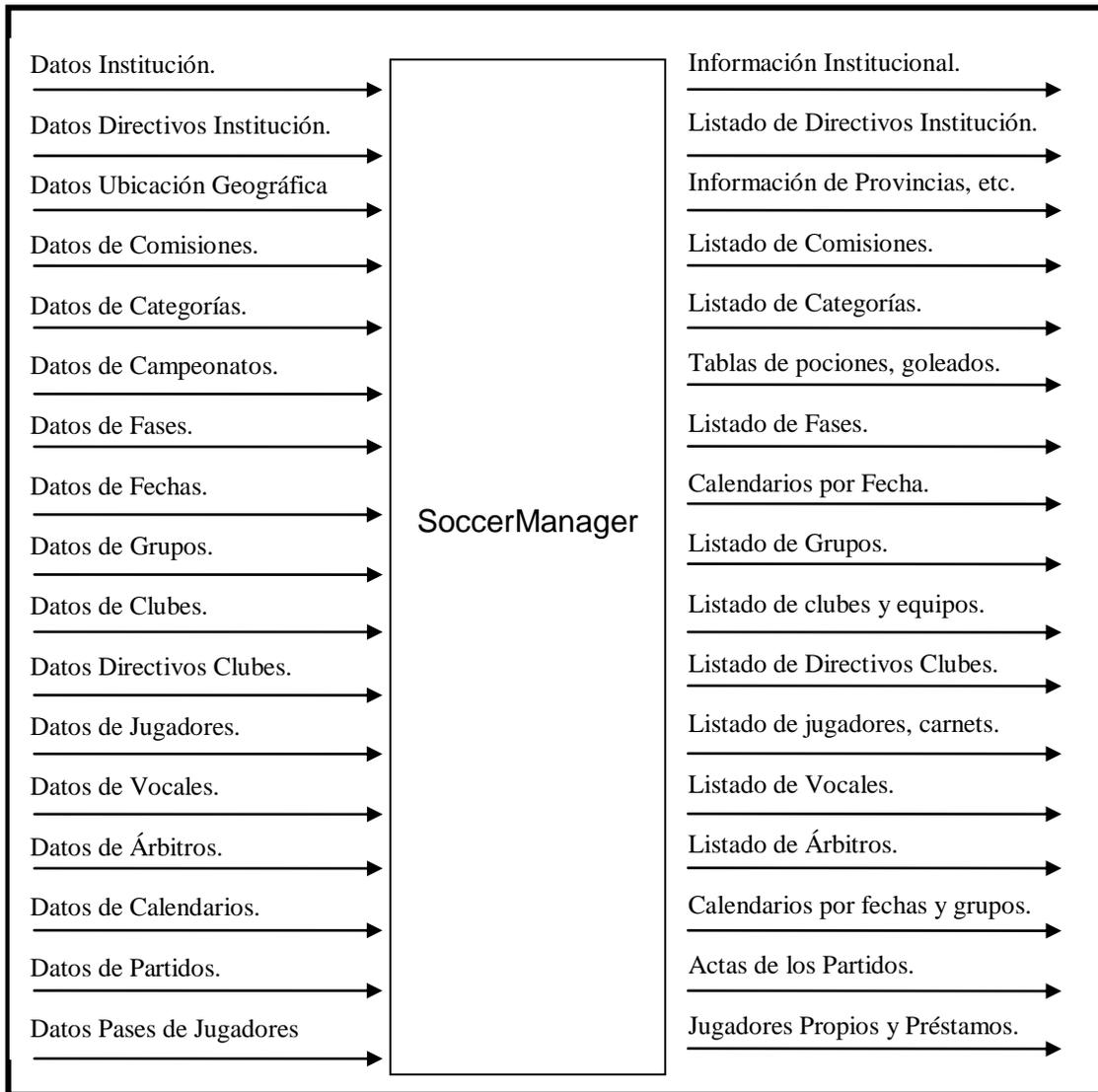


Fig. 15: Diagrama de flujo de datos nivel
Elaborador por: El investigador.

4.3.4 Diseño de los diagramas de estados

Los diagramas de estados permiten modelar el ciclo de vida que tiene y objeto y su comportamiento en sus diferentes etapas. Así como también permiten determinar cuáles son los estados que tiene un determinado objeto, que acción se puede tomar en cada estado, también nos ayudan para el diseño de la base de datos. Estos son los diagramas de estados de los principales objetos.

Diagrama de estado crear equipo.

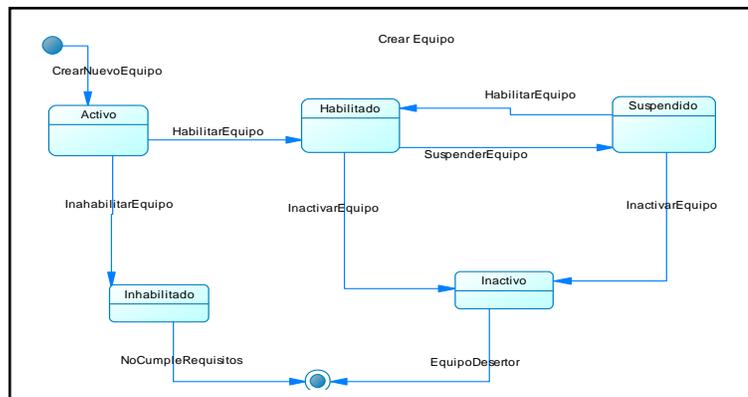


Fig. 16: Diagrama de estado crear equipo
Elaborador por: El investigador.

Diagrama de estado registro de jugador.

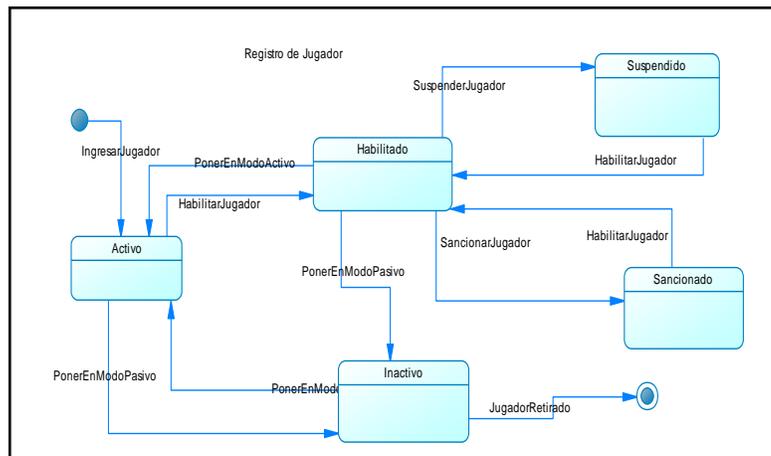


Fig. 17: Diagrama de estado registro de jugador
Elaborador por: El investigador.

Diagrama de estado crear calendario.

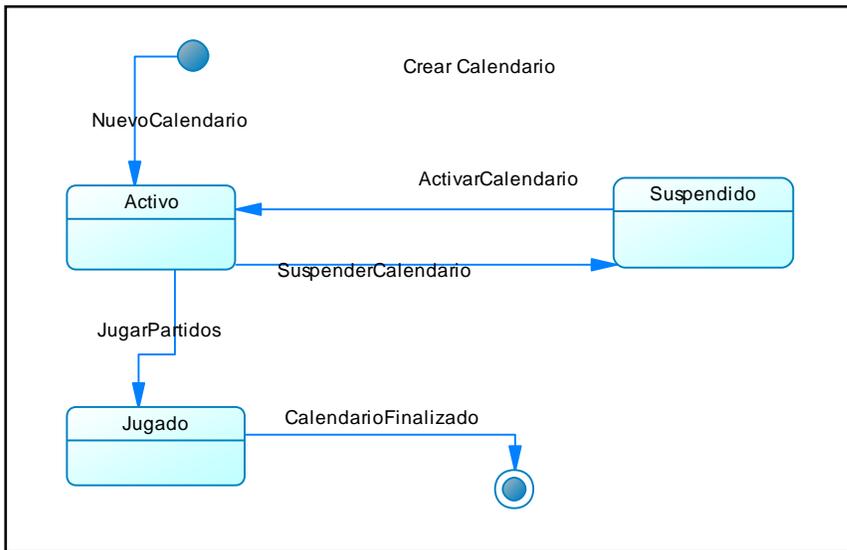


Fig. 18: Diagrama de estado crear calendario
Elaborador por: El investigador.

Diagrama de estado nueva multa.

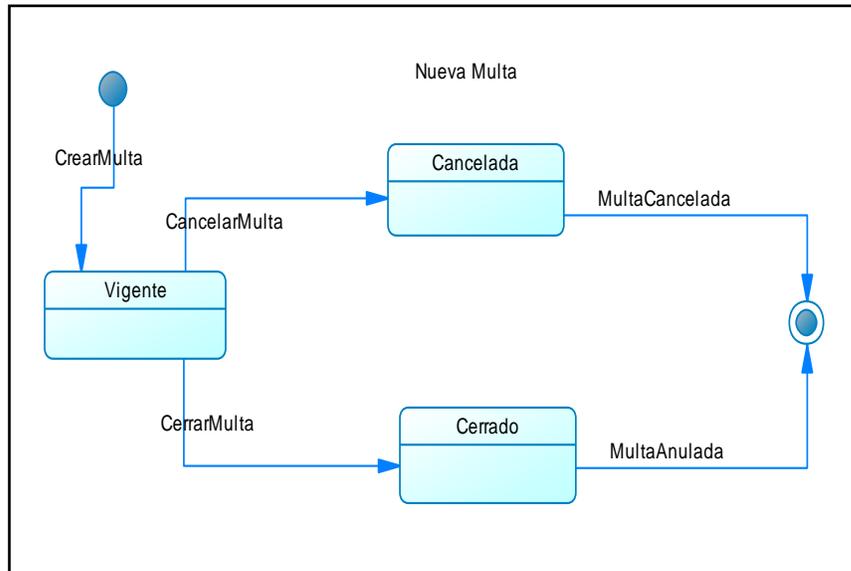


Fig. 19: Diagrama de estado nueva multa
Elaborador por: El investigador.

Diagrama de estado nueva sanción.

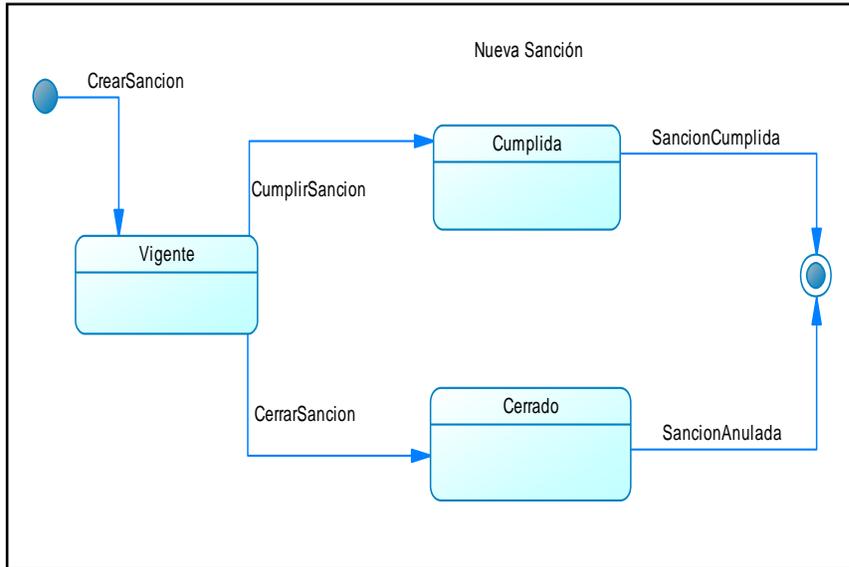


Fig.20: Diagrama de estado nueva sanción
Elaborador por: El investigador.

Diagrama de estado creación de empleado.

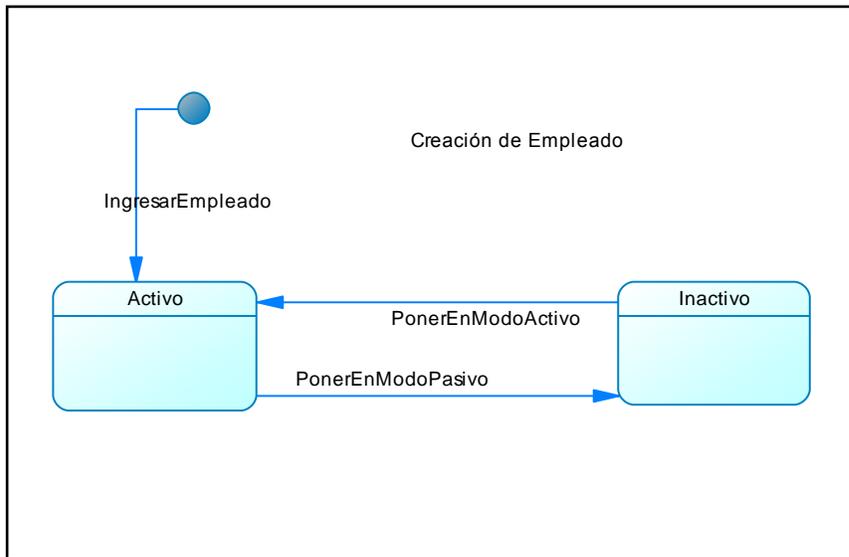


Fig. 21: Diagrama de estado creación de empleado
Elaborador por: El investigador.

Diagrama de estado creación de directivo.

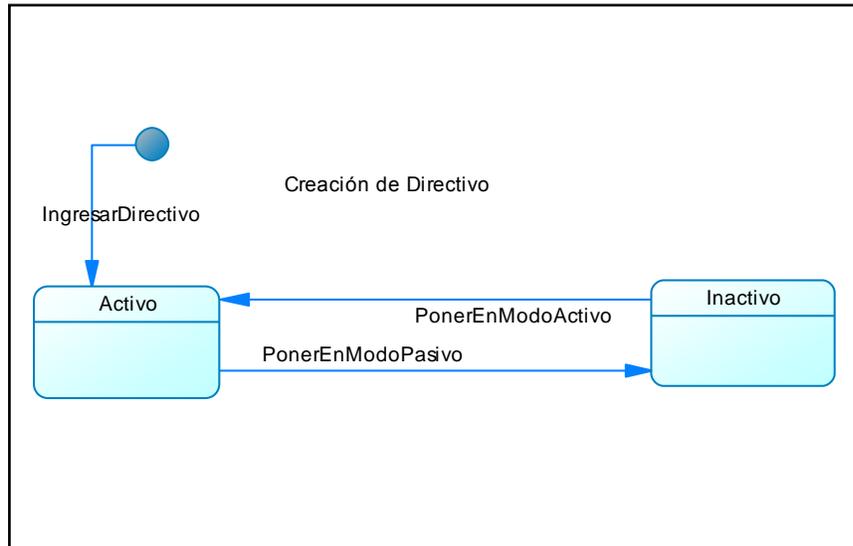


Fig. 22: Diagrama de estado creación de directivo
Elaborador por: El investigador.

Diagrama de estado creación de vocal.

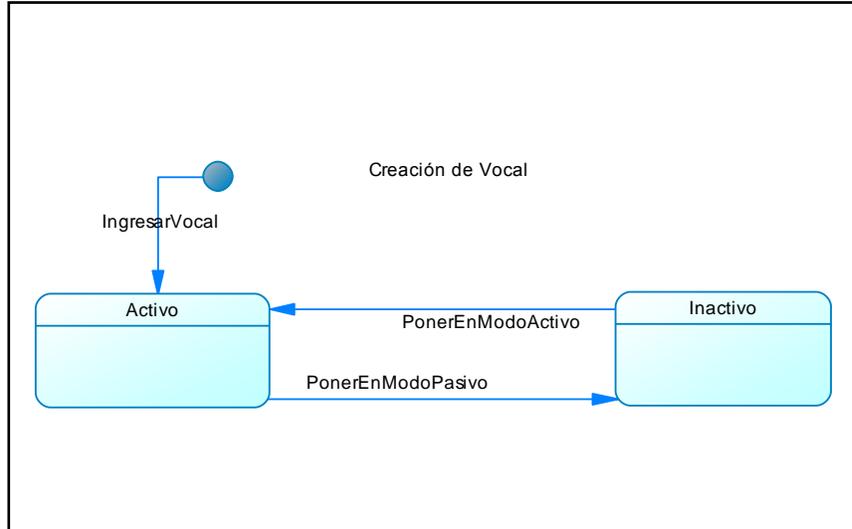


Fig. 23: Diagrama de estado creación de vocal
Elaborador por: El investigador.

4.3.5 Diseño de la base de datos

El correcto diseño de una base de datos, permite tener acceso a la información de forma rápida y ágil. Ya que un diseño correcto nos permite alcanzar los objetivos trazados en el proyecto.

Para el diseño de la base de datos del presente proyecto se ha considerado conveniente dividirla por módulos, esto con el fin de facilitar el diseño de la misma y comprender de mejor manera los requerimientos establecidos y especificados por el usuario, los cuales se encuentran detallados en la parte de especificación de requerimientos. También se ha considerado necesario establecer algunas pequeñas reglas para los nombres de los objetos, para los atributos de dichos objetos y en general para el modelado de la base de datos, con el fin de estandarizar y mejorar la comprensión de los diseños de la base de datos.

Reglas para el modelado de la base de datos:

- Se dividirá en módulos de acuerdo a su funcionalidad.
- A cada módulo se le pondrá un prefijo correspondiente el cual será de tres letras seguido de un guion bajo y luego el nombre del objeto.
- Los nombres de las tablas (Objetos) serán de 8 caracteres máximo.
- Los nombres de los campos serán: si el nombre del campo está compuesto de una sola palabra este será de 4 caracteres máximo, si está compuesto de dos palabras el nombre del campo será de 6 caracteres máximo tres caracteres de la una y tres de la otra palabra, si el nombre del campo tiene tres palabras el nombre del campo será de 9 caracteres máximo.
- Las claves primaria de todas la tablas será un ID auto numérico.
- Los nombres de las vistas tendrán el prefijo vw seguido de un guion bajo, luego el prefijo del módulo correspondiente, luego un guion bajo seguido del nombre de la consulta.

La base de datos está compuesta por los siguientes módulos:

- **Módulo General:** en este módulo se encontraran todas la tablas que contengan los parámetros necesarios para el funcionamiento del sistema, además estas tablas contendrán datos de forma general los cuales podrían ser utilizados por otras aplicaciones ya que estas tablas tendrán un diseño generalizado por ello estas tablas estarán dentro del módulo general, el prefijo de estas tablas será **GEN_NombreObjeto**.
- **Módulo Ligas:** el cual contendrá los objetos es decir las tablas específicas para la administración, manejo y control de los campeonatos de fútbol, los cuales controlaran la fiabilidad de la información, el prefijo de las tablas que se encuentren en este módulo será **LIG_NombreObjeto**.
- **Módulo de Recurso Humanos:** aquí estarán todas las tablas que contenga información del personal que labora en la institución, es decir información de los empleados, el prefijo de este módulo será **RHH_ NombreObjeto**.
- **Módulo de Seguridad:** el cual contendrá todas las tablas y objetos en los cuales se guarda la información con la cual se brinda seguridad al sistema. Este módulo no tendrá ningún prefijo ya que los objetos son generados de forma automática por la herramienta de desarrollo, en este caso por Microsoft LigthSwitch, esta herramienta utiliza los Member Ship de SQL Server para dar seguridad a la aplicación.

4.3.6 Creación del modelo lógico de la base de datos

El modelo lógico del presente proyecto se encuentra dividido en 4 módulos que son: el módulo general, el módulo ligas, el módulo de seguridad que es generado de forma automática por la propia herramienta de desarrollo y el módulo de recursos humanos.

| | | | | |
|--------|---------------|------------|----------------|----------------|
| Visión | Planificación | Desarrollo | Estabilización | Implementación |
|--------|---------------|------------|----------------|----------------|

Módulos de la base de datos.

| Nombres de los módulos |
|----------------------------|
| Módulo General |
| Módulo Ligas |
| Módulo de Seguridad |
| Módulo de Recursos Humanos |

Tabla 32: Módulos de la base de datos
Elaborador por: El investigador.

Modelo Lógico del Módulo Liga

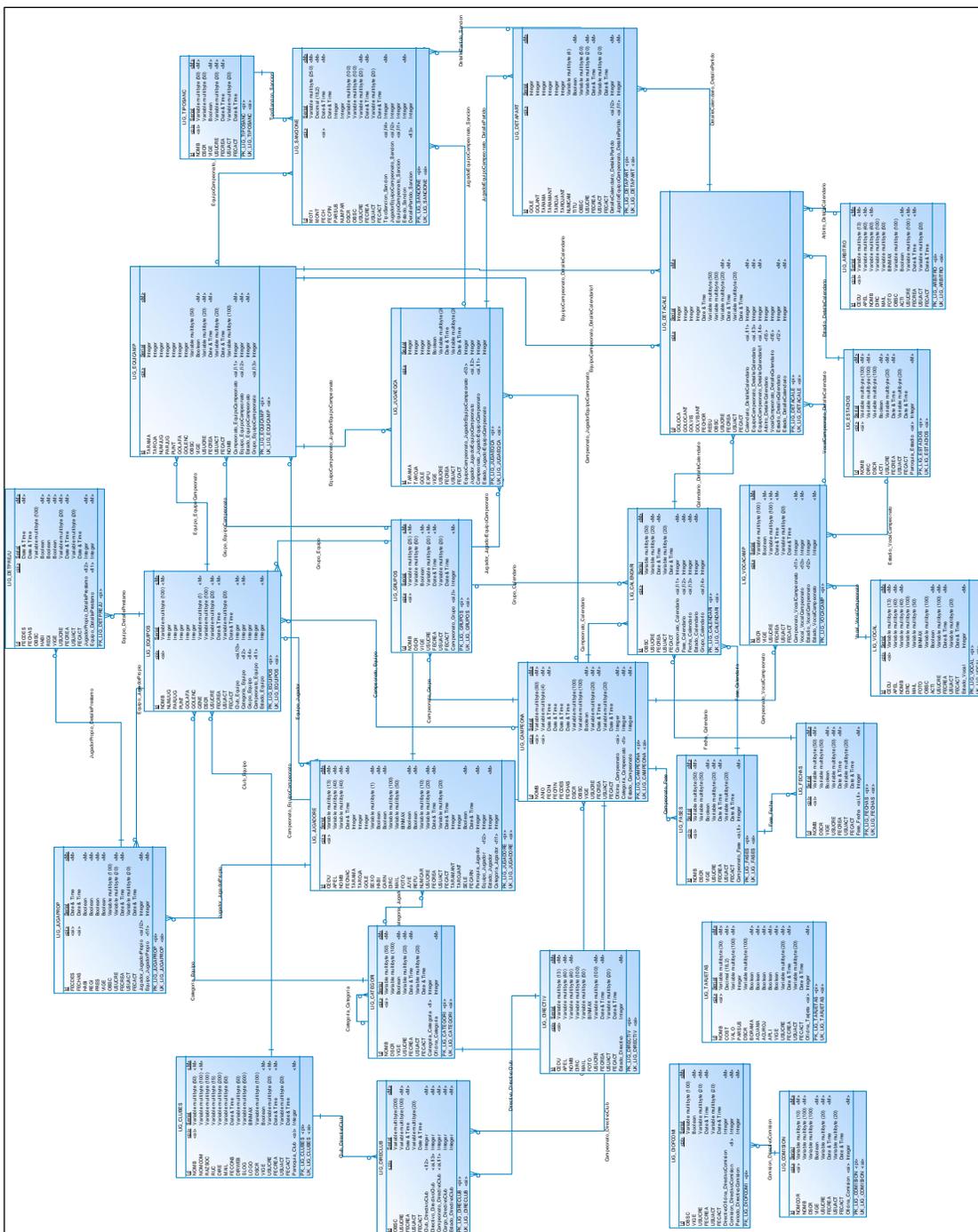


Fig. 25: Modelo Lógico del Módulo Liga
Elaborador por: El investigador.

Modelo Lógico del Módulo Seguridad.

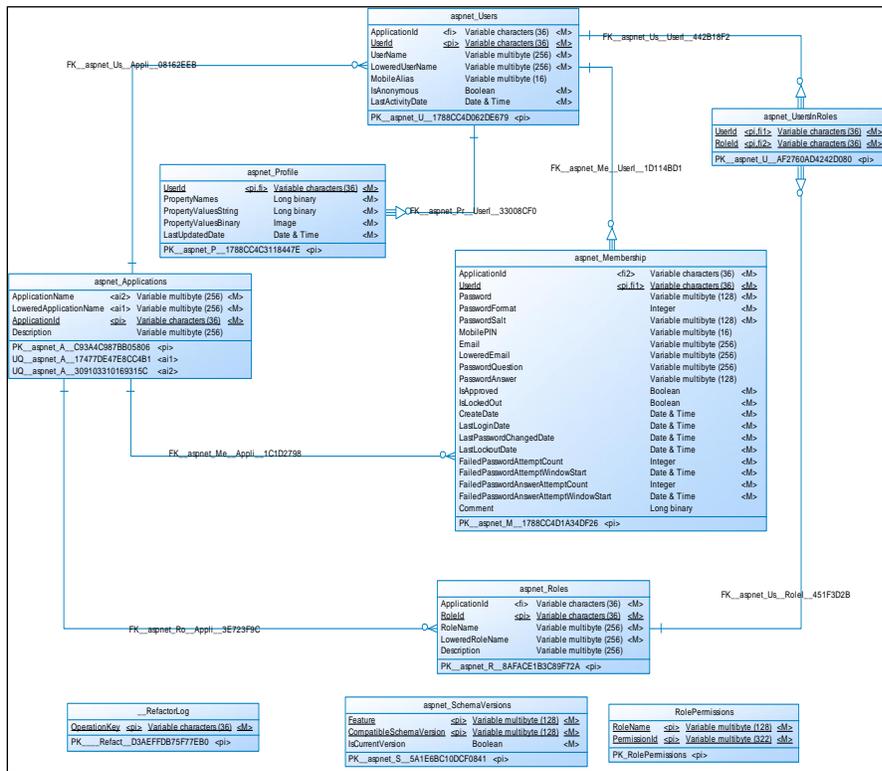


Fig. 26: Modelo Lógico del Módulo de Seguridad
Elaborador por: El investigador

Modelo Lógico del Módulo de Recursos Humanos.

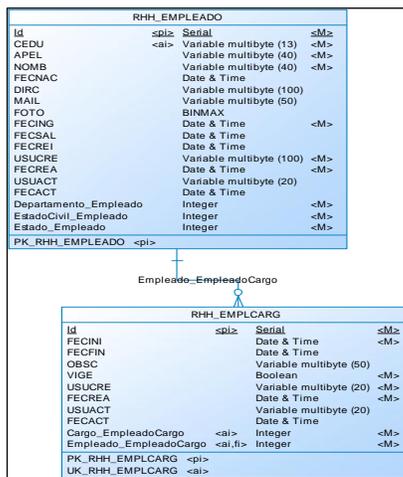


Fig. 27: Modelo Lógico del Módulo de Recursos Humanos
Elaborador por: El investigador.

4.3.7 Diseño del modelo físico de la base de datos

El modelo físico es una descripción de la implementación de una base de datos, este depende del tipo de SGDB y describe las estructuras de almacenamiento y los métodos de acceso a los datos. El modelo físico también es el proceso de traducción del modelo lógico de base de datos.

Modelo Físico del Módulo General.

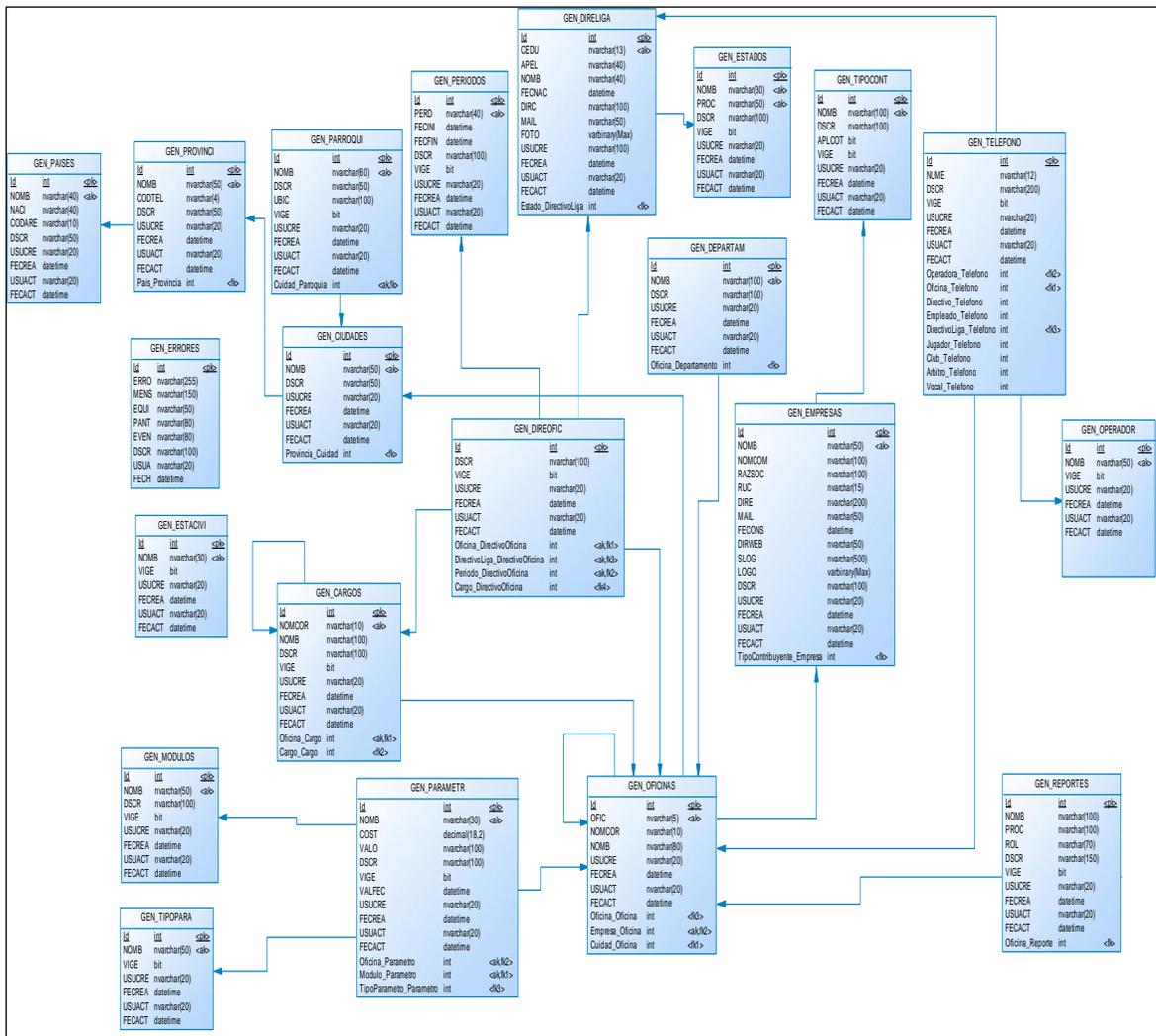


Fig. 28: Modelo Físico del Módulo General
Elaborador por: El investigador.

Modelo Físico del Módulo Liga.

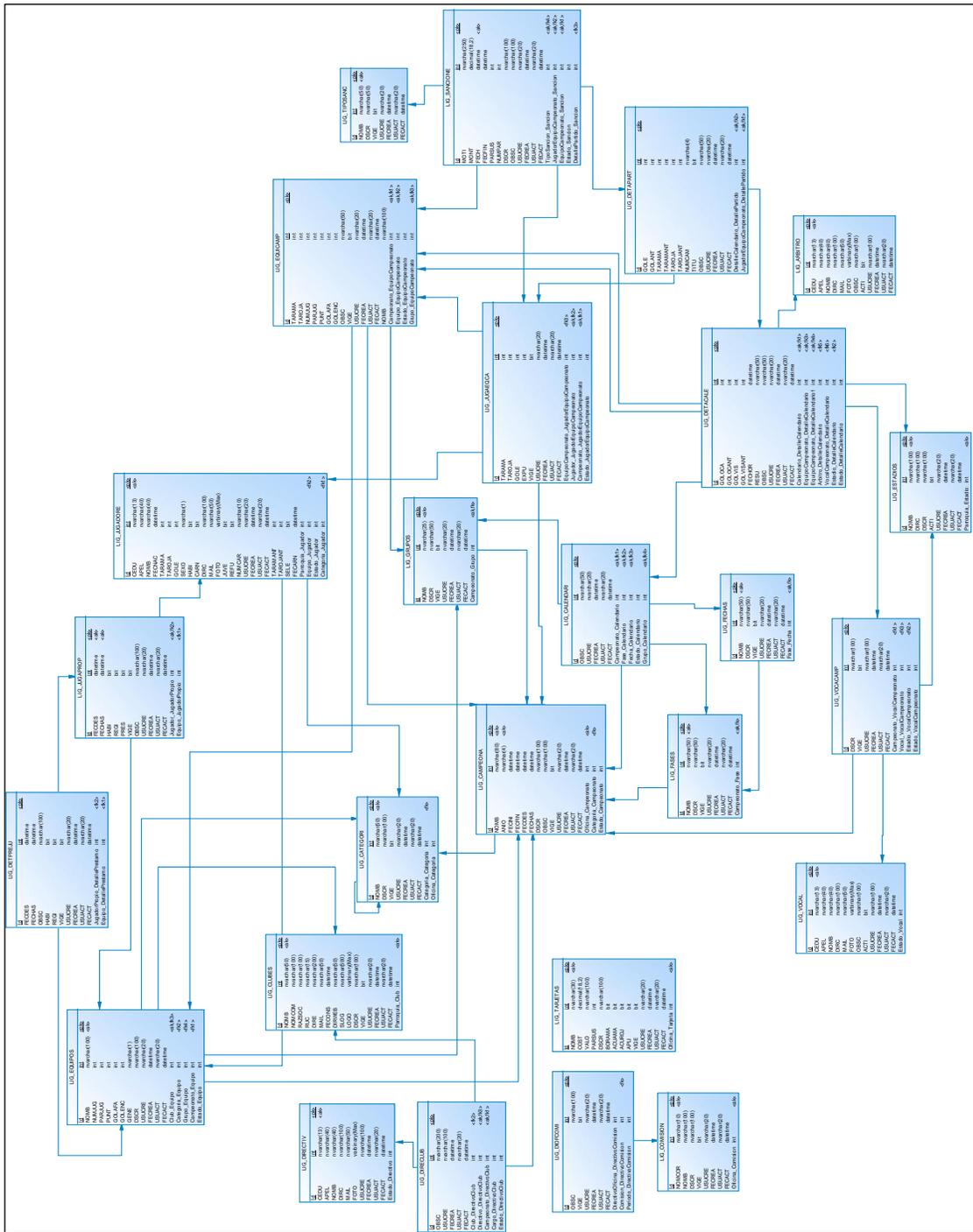


Fig. 29: Modelo Físico del Módulo Liga
Elaborador por: El investigador.

Modelo Físico del Módulo de Seguridad.

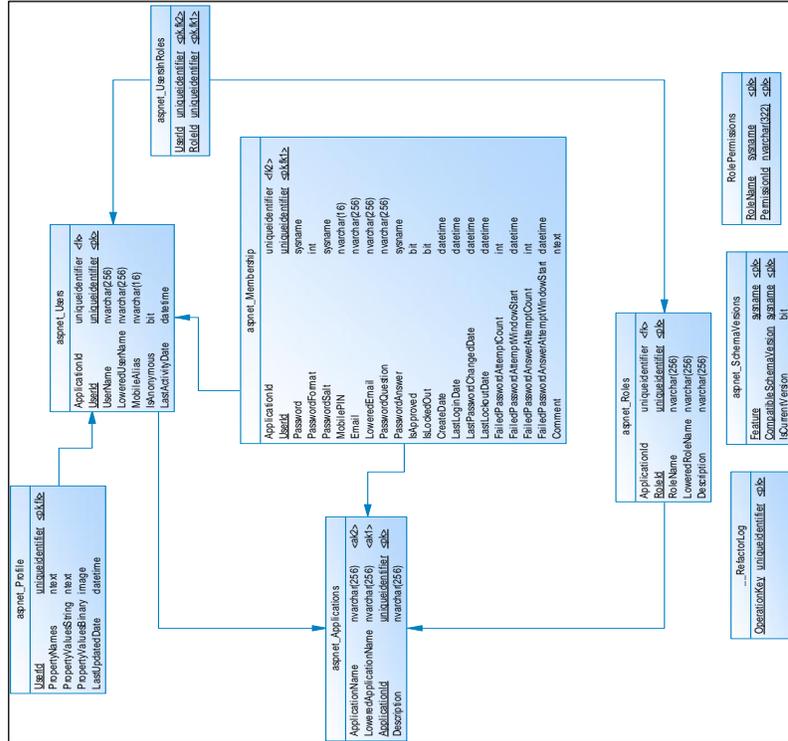


Fig. 30: Modelo Físico del Módulo de Seguridad
Elaborador por: El investigador.

Modelo Físico del Módulo de Recursos Humanos.

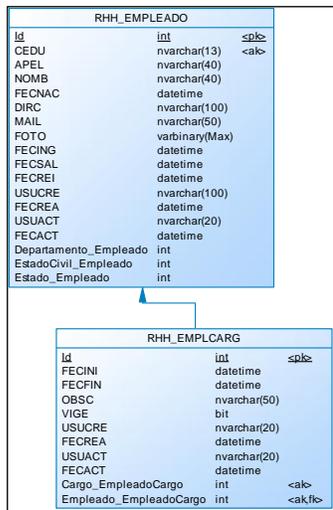


Fig. 31: Modelo Físico del Módulo de Recursos Humanos
Elaborador por: El investigador.

4.3.8 Diseño de las interfaces de usuario

El diseño de la interfaz gráfica de usuario permite determinar cómo estará distribuida la información y los diferentes controles en cada una de las pantallas de la aplicación. Esto permite obtener una vista previa de como quedara la aplicación.

Interfaz de Inicio de Sesión.



Fig. 32: Interfaz de Inicio de Sesión
Elaborador por: El investigador.

- a) **Nombre de la Institución:** En esta área ira el nombre de la institución en nuestro caso ira “LIGA DE PORTIVA CANTONAL PELILEO”.
- b) **Lema de la Institución:** En esta sesión se escribirá un lema que vaya orientado a fomentar la práctica del deporte.
- c) **Banner de imágenes:** Aquí se ubicarán imágenes referentes al deporte en este caso al fútbol o imágenes institucionales.

d) Mensajes de las imágenes: En esta sesión se escribirán mensajes que hagan referencia a las imágenes o a las funcionalidades del sistema.

e) Información: Es como un pequeño banner para poner información y una imagen.

f) Inicio de sesión: En esta área se ubican los componentes necesarios para que permita al usuario iniciar una sesión en el sistema.

g) Información del desarrollador de la aplicación: En esta área se pondrán datos de la persona que desarrolló la aplicación.

Interfaz de pantallas de consultas y manipulación de datos.

Todas las pantallas ya sean de consulta o de manipulación de datos tendrán el siguiente aspecto. Y las siguientes partes.

Pantallas de consultas y manipulación de datos

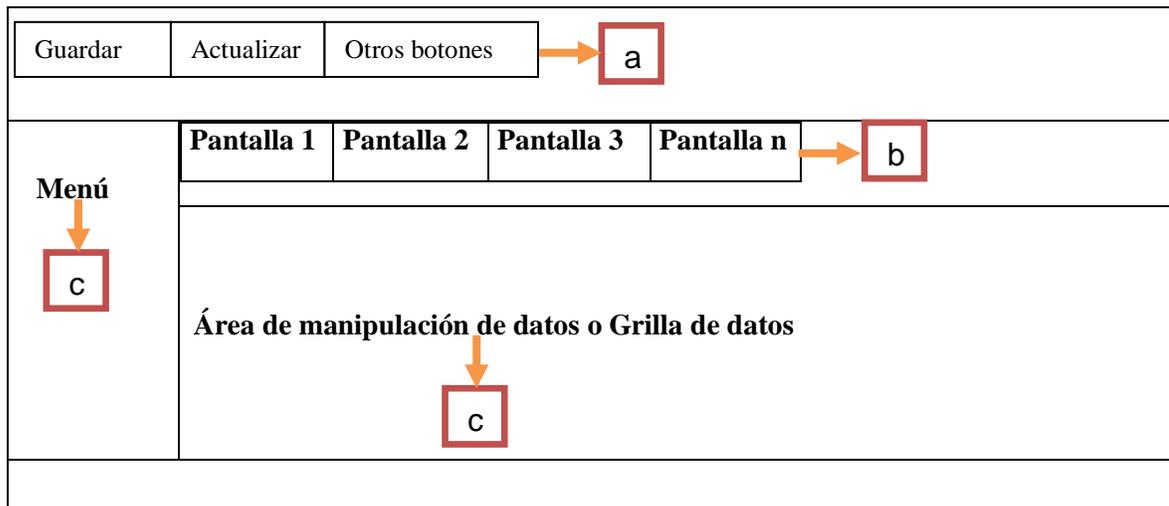


Fig. 33 Interfaz de pantallas de consultas y manipulación de datos
Elaborador por: El investigador.

a) Botones de acciones: Ejecutan una determina acción ya sea sobre un objeto específico o sobre toda la aplicación.

Guardar: el cual como su nombre lo indica guarda los datos en la base de datos de la pantalla que está activa, en caso de existir algún error al guardar los datos ya sea

por campos que son obligatorios o por cualquier otro motivo estos errores se despliegan en la cabecera de la grilla de datos.

Actualizar: este botón refresca los datos de la pantalla que está activa.

Otros botones: en esta área se ubicaran botones adicionales creados en dependencia de las necesidades que requiera cada pantalla.

b) Área de Pantallas activas: En esta área están todas las pantallas activas que han sido levantadas por el usuario.

c) Menú: Contiene el menú de la aplicación agrupado en diferentes ítems o categorías.

d) Área de trabajo o de manipulación de datos: En esta área se encuentran los datos, estos datos pueden estar en diversos compones como pueden ser, combo box, grillas de datos, text box o cualquier otro componente que permita la visualización y la manipulación de datos.

Interfaz área de trabajo o de manipulación de datos.

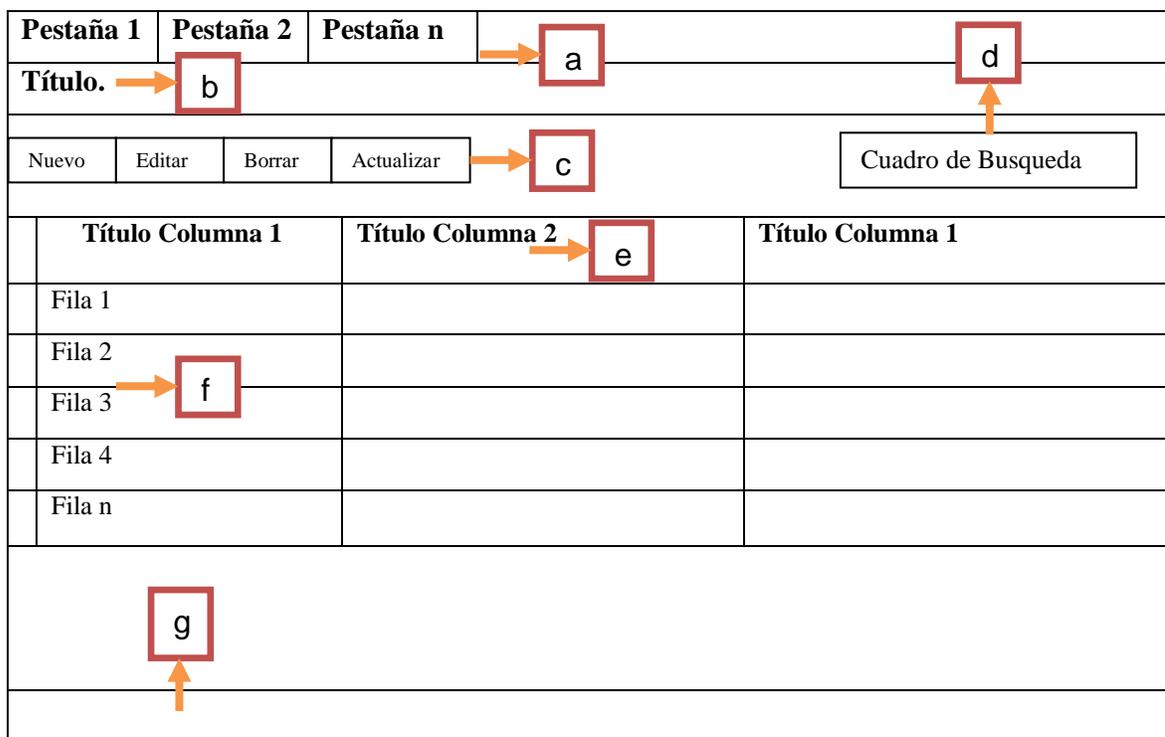


Fig. 34 Interfaz área de trabajo o de manipulación de datos
Elaborador por: El investigador.

a) Pestañas del área de manipulación de datos: Son las pestañas que tiene una grilla, cabe indicar que no todas las grillas tienen estas pestañas. Solo tendrán aquellas en las que se necesite agrupar los datos.

b) Título: Título de la grilla de datos.

c) Botones de manipulación de datos: Estos botones realizan acciones directamente sobre los datos es decir adicionan un nuevo registro, modifican un registro, eliminan un registro y refrescan los datos. Los botones de manipulación de datos necesitan que estas acciones se guarden en la base de datos.

- ✓ **Nuevo:** Permite agregar un nuevo registro.
- ✓ **Editar:** Permite modificar un registro.
- ✓ **Borrar:** Permite eliminar un registro.
- ✓ **Actualizar:** Permite refrescar los datos de la pantalla.

d) Cuadro de búsqueda: En este cuadro se pueden realizar búsquedas sobre algunos campos de una grilla que sean de tipo texto.

e) Título de las columnas: Se ubican los títulos de las columnas de la grilla que contiene datos.

f) Filas de datos: Filas de datos mostradas en forma de tablas o grilla, las cuales pueden ser editadas como en una hoja de cálculo.

g) Índices de navegación: Estos índices permiten navegar entre páginas de una pantalla, estas páginas se crean automáticamente en dependencia del volumen de dato por defecto una página contiene 45 registros.

Interfaz de pantallas de consultas y manipulación de datos con filtros en cascada.

Las pantallas con filtros en cascada tendrán una o varias listas de selección también llamados combo box, los cuales están enlazados el uno con el otro es decir hay que seleccionar un valor en el combo box 1 para que se carguen los valores de combo box dos y así sucesivamente hasta que se carguen datos en la grilla.

Pantallas de consultas y manipulación de datos con filtros en cascada.

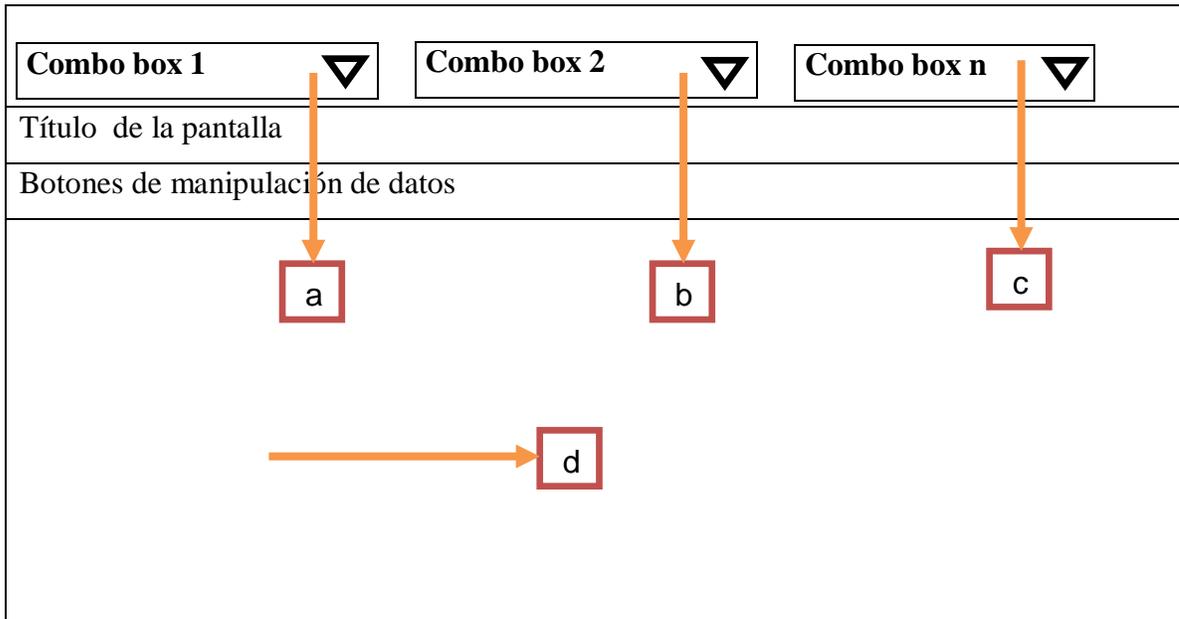


Fig. 35 Interfaz de pantallas de consultas y manipulación de datos con filtros en cascada
Elaborador por: El investigador.

- a) Combo box 1:** Esta lista contiene datos que al ser seleccionado un dato filtra los datos del siguiente combo box en este caso del combo box 2.
- b) Combo box 2:** Esta lista contiene datos que son filtrados en base al dato seleccionado en el combo box uno. Al ser seleccionado un dato de esta lista filtra los datos para el combo box n.
- c) Combo box n:** Contiene datos que son filtrados en base a los datos seleccionados en los combo box 1 y 2, esta a su vez filtra los datos de la grilla principal.
- d) Grilla principal de datos:** Los datos son filtrados en base a los elementos seleccionados en los combo box, estos datos pueden ser se lectura – escritura o solo de lectura.

Interfaz de pantallas de reportes.

Las pantallas de reportes del sistema tendrán el siguiente aspecto.

Pantallas de reportes

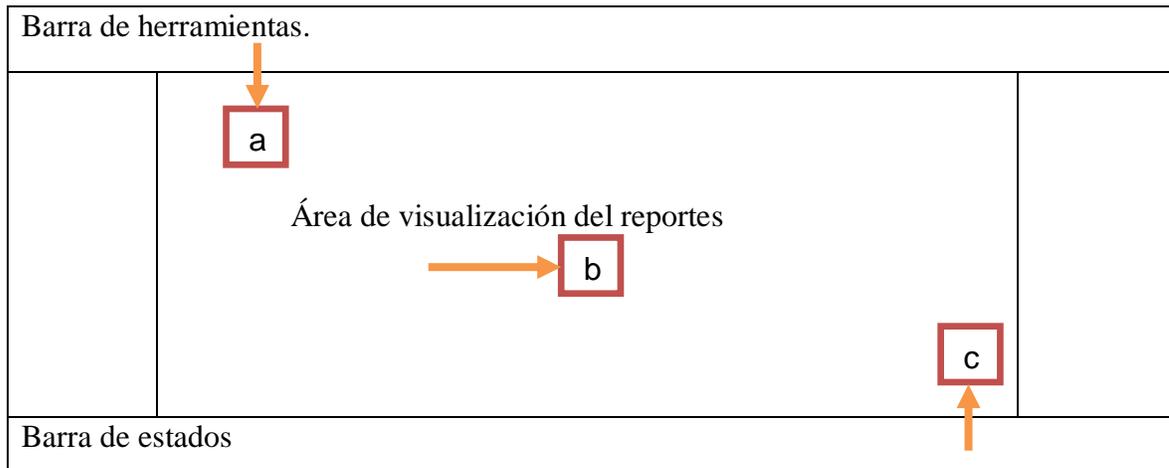


Fig. 36 Interfaz de pantallas de reportes

Elaborador por: El investigador.

a) Barra de herramientas: Esta barra contiene diferentes herramientas que permite la manipulación del reporte, los principales elementos que contenga esta barra serán:

- ✓ Ingreso de parámetros del reporte.
- ✓ Impresión del reporte.
- ✓ Configuración de formato de hoja en la cual se desea imprimir o exportar el reporte.
- ✓ Botones de navegación entre páginas del reporte.
- ✓ Exportar en diferentes formatos el reporte.

b) Área del reporte: Esta el área donde se despliega o se visualiza el reporte.

c) Barra de estados: Contiene información del número de páginas que tiene un reporte, así como también tamaño del Zoom del reporte con el cual se encuentra actualmente.

4.3.9 Creación de la capa de negocios en LightSwitch.

Para crear la capa de negocios en LightSwitch tenemos que pasar todas las tablas con cada uno de sus atributos del modelo físico de la base de datos a objetos los cuales son la representación de una tabla, así como también deberemos pasar las relaciones que tiene las

tablas, las únicas tablas que no pasaremos a la capa de negocios serán las del módulo de seguridad ya que estas son generadas de forma automática por la herramienta de desarrollo, es decir por LightSwitch.

Los objetos que representan a las tablas deberán tener nombres descriptivos de acuerdo a la tabla a la cual representan, estos no podrán tener caracteres especiales.

Capa de negocio de LightSwitch.

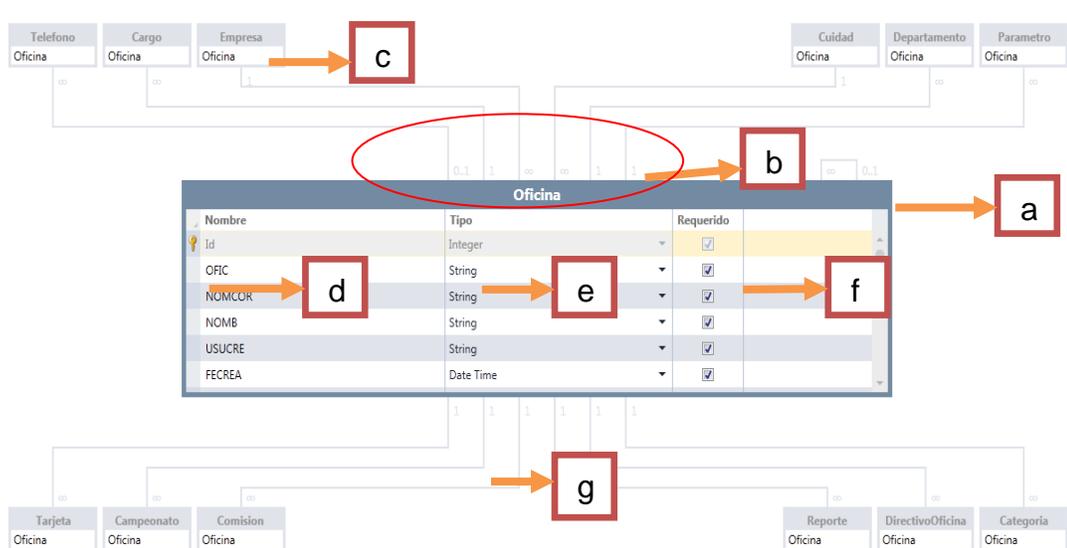


Fig. 37 Capa de negocios en LightSwitch
Elaborador por: El investigador

a) Tabla: Al crear la capa de negocios una tabla se convierte en un objeto y es la representación de una tabla de base de datos, este objeto puede tener un nombre diferente al nombre de la tabla de base de datos que representa.

Este objeto tiene propiedades las cuales representan a los atributos de una tabla, estas propiedades a su vez también tienen propiedades las cuales representan a las características que tienen los atributos de una tabla.

b) Cardinalidad: Es la forma en la que se relacionan las entidades en nuestro caso sería la forma en la cual se relacionan los objetos, que puede ser (1 - ∞, 0.1 - ∞). La primera relación es uno a muchos de forma obligatoria, el segundo un objeto hijo puede como no

puede tener referencia del objeto padre es una relación no obligatoria. Y representa como están relacionados los objetos entre sí.

c) **Objetos:** Son los objetos con los cuales se relaciona un objeto. Estos pueden ser los objetos padres o los objetos hijos.

d) **Propiedades de un objeto o campos de una tabla:** Representan los campos que tiene una tabla en la base de datos, en un objeto representa las propiedades que tiene este objeto que su vez estas propiedades pueden tener también sus propias propiedades.

e) **Tipo:** Es una propiedad de un campo, indica el tipo de dato que tiene ese campo en un objeto indica el tipo de dato que tiene una propiedad.

f) **Requerido:** Indica si un campo es requerido o no es requerido.

g) **Relación:** Indica con que objetos tiene relación o que objetos están relacionados.

Una vez pasadas todas las tablas y relaciones del modelo físico a la capa de negocios tendremos lo siguiente.

Tablas de la capa de negocio.

| Tabla | Tabla |
|---------------|--------------|
| GEN_CARGOS | LIG_DETACALE |
| GEN_CIUDADES | LIG_DETAPART |
| GEN_DEPARTAM | LIG_DETPREJU |
| GEN_DIRELIGA | LIG_DIOFCOMI |
| GEN_DIREOFIC | LIG_DIRECLUB |
| GEN_EMPRESAS | LIG_DIRECTIV |
| GEN_ERRORES | LIG_EQUICAMP |
| GEN_ESTACTIVI | LIG_EQUIPOS |
| GEN_ESTADOS | LIG_ESTADIOS |
| GEN_MODULOS | LIG_FASES |
| GEN_OFICINAS | LIG_FECHAS |
| GEN_OPERADOR | LIG_GRUPOS |
| GEN_PAISES | LIG_JUGADORE |
| GEN_PARAMETR | LIG_JUGAEQCA |
| GEN_PARROQUI | LIG_JUGAPROP |
| GEN_PERIODOS | LIG_SANCIONE |
| GEN_PROVINCI | LIG_TARJETAS |
| GEN_REPORTES | LIG_TIPOSANC |
| GEN_TELEFONO | LIG_VOCACAMP |
| GEN_TIPOCONT | LIG_VOCAL |
| GEN_TIPOPORA | RHH_EMPLCARG |
| LIG_ARBITRO | RHH_EMPLEADO |
| LIG_CALENDARI | |
| LIG_CAMPEONA | |
| LIG_CATEGORI | |
| LIG_CLUBES | |
| LIG_COMISION | |

Fig. 38 Listado de tablas en la capa de negocios en LightSwitch
Elaborador por: El investigador.

Cada tabla en la capa de negocios tienen los siguientes métodos. El primer método registra el usuario y la fecha en la que se creó el registro. El segundo método almacena información de cuándo y quién actualizó un registro.

```

/// <summary>
/// Se dispara cuando se crea un nuevo Estado
/// </summary>
partial void Estado_Created()
{
    try
    {
        USUCRE = Application.User.Name;
        VIGE = true;
    }
    catch (Exception exception)
    {

    }
}
/// <summary>
/// Se dispara justo antes de actualizar los datos en el almacen de datos
/// </summary>
/// <param name="entity"></param>
partial void GEN_ESTADOS_Updating(Estado entity)
{
    try
    {
        entity.FECACT = DateTime.Now;
        entity.USUACT = Application.User.Name;
        //SaveChanges();
    }
    catch (Exception exception)
    {

    }
}

```

En la capa de negocios tenemos varios métodos que se ejecutan antes o después de insertar, modificar o borrar un dato, estos métodos funcionan como si fueran triggers de base de datos. Así como también al instante de validar la información ingresada en cada registro. Algunos de estos métodos se describen a continuación.

El siguiente método actualiza en la tabla LIG_JUGADORE los campos de auditoría y la fecha de carnetización si en campo carnetiza en verdadero (true).

```

/// <summary>

```

```

/// Se dispara justo antes de actualizar los datos en el almacen de datos
/// </summary>
/// <param name="entity"></param>
partial void LIG_JUGADORE_Updating(Jugador entity)
{
    try
    {
        //Campos de auditorias
        entity.FECACT = DateTime.Now;
        entity.USUACT = Application.User.Name;
        //Campo para carnetizar
        if (entity.CARN)
        {
            entity.FECARN = DateTime.Now;
        }
    }
    catch (Exception exception)
    {
    }
}

```

A continuación se describen los métodos que actúan antes de hacer un insert, update o delete en la tabla LIG_EQUICAMP, lo que hacen estos métodos es actualizar algunos campos como goles, puntos, partidos jugados y número de jugadores.

#region Equipos en Campeonatos

```

/// <summary>
/// Se dispara justo antes de insertar un elemento en el almacen de datos (EquipoCampeonato)
/// </summary>
/// <param name="entity"></param>
partial void LIG_EQUICAMP_Inserting(EquipoCampeonato entity)
{
    try
    {
        //Actualizar el campeonato el equipo
        entity.Equipo.Campeonato = entity.Campeonato;
        //entity.Equipo.PARJUG = entity.PartidosJugados;
        //entity.Equipo.NUMJUG = entity.NumeroJugadores;
        entity.Equipo.PUNT = entity.PUNT;
        entity.Equipo.GOLAFa = entity.GOLAFa;
        entity.Equipo.GOLENC = entity.GOLENC;
        //Escribir el campo de ordenacion (solo para ordenar)
        entity.NOMB = entity.Equipo.NombreCompleto;
        //Actualizar el numero de jugadores
        //entity.NUMJUG = entity.NumeroJugadores;
    }
    catch (Exception exception)
    {
    }
}

```

```

}
}

/// <summary>
/// Se dispara justo antes de actualizar los datos en el almacen de datos
/// </summary>
/// <param name="entity"></param>
partial void LIG_EQUICAMP_Updating(EquipoCampeonato entity)
{
    try
    {
        //Campos de auditoria
        entity.FECACT = DateTime.Now;
        entity.USUACT = Application.User.Name;
        //Campos de referencia
        if (entity.Estado.NOMB.ToLower() != "Cerrado".ToLower())
        {
            //entity.Equipo.Campeonato = entity.Campeonato;
            entity.Equipo.PARJUG = entity.PartidosJugados;
            entity.Equipo.NUMJUG = entity.NumeroJugadores;
            entity.Equipo.PUNT = entity.PUNT;
            entity.Equipo.GOLAFa = entity.GOLAFa;
            entity.Equipo.GOLENC = entity.GOLENC;
            //Actualizar el numero de jugadores
            entity.NUMJUG = entity.NumeroJugadores;
        }
    }
    catch (Exception exception)
    {
    }
}

/// <summary>
/// Se dispara justo antes de borrar los datos del almacen de datos
/// </summary>
/// <param name="entity"></param>
partial void LIG_EQUICAMP_Deleting(EquipoCampeonato entity)
{
    try
    {
        //Campos de auditoria
        entity.FECACT = DateTime.Now;
        entity.USUACT = Application.User.Name;
        //Campo de referencia
        entity.Equipo.Campeonato = null;
        entity.Equipo.PARJUG = entity.PartidosJugados;
        entity.Equipo.NUMJUG = entity.NumeroJugadores;
        entity.Equipo.PUNT = entity.PUNT;
        entity.Equipo.GOLAFa = entity.GOLAFa;
        entity.Equipo.GOLENC = entity.GOLENC;
        //Actualizar el numero de jugadores
        entity.NUMJUG = entity.NumeroJugadores;
    }
}

```

```

catch (Exception exception)
{
}
}

```

```
#endregion
```

Para controlar si un jugador es juvenil se define un parámetro y en base a ese parámetro y la fecha de nacimiento se determina si es juvenil o no. Existe un parámetro de juvenil por categoría.

```
#region Jugador en equipo campeonato
```

```

/// <summary>
/// Se dispara justo antes de insertar un elemento en el almacen de datos (JugadorEquipoCampeonato)
/// </summary>
/// <param name="entity"></param>
partial void LIG_JUGAEQCA_Inserting(JugadorEquipoCampeonato entity)
{
    try
    {
        //Variables locales
        DateTime parametroFechaJuvenil = DateTime.Now.Date;
        //Campo de clave primaria
        entity.Jugador.Equipo = entity.EquipoCampeonato.Equipo;
        //Campo de referencia
        entity.Campeonato = entity.EquipoCampeonato.Campeonato;
        //Obtener la fase activa del campeonato
        var registro = (from fase in entity.EquipoCampeonato.Campeonato.Fase.Where(x => x.VIGE == true)
            select fase).FirstOrDefault();
        //Asignar el valor
        entity.Fase = registro;
        //Contar el número de jugadores que tiene un equipos
        entity.EquipoCampeonato.NUMJUG = entity.EquipoCampeonato.NumeroJugadores;
        entity.EquipoCampeonato.Equipo.NUMJUG = entity.EquipoCampeonato.NumeroJugadores;
        //Para verificar si es juvenil
        //Buscar el parámetro para ver si es juvenil o no
        var parametroJuvenil = from parametro in GEN_PARAMETR.Where(x => x.NOMB == "Juvenil"
            && x.Categoria.Id == entity.Campeonato.Categoria.Id)
            select parametro;
        //Recorrer
        foreach (var filaObtenida in parametroJuvenil)
        {
            Parametro datosFilaObtenida = (Parametro)filaObtenida;
            parametroFechaJuvenil = datosFilaObtenida.VALFEC.Value;
        }
        //Verificar si es juvenil
        if (entity.Jugador.FECNAC > parametroFechaJuvenil)
        {

```

```

entity.JUVE = true;
entity.Jugador.JUVE = true;
}
//Habilitar
entity.Jugador.CARN = entity.CARN;
}
catch (Exception exception)
{
}
}

/// <summary>
/// Se dispara justo antes de actualizar los datos en el almacen de datos
/// </summary>
/// <param name="entity"></param>
partial void LIG_JUGAEQCA_Updating(JugadorEquipoCampeonato entity)
{
    try
    {
        entity.FECACT = DateTime.Now;
        entity.USUACT = Application.User.Name;
        //Asignar el campeonato en jugadorEnEquipoCampeonato
        if (entity.VIGE)
        {
            //Contar el numero de jugadores que tiene un equipos
            entity.EquipoCampeonato.NUMJUG = entity.EquipoCampeonato.NumeroJugadores;
            entity.EquipoCampeonato.Equipo.NUMJUG = entity.EquipoCampeonato.NumeroJugadores;
        }
        //Para generar listado de jugadores a ser cernetizados
        if (entity.CARN)
        {
            entity.Jugador.CARN = entity.CARN;
        }
    }
    catch (Exception exception)
    {
    }
}

#endregion

```

El detalle del calendario tiene algunos estados y este a su vez pasa a ser la cabecera del partido. Dependiendo del estado en el que se encuentre se asignara los puntos que le correspondan a cada equipo según el resultado final del partido.

```

#region Detalle del Calendario
/// <summary>
/// Se dispara justo antes de actualizar los datos en el almacen de datos

```

```

/// </summary>
/// <param name="entity"></param>
partial void LIG_DETACALE_Updating(DetalleCalendario entity)
{
    try
    {
        //Variables locales
        //Para determinar si existen registros en el detalle del partido
        int registrosDetallePartido = entity.DetallePartido.Count(x => x.DetalleCalendario.Id == entity.Id);
        //Actualizar el estado del partido automaticamente
        if (registrosDetallePartido > 0 )
        {
            if (entity.Estado.NOMB == "Activo")
            {
                var estadoDetalleCalendario = from estados in GEN_ESTADOS.Where(x => x.NOMB ==
                    "Jugado" && x.PROC == "DetalleCalendario")
                    select estados;
                foreach (var x in estadoDetalleCalendario)
                {
                    entity.Estado = (Estado)x;
                }
            }
        }

        //Campos de auditoria
        entity.FECACT = DateTime.Now;
        entity.USUACT = Application.User.Name;

        //Hacer esto solo si el estado del detalle del calendario esta en: Jugado
        if (entity.Estado.NOMB == "Jugado" && entity.Estado.PROC == "DetalleCalendario")
        {
            //Si existen registros en el detalle del partido
            if (registrosDetallePartido >= 0)
            {
                //Para restar los puntos de los equipos
                if (entity.GOLOCANT != null && entity.GOLVISANT != null)
                {
                    //Al equipo local
                    if (entity.GOLOCANT > entity.GOLVISANT)
                        entity.EquipoCampeonatoLocal.PUNT -= 3;
                    //Alequipo visitante
                    if (entity.GOLOCANT < entity.GOLVISANT)
                        entity.EquipoCampeonatoVisitante.PUNT -= 3;
                    //A los dos equipos
                    if (entity.GOLOCANT == entity.GOLVISANT)
                    {
                        entity.EquipoCampeonatoLocal.PUNT -= 1;
                        entity.EquipoCampeonatoVisitante.PUNT -= 1;
                    }
                }
                //Para sumar o asignar los puntos
                //Al equipo local
                if (entity.GOLOCA > entity.GOLVIS)
                    entity.EquipoCampeonatoLocal.PUNT += 3;
            }
        }
    }
}

```

```

//Al equipo visitante
if (entity.GOLOCA < entity.GOLVIS)
    entity.EquipoCampeonatoVisitante.PUNT += 3;
//A los dos equipos
if (entity.GOLOCA == entity.GOLVIS)
{
    entity.EquipoCampeonatoLocal.PUNT += 1;
    entity.EquipoCampeonatoVisitante.PUNT += 1;
}
//Asignar los valores anteriores
entity.GOLOCANT = entity.GOLOCA;
entity.GOLVISANT = entity.GOLVIS;
}
}
//Cuando en el detalle del partido no existen registro y el estado del detalle del calendario es Jugador
if (registrosDetallePartido == 0 && entity.Estado.NOMB == "Activo")
{
    if (entity.GOLOCANT != null && entity.GOLVISANT != null)
    {
        entity.EquipoCampeonatoLocal.PUNT -= 1;
        entity.EquipoCampeonatoVisitante.PUNT -= 1;
    }
}
}
}
catch (Exception exception)
{
}
}
}
#endregion

```

En el método **LIG_DETAPART_Inserting** del detalle del partido se encuentra programada gran parte de la lógica del sistema, este método se ejecuta cuando se inserta un registro en dicha tabla, desde este método se crean multas, sanciones, se ponen tarjetas amarillas y rojas, se asignan los goles tanto a jugadores como a sus equipos, se genera el resultado del partido, etc. En el código se explica que hace cada línea.

```

#region Detalle del partido
/// <summary>
/// Se dispara justo antes de crear el registro en el elmacen de datos
/// </summary>
/// <param name="entity"></param>
partial void LIG_DETAPART_Inserting(DetallePartido entity)
{
    //Variables locales
    int numeroTarjetasAmarillas = 0, numeroPartidosSuspendidos=0;
    string motivoSuspension = "";
    bool swSancion = true, borrarAmarillas = false, acumularAmarillas = false, acumularRojas = false;

```

```

//Si tienes dos amarillas automaticamente tiene una roja
if (entity.TARAMA == 2)
{
    entity.TAROJA = 1;
}
if (entity.DetalleCalendario.GOLOCA == null)
{
    entity.DetalleCalendario.GOLOCA = 0;
}
if (entity.DetalleCalendario.GOLVIS == null)
{
    entity.DetalleCalendario.GOLVIS = 0;
}
//Para el detalle de calendario
//Sumar los goles de local y visitante en la entidad detalle_calendario
if (entity.DetalleCalendario.EquipoCampeonatoLocal.Id ==
entity.JugadorEquipoCampeonato.EquipoCampeonato.Id)
{
    //Para el equipo local
    entity.DetalleCalendario.GOLOCA += entity.GOLE;
    //En la entidad EquipoEnCampeonato
    entity.DetalleCalendario.EquipoCampeonatoLocal.GOLAFa += entity.GOLE;
    entity.DetalleCalendario.EquipoCampeonatoVisitante.GOLENC += entity.GOLE;
    entity.DetalleCalendario.EquipoCampeonatoLocal.TARAMA += entity.TARAMA;
    entity.DetalleCalendario.EquipoCampeonatoLocal.TAROJA += entity.TAROJA;
    //Actualizar los goles a favor y en contra del equipo local
    entity.DetalleCalendario.EquipoCampeonatoLocal.Equipo.GOLAFa += entity.GOLE;
    entity.DetalleCalendario.EquipoCampeonatoVisitante.Equipo.GOLENC += entity.GOLE;
}
else
{
    if (entity.DetalleCalendario.EquipoCampeonatoVisitante.Id ==
entity.JugadorEquipoCampeonato.EquipoCampeonato.Id)
    {
        //Para el equipo visitante
        entity.DetalleCalendario.GOLVIS += entity.GOLE;
        //En la entidad EquipoEnCampeonato
        entity.DetalleCalendario.EquipoCampeonatoVisitante.GOLAFa += entity.GOLE;
        entity.DetalleCalendario.EquipoCampeonatoLocal.GOLENC += entity.GOLE;
        entity.DetalleCalendario.EquipoCampeonatoVisitante.TARAMA += entity.TARAMA;
        entity.DetalleCalendario.EquipoCampeonatoVisitante.TAROJA += entity.TAROJA;
        //Actualizar los goles a favor y en contra del equipo visitante
        entity.DetalleCalendario.EquipoCampeonatoVisitante.Equipo.GOLAFa += entity.GOLE;
        entity.DetalleCalendario.EquipoCampeonatoLocal.Equipo.GOLENC += entity.GOLE;
    }
}
//Sumar los goles en la entidad jugador
entity.JugadorEquipoCampeonato.Jugador.GOLE += entity.GOLE;
//Para la entidad jugadorEnEquipoCampeonato
//Sumar las tarjetas amarillas en la entidad JugadorEnEquipoCampeonato
entity.JugadorEquipoCampeonato.TARAMA += entity.TARAMA;
//Sumar las tarjetas rojas
entity.JugadorEquipoCampeonato.TAROJA += entity.TAROJA;

```

```

//Sumar los goles
entity.JugadorEquipoCampeonato.GOLE += entity.GOLE;
//Actualizar las tarjetas amarillas anteriores
entity.JugadorEquipoCampeonato.Jugador.TARAMANT =
entity.JugadorEquipoCampeonato.Jugador.TARAMA;
entity.JugadorEquipoCampeonato.Jugador.TAROJANT =
entity.JugadorEquipoCampeonato.Jugador.TAROJA;
#region Multa para una tarjeta amarilla
//Generar sanciones y multas por las tarjetas
if (entity.TARAMA == 1 && entity.TAROJA == 0)
{
    //Instanciar un nuevo objeto para generar la multa
    Sancion multa = new Sancion();
    //Buscar los parametros cuando de una tarjeta amarilla
    var parametroTarjetaAmarilla = from parametros in LIG_TARJETAS.Where(x => x.NOMB ==
"TAMARILLA" && x.Oficina.Id == entity.DetalleCalendario.Calendario.Campeonato.Oficina.Id)
    select parametros;
    //Recorer todos los registros
    foreach (var x in parametroTarjetaAmarilla)
    {
        //numeroTarjetasAmarillas = (int)x.VALAMA;
        Tarjeta parametroTarjeta = (Tarjeta)x;
        multa.MONT = parametroTarjeta.COST;
        multa.MOTI = parametroTarjeta.DSCR;
    }
    //Buscar el estado para la sancion
    var estadoSancion = from estado in GEN_ESTADOS.Where(x => x.NOMB == "Vigente" &&
x.PROC == "Multa")
    select estado;
    //Recorrer todos los registros
    foreach (var x in estadoSancion)
    {
        multa.Estado = (Estado)x;
    }
    //Buscar el tipo de la sancion
    var tipoSancion = from tipos in LIG_TIPOSANC.Where(x => x.NOMB == "Multa")
    select tipos;
    //Recorrer todos los registros
    foreach (var x in tipoSancion)
    {
        multa.TipoSancion = (TipoSancion)x;
    }
    //Asignar valores a la sancion
    multa.FECH = Convert.ToDateTime(entity.DetalleCalendario.FECHOR);
    multa.EquipoCampeonato = entity.JugadorEquipoCampeonato.EquipoCampeonato;
    multa.JugadorEquipoCampeonato = entity.JugadorEquipoCampeonato;
    //Sumar las tarjetas amarillas en la entidad jugadores
    entity.JugadorEquipoCampeonato.Jugador.TARAMA += entity.TARAMA;
    multa.DetallePartido = entity;
}
}
#endregion
#region Multa y Sancion por dos tarjetas amarillas y una roja
if (entity.TARAMA == 2 && entity.TAROJA == 1)
{

```

```

//Instanciar un nuevo objeto para generar la multa
Sancion multa = new Sancion();
Sancion sancion = new Sancion();
//Buscar los parametros cuando son dos tarjetas amarillas y una roja
var parametroDosAmarillasyRoja = from parametros in LIG_TARJETAS.Where(x => x.NOMB ==
"T2AMAYROJA" && x.Oficina.Id == entity.DetalleCalendario.Calendario.Campeonato.Oficina.Id)
select parametros;
//Recorer todos los registros
foreach (var x in parametroDosAmarillasyRoja)
{
    //Datos de tipo Tarjeta
    Tarjeta parametroTarjeta = (Tarjeta)x;
    //Para generar la multa
    multa.MONT = parametroTarjeta.COST;
    multa.MOTI = parametroTarjeta.DSCR;
    //Para generar la sancion
    sancion.MONT = 0;
    sancion.MOTI = parametroTarjeta.DSCR;
    //Numero de partidos suspendidos
    sancion.NUMPAR = parametroTarjeta.PARSUS;
    //Ver si se acumulan, borran, o no suman las tarjetas
    borrarAmarillas = parametroTarjeta.BORAMA;
    acumularAmarillas = parametroTarjeta.ACUAMA;
    acumularRojas = parametroTarjeta.ACUROJ;
}
//Asignar valores a la sancion
sancion.FECH = Convert.ToDateTime(entity.DetalleCalendario.FECHOR).Date;
sancion.EquipoCampeonato = entity.JugadorEquipoCampeonato.EquipoCampeonato;
sancion.JugadorEquipoCampeonato = entity.JugadorEquipoCampeonato;
//Multa
multa.FECH = Convert.ToDateTime(entity.DetalleCalendario.FECHOR).Date;
multa.EquipoCampeonato = entity.JugadorEquipoCampeonato.EquipoCampeonato;
multa.JugadorEquipoCampeonato = entity.JugadorEquipoCampeonato;
//Buscar el estado para la sancion
var estadoMulta = from estado in GEN_ESTADOS.Where(x => x.NOMB == "Vigente" &&
x.PROC == "Multa")
select estado;
//Recorrer todos los registros
foreach (var x in estadoMulta)
{
    multa.Estado = (Estado)x;
}
//Buscar el estado para la sancion
var estadoSancion = from estado in GEN_ESTADOS.Where(x => x.NOMB == "Vigente" &&
x.PROC == "Sancion")
select estado;
//Recorrer todos los registros
foreach (var x in estadoSancion)
{
    sancion.Estado = (Estado)x;
}
//Para obtener el tipo de sancion (MULTA)
var tipoMulta = from tipo in LIG_TIPOSANC.Where(x => x.NOMB == "Multa")
select tipo;

```

```

//Asignar el tipo de la sancion
foreach (var x in tipoMultas)
{
    //Asignar el tipo de sancion
    multa.TipoSancion = (TipoSancion)x;
}
//Para obtener el tipo de sancion (SANCION)
var tipoSancion = from tipo in LIG_TIPOSANC.Where(x => x.NOMB == "Sancion")
                  select tipo;
//Asignar el tipo de la sancion
foreach (var x in tipoSancion)
{
    //Asignar el tipo de sancion
    sancion.TipoSancion = (TipoSancion)x;
}
//Buscar el partido en el fue sancionado
if (entity.DetalleCalendario.EquipoCampeonatoLocal.Id ==
entity.JugadorEquipoCampeonato.EquipoCampeonato.Id)
{
    sancion.PARSUS = entity.DetalleCalendario.EquipoCampeonatoLocal.PartidosJugados;
}
else
{
    if (entity.DetalleCalendario.EquipoCampeonatoVisitante.Id ==
entity.JugadorEquipoCampeonato.EquipoCampeonato.Id)
    {
        sancion.PARSUS = entity.DetalleCalendario.EquipoCampeonatoVisitante.PartidosJugados;
    }
}
//Buscar el estado de la sancion de jugador
var estadoSancionJugador = from estado in GEN_ESTADOS.Where(x => x.NOMB ==
"Suspendido" && x.PROC == "Jugador")
                           select estado;
//Recorrer todos los registros
foreach (var x in estadoSancionJugador)
{
    entity.JugadorEquipoCampeonato.Estado = (Estado)x;
    entity.JugadorEquipoCampeonato.Jugador.Estado = (Estado)x;
}
if (borrarAmarillas)
{
    entity.JugadorEquipoCampeonato.Jugador.TARAMA = 0;
}
if (acumularAmarillas)
{
    //Sumar las tarjetas amarillas en la entidad jugadores
    entity.JugadorEquipoCampeonato.Jugador.TARAMA += entity.TARAMA;
}
if (acumularRojas)
{
    entity.JugadorEquipoCampeonato.Jugador.TAROJA += entity.TAROJA;
}
multa.DetallePartido = entity;
sancion.DetallePartido = entity;

```

```

//Variable de control
swSancion = false;
}
#endregion
#region Multas y Sanciones por una tarjeta amarilla y una roja
if (entity.TARAMA == 1 && entity.TAROJA == 1)
{
    //Instanciar un nuevo objeto para generar la multa
    Sancion multa = new Sancion();
    Sancion sancion = new Sancion();
    //Buscar los parametros cuando son una tarjeta amarilla y una roja
    var parametroUnaAmarillayRoja = from parametros in LIG_TARJETAS.Where(x => x.NOMB ==
    "TAMAYROJA" && x.Oficina.Id == entity.DetalleCalendario.Calendario.Campeonato.Oficina.Id)
    select parametros;
    //Buscar la sancion a ser aplicada segun el parametro
    foreach (var x in parametroUnaAmarillayRoja)
    {
        //Datos de tipo Tarjeta
        Tarjeta parametroTarjeta = (Tarjeta)x;
        //Mirar si se acumulan las tarjetas rojas y generar la sancion
        if (parametroTarjeta.ACUROJ)
        {
            //Buscar que sancion aplicar
            if (parametroTarjeta.APLI)
            {
                //Si aplica la sancion de una amarilla y una roja directamente
                sancion.MONT = 0;
                sancion.MOTI = parametroTarjeta.DSCR;
                //Numero de partidos que fue sancionado
                sancion.NUMPAR = Convert.ToInt32(parametroTarjeta.PARSUS);
            }
            else
            {
                bool sw = true;
                ///Seguir buscando que sancion aplicar, para ello buscar el numero de tarjetas rojas
                acumuladas que tiene este jugador,
                ///y sumarle una es decir si tiene 1+1=2 para buscar un parametro en las tarjetas que indique
                que sancion aplicar
                var parametroUnaAmarillayRojaAplicar = from parametros in LIG_TARJETAS.Where(y
                => y.NOMB == "TROJAS" + (entity.JugadorEquipoCampeonato.Jugador.TAROJA +
                1).ToString() && y.Oficina.Id ==
                entity.DetalleCalendario.Calendario.Campeonato.Oficina.Id)
                select parametros;
                //Buscar la sancion a aplicar
                foreach (var y in parametroUnaAmarillayRojaAplicar)
                {
                    //Datos de tipo Tarjeta
                    Tarjeta parametroTarjetaRojaX = (Tarjeta)x;
                    sancion.MONT = 0;
                    sancion.MOTI = parametroTarjetaRojaX.DSCR;
                    //Numero de partidos que fue suspendido
                    sancion.NUMPAR = Convert.ToInt32(parametroTarjetaRojaX.PARSUS);
                    //Encontro un parametro para aplicar sancion
                    sw = false;
                }
            }
        }
    }
}

```

```

}
//Si no encuentro una sancion que aplicar aplique la sancion por defecto asi no este marcada
que aplique esa sancion
if (sw)
{
    sancion.MONT = 0;
    sancion.MOTI = parametroTarjeta.DSCR;
    //Numero de partidos que fue suspendido
    sancion.NUMPAR = Convert.ToInt32(parametroTarjeta.PARSUS);
}
}
}
else
{
    //Si no se acumulan las tarjetas rojas, aplicar la sancion establecida para una amarilla y una roja
    //Si aplica la sancion de una amarilla y una roja directamente
    sancion.MONT = 0;
    sancion.MOTI = parametroTarjeta.DSCR;
    //Numero de partidos que fue suspendido
    sancion.NUMPAR = Convert.ToInt32(parametroTarjeta.PARSUS);
}
//Generar la multa
multa.MONT = parametroTarjeta.COST;
multa.MOTI = parametroTarjeta.DSCR;
//Ver si se acumulan, borran, o no suman las tarjetas
borrarAmarillas = parametroTarjeta.BORAMA;
acumularAmarillas = parametroTarjeta.ACUAMA;
acumularRojas = parametroTarjeta.ACUROJ;
}
//Asignar valores para generar la sancion
sancion.FECH = Convert.ToDateTime(entity.DetalleCalendario.FECHOR).Date;
sancion.JugadorEquipoCampeonato = entity.JugadorEquipoCampeonato;
sancion.EquipoCampeonato = entity.JugadorEquipoCampeonato.EquipoCampeonato;
//Asignar valores para generar la multa
multa.FECH = Convert.ToDateTime(entity.DetalleCalendario.FECHOR).Date;
multa.EquipoCampeonato = entity.JugadorEquipoCampeonato.EquipoCampeonato;
multa.JugadorEquipoCampeonato = entity.JugadorEquipoCampeonato;
//Buscar el estado para la sancion
var estadoMulta = from estado in GEN_ESTADOS.Where(x => x.NOMB == "Vigente" &&
x.PROC == "Multa")
select estado;
//Recorrer todos los registros
foreach (var x in estadoMulta)
{
    multa.Estado = (Estado)x;
}
//Buscar el estado para la sancion
var estadoSancion = from estado in GEN_ESTADOS.Where(x => x.NOMB == "Vigente" &&
x.PROC == "Sancion")
select estado;
//Recorrer todos los registros
foreach (var x in estadoSancion)
{
    sancion.Estado = (Estado)x;
}

```

```

}
//Para obtener el tipo de sancion (MULTA)
var tipoMulta = from tipo in LIG_TIPOSANC.Where(x => x.NOMB == "Multa")
                select tipo;
//Asignar el tipo de la sancion
foreach (var x in tipoMulta)
{
    //Asignar el tipo de sancion
    multa.TipoSancion = (TipoSancion)x;
}
//Para obtener el tipo de sancion (SANCION)
var tipoSancion = from tipo in LIG_TIPOSANC.Where(x => x.NOMB == "Sancion")
                  select tipo;
//Asignar el tipo de la sancion
foreach (var x in tipoSancion)
{
    //Asignar el tipo de sancion
    sancion.TipoSancion = (TipoSancion)x;
}
//Buscar el partido en el fue sancionado
if (entity.DetalleCalendario.EquipoCampeonatoLocal.Id ==
entity.JugadorEquipoCampeonato.EquipoCampeonato.Id)
{
    sancion.PARSUS = entity.DetalleCalendario.EquipoCampeonatoLocal.PartidosJugados;
}
else
{
    if (entity.DetalleCalendario.EquipoCampeonatoVisitante.Id ==
entity.JugadorEquipoCampeonato.EquipoCampeonato.Id)
    {
        sancion.PARSUS = entity.DetalleCalendario.EquipoCampeonatoVisitante.PartidosJugados;
    }
}

//Buscar el estado de la sancion de jugador
var estadoSancionJugador = from estado in GEN_ESTADOS.Where(x => x.NOMB ==
"Suspendido" && x.PROC == "Jugador")
                            select estado;
//Recorrer todos los registros
foreach (var x in estadoSancionJugador)
{
    entity.JugadorEquipoCampeonato.Estado = (Estado)x;
    entity.JugadorEquipoCampeonato.Jugador.Estado = (Estado)x;
}
//Encerar las tarjetas amarillas y rojas
if (borrarAmarillas)
{
    entity.JugadorEquipoCampeonato.Jugador.TARAMA = 0;
}
if (acumularAmarillas)
{
    //Sumar las tarjetas amarillas en la entidad jugadores
    entity.JugadorEquipoCampeonato.Jugador.TARAMA += entity.TARAMA;
}

```

```

if (acumularRojas)
{
    entity.JugadorEquipoCampeonato.Jugador.TAROJA += entity.TAROJA;
}
multa.DetallePartido = entity;
sancion.DetallePartido = entity;
//Variable de control
swSancion = false;
}
#endregion
#region Multas y Sanciones por una tarjeta roja
if (entity.TARAMA == 0 && entity.TAROJA == 1)
{
    //Instanciar un nuevo objeto para generar la multa
    Sancion multa = new Sancion();
    Sancion sancion = new Sancion();

    //Buscar los parametros cuando son una tarjeta amarilla y una roja
    var parametroUnaAmarillayRoja = from parametros in LIG_TARJETAS.Where(x => x.NOMB ==
    "TROJAS1" && x.Oficina.Id == entity.DetalleCalendario.Calendario.Campeonato.Oficina.Id)
        select parametros;
    //Buscar la sancion a ser aplicada segun el parametro
    foreach (var x in parametroUnaAmarillayRoja)
    {
        //Datos de tipo Tarjeta
        Tarjeta parametroTarjeta = (Tarjeta)x;
        //Mirar si se acumulan las tarjetas rojas y generar la sancion
        if (parametroTarjeta.ACUIROJ)
        {
            //Buscar que sancion aplicar
            if (parametroTarjeta.APLI)
            {
                //Si aplica la sancion de una amarilla y una roja directamente
                sancion.MONT = 0;
                sancion.MOTI = parametroTarjeta.DSCR;
                //Numero de partidos que fue sancionado
                sancion.NUMPAR = Convert.ToInt32(parametroTarjeta.PARSUS);
            }
        }
        else
        {
            bool sw = true;
            ///Seguir buscando que sancion aplicar, para ello buscar el numero de tarjetas rojas
            acumuladas que tiene este jugador,
            ///y sumarle una es decir si tiene 1+1=2 para buscar un parametro en las tarjetas que indique
            que sancion aplicar
            var parametroUnaAmarillayRojaAplicar = from parametros in LIG_TARJETAS.Where(y
            => y.NOMB == "TROJAS" + (entity.JugadorEquipoCampeonato.Jugador.TAROJA +
            1).ToString() && y.Oficina.Id ==
            entity.DetalleCalendario.Calendario.Campeonato.Oficina.Id)
                select parametros;
            //Buscar la sancion a aplicar
            foreach (var y in parametroUnaAmarillayRojaAplicar)
            {
                //Datos de tipo Tarjeta

```

```

Tarjeta parametroTarjetaRojaX = (Tarjeta)x;
sancion.MONT = 0;
sancion.MOTI = parametroTarjetaRojaX.DSCR;
//Numero de partidos que fue suspendido
sancion.NUMPAR = Convert.ToInt32(parametroTarjetaRojaX.PARSUS);
//Encontro un parametro para aplicar sancion
sw = false;
}
//Si no encontro una sancion que aplicar aplique la sancion por defecto asi no este marcada
que aplique esa sancion
if (sw)
{
sancion.MONT = 0;
sancion.MOTI = parametroTarjeta.DSCR;
//Numero de partidos que fue suspendido
sancion.NUMPAR = Convert.ToInt32(parametroTarjeta.PARSUS);
}
}
else
{
//Si no se acumulan las tarjetas rojas, aplicar la sancion establecida para una amarilla y una roja
//Si aplica la sancion de una amarilla y una roja directamente
sancion.MONT = 0;
sancion.MOTI = parametroTarjeta.DSCR;
//Numero de partidos que fue suspendido
sancion.NUMPAR = Convert.ToInt32(parametroTarjeta.PARSUS);
}
//Generar la multa
multa.MONT = parametroTarjeta.COST;
multa.MOTI = parametroTarjeta.DSCR;
//Ver si se acumulan, borran, o no suman las tarjetas
borrarAmarillas = parametroTarjeta.BORAMA;
acumularAmarillas = parametroTarjeta.ACUAMA;
acumularRojas = parametroTarjeta.ACUROJ;
}

//Asignar valores a la sancion
sancion.FECH = Convert.ToDateTime(entity.DetalleCalendario.FECHOR).Date;
sancion.JugadorEquipoCampeonato = entity.JugadorEquipoCampeonato;
sancion.EquipoCampeonato = entity.JugadorEquipoCampeonato.EquipoCampeonato;
//Multa
multa.FECH = Convert.ToDateTime(entity.DetalleCalendario.FECHOR).Date;
multa.EquipoCampeonato = entity.JugadorEquipoCampeonato.EquipoCampeonato;
multa.JugadorEquipoCampeonato = entity.JugadorEquipoCampeonato;

//Buscar el estado para la sancion
var estadoMulta = from estado in GEN_ESTADOS.Where(x => x.NOMB == "Vigente" &&
x.PROC == "Multa")
select estado;
//Recorrer todos los registros
foreach (var x in estadoMulta)
{
multa.Estado = (Estado)x;
}

```

```

}
//Buscar el estado para la sancion
var estadoSancion = from estado in GEN_ESTADOS.Where(x => x.NOMB == "Vigente" &&
x.PROC == "Sancion")
    select estado;
//Recorrer todos los registros
foreach (var x in estadoSancion)
{
    sancion.Estado = (Estado)x;
}
//Para obtener el tipo de sancion (MULTA)
var tipoMulta = from tipo in LIG_TIPOSANC.Where(x => x.NOMB == "Multa")
    select tipo;
//Asignar el tipo de la sancion
foreach (var x in tipoMulta)
{
    //Asignar el tipo de sancion
    multa.TipoSancion = (TipoSancion)x;
}
//Para obtener el tipo de sancion (SANCION)
var tipoSancion = from tipo in LIG_TIPOSANC.Where(x => x.NOMB == "Sancion")
    select tipo;
//Asignar el tipo de la sancion
foreach (var x in tipoSancion)
{
    //Asignar el tipo de sancion
    sancion.TipoSancion = (TipoSancion)x;
}
//Buscar el partido en el fue sancionado
if (entity.DetalleCalendario.EquipoCampeonatoLocal.Id ==
entity.JugadorEquipoCampeonato.EquipoCampeonato.Id)
{
    sancion.PARSUS = entity.DetalleCalendario.EquipoCampeonatoLocal.PartidosJugados;
}
else
{
    if (entity.DetalleCalendario.EquipoCampeonatoVisitante.Id ==
entity.JugadorEquipoCampeonato.EquipoCampeonato.Id)
    {
        sancion.PARSUS = entity.DetalleCalendario.EquipoCampeonatoVisitante.PartidosJugados;
    }
}

//Buscar el estado de la sancion de jugador
var estadoSancionJugador = from estado in GEN_ESTADOS.Where(x => x.NOMB ==
"Suspendido" && x.PROC == "Jugador")
    select estado;
//Recorrer todos los registros
foreach (var x in estadoSancionJugador)
{
    entity.JugadorEquipoCampeonato.Estado = (Estado)x;
    entity.JugadorEquipoCampeonato.Jugador.Estado = (Estado)x;
}
//Encerar las tarjetas amarillas y rojas

```

```

if (borrarAmarillas)
{
    entity.JugadorEquipoCampeonato.Jugador.TARAMA = 0;
}
if (acumularAmarillas)
{
    //Sumar las tarjetas amarillas en la entidad jugadores
    entity.JugadorEquipoCampeonato.Jugador.TARAMA += entity.TARAMA;
}
if (acumularRojas)
{
    entity.JugadorEquipoCampeonato.Jugador.TAROJA += entity.TAROJA;
}
multa.DetallePartido = entity;
sancion.DetallePartido = entity;

//Variable de control
swSancion = false;
}
#endregion
#region Verificar si existen tarjetas amarillas acumuladas

//Buscar en los parametros de las tarjetas cuantas tarjetas amarillas puedo acumular
var numeroAmarillas = from parametroAcumulacionTarjetasAmarillas in
LIG_TARJETAS.Where(x => x.NOMB == "TARAMACUM" && x.VIGE == true && x.Oficina.Id
== entity.DetalleCalendario.Calendario.Campeonato.Oficina.Id)
select parametroAcumulacionTarjetasAmarillas;
//Recorrer todos registros
foreach (var x in numeroAmarillas)
{
    Tarjeta parametroTarjetas = (Tarjeta)x;
    numeroTarjetasAmarillas = Convert.ToInt32(parametroTarjetas.VALO);
    numeroPartidosSuspendidos = (int)parametroTarjetas.PARSUS;
    motivoSuspension = Convert.ToString(parametroTarjetas.DSCR);
}
//Verificar el numero de tarjetas amarillas para sancionar al jugador por acumulacion de tarjetas
amarillas
if (entity.JugadorEquipoCampeonato.Jugador.TARAMA == numeroTarjetasAmarillas)
{
    if (swSancion)
    {
        //Instanciar un nuevo objeto para generar la sancion por la acumulacion de tarjetas amarillas
        Sancion sancion = new Sancion();
        //Asignar valores a la sancion
        sancion.FECH = Convert.ToDateTime(entity.DetalleCalendario.FECHOR).Date;
        sancion.JugadorEquipoCampeonato = entity.JugadorEquipoCampeonato;
        sancion.EquipoCampeonato = entity.JugadorEquipoCampeonato.EquipoCampeonato;
        //Numero de partidos suspendidos
        sancion.NUMPAR = numeroPartidosSuspendidos;
        //Motivo de la sancion
        sancion.MOTI = motivoSuspension;
        sancion.MONT = 0;
        //Buscar el estado para la sancion
    }
}

```

```

var estadoSancion = from estado in GEN_ESTADOS.Where(x => x.NOMB == "Vigente" &&
x.PROC == "Sancion")
    select estado;
//Recorrer todos los registros
foreach (var x in estadoSancion)
{
    sancion.Estado = (Estado)x;
}
//Buscar el tipo de la sancion
var tipoSancion = from tipos in LIG_TIPOSANC.Where(x => x.NOMB == "Sancion")
    select tipos;
//Recorrer todos los registros
foreach (var x in tipoSancion)
{
    sancion.TipoSancion = (TipoSancion)x;
}
//Buscar el partido en el fue sancionado
if (entity.DetalleCalendario.EquipoCampeonatoLocal.Id ==
entity.JugadorEquipoCampeonato.EquipoCampeonato.Id)
{
    sancion.PARSUS = entity.DetalleCalendario.EquipoCampeonatoLocal.PartidosJugados;
}
else
{
    if (entity.DetalleCalendario.EquipoCampeonatoVisitante.Id ==
entity.JugadorEquipoCampeonato.EquipoCampeonato.Id)
    {
        sancion.PARSUS = entity.DetalleCalendario.EquipoCampeonatoVisitante.PartidosJugados;
    }
}

//Buscar el estado de la sancion de jugador
var estadoSancionJugador = from estado in GEN_ESTADOS.Where(x => x.NOMB ==
"Suspendido" && x.PROC == "Jugador")
    select estado;
//Recorrer todos los registros
foreach (var x in estadoSancionJugador)
{
    entity.JugadorEquipoCampeonato.Estado = (Estado)x;
    entity.JugadorEquipoCampeonato.Jugador.Estado = (Estado)x;
}
//Borrar las tarjetas amarillas acumuladas
entity.JugadorEquipoCampeonato.Jugador.TARAMA = 0;
sancion.DetallePartido = entity;
}
else
{
    //Borrar las tarjetas amarillas acumuladas
    entity.JugadorEquipoCampeonato.Jugador.TARAMA = 0;
}
}
#endregion
//Variables para el control cuando se actualizan estos campos
//Asignar los goles en goles_antes para ser utilizado se actualizan

```

```

entity.GOLANT = entity.GOLE;
//Asignar las tarjetas amarillas anteriores
entity.TARAMANT = entity.TARAMA;
//Asignar las tarjetas rojas anteriores
entity.TAROJANT = entity.TAROJA;
}
#endregion

```

Como en la capa de negocios cada tabla es un objeto para hacer una actualización de los datos en una tabla no se necesita hacerlo por SQL.

4.3.10 Desarrollo del sistema (Pantallas, Formas y el Menú)

Una vez terminada la capa de negocios, procedemos con la creación de las pantallas, para los cual se establecen las siguientes reglas:

- Los nombres de las pantallas que en la herramienta de desarrollo es un objeto más, deberán tener los prefijos establecidos para cada módulo (GEN_, LIG_, RHH_, SEG_), seguido del nombre del objeto (Tabla) pluralizado sobre el cual se manipulan los datos o las propiedades, más la palabra **listado** si es una pantalla que permite la manipulación de los datos y **lista** si es una pantalla de consulta, la segunda parte que hace referencia al nombre del objeto pluralizado se aplicara para pantallas que permitan la manipulación de los datos de un solo objeto.
- Las pantallas que tengan pestañas o que permitan la manipulación de datos de varios objetos deberán tener en la parte que corresponda el nombre del objeto pluralizado un nombre descriptivo a la agrupación de objetos que hace referencia.
- Finalmente las pantallas que hacen referencia a los reportes o que generan reportes tendrán la siguiente nomenclatura para la formación de su nombre, el prefijo será (GER_, LIR_, RHR_, SER_) seguido de un nombre descriptivo del reporte al cual hace referencia.

Página de inicio de sesión o página de login del sistema.

Esta página proporciona información de algunas funcionalidades del sistema, las cuales están representadas con imágenes y botones. A su vez también permite iniciar sesión a los diferentes usuarios del sistema con sus respectivos roles.

Inicio de sesión en el sistema.



Fig. 39: Página de inicio de sesión o página de login del sistema
Elaborador por: El investigador.

a) Banner de Imágenes de las funcionalidades del sistema: Contiene cinco imágenes las cuales describen algunas funcionalidades del sistema, estas imágenes cambian automáticamente cada cierto tiempo.

b) Inicio de sesión: Aquí se encuentran controles necesarios que permiten la validación de los datos de los usuarios del sistema, y permiten el ingreso al mismo.

c) Información del desarrollador del sistema: Despliega información de quien desarrollo en sistema.

En el siguiente código HTML se pueden cambiar las imágenes del banner.

```
<div id="featured_content">
  <div class="featured_box" id="fc1">
</div>
  <div class="featured_box" id="fc2">
</div>
  <div class="featured_box" id="fc3">
</div>
  <div class="featured_box" id="fc4">
</div>
  <div class="featured_box" id="fc5">
</div>
</div>
```

Página principal del sistema.

Esta es la página principal del sistema, sobre ella se desplegará las pantallas que tiene el sistema, así como también el menú, botones de acciones e información del usuario que está conectado.

Página principal del sistema o página maestra.

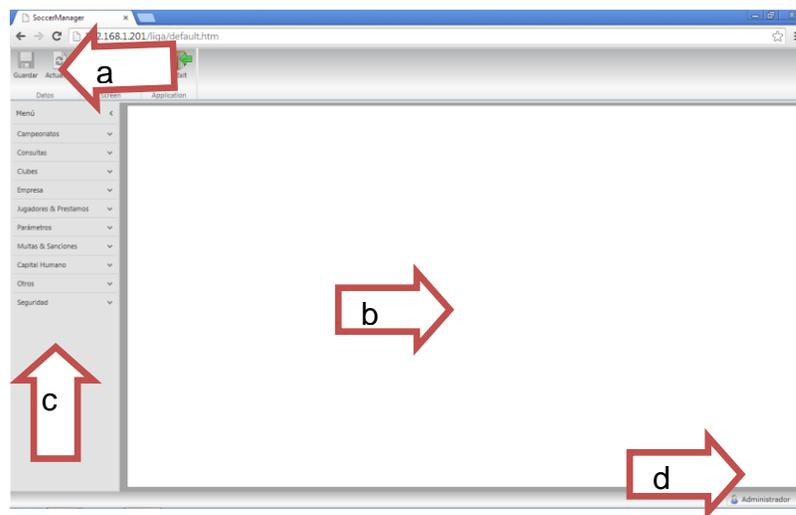


Fig. 40: Página principal del sistema o página maestra
Elaborador por: El investigador.

a) Botones de acción del sistema: En esta sección de la página estarán los botones de acción tanto de la aplicación como de las ventanas de las páginas que sobre ella se despliegan.

b) Área de trabajo: Aquí se desplegaran las pantallas del sistema, esta es el área de visualización y manipulación de datos.

c) Menú del sistema: Área donde se encuentra el menú agrupado por categorías y filtrado en dependencia de los permisos que tenga el rol que se le haya asignado al usuario. Estas son las categorías y las opciones del menú.

- Campeonatos
 - ✓ Categorías
 - ✓ Calendarios
 - ✓ Campeonato
 - ✓ Equipos – Campeonatos
 - ✓ Jugadores – Campeonatos
 - ✓ Partidos
 - ✓ Vocales Campeonatos
- Consultas
 - ✓ Equipo – Campeonato
 - ✓ Jugador – Campeonato
 - ✓ Tabla Goleadores
- Clubes
 - ✓ Club
 - ✓ Directivos
 - ✓ Directivos Club
 - ✓ Equipo Jugadores
- Empresa
 - ✓ **Bienvenida**
 - ✓ Cargos

- ✓ Comisiones
- ✓ Directivos Liga
- ✓ Empresa y Oficina
- ✓ Geografía
- ✓ Listado Reportes
- ✓ Vocales
- Jugadores & Prestamos
 - ✓ Jugadores Propios
 - ✓ Prestamos Jugadores
 - ✓ Regresar Prestamos
 - ✓ Jugadores con Equipo
- Parámetros
 - ✓ Específicos
 - ✓ Generales
- Multas & Sanciones
 - ✓ Cobrar Multas
 - ✓ Levantar Sanciones
 - ✓ Multas Generadas
 - ✓ Sanciones Generadas
- Capital Humano
 - ✓ Empleados
 - ✓ Cargos Empleados
- Otros
 - ✓ Árbitros
- Seguridad
 - ✓ Roles
 - ✓ Usuarios

Con el siguiente método se verifica si un determinado usuario tiene o no acceso a una pantalla. Todas las pantallas que necesiten este control tienen este método.

```

partial void Lig_JugadoresListado_CanRun(ref bool result)
{
    // Establece el resultado en el valor del campo deseado
    result = Application.Current.User.HasPermission(Permissions.LigJugadoresListado);
}
    
```

d) Información del usuario: En esta área se visualiza el nombre completo del usuario, cabe señalar que este no es el nombre de inicio de sesión del usuario.

Pantalla de inicio del sistema.

Contiene algunos accesos directos de los procesos que se utilizan con mayor frecuencia, algo similar a los iconos del escritorio de un sistema operativo. Esta es la primera pantalla que ve el usuario una vez que se autentifique en el sistema. A esta pantalla podrán tener accesos todos los usuarios que se autentifiquen en el sistema sin importar el rol que tengan, pero no podrá ejecutar los enlaces de los accesos directos si no tiene permisos de hacerlo.

Pantalla de accesos rápidos a procesos.

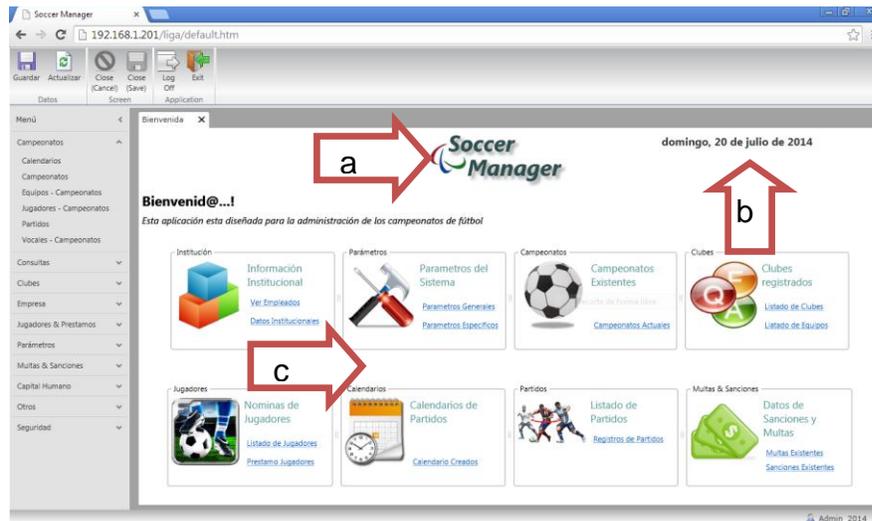


Fig. 41: Pantalla de inicio del sistema o accesos rápidos a procesos
Elaborador por: El investigador.

a) Nombre del sistema: Contiene el nombre de sistema conjuntamente con el logotipo del mismo.

b) **Fecha actual:** Despliega la fecha actual del sistema.

c) **Accesos directos:** Enlaces directos a los procesos que son utilizados con mayor frecuencia agrupados por categorías.

Pantalla de manipulación de datos simple.

Este tipo de pantallas visualizan y permiten la manipulación directa de los datos de una sola entidad o tabla.

Manipulación de datos en grillas.

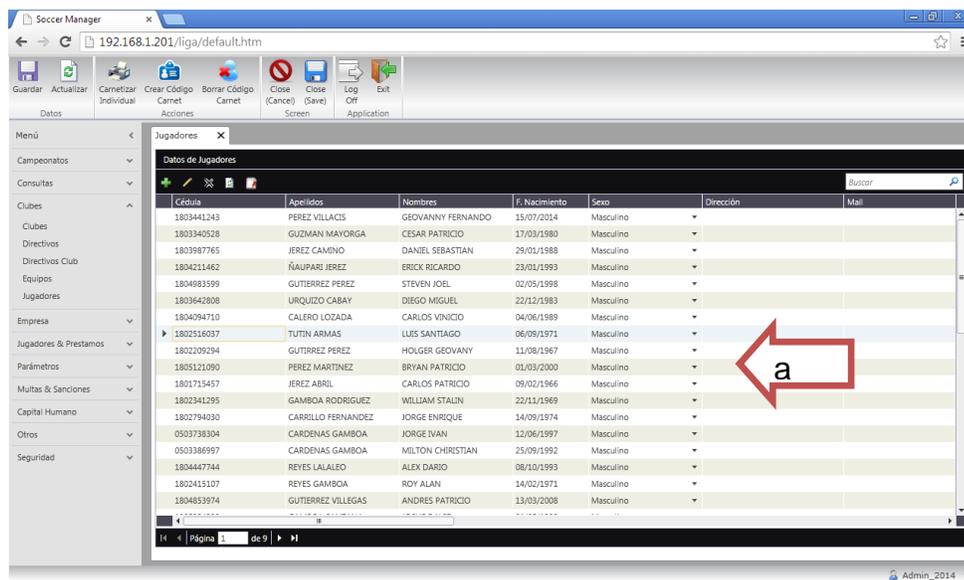


Fig. 42: Pantalla de manipulación de datos simple
Elaborador por: El investigador.

a) **Grilla de datos:** Contiene datos de una sola entidad los cuales pueden ser manipulados, es decir se puede buscar, agregar, modificar y eliminar datos.

La programación de los botones Carnetizar Individual, Crear Código Carnet y Borrar Código Carnet se describen a continuación.

```
public partial class Lig_JugadoresListado
{
```

```
//Llama a la pantalla de reportes para generar los carnets
partial void CarnetizarIndividual_Execute()
{
    try
    {
        DateTime fechaCarnetizar = DateTime.Now;
        // Obtener la fecha a partir de la cual se desea carnetizar de forma individual
        var parametroFechaCarnetizar = from parametro in Parametro.Where(x => x.NOMB ==
"CarnetIndividual" && x.VIGE == true)
select parametro;

        //Buscar la fecha
        foreach (var filaObtenida in parametroFechaCarnetizar)
        {
            fechaCarnetizar = Convert.ToDateTime(filaObtenida.VALO);
        }

        //Llamar al reporte para carnetizar por jugador
        this.Application.ShowLir_CarnetConParametros(0, 0, "Si", fechaCarnetizar, "Carnetizar
Individualmente");
    }
    catch (Exception exception)
    {
    }
}
/// <summary>
/// Crea los codigos de los carnets
/// </summary>
partial void CrearCodigoCarnet_Execute()
{
    // Write your code here.
    try
    {
        //Variable para la primera parte del codigo del carnet
        String codigoCarnet = "";
        // Obtener el parametro para formar el codigo del carnet
        var parametroFechaCarnetizar = from parametro in Parametro.Where(x => x.NOMB ==
"CodigoCarnet" && x.VIGE == true)
select parametro;

        //Buscar el codigo
        foreach (var filaObtenida in parametroFechaCarnetizar)
        {
            codigoCarnet = filaObtenida.VALO;
        }

        //Seleccionar los jugadores para actualizar el codigo del carnet
        var listadoJugadores = from jugadores in Jugador.Where(x => x.NUMCAR == "" ||
x.NUMCAR.Length <= 1 || x.NUMCAR == null)
select jugadores;

        //Recorer todos los jugadores
        foreach (var filaObtenida in listadoJugadores)
        {
            //Si el codigo del carnet es diferente de null

```

```

if (codigoCarnet != "")
{
    //si el codigo del carnet solo tiene una letra
    if (codigoCarnet.Length == 1)
    {
        //De 1 al 9 agregar 00
        if (filaObtenida.Id < 10)
        {
            filaObtenida.NUMCAR = codigoCarnet + "00" + filaObtenida.Id.ToString();
        }
        //Del 10 al 99 agregar 0
        if (filaObtenida.Id >= 10 && filaObtenida.Id < 100)
        {
            filaObtenida.NUMCAR = codigoCarnet + "0" + filaObtenida.Id.ToString();
        }
        if (filaObtenida.Id >= 100)
        {
            filaObtenida.NUMCAR = codigoCarnet + filaObtenida.Id.ToString();
        }
    }
}
//Si tiene mas de una letra
else
{
    filaObtenida.NUMCAR = codigoCarnet + filaObtenida.Id.ToString();
}
}
//Si el valor del paramatro para armar el codigo del carnet es nulo, formarlo solo con el ID
else
{
    //Si el ID esta del 1 al 9 poner 00
    if (filaObtenida.Id < 10)
    {
        filaObtenida.NUMCAR = "00" + filaObtenida.Id.ToString();
    }
    //Si el ID esta entre 10 y 99 poner 0
    if (filaObtenida.Id >= 10 && filaObtenida.Id < 100)
    {
        filaObtenida.NUMCAR = "0" + filaObtenida.Id.ToString();
    }
    if (filaObtenida.Id >= 100)
    {
        filaObtenida.NUMCAR = filaObtenida.Id.ToString();
    }
}
}
}
//Guardar los datos
Save();
}
catch (Exception exception)
{
}
}
}
/// <summary>
/// Borra los codigos de los carnets

```

```

/// </summary>
partial void BorrarCodigoCarnet_Execute()
{
    try
    {
        //Seleccionar los jugadores para borrar el codigo del carnet
        var listadoJugadores = from jugadores in Jugador.Where(x => x.NUMCAR != "" ||
x.NUMCAR.Length >= 1 || x.NUMCAR != null) select jugadores;
        //Recorer todos los jugadores
        foreach (var filaObtenida in listadoJugadores)
        {
            //Borrar el codigo de los carnets
            filaObtenida.NUMCAR = "";
        }
        //Guardar los datos
        Save();
    }
    catch (Exception exception)
    {
    }
}

```

Pantalla de manipulación de datos con filtros.

Son pantallas que al ser ejecutadas no muestran datos por defecto, sino que hay que seleccionar datos de las listas de valores para que se pueda visualizar los datos en la grilla.

Manipulación de datos con filtros.

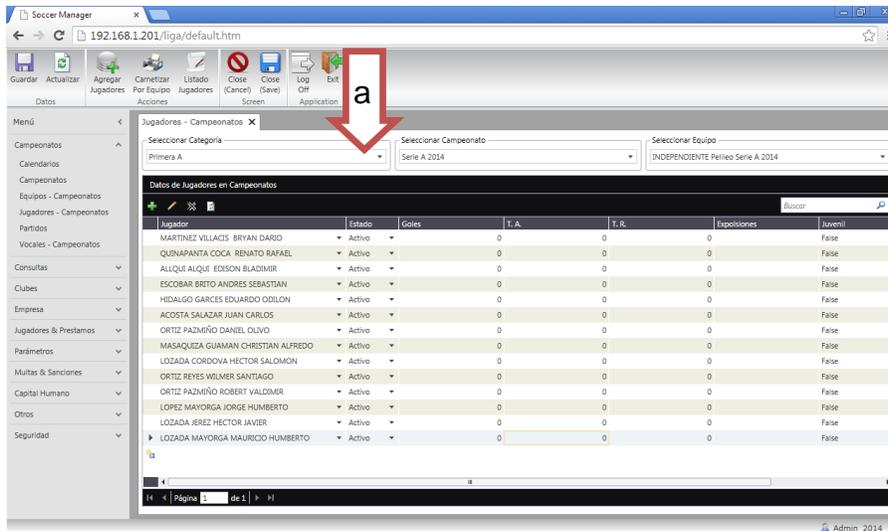


Fig. 43: Pantalla de manipulación de datos con filtros
Elaborador por: El investigador.

a) **Listas de valores:** Estas listas son en cascada, es decir que para que se carguen los datos de la siguiente lista hay que seleccionar un valor de la lista anterior. Al seleccionar valores de estas listas se cargan los datos de la grilla principal.

Los métodos que permiten Carnetizar por Equipo y el Listado de jugadores se describe a continuación.

```

partial void CarnetizarPorEquipo_Execute()
{
    this.Application.ShowLir_CarnetConParametros(PropertyEquipoCampeonato.Id, 0, " ",
    DateTime.Now, "Carnetizar Por Equipo");
}
partial void ListadoJugadores_Execute()
{
    this.Application.ShowLir_JugadoresPorEquipoCampeonato(PropertyEquipoCampeonato.Id, 0);
}

```

Pantallas con pestañas.

Las pantallas con pestañas permiten agrupar en una misma pantalla varios procesos similares. Estas pestañas pueden ser tanto en cascada como simple.

Pantallas con pestañas.

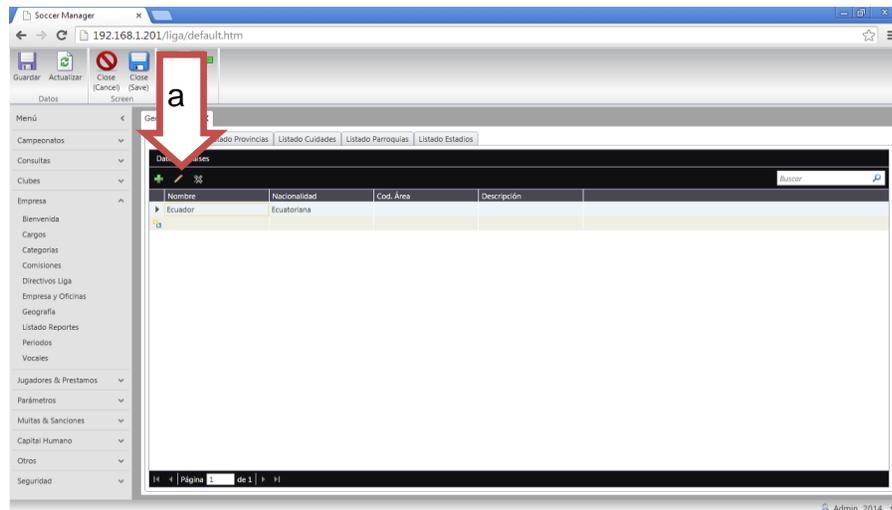


Fig. 44: Pantallas con pestañas
Elaborador por: El investigador.

a) **Pestañas:** Permiten agrupar varios procesos, si estas son en cascada al seleccionar un valor en la primera se cargaran los valores relacionados a esta en la siguiente.

Pantalla de Ingreso o Modificación de datos.

Son pantallas modales las cuales se despliegan sobre la grilla principal, estas permiten ingresar nuevos datos o modificar datos existentes.

Ingreso o Modificación de datos.

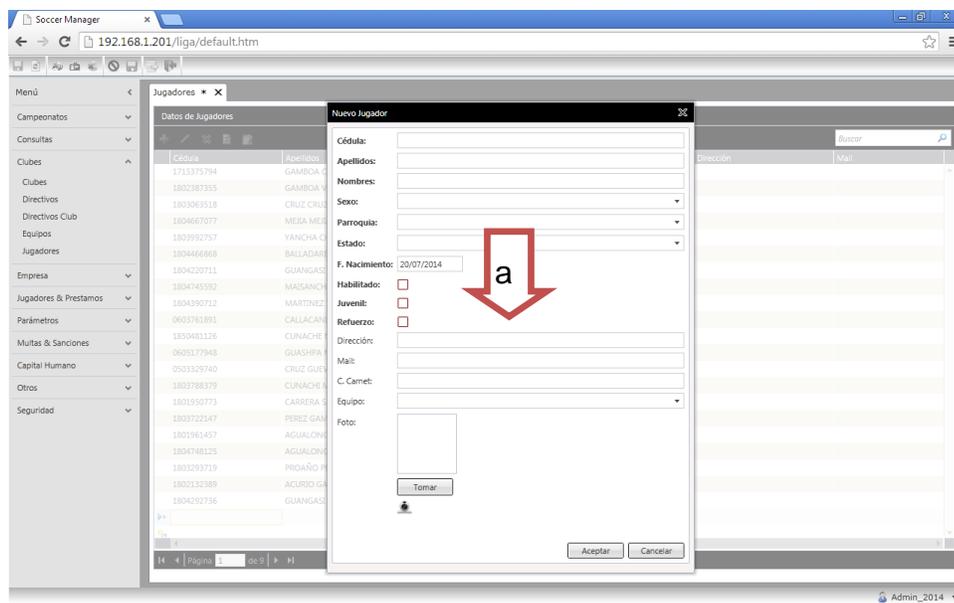


Fig. 45: Pantalla de Ingreso o Modificación de datos
Elaborador por: El investigador.

a) **Pantalla modal:** Casi todas las pantallas tienen implícitamente una ventana modal, la cual es utilizada por lo general para ingresar nuevos datos. Esta se despliega sobre la pantalla desde la cual se la invocó.

Los siguientes métodos permiten ingresar o modificar datos en una ventana modal. También se describe el método que permite utilizar la cámara web del computador.

```

//Variables
private WebcamControl WebCam;
//Nuevo
partial void JugadorAddAndEditNew_CanExecute(ref bool result)
{
    result = this.Jugador.CanAddNew;
    //Toma un valor de verdadero si es nuevo registro
    nuevoRegistro = true;
}
partial void JugadorAddAndEditNew_Execute()
{
    Jugador nuevoJugador = this.Jugador.AddNew();
    this.Jugador.SelectedItem = nuevoJugador;
    this.OpenModalWindow("JugadorEditDialog");
}
//Cambio
partial void JugadorEditSelected_CanExecute(ref bool result)
{
    result = this.Jugador.CanEdit && this.Jugador.SelectedItem != null;
    //jugadorSeleccionado = this.Jugador.SelectedItem;
    //Toma un valor de falso si solo se va a editar el registro
    nuevoRegistro = false;
}
partial void JugadorEditSelected_Execute()
{
    //Abrir pantalla modal
    this.OpenModalWindow("JugadorEditDialog");
}
/// <summary>
/// Cuando se crea la pantalla en memoria
/// </summary>
partial void Lig_JugadoresListado_Created()
{
    // Crear componente
    var Ctrl = this.FindControl("ScreenContent");
    Ctrl.ControlAvailable += new EventHandler<ControlAvailableEventArgs>(WebCamHandler);
}
public void WebCamHandler(object sender, ControlAvailableEventArgs e)
{
    this.WebCam = (WebcamControl)e.Control;
    this.WebCam.CaptureCompleted += new
        WebcamControl.CaptureCompletedEventHandler(WebCam_CaptureCompleted);
}
public void WebCam_CaptureCompleted()
{
    this.Jugador.SelectedItem.FOTO = this.WebCam.CapturedImage;
}
/// <summary>
/// Se dispara cuando el boton Cancelar es precionado
/// </summary>
partial void Cancelar_Execute()
{
    Jugador.SelectedItem.Details.DiscardChanges();
}

```

```
this.CloseModalWindow("JugadorEditDialog");
}
```

4.3.11 Creación de reportes e informes que tendrá el sistema

Los reportes con los que cuenta el sistema están diseñados para ser impresos en un formato A4, pero si el usuario desea puede cambiar el formato de impresión aunque el reporte no tendrá un diseño óptimo. Estos reportes también podrán ser exportados a archivos digitales en algunos formatos disponibles como pdf, html, etc.

Pantallas de visualización y generación de reportes

Estas pantallas permiten la visualización de los reportes que tiene el sistema, así como también contienen una barra de herramientas propia desde la cual se pueden realizar tareas como: imprimir, cambiar el formato del papel, exportar el documento, etc.

Visualización y generación de reportes.

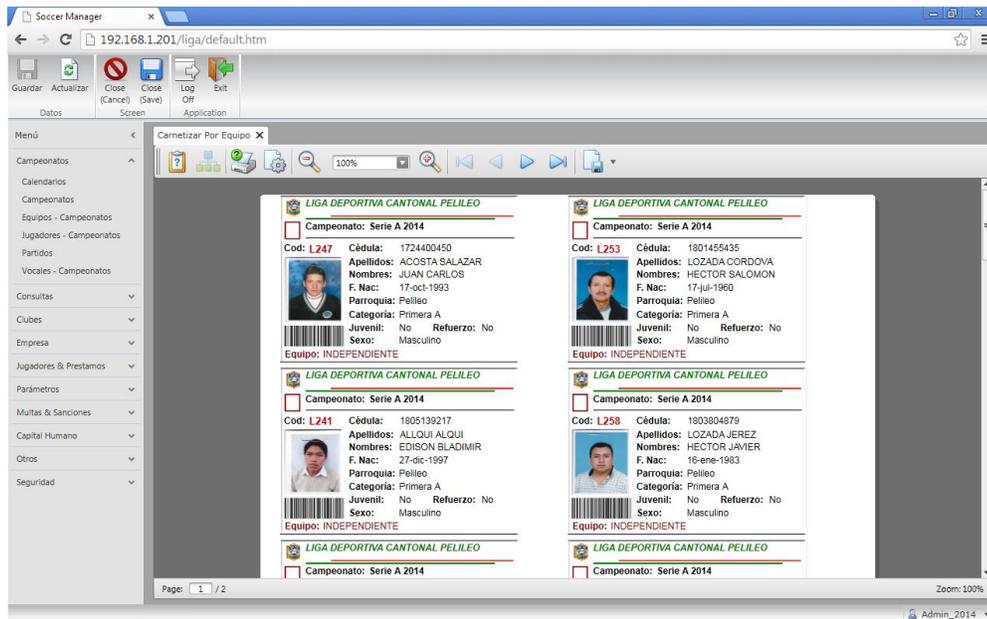


Fig. 46: Pantallas de visualización y generación de reportes
Elaborador por: El investigador.

Para generar un reporte con parámetros la pantalla necesita de los siguientes métodos.

```

public void CustomizeReportPreviewModel(DevExpress.Xpf.Printing.ReportPreviewModel model)
{
    try
    {
        //Pasar los paramtros
        model.Parameters["parametroIdEquipoCampeonato"].Value = parametroIdEquipoCampeonato;
        model.Parameters["parametroIdCampeonato"].Value = parametroIdCampeonato;
        model.Parameters["parametroCarnetizado"].Value = parametroCarnetizado;
        model.Parameters["parametroFechaCarnetiza"].Value = parametroFechaCarnetiza;
    }
    catch (Exception exception)
    {
    }
}
partial void Lir_CarnetConParametros_Activado()
{
    try
    {
        //Variables locales
        string nombreReporte = "";
        // Assign the name of the report, which you want to preview in this screen.
        this.DisplayName = TituloPantalla;
        //Buscar el nombre de reporte
        var reporte = from reportes in this.DataWorkspace.ApplicationData.GEN_REPORTES.Where(x =>
x.PROC.ToUpper() == "Carnet".ToUpper())
                    select reportes;
        //Recorer los datos obtenidos
        foreach (var x in reporte)
        {
            Reporte repo = (Reporte)x;
            nombreReporte = repo.NOMB;
        }
        //this.ReportTypeName = "CarnetGeneral_LDCP";
        //this.ReportTypeName = "CarnetGeneral_LDCF";
        this.ReportTypeName = nombreReporte;
    }
    catch (Exception exception)
    {
    }
}

```

4.3.12 Ingreso y documentación de los parámetros del sistema

El sistema cuenta con algunos parámetros, los cuales permiten que algunos procesos internos puedan ejecutarse de forma automática.

El siguiente cuadro muestra los parámetros que contiene la tabla GEN_PARAMETRO, estos parámetros son utilizados en procesos internos.

Listado de parámetros del sistema.

| Nombre(Nombre del parámetro) | Costo(en \$) | Valor (en texto) | V. Fecha (Valor en fecha) | Tipo Parámetro | Descripción | Oficina | Módulo | Categoría(Puede o no tener) | Vg. |
|------------------------------|--------------|---------------------|---------------------------|----------------|---|---------|--------|-----------------------------|-----|
| Juvenil | \$ 0,00 | 0 | 01/01/1997 | Fecha | Determinar el juvenil de la serie A | MAT-001 | Liga | Serie A | Si |
| CarnetIndividual | \$ 0,00 | 02/08/2014 18:40:09 | | Fecha | Para carnetizar con una fecha de inicio | MAT-001 | Liga | | Si |
| ListadoCarnets | \$ 0,00 | si | | Texto | Para carnetizar por listas | MAT-001 | Liga | | Si |
| CodigoCarnet | \$ 0,00 | L | | Texto | Valor inicial del código del carnet | MAT-001 | Liga | | Si |
| Juvenil | \$ 0,00 | 0 | 01/01/1996 | Fecha | Determinar el juvenil de la serie B | MAT-001 | Liga | Serie B | Si |

Tabla 33: Parámetros del sistema
Elaborador por: El investigador.

Estos parámetros son para aplicar multas y sanciones a los jugadores por conceptos de tarjetas amarillas y rojas. La entidad que almacena esta información es LIG_TARJETAS.

Parámetros para tarjetas amarillas y rojas.

| Nombre | Costo | Valor | Partidos de Suspensión | Descripción | Borrar T. A. | Acumular T. A. | Acumular T. R. | Aplica | Vg. | Oficina |
|------------|---------|-------|------------------------|---------------------------------------|--------------|----------------|----------------|--------|-----|---------|
| TAMARILLA | \$ 0,50 | 0 | 0 | Por una tarjeta amarilla | No | No | No | No | Si | MAT-001 |
| T2AMAYROJA | \$ 2,00 | 0 | 1 | Por dos tarjetas amarillas y una roja | No | Si | No | No | Si | MAT-001 |
| TAMAYROJA | \$ 1,50 | 0 | 1 | Por una tarjeta amarilla y una roja | No | Si | Si | Si | Si | MAT-001 |
| TROJASI | \$ 1,00 | 0 | 2 | Por una tarjeta roja directa | No | No | Si | Si | Si | MAT-001 |
| TARAMACUM | \$ 0,00 | 5 | 1 | Por acumulación de tarjetas amarillas | Si | No | No | No | Si | MAT-001 |
| TAROJACUM2 | \$ 0,00 | 2 | 4 | Por acumulación de dos rojas | No | No | Si | No | Si | MAT-001 |
| TAROJACUM3 | \$ 0,00 | 3 | 6 | Por acumulación de tres rojas | No | No | Si | No | Si | MAT-001 |

Tabla 34: Parámetros para tarjetas
Elaborador por: El investigador.

En el sistema estos parámetros y datos se encuentran en el menú parámetros, parámetros específicos en las pestañas. Parámetros y Parámetros de tarjetas. Cabe mencionar también que los nombres de estos parámetros deber estar como se muestran en las tablas. No se pueden agregar más parámetros de los existentes.

Los tipos de sanción también son tipos de parámetros que permiten conjuntamente con los datos de la tabla anterior generar sanción o multas según sea el caso. Estos datos se almacenan en la tabla `LIG_TIPOSANC`.

Tipos de sanciones.

| Nombre | Descripción | Vg. |
|----------------|--------------------------------------|-----|
| Multa | Cuando se genera una multa económica | Si |
| Sanción | Cuando se genera un sanción | Si |

Tabla 35: Tipos de sanción
Elaborador por: El investigador

Los elementos que tiene una entidad pueden pasar por varios estados a lo largo de su vida en el sistema. La siguiente tabla muestra los estados que estos pueden tener y los procesos con los cuales se relacionan. Esta información se encuentra en la tabla `GEN_ESTADOS`.

Estados de los procesos.

| Nombre | Proceso | Descripción | Vg. |
|-------------------|-------------------|---|-----|
| Activo | General | Estado que aparece en todos los procesos | Si |
| Inactivo | Jugador | Estado solo para el Jugadores | Si |
| Postergado | Calendario | Estado solo para el Calendarios | Si |
| Suspendido | DetalleCalendario | Estado solo para el Detalles de los Calendarios | Si |
| Sancionado | Equipo | Estado solo para el Equipos | Si |
| Jugado | DetalleCalendario | Estado solo para el Detalles de los Calendarios | Si |
| Vigente | Multa | Estado solo para las Multas | Si |
| Vigente | Sancion | Estado solo para las Sanciones | Si |

| | | | |
|-------------------|---------|--|----|
| Suspendido | Jugador | Estado solo para el Jugadores | Si |
| Cancelado | Multa | Estado solo para las Multas | Si |
| Cumplida | Sancion | Estado solo para las Sanciones | Si |
| Cerrado | General | Estado que aparece en todos los procesos | Si |

Tabla 36: Listado de los estados con sus respectivos procesos
Elaborador por: El investigador.

Los nombres de los estados pueden variar así como también pueden ser creados nuevos estado, solo en la columna **Proceso** no se puede incrementar nuevos proceso, es decir se pueden crear nuevos estados pero solo para los procesos existentes. En el sistema estos parámetros y datos se encuentran en el menú parámetros, parámetros generales en las pestañas. Tipo de Sanción y Estados.

4.4 Estabilización

La solución que hemos diseñado pasa a un entorno real de explotación en la cual se restringe el número de usuarios así como también las condiciones de tal forma que se pueda llevar a cabo un control efectivo.

4.4.1 Configuración del servidor web IIS en un ámbito de pruebas

Todas la versiones de Windows sean estas para Desktop o Servidor incluyen un servidor web llamado Internet Information Service conocido por sus siglas como IIS. Para instalar IIS sigue los siguientes pasos.

- Ir al Panel de control y abrir la opción "Programas y características".
- En el panel del lado izquierdo escoger "Activar o desactivar las características de Windows".

- En la lista de funciones de Windows, despliega el árbol y marca todas las casillas correspondiente a "Internet Information Services" y haz clic en Aceptar.

Instalación de Internet Information Server.



Fig. 47: Instalación de Internet Information Services
Elaborador por: El investigador.

Para Administrar y configurar el servidor ISS, escribe en el cuadro de Inicio **inetmgr** y presiona la tecla enter. También lo podemos hacer por el panel de control -> Herramientas administrativas, ahí nos aparece Administrador de Internet Information Services (IIS).

Panel de Administración de Internet Information Server.

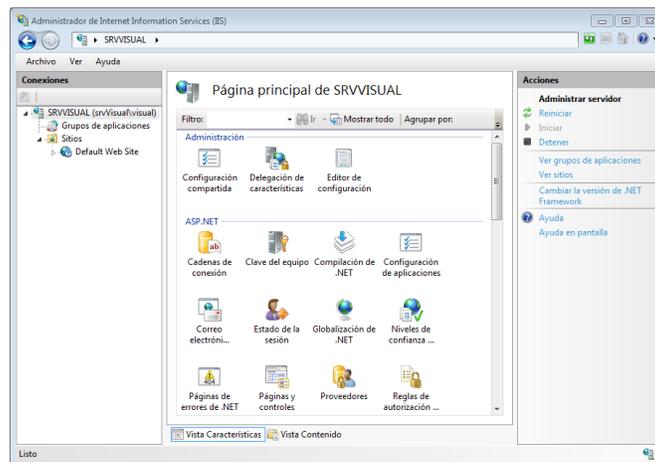


Fig. 48: Pantalla de Administración de Internet Information Service
Elaborador por: El investigador.

Una vez instalado es necesario registrar el Framework 4.0, para ello es necesario ejecutar el símbolo del sistema como administrador mediante línea de comandos. Para esto deberíamos estar ubicados en la siguiente dirección.

C:\Windows\Microsoft.NET\Framework\v4.0.30319

Una vez ubicados en este directorio escribir el comando **aspnet_regiis.exe -i** y presionar la tecla enter.

C:\Windows\Microsoft.NET\Framework\v4.0.30319>aspnet_regiis.exe -i

Esto permite registrar el Framework 4.0 para que trabaje por defecto con el IIS. Y podamos ejecutar nuestra aplicación.

Ahora copiamos el directorio que contiene nuestra aplicación en **C:\inetpub\wwwroot** luego abrimos el administrador de IIS, expandimos el árbol de navegación y damos clic derecho sobre el directorio que acabamos de copiarlo y lo convertimos en aplicación.

Convertir en una aplicación web.

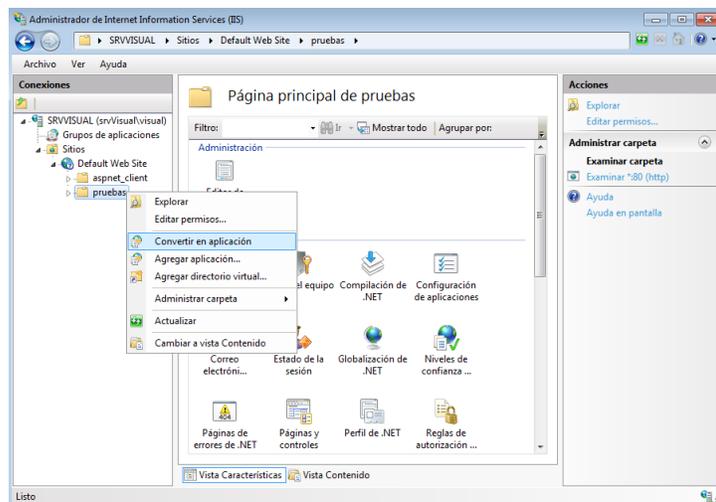


Fig. 49: Convertir en una aplicación web el aplicativo
Elaborador por: El investigador.

Esto nos permite acceder desde un navegador a nuestra aplicación, si estamos en una LAN podremos acceder por http://ip_servidor/pruebas y si nos queremos conectar desde el Internet necesitamos tener un DNS y una Ip pública.

4.4.2 Realización de las pruebas BETA

Las pruebas BETA son realizadas por el usuario final, es decir que se desarrollan en el entorno del cliente, un entorno que esta fuera del control de los desarrolladores. El usuario final empieza a utilizar el sistema en un ambiente real y trata de encontrar fallos en el sistema, los fallos encontrados deben ser reportados por escrito al desarrollador.

Para que el usuario reporte los errores encontrados en el software se ha creado la siguiente tabla, en la cual podrá describir el error y también podrá incluir sugerencias en cuanto a la interacción y manipulación del sistema.

Pruebas Beta realizadas.

| PRUEBAS BETA | |
|--------------------------------------|--|
| Usuario : Secretaría | |
| Fecha: 19-07-2014 | |
| Prueba realizada | Observaciones |
| Ingreso y modificación de equipos. | Al ingresar o modificar la información de los equipo en la grilla en el estado y grupo aparecen los grupos y estado que deberían ser. Pero al ingresar o modificar desde los botones en la que parece otra ventana aparecen más estados y grupos de otras categorías y campeonatos. Los grupos deberían estar solo los de la categoría a la pertenece el equipo. |
| Ingreso y modificación de jugadores. | El campo juvenil toca llenarlo manualmente, creo que sería mejor que este sea automático al ingresar la fecha de nacimiento. Tomando en cuenta que hay un juvenil por cada categoría. |
| Ingreso y modificación de | En las listas de valores no se están filtrando las fechas y los |

| | |
|--------------------------------------|--|
| calendarios. | grupos que pertenecen a esa fase. |
| Ingreso del detalle de los partidos. | Las tarjetas amarillas no pueden ser mayores que dos y las tarjetas rojas no pueden ser mayores que uno. |
| Finalizar Campeonatos. | Al finalizar un campeonato se debe cerrarse todo lo relacionado al mismo como (fases, fechas, grupos, calendarios, partidos, entre otros). |

Tabla 37: Pruebas beta
Elaborador por: El investigador.

4.4.3 Configuración del servidor web IIS en ambiente de producción

Para configurar un servidor web IIS en un ambiente de producción debemos seguir los mismo pasos que con los cuales configuramos un servidor en ambiente de pruebas, es decir solo agregamos otro directorio y lo convertimos en aplicación, la diferencia radica en la URL de acceso que para este ambiente será http://ip_servidor/liga.

Configuración de Internet Information Server.

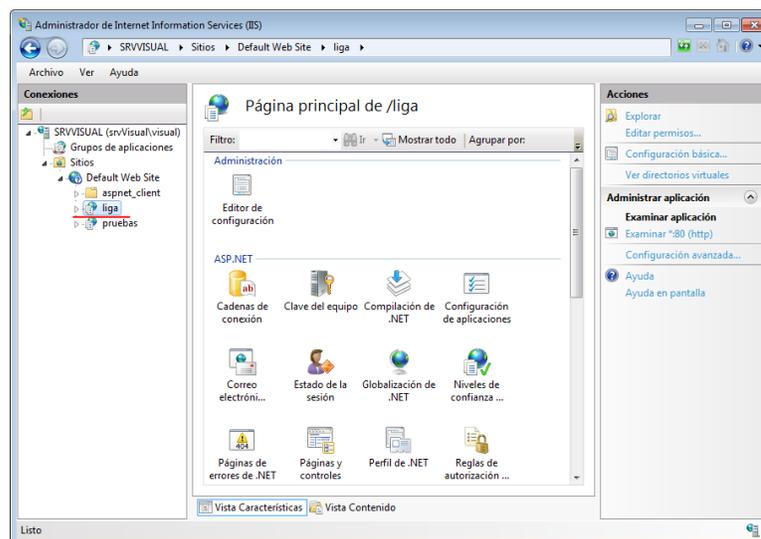


Fig. 50: Configuración del servidor web IIS
Elaborador por: El investigador.

4.4.4 Gestionar las incidencias que presente el sistema

La gestión de incidencias permite detectar los procesos que no están funcionando de una forma correcta o adecuada, esto permite llevarlos de una forma ordena y detallada, para posteriormente solucionar los problemas presentado.

Para gestionar las incidencias que presente el sistema en esta etapa y poder llevar un control sobre las mismas hemos creado la siguiente tabla, en la cual se detallaran las incidencias de una forma ordenada.

Incidencias del sistema.

| # | Pantalla | Proceso | Observación | Usuario | Corregido |
|---|-------------------|---------------------------------------|---|----------------|-----------|
| 1 | Jugadores. | Ingreso de jugadores. | El juvenil es por categoría. | Administrador. | Si |
| 2 | Campeonato. | Finalización de un campeonato. | Los goles, partidos jugadores, número de jugadores de los equipos deben quedar en cero. | Administrador. | Si |
| 3 | Calendarios. | Creación y modificación. | Las fechas y los grupos deben estar solo los que corresponden a esa fase y categoría. | Administrador. | Si |
| 4 | Campeonato. | Tabla de posiciones. | Deber ser agrupado por grupos y solo de las fases vigentes. | Administrador. | Si |
| 5 | Tabla Goleadores. | Generación de la tabla de goleadores. | La tabla de goleadores debe ser por campeonato y no uno por cada fase. | Administrador. | Si |

Tabla 38: Incidencias del sistema
Elaborador por: El investigador.

Estas fueron las principales incidencias generadas en el sistema en la fase pruebas y estabilización del sistema. Las incidencias presentadas han sido corregidas en un 100%.

4.4.5 Elaboración del plan de despliegue o implementación

Una vez que se ha finalizado con la fase de pruebas, se han corregido las incidencias presentadas en las mismas. Y la solución final cumple condiciones de calidad para su liberación final, se procede con el siguiente plan de despliegue.

Plan de despliegue e implementación.

| Fase | Estrategia | Tarea | Responsable |
|------------------------|---|---|--------------|
| Implementación. | Implementar la versión completa del sistema con la base de datos previamente parametrizada. | <ul style="list-style-type: none"> ✓ Instalación de la base de datos. ✓ Parametrizar la base de datos. ✓ Asignar permisos de ingreso a todas las formas al usuario Administrador. ✓ Instalación del plugin Silverlight en los clientes. | Investigador |
| Capacitación. | Capacitar uno por uno a los usuarios incluyendo al administrador. | <ul style="list-style-type: none"> ✓ Enseñar el funcionamiento del sistema. ✓ Capacitar en la generación de reportes. ✓ Enseñar a sacar respaldos de la base de datos. | Investigador |

Tabla 39: Plan de despliegue e implementación
Elaborador por: El investigador.

Las fases que se consideran dentro del plan de despliegue o implementación son las de implementación del sistema en sí y posterior a esta fase se iniciara con la capacitación a los usuarios.

4.5 Despliegue o Implementación

Esta es la etapa final del proyecto ya se ha comprobado la calidad de la solución la cual se encuentra lista para ser publicada, conjuntamente con la liberación de la solución final se entregaran los respectivos manuales e instructivos.

4.5.1 Liberación del sistema en producción

Para la liberación final del sistema, seguimos el plan de despliegue en sus dos etapas. Primero con la implementación del mismo. Para ello ya tenemos parametrizada la base de datos en el equipo de desarrollo, así como también tenemos instalado y configurado el Internet Information Server que es el servidor de la aplicación, la instalación de la base de datos se hace en el equipo de tiene asignada la Secretaría.

Una vez instalado y configurado el sistema procedemos con la capacitación a los usuarios finales y también a la persona encargada de sistemas de Liga Deportiva Cantonal Pelileo, de acuerdo con la planificación realizada en el plan de despliegue o implementación.

4.5.2 Elaboración de guías y manuales del sistema

El manual de usuario final y el manual técnico se encuentran en los **Anexos 3 y 4** respectivamente.

4.5.3 Creación de un registro de mejoras y sugerencias

Una vez que se ha implementado el proyecto, se procedió a la creación de una bitácora de registro de mejoras y sugerencias en el cual el usuario podrá ingresar a su criterio mejoras o sugerencias orientadas a los procesos del mismo. Dicha bitácora tiene el siguiente formato.

Registro de mejoras y sugerencias.

| # | Usuario | Procesos | Mejora | Sugerencia | Aplica |
|---|---------------|------------------------|--------------------------------------|--|--------|
| 1 | Administrador | Consulta de jugadores | Poder consultar jugadores por equipo | | Si |
| 2 | Administrador | Cierre de campeonatos | | Un reporte del todas de tarjetas amarillar y rojas de un campeonato | No |
| 3 | Administrador | Prestamos de jugadores | | Un enlace desde el préstamo del jugador para poder editar los datos del jugador. | No |

Tabla 40: Registro de mejoras y sugerencias
Elaborador por: El investigador.

4.5.4 Entrega del proyecto final

La entrega del proyecto final se hace conjuntamente con los respectivos manuales de usuario final los cuales se encuentran en el **Anexo 3**, un manual técnico se encuentra en el **Anexo 4** y una acta de configuración del sistema implementado, dicha acta se encuentra en el **Anexo 5**.

4.6. Discusión y resultados

Una vez implementada la aplicación web para el control de la fiabilidad de la información en la gestión de los campeonatos de fútbol en Liga Deportiva Cantonal Pelileo como propuesta de solución al problema planteado, se realizan las siguientes observaciones en un cuadro de dialogo comparativo en el cual se puede notar las diferencias entre un estado anterior y un estado posterior a la implementación del sistema.

Tabla comparativa de aspectos anteriores y posteriores a la implementación del sistema.

| Aspecto | Condiciones anteriores | Condiciones posteriores |
|--|---|--|
| Almacenamiento de la información. | Información almacenada en archivadores físicos, en archivos digitales y de forma descentralizada. | La información se almacena en una base de datos y está centralizada. |
| Organización de la información. | En archivadores físicos y digitales en diferentes lugares descentralizada. | En una sola base de datos de forma centralizada y organizada. |
| Integridad de la información. | Acceso y manipulación de la información por cualquier persona. | Acceso y manipulación de a información solo por personas debidamente autorizadas. |
| Búsqueda de información. | Búsqueda y revisión manual de la información en los archivadores y en los directorios digitales. | Búsqueda automática de información en base a determinados criterios de búsqueda. |
| Disponibilidad de la información. | Solo para las personas que conozcan la ubicación física de la información. | Para todas las personas que tengan acceso al sistema tomando en cuentas de debidas medidas de seguridad. |
| Retrasos en los procesos. | El volumen de información física y la descentralización de la misma generan retrasos en los procesos. | La organización y la centralización de la información agiliza los procesos. |
| Generación de informes. | En documentos de texto y en hojas de cálculo. | Generación automática de documentos con filtros y seguridades en varios formatos. |
| Cumplimiento de leyes y reglamentos por parte de los equipos y jugadores. | Control manual e insuficientes recursos tecnológicos y humanos para el control. | Una base de datos relacional y parametrizable que permite un mejor control de la información. |
| Información de cartas pase no actualizada. | Registro manual de cartas pase de jugadores año a año. | Registro automático de cartas pase de jugadores y gestión automática de los mismos. |

Tabla 41: Aspectos anteriores y posteriores a la implementación del sistema
Elaborador por: El investigador.

CAPÍTULO V

CONCLUSIONES Y RECOMENDACIONES

5.1. CONCLUSIONES

- ✓ La forma manual utilizada para la gestión y administración de los campeonatos de fútbol en Liga Deportiva Cantonal Pelileo, genera un ineficiente procesamiento de la información y no permite un control adecuado y eficiente de la misma.
- ✓ Los insuficientes recursos tecnológicos adecuados para el control de la fiabilidad de la información y la gestión de los campeonatos de fútbol genera que los jugadores y equipos no cumplan a cabalidad con los requisitos y exigencias de Liga Deportiva Cantonal Pelileo.
- ✓ La utilización de la metodología MSF conjuntamente con los diagramas UML permiten y facilitan la comprensión de la interacción entre los usuarios y la aplicación en la etapa de diseño.
- ✓ La metodología MSF se centra en la gestión y administración del proyecto para lograr el impacto deseado y obtener los resultados esperados.
- ✓ Los equipos hardware y el software que poseen los mismos cumplen con los requisitos mínimos para el funcionamiento de la aplicación web.
- ✓ El sistema desarrollado como propuesta de solución se ajusta fácilmente a las diversas formas existentes para administrar y gestionar un campeonato de fútbol, esto gracias a los parámetros con los que cuenta el sistema.

- ✓ La implementación de la aplicación web permitió optimizar y agilizar los procesos en la organización, administración y control de los campeonatos de fútbol en Liga Deportiva Cantonal Pelileo.

5.2. RECOMENDACIONES

- ✓ Se recomienda a la persona encargada de la administración de la aplicación obtener de forma constante y periódica respaldos de la base de datos para evitar pérdida de información.
- ✓ Todos y cada uno de los usuarios de la aplicación deberán ser cuidadosos con el manejo de sus contraseñas para evitar que estas caigan en manos de personas no autorizadas en el manejo y manipulación de la información. Para de esta forma mantener la información consistente y segura.
- ✓ Los usuarios de la aplicación deberán cambiar sus contraseñas de forma periódica, esto con el fin de mantener buenas políticas de seguridad de la información. Dichas contraseñas deberán ser complejas es decir alfa numéricas y con caracteres especiales.
- ✓ Antes de emitir los carnets de los jugadores se deberán revisar los parámetros en el sistema para evitar inconstancias en los mismos.
- ✓ Se recomienda a los usuarios del aplicativo revisar los manuales de funcionamiento ante cualquier duda en el manejo de los procesos en el aplicativo.

BIBLIOGRAFÍA

- [1] María Alexandra Sánchez Díaz “Sistema de Gestión para el control de procesos administrativos en la Asociación de Profesores de la Universidad Técnica de Ambato (APUA)”, Previo a la obtención de título de Ingeniero en Sistemas Computacionales a informáticos, Universidad Técnica de Ambato, Ambato – Ecuador, 2013.
- [2] Byron B. Cárdenas V. “Sistema automatizado para el registro de sanciones de los jugadores en la Liga Deportiva Parroquial Huachi Grande”, Previo a la obtención de título de Ingeniero en Sistemas Computacionales a informáticos, Universidad Técnica de Ambato, Ambato – Ecuador, 2012.
- [3] Juan P. Baltar, “Que es sistema automatizado”, [slideshare.net/JuanPabloBaltazarTorres](http://www.slideshare.net/JuanPabloBaltazarTorres), May. 12, 2011 [Online]. Disponible en: <http://www.slideshare.net/JuanPabloBaltazarTorres/que-es-un-sistema-automatizado> [Accedido: Jun. 28, 2013].
- [4] Auditoria, “Ventajas Y Desventajas de los Sistemas Automatizados”, auditoria.obolog.com, Feb. 23, 2008, [Online]. Disponible en: <http://auditoria.obolog.com/ventajas-desventajas-sistemas-automatizados-63189> [Accedido: Jul. 03, 2013].
- [5] Anónimo, “Control de la Información”, [securityartwork.es](http://www.securityartwork.es), Mar. 15, 2010, [Online]. Disponible en: <http://www.securityartwork.es/2010/03/15/control-de-la-informacion/> [Accedido: Jul. 08, 2013].
- [6] MarkusErb, “Gestión de Riesgo en la Seguridad Informática”, protejete.wordpress.com, [Online]. Disponible en: http://protejete.wordpress.com/gdr_principal/seguridad_informacion_proteccion/ [Accedido: Jul. 12, 2013].

- [7] Macarena Pereyra Rozas, “¿Cuáles son las medidas para proteger la información confidencial?”, iprofesional.com, Jul. 23, 2013, [Online]. Disponible en: <http://www.iprofesional.com/notas/91106-Cules-son-las-medidas-para-proteger-la-informacin-confidencial> [Accedido: Ago. 05, 2013].
- [8] BBVA, “Control de la Información”, inversores.bbva.com, 2008, [Online]. Disponible en: http://inversores.bbva.com/TLBB/fbinir/mult/RIC_Tit4_tcm926-348029.pdf [Accedido: Ago. 20, 2013].
- [9] Instituto de estudios deportivos (**IESPORT**), “GESTIÓN DE UN CAMPEONATO DE FÚTBOL 7 PARA AFICIONADOS”, iesport.es, [Online]. Disponible en: <http://www.iesport.es/es/comunidad-docente/documentosiesport.html?id=238> [Accedido Ago. 22, 2013].
- [10] La Bitacora de AudieMan, “Metodología Ágil MSF (Microsoft Solution Framework)”, audiemangt.blogspot.com, May 2010, [Online]. Disponible en: <http://audiemangt.blogspot.com/2010/05/metodologia-agil-msf-microsoft-solution.html> [Accedido Nov. 15, 2013].
- [11] Felipe Lozano Leon (Slideshare), “Presentación MSF”, es.slideshare.net, Jul 10, 2012, [Online]. Disponible en: <http://es.slideshare.net/GFLL/presentacion-msf-13589379> [Accedido: Nov. 17, 2013].
- [12] Carlos Julio Landázuri Ortiz, “ESTUDIO DE LA METODOLOGÍA MSF “MICROSOFT SOLUTION FRAMEWORK” APLICADA AL DESARROLLO DE UN SISTEMA DE FACTURACIÓN “SOFTWARE LIBRE” PARA LA UNIÓN DE PAPELERÍAS DE LA CIUDAD DE IBARRA.”, Carrera de Ingeniería en Sistemas Computacionales, Universidad Técnica del Norte Facultad de Ingeniería en Ciencias Aplicadas, Ibarra – Ecuador, 2013.

- [13] Alex Merino (Slideshare), “NORMA IEEE 830 PARA ESPECIFICACIÓN DE REQUERIMIENTOS DE SOFTWARE”, es.slideshare.net, Nov 17, 2012, [Online]. Disponible en: <http://es.slideshare.net/amerino2010/ieee-830> [Accedido: Dic. 05, 2013].
- [14] Microsoft, “Diagramas de casos de uso de UML: Instrucciones”, es.slideshare.net, Ene 28, 2011, [Online]. Disponible en: <http://es.slideshare.net/james1360/ieee-830-srs-6739034> [Accedido: Dic. 07, 2013].
- [15] BBVA, “Control de la Información”, msdn.microsoft.com, [Online]. Disponible en: <http://msdn.microsoft.com/es-es/library/dd409432.aspx> [Accedido: Mar. 20, 2014].
- [16] Pantusín Lema, Willian Fernando, Viracocha Bedoya, Edwin Mauricio “Análisis de la metodología MSF (Microsoft Solutions Frameworks) para el desarrollo del software, prototipo: sistema de administración en la compañía de transporte público urbano popular de pasajeros “Latacunga S.A. Citulasa” ubicada en la ciudad de Latacunga provincia de Cotopaxi.”, Previo a la obtención de título en Ingeniería Informática y Sistemas, Universidad Técnica de Cotopaxi, Latacunga – Ecuador, 2009.

AneXos

ANEXO 1: Especificación de requerimientos según estándar IEEE 830

1. Introducción

Liga Deportiva Cantonal Pelileo y el comité de fútbol, son los entes que se encargan año tras año de organizar el tormento de futbol, con el propósito de integrar a la comunidad pelileña, mejorar e incentivar el desarrollo físico y mental del ser humano.

1.1 Propósito

El comité de fútbol que es el encargado de la organización y control del campeonato de fútbol, abalizado por Liga Deportiva Cantonal Pelileo, se ha visto en la imperiosa necesidad de contar con un sistema de software que permita un mejor control de los sistemas de campeonato, ya que se ha tenido algunos inconvenientes tales como:

- Jugadores que estando sancionados e impedidos de jugar en sus clubes o en cualquier otro club perteneciente a LDPC, han sido habilitados y han jugado, esto debido a que se lleva un control de forma manual.
- Jugadores que han jugado en más de un equipo en diferentes categorías, estando eso prohibido y debidamente reglamentado en que un jugador podrá jugar únicamente en un determinado club.
- Otro de los problemas que se ha venido suscitando es con respecto a los pases de los jugadores, que perteneciendo a un determinado club y siendo el club propietario de sus derechos deportivos, se han vinculado a otros clubes sin arreglar sus derechos deportivos con el club propietario de sus derechos deportivos.

El propósito del desarrollo y la implementación de un sistema es que este permita controlar estos y otros aspectos, así como también permita agilizar los procesos relacionados con los campeonatos de fútbol.

Ámbito del sistema

- El sistema de control del campeonato de futbol se llamara **SOCCER MANAGER**
- Este sistema se encargará del control de los campeonatos de fútbol, en lo referente a: clubes, jugadores, árbitros, estadios, sanciones, multas, vocales, calendarios, partidos, etc. Y todo lo relacionado con lo que tiene que ver con un campeonato de fútbol.
- El sistema controlará con respecto a los clubes lo siguiente:
 - ✓ Las tablas de posiciones.
 - ✓ Controlará que los jugador que pertenezcan a un club, no se puedan inscribir en otro club sin la previa autorización del club dueño de sus derechos deportivos. Gestionará los pases de los jugadores.
 - ✓ Número de partidos jugados, ganados, empatados o perdidos.
 - ✓ Número de goles marcados y recibidos.
 - ✓ Multas y Sanciones impuestas a los clubes.
- El sistema controlará respecto a los jugadores lo siguiente:
 - ✓ Goles marcados por cada uno de los jugadores, en los clubes en los que milito y en los campeonato en los jugó.
 - ✓ El sistema se encargará de controlar que los jugadores que hayan sido suspendido no puedan jugar, ni en el club al que pertenece ni en ningún otro club que se encuentre inscrito en Liga Deportiva Cantonal Pelileo, hasta que haya cumplido su respectiva sanción.
 - ✓ Con una base de datos se podrá controlar las sanciones de los jugadores que hayan sido impedidos de jugar, cualquiera que haya sido el motivo.
 - ✓ Controlará que los jugadores una vez que hayan sido inscritos y formen parte de la nómina de un club, estos no puedan ser inscritos por otro club, es decir que un jugador solo puede ser inscrito una sola vez y por un solo club, para ese campeonato.

- ✓ Se llevará un registro del número de tarjetas amarillas, rojas y expulsiones.
 - ✓ Solo los jugadores que pertenezcan al mismo y que este club participe en las categorías inferiores podrá jugar en las categorías y sea esta primera o segunda categoría, este jugador deberá ser juvenil.
- En cuanto a los estadios el sistema contará con un listado de los estadios que han sido calificados y habilitados, para que en dichos estadios se puedan realizar los diferentes partidos de fútbol.
 - El sistema en cuanto a los árbitros solo llevará un control de quienes han pitado los partidos de fútbol. Los árbitros no son competencia de Liga Deportiva Cantonal Pelileo ya que ellos tienen su propio gremio.
 - En cuanto a las sanciones el sistema controlará las sanciones impuestas a los dirigentes de los clubes, jugadores y clubes según lo estipulado en el reglamento del campeonato interno de fútbol, o las que sean impuestas por el Comité de Fútbol o por LDCP.
 - El sistema no podrá programar las fechas de los partidos de forma automática, puesto que esto es hecho por el comité de fútbol, según lo establecido y especificado en el reglamento interno del campeonato de fútbol.
 - El sistema controlará las multas impuestas ya sea a clubes, jugadores, dirigentes, etc. Pero solo como información, ya que este sistema de control es totalmente independiente del sistema contable que posee LDCP.
 - Uno de los principales objetivos de este sistema es que se pueda controlar las sanciones impuestas a los jugadores, y que estos cumplan con las sanciones que se les han sido impuestas, ya en la actualidad como se lo hace de forma manual es imposible controlar a todos los jugadores de las diferentes categorías y clubes.
 - Los beneficios de este sistema, pues van a hacer que el campeonato se desarrolle con normalidad, para los clubes en cuanto a los derechos deportivos de sus jugadores, para el personal administrativo con disminución de la carga de trabajo y poder llevar los registros de partidos, jugadores, clubes, dirigentes o directivos de los clubes de una forma automatizada y de este modo no se puedan producir irregularidades como las que han venido suscitando. Como el cambio

de fotografías en los carnets con datos de otros jugadores, para solucionar este problema el sistema generará carnets con fotografías impresas en el mismo carnet, además de un código de barras con el número de la cédula de ciudadanía.

- En general el sistema se encargará de todo el control de los campeonatos de fútbol en todas sus fases y etapas.

1.2 Definiciones, Acrónimos y Abreviaturas

- **BDD** (base de datos).
- **LDCP** (Liga Deportiva Cantonal Pelileo).
- **IIS 7** (Internet Information Server 7).
- **MSF** (Microsoft Solution Framework)

1.3 Referencias

- Reglamento para el campeonato interno de fútbol.
- Acta de juegos (un acta para el registro de un partido)
- Acta para el informe arbitral(solo para árbitros de la federación nacional de arbitraje)
- Formulario de inscripciones de jugadores.
- Formularios para la inscripción de los clubes (una hoja para los directivos del club, y otra para la nómina de los jugadores propios del club).

1.4 Visión General del Documento

El sistema tendrá los listados de los clubes que se inscriban con sus respectivos jugadores y directivos en el presente campeonato, estarán agrupados de acuerdo a las siguientes categorías las cuales se conforman de acuerdo a l Art. 18 del reglamento del

campeonato interno de fútbol: Primera categoría serie A, Primera categoría serie B, segunda categoría, categorías sub 10, sub 12, sub 14 y sub 16. El sistema contará con parámetros, con el fin de brindar flexibilidad.

Las sanciones que les hayan sido impuestas en campeonatos pasados a: clubes, jugadores y directivos que aún no han sido cumplidas, estas sanciones constaran en la base de datos del sistema.

Los clubes, jugadores y directivos que hayan sido sancionados por cualquier motivo deberán cumplir las sanciones que se les hayan sido impuestas según el reglamento, para que puedan actuar y ser habilitados en sus respectivos clubes o en otros clubes si así es su deseo.

Cualquier club nuevo que decida participar en el campeonato lo podrá hacer y jugara en la segunda categoría. Si un club de primera categoría A y B no se inscribe para su participación en el presenta campeonato automáticamente perderá su categoría. Los jugadores y clubes podrán ser inscritos en el presente campeonato siempre y cuando no hayan sido suspendidos un ano calendarios y posean deudas con la LDGP, los jugadores serán inscritos una sola vez y por un solo club.

El sistema contará también con un listado de todos los estadios en los que se jugaran los diferentes campeonatos en las distintas categorías.

2. Descripción General

2.1 Perspectiva del Producto

Con la implementación de este sistema los campeonatos se desarrollaran sin mayores inconvenientes, y se lograra corregir todas aquellas irregularidades y anomalías que han venido suscitándose a los largo de varios años.

Se lograra que las sanciones que sean impuestas ya sea a jugadores, directivos a los mismos clubes se cumplan de acuerdo con el reglamento. También se pretende que este sistema no solamente pueda ser utilizado en LDGP, si no que pueda ser usado en cualquier Liga Deportiva, ya sea esta cantonal, parroquial o barrial, ya que el sistema contará con parámetros, los cuales pueden ser acoplados a los distintos sistemas de campeonato de fútbol.

Este sistema es totalmente independiente de la parte contable o de cualquier sistema contable y que se trata de un sistema de administración y control de campeonatos de fútbol. El sistema únicamente registrara los valores de las multas impuestas ya sea a equipos o jugadores.

2.2 Funciones del Producto

El sistema estará conformado por una base de datos de será la parte macro del sistema la cumplirá las siguientes funciones:

- Mostrar todos los clubes que se encuentren en las diferentes categorías y campeonatos.
- Una lista con todos los jugadores que hayan sido inscritos por sus respectivos clubes.
- También tendrá una lista de los directivos de cada uno de los clubes, ya que estos serán los que representen a sus respectivos clubes.
- Se mantendrá un registro de los pases de los jugadores.
- Registrar las sanciones y multas que se les haya sido impuestas tanto a jugadores, directivos y clubes.
- Permitirá registrar las actas que se levanten en cada partido, en ellas constara: El lugar en donde se desarrolló, la fecha, la hora, la categoría, el o los árbitros que pitaron el encuentro, los nombres de los clubes, el listado de jugadores titulares y suplentes, número de goles marcados por cada jugadores, marcador del partido y

el informe arbitral en cual constará todas las circunstancias que se hayan producido en el desarrollo del partido.

- Logra corregir todas las anomalías que se han venido dando a lo largo de muchos años.
- El sistema estará en la capacidad de generar los siguientes informes: Tabla de goleadores por campeonato, tabla de posiciones por grupos y fases, tabla de posiciones acumulada, Calendarios por fases, fechas y grupos esto previo ingreso de los mismo, carnets de los jugadores (por campeonato, por equipo y de forma individual de un listado de jugadores). Listado de jugadores propios y refuerzos de un determinado club.
- El sistema estará en la capacidad de trabajar con roles a los cuales se tendrá que asignar los usuarios.

2.3 Características de los Usuarios

Básicamente tendremos dos tipos de usuarios, un usuario que será el administrador del sistema y los otros usuarios será el personal administrativo de LDGP, el administrador del sistema tendrá todos los privilegios de un administrador del sistema, mientras que el personal administrativo tendrá algunas restricciones sobre el sistema, estos usuarios no podrán realizar ninguna modificación ya sea física o lógica a la base de datos o al mismo sistema.

El administrador será la persona encargada de dar mantenimiento al sistema y de asignar los permisos que sean necesarios tanto a roles y asignar un rol a un o unos usuarios.

2.4 Restricciones

- Las restricciones en cuanto a las funciones del sistema se rigen al reglamento interno del campeonato de futbol.

- En cuanto a restricciones de hardware la LDCP no cuenta con equipos servidores, por lo que el sistema será instado en un computador de escritorio, bajo el sistema operativo Windows XP o 7.
- En cuanto a lenguajes de programación no existen ninguna restricción, los protocolos de comunicación que se utilicen serán el conjunto de protocolos de la familia TCP/IP, puesto que existen computadores los cuales tienen que estar intercomunicados.
- El software que se utilice para desarrollar el sistema queda a consideración de los desarrolladores puesto que no existe ningún tipo de restricción.
- Las interfaces de usuario serán fáciles de manejar, se aplicará interfaces de entorno web.
- La seguridad del sistema deberá estar acorde a estándares de seguridad de la información, tanto para aplicaciones de escritorio como para aplicaciones web.
- LDCP no cuenta con un dominio de internet, ni con una dirección IP pública, por lo cual el sistema funcionará en una Intranet. El sistema estará disponible únicamente en plataformas Windows, y bajo el servidor web Internet Information Server.

2.5 Suposiciones y Dependencias

Suposiciones

Los datos que estén en el sistema tendrán un alto grado de seguridad y confiabilidad, los cuales únicamente serán manipulados por el personal administrativo de LDCP.

Cualquier petición de cambio en el sistema estará sujeta a la revisión y aprobación del equipo de desarrolladores.

Dependencias

Algunas de las reglas pueden cambiar en el reglamento interno del campeonato de futbol, pero estas no afectaran al sistema lo único que se tendrá que hacer es modificar los parámetros, ya que este sistema será desarrollado para que sea configurable para que se pueda adaptar fácilmente a cualquier sistema de campeonato, siempre y cuando las reglas base.

El sistema será desarrollado solamente para que corra bajo la plataforma de Windows, en cualquiera de sus versiones.

2.6 Requerimientos futuros

Los requerimientos y perspectivas de LDCP son el poder adquirir un servidor y un dominio para que este sistema esté disponible desde el internet, para de esta formar mejorar el servicio a la colectividad pelileña.

3. Requerimientos Especificos

3.1 Interfaces Externas

Interfaces de usuario

La interfaz de usuario deberá ser desarrollada de forma muy sencilla, de forma que cualquier usuario con conocimientos básico de informática puede manejar.

Interfaces de hardware

Las interfaces deberán ser diseñadas para poder ser manejadas ya sea con el mouse, con el teclado e interactuar con un lector de código de barras. Esto por el código de barras que tendrán los carnets de los jugadores.

Interfaces de software

El sistema será instalado bajo una plataforma windows, en un sistema operativo windows 7, 8 o XP ya que el sistema solo tendrá 3 usuarios, yo como diseñador recomiendo que el sistema sea instalado en la misma plataforma de windows pero en un sistema operativo cliente/servidor, para que el sistema funcione de una manera más eficiente y pueda ser explotado el 100% de sus funcionalidades.

3.2 Funciones

Procesos:

Proceso de inscripciones y requisitos

Las inscripciones de los clubes se receptaran en las oficinas de LDCP, dentro de las fechas establecidas en todas las categorías.

- De los clubes nuevos: Todos los que decidan participar en el campeonato interno de futbol lo podrán hacer cumpliendo los siguientes requisitos.
 - ✓ No ser un club desertor del campeonato pasado
 - ✓ Sobre el club no debe pesar ninguna sanción.
 - ✓ Los clubes participaran en la segunda categoría o en las. categorías inferiores.
 - ✓ Cada uno de los clubes que decida participar en la segunda categoría cancelarán un monto por derecho de afiliación, también depositarán una determinada cantidad de dinero como garantía, la misma que se hará efectiva en el caso de que resultan desertores, estos montos serán determinados por el Comité de Fútbol, así también se Comité determinara los montos para las categorías inferiores.

- De los clubes que formen parte de la primera categoría.
 - ✓ No haber sido suspendido en el campeonato pasado.
- Deberán presentar un la nómina de sus jugadores que pertenecen al club, caso contrario deberán presentar los pases definitivos o cartas de préstamo legalizadas por el Comité de Fútbol, cada club también presentara sus directivos en los formularios entregados por el Comité.
- Todos los clubes están obligados a presentar la nómina de los jugadores de su pertenencia.
- En las nóminas de los jugadores constará: Nombres, Apellidos, Número de cédula y las respectivas firmas de los jugadores.
- En la nómina de los directivos constará: El nombre del club, la dirección y teléfonos de la sede del club, los nombres, dirección y teléfono de los directivos del club, en sus respectivos cargos.
- Deberán presentar un certificado de no adeudar a LDGP.
- Ninguno de los clubes podrán inscribir a jugadores que hayan sido sancionados.
- Cada equipo podrá jugar con jugadores pelileños y dos refuerzos foráneos o extranjeros (opcional).
- Los clubes podrán inscribir un mínimo de 16 jugadores en todas sus categorías.
- Los jugadores serán inscritos una sola vez y por un solo club, es decir que un jugador puede participar en un solo club. En caso de hacerlo será sancionado de acuerdo al reglamento.

Inauguración del campeonato

Cada club deberá presentarse con un mínimo de 11 jugadores con sus respectivos uniformes y sus directivos.

Sistema de campeonato

- El comité de fútbol presentara el calendario de juegos al inicio de temporada, el mismo que será enviado a todos los clubes participantes.
- El lugar, el día y la hora de los partidos serán determinados por el comité de fútbol en las sesiones de los días lunes.
- El ascenso de los clubes se realizará de acuerdo a las siguientes normas:
 - ✓ Los clubes de la categoría primera B que alcancen los cuatro primeros lugares, ascenderán a la primera categoría A, para completar 20 clubes en primera A. En caso de haber un mayor número de clubes que desciendan, ascenderá el siguiente inmediato superior.
 - ✓ Los clubes de la segunda categoría que alcancen los seis primeros lugares, ascenderán a la primera categoría B, para completar 22 clubes en primera B. En caso de haber un mayor número de clubes que desciendan, ascenderá el siguiente inmediato superior.
- El descenso de los clubes participantes, se realizará de acuerdo a las siguientes normas:
 - ✓ Descenderán de la categoría primera A, los equipos que ocupen los tres últimos lugares, más los equipos sancionados y desertores.
 - ✓ Descenderán de la categoría primera B, los equipos que ocupen los cinco últimos lugares, incluidos los equipos sancionados y desertores.
- No existirá compra ni venta de categorías.
- Los encuentros de fútbol únicamente serán suspendidos de acuerdo con lo estipulado en el reglamento.
- Una vez iniciado el campeonato los clubes participantes no podrán contraer compromisos deportivos en otros campeonatos.
- Para dar inicio a un partido, cada club deberá tener en cancha un mínimo de 7 jugadores.

Modalidad del campeonato

- Tanto en la primera A como en la primera B jugaran todos contra todos una vuelta, clasificaran los ocho primeros en cada categoría, los tres primeros lugares tendrán 3, 2 y 1 punto de bonificación respectivamente.
- La semifinal se jugara con los 8 clubes, de los cuales clasificaran cuatro los mismos que jugaran el primero con el cuarto y el segundo con el tercero, los ganadores jugaran por el primero y segundo lugar, mientras que los perdedores jugaran por el tercer y cuarto lugar.
- En la segunda categoría se harán series, siendo los recién descendidos cabecillas de las series, jugaran todos contra todos clasificando 16 de todas las series.
- Se dividirán en series de acuerdo al número de clubes inscritos.
- La segunda fase se jugara con los 16 clubes, en 2 series de las cuales clasificaran el primero y el segundo de cada serie, y jugaran el primero de la serie A con el segundo de la serie B en forma cruzada y los ganadores irán a disputar el primero y segundo lugar.
- Los partidos que se desarrollen en el Estadio Ciudad de Pelileo, el ingreso máximo de 20 personas entre ellos: 17 jugadores, 1 entrenador, 1 masajista y el entrenador.
- Al término del campeonato se entregaran los premios que se estipulan en el reglamento.

En cuanto a los árbitros

- Los partidos de la serie A serán dirigidos por Jueces de la Asociación de Árbitros de Fútbol Profesional de Tungurahua, en el resto de categorías los encuentros serán dirigidos por Jueces del Comité Amateur de Árbitros San Pedro de Pelileo.

- En caso de inconformidad del club o de los clubes tanto de primera B como de la segunda categoría, estos podrán solicitar arbitraje de la Asociación de Árbitros de Fútbol Profesional de Tungurahua.
- Un Árbitro no podrá ser dirigente ni jugador de ningún club.
- Un Árbitro no podrá dirigir más de dos encuentros en el mismo día, solo en las categorías inferiores podrá dirigir un máximo de tres partidos.
- El Comité Amateur de Árbitros San Pedro de Pelileo, deberá presentar la designación de los árbitros en la Secretaría de LDCP.

Los cotejos

- En cada uno de los partidos se levantará un acta.
- Los jueces del Comité Amateur de Árbitros San Pedro de Pelileo, llenaran el acta que les proporciona LDCP, mientras que los Jueces de la Asociación de Árbitros de Fútbol Profesional de Tungurahua llenaran dos actas.
- Las actas deberán ser llenas de forma obligatoria, cada uno de sus puntos.

Los estadios en los cuales se jugaran los encuentros.

El comité de Fútbol designara el lugar la hora y la fecha de los encuentros en cada una de las categorías.

Las categorías

El campeonato interno de fútbol cuenta con las siguientes categorías:

- Primera categoría A
- Primera categoría B
- Segunda categoría.

- Categoría Sub 16
- Categoría Sub 14
- Categoría Sub 12
- Categoría Sub 10

Solo los clubes que tengas categorías inferiores, sus jugadores podrán jugar en las categorías superiores.

Las sanciones

- Las sanciones podrán ser tanto para clubes como para jugadores de un año calendario.
- Los clubes que inscriban jugadores que hayan sido sancionados, el club, sus directivos y el jugador será sancionados a acuerdo con el reglamento.
- Los jugadores que acumulen cinco tarjetas amarillas deberán descansar un partido obligatoriamente.
- Los jugadores que tengan una tarjeta roja de forma directa en un partido darán descansar dos partidos y si es reincidente deberá descansar el doble.
- Los jugadores que sean expulsados por doble tarjeta amarilla descansara un partido.
- Las demás sanciones que se establecen tanto para jugadores, clubes, directivos, etc. Se encuentran estipulados en el reglamento.

Los derechos deportivos

- Un jugador juvenil que juegue dos años consecutivos en el mismo club el club pasa a ser dueño de sus derechos deportivos, salvo el caso que el jugador sea dueño de su pase, el pase pertenezca a otro club o tenga un representante.

- Jugador que participe dos años consecutivos en el club, este pasa a ser el dueño del pase.
- Jugador que desee separarse del club para obtener su carta pase de libertad, deberá estar inactivo un año en las categorías inferiores, y dos años en las categorías de mayores.
- El equipo que no participe en el campeonato sus jugadores son libres de actuar en cualquier club.
- Los jugadores deberán presentarse en cancha con su respectivo carnet, el mismo que es extendido por LDCP en el momento de la inscripción.

3.3 Requerimientos de Rendimiento

En cuanto a los tiempos de rendimiento del sistema y de la base de datos, se basan en los estándares internacionales, los cuales nos dicen que todo sistema deben tener un tiempo de respuesta inferior a los 5 segundo, de igual forma la base de datos debe cumplir con los mismos requerimientos, para que un sistema sea calificado como óptimo en su rendimiento debe cumplir esta y otras normas de desarrollo [14].

3.4 Restricciones de Diseño

Seguridad

El sistema debe cumplir con los máximos niveles de seguridad para que la información que se ingrese al sistema pueda ser confiable y útil. Para esto el sistema contará con roles y usuarios.

Base de datos

Solo el administrador podrá realizar cambios en la base de datos todos los demás roles o usuarios se les restringirá el acceso o la modificación a determinadas tablas u objetos que se encuentren a la BDD.

Políticas de empresa

Liga Deportivo Cantonal de Pelileo no ha hecho ninguna restricción respecto a las herramientas que se utilizaran para el desarrollo del sistema.

3.5 Atributos del Sistema

Requisitos de usuarios

Los requerimientos son en cuanto a las interfaces, ya que estas deben ser diseñadas y desarrolladas, de forma que sean muy amigables, fáciles de manejar y entender.

La interfaz será una sola para todos los usuarios, restringidas determinadas funciones y páginas dependiendo el tipo de usuario.

Solo el administrador tendrá todos los privilegios sobre el sistema.

Requerimientos tecnológicos

El sistema será desarrollado bajo la plataforma Windows y podrá ser instalado de preferencia en los siguientes sistemas operativos: Windows 7 Windows 8, para que el sistema preste una mayor funcionabilidad y velocidad, se recomienda que sea instalado en un sistema operativo de tipo servidor como: Windows server 2008 o 2012. En el sistema operativo deberá estar configurado IIS 7 con net framework 4.0.

3.6 Otros Requisitos

El método de desarrollo para llevar a cabo este proyecto será el método MSF (Microsoft Solution Framework).

4. Apéndices

- **Apéndice 1:** Entrevista. Se encuentra en el Anexo 2.

Anexo 2: Resumen de entrevista.

UNIVERSIDAD TÉCNICA DE AMBATO

FACULTAD DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA E INDUSTRIAL

ENTREVISTA

Objetivo: Recolectar información sobre los principales inconvenientes que tiene Liga Deportiva Cantonal Pelileo con la organización y desarrollo de los campeonatos de fútbol.

Nombre del entrevistado: Secretaria. Sra. Catalina Llerena.

Persona encargada de la entrevista: Investigador. Byron Moreno.

Fecha: Viernes 11 de noviembre del 2013.

En la entrevista realizada a la secretaría de Liga Deportiva Cantonal Pelileo, podemos destacar las siguientes preguntas y un resumen de sus respectivas respuestas. La entrevista que se realizó fue una entrevista no estructurada.

¿Cuáles son los principales problemas que se han venido suscitando?

Bueno los principales problemas que tenemos son que los jugadores que juegan en un club en una determinada categoría, han jugado en otro club ya sea de la misma categoría o de otra categoría, los clubes han inscrito a jugadores sancionados, hay jugadores que para jugar han cambiado las fotos de su carnet o algún otro dato.

¿Cuáles son los requisitos que los clubes deben cumplir para inscribirse?

Los clubes que decidan participar deberán cumplir los requisitos establecidos en el reglamento interno según el artículo correspondiente a la inscripción de clubes.

¿Quién revisa la información de la documentación que presenta un club para un determinado campeonato?

La comisión de fútbol es la encargada de revisar la documentación de los jugadores que presenta un club, ellos son los que habilitan a los jugadores.

¿De qué forma se lleva el control sobre los equipos y jugadores que participan en un campeonato de fútbol?

Se lo lleva de forma manual en hojas de Excel se registran los goles de cada equipo, así como también las tarjetas y las sanciones que se aplican a los jugadores. Hay que tener en cuenta que al hacer esto podemos cometer errores ya que son más de cien equipos y unos cinco mil jugadores sumados de todas las categorías, y no se puede controlar de una forma eficiente.

Anexo 3: Manual de usuario

El objetivo del manual de usuario es el de proporcionar a los usuario finales una guía para el correcto manejo de los procesos en el sistema, en este manual se explica los controles y las funcionalidades de las diferentes pantallas que tiene la aplicación.

La aplicación puede ser ejecutada en cualquier navegador con solo instalar el plugin de **Silverlight**, es todo lo que se necesita para ejecutar la aplicación, el usuario puede acceder al aplicativo a través de la siguiente URL http://ip_equipo_servidor/liga esto en una intranet, si se desea acceder desde el internet habrá que habilitar una IP pública.

Página de inicio de sesión.

Esta es la página de inicio de sesión del sistema a través de la cual el usuario podrá autenticarse ingresando un nombre de usuario y una contraseña.

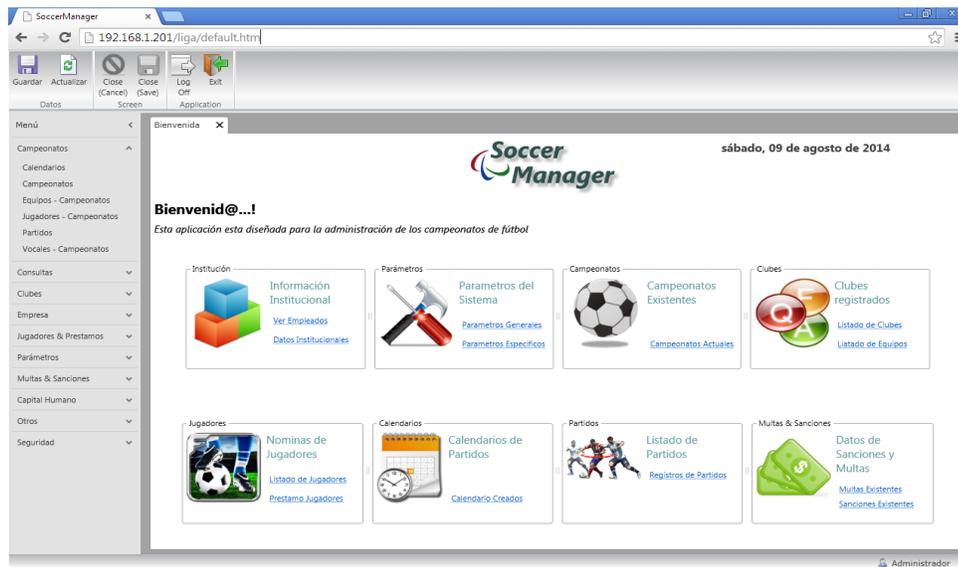


Pantalla 1: Inicio de sesión.
Fuente: Sistema SOCCER MANAGER.

a) **Inicio de sesión:** Contiene controles que permite iniciar una sesión digitando el nombre de usuario y la contraseña, luego estos datos son validados por el sistema.

Pantalla principal

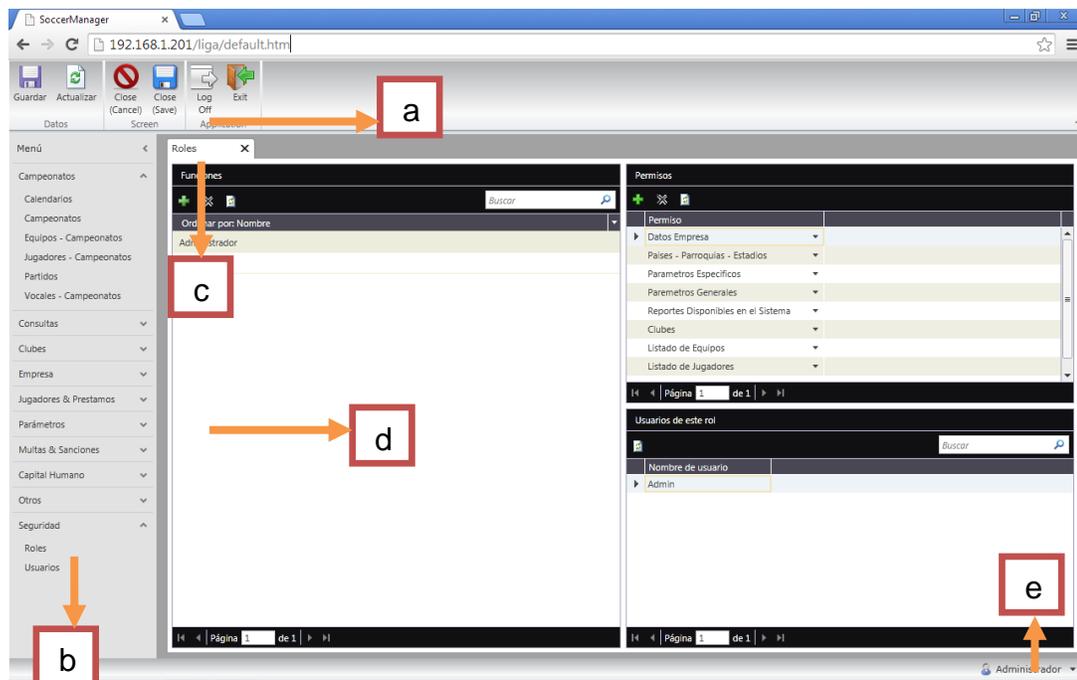
La pantalla de la aplicación contiene, el nombre del sistema, la fecha del sistema y algunos enlaces directos a pantallas de procesos que se los utiliza con mayor frecuencia.



Pantalla 2: Home del sistema.
Fuente: Sistema SOCCER MANAGER.

Pantallas de creación y mantenimiento de roles y usuarios.

Estas pantallas son de creación y mantenimiento de roles, permisos de accesos a pantallas y usuarios. Una ventana contiene las siguientes partes.



Pantalla 3 : Creación y mantenimiento de roles.
Fuente: Sistema SOCCER MANAGER.

a) Botones de acción: Contiene botones que realizan acciones sobre los datos o sobre la aplicación. Agrupadas en tres partes.

Datos: Contiene los siguientes botones, y aparecen en todas las pantallas.



Guardar: Permite almacenar los cambios realizados de la pantalla que este activa.



Actualizar: Refresca los datos de la pantalla que este activa.

Acciones: En dependencia de cada pantalla. Unas pantallas tendrán algún botón y otras no.

Screen: Contiene los siguientes botones, ejecutan acciones sobre cada pantalla activa.



Close
(Cancel)

Close (Cancel): Cierra la ventana activa sin guardar los datos.



Close
(Save)

Close (Save): Cierra la ventana activa guardando los datos.

Aplicación: Contiene los siguientes botones, ejecutar una acción sobre toda la aplicación.



Log
Off

Log Off: No realiza ninguna acción, ya que presenta un error propio del componente.



Exit

Exit: Cierra toda la aplicación en ocasiones puede presentar algún error propio del componente.

b) Menú del sistema: Contiene los accesos a las diferentes pantallas que tiene el sistema agrupados por categorías, el menú varía en dependencia de los permisos que tenga cada rol.

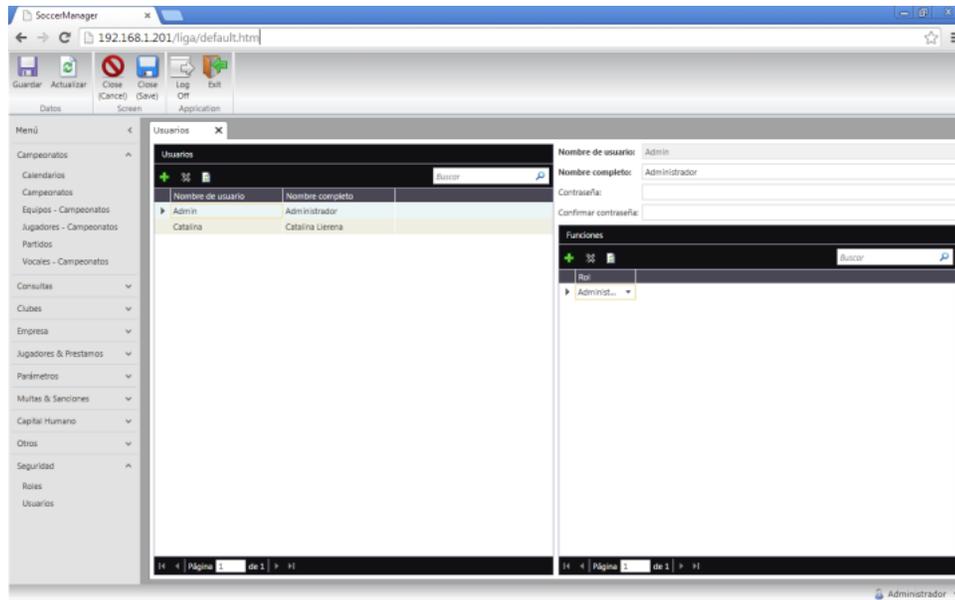
c) Pantalla Activa: Es la pantalla sobre la cual estamos trabajando, si hay datos que deben ser guardados aparecerá un (*) en la título de la pantalla.

d) Área de trabajo: Es el lugar donde se despliegan los datos.

e) Nombre del usuario conectado: Despliega el nombre completo del usuario que esta conecto en el sistema. Al dar clic en el botón que aparece al final del nombre, el usuario puede cambiar su contraseña.

En la grilla de permisos podemos asignar los permisos de acceso a las pantallas, para ello tenemos una lista de valores con los permisos que se pueden asignar.

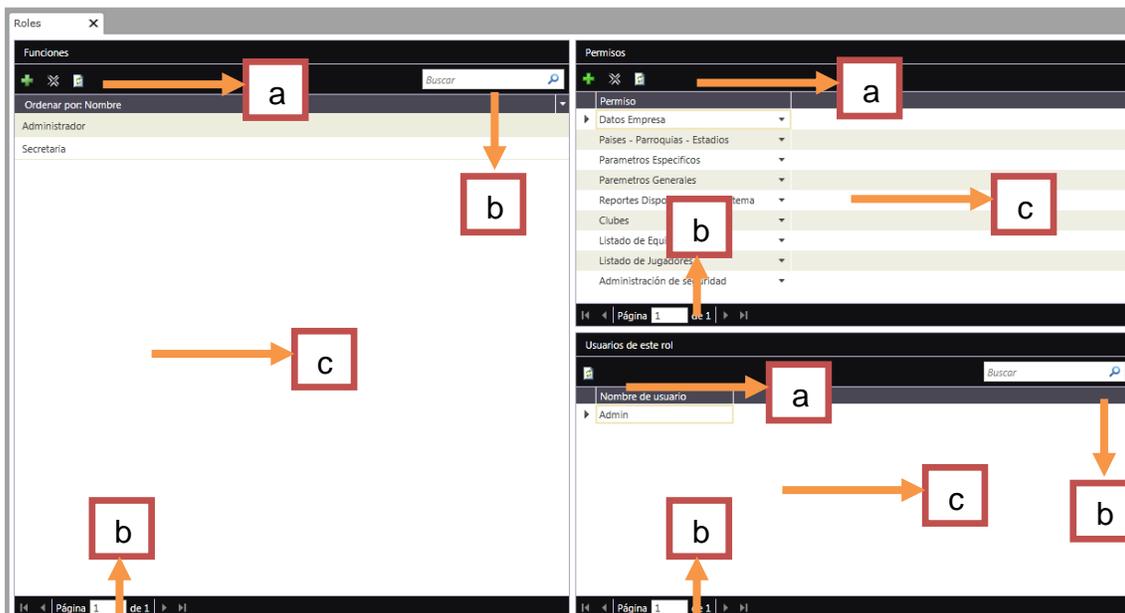
En la siguiente pantalla podemos ingresar y modificar los usuarios de la aplicación, así como también asignarles un rol o función.



Pantalla 4: Creación y mantenimiento de usuarios.
Fuente: Sistema SOCCER MANAGER.

Componentes de una pantalla

Los componentes de una pantalla de forma general son los siguientes. Algunas pantallas pueden o no pueden tener todos estos componentes. Varían de acuerdo a la funcionalidad de la pantalla. En la aplicación la diferencia entre una ventana y una pantalla es que una ventana puede contener muchas pantallas. Pero solo la pantalla permite al usuario interactuar con los datos.



Pantalla 5: Componentes de una pantalla.
Fuente: Sistema SOCCER MANAGER.

a) Botones de manipulación de datos: Cada grilla en dependía de su funcionalidad tiene botones de manipulación de datos estos pueden ser.

-  Permite agregar un nuevo registro ya sea en la grilla o en una ventana modal.
-  Permite modificar un registro en la misma grilla o en una ventana modal.
-  Permite eliminar un registro.
-  Permite refrescar los datos de la pantalla.

b) Cuadro de búsqueda: Permite realizar búsquedas en campos tipo texto en la grilla de datos.

c) Grilla de datos: Contiene datos que pueden ser manipulados.

d) Índices de navegación: Estos índices permiten navegar entre páginas de una pantalla, estas páginas se crean automáticamente en dependencia del volumen de dato.

Pantallas de parámetros.

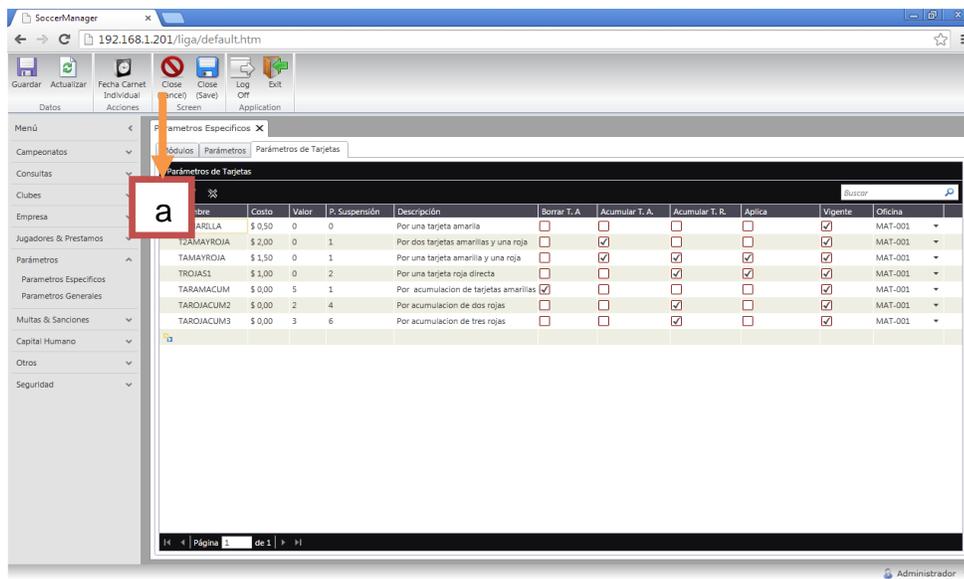
Las pantallas de parámetros de la aplicación se encuentran en la categoría **Seguridad** dentro del menú. En estas pantallas se deben definir los parámetros que utiliza la aplicación. Existen dos pantallas una de parámetros generales y otra de parámetros específicos, las cuales tienen varias pestañas.

La pantalla de parámetros generales contiene las siguientes pestañas.

| Nombre Pestaña | Descripción | Obligatorio |
|------------------------------|--|-------------|
| Estados | Contiene un listado de estados para los siguientes procesos, cabe mencionar que solo el proceso General aparece en todos los otros procesos (General, Jugador, Calendario, DetalleCalendario, Equipo, Multa, Sancion). | Si |
| Estado Civil | Listado de estados civiles (al menos uno) | Si |
| Operadoras Telefónica | Listado de las operadoras telefónicas | No |
| Tipo Contribuyente | Listado de los tipos de contribuyentes(al menos uno) | Si |
| Tipo Parámetro | Estos pueden ser de tipo fecha, número o texto | Si |
| Tipo Sanción | Solo pueden ser de dos tipos Multa o Sanción | Si |

Matriz 1: Listado de pestañas de la pantalla parámetros generales.
Fuente: Sistema SOCCER MANAGER.

La otra pantalla es de parámetros específicos, la cual tiene tres pestañas (Módulos, Parámetros, Parámetros de tarjetas). Para obtener mayor información sobre estos parámetros revisar la parte de parametrización de sistema.

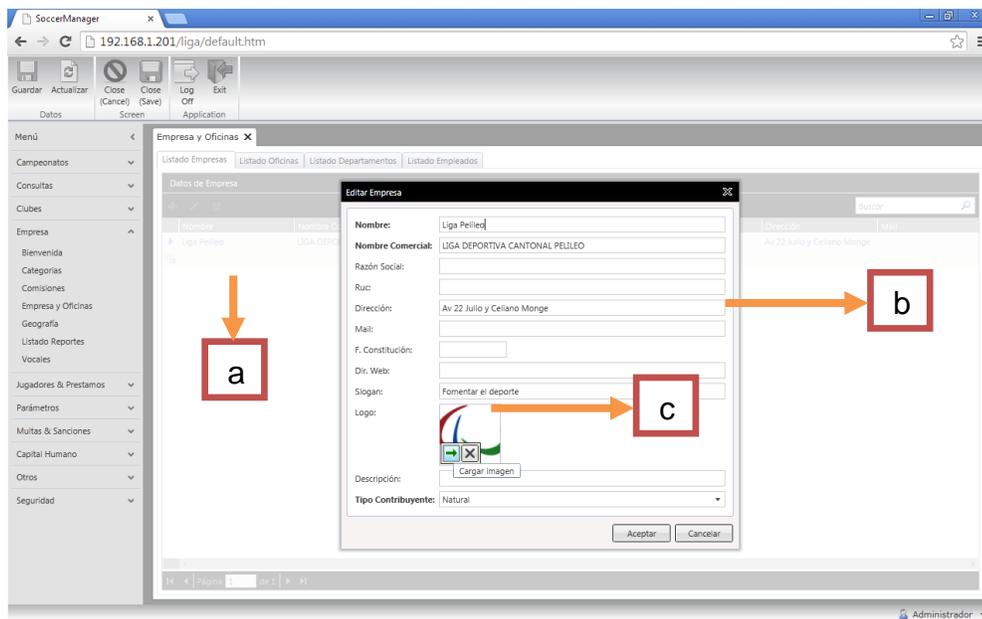


Pantalla 6: Parámetros específicos.
Fuente: Sistema SOCCER MANAGER.

a) **Fecha carnet individual:** Establece la fecha y la hora actual en la columna **valor** del parámetro **CarnetIndividual**, esto permite crear un punto de partida para poder carnetizar jugadores de forma individual, se carnetizaran todos los jugadores que tengan marcada la columna **Carnetizado** en la pantalla **Jugadores**, la cual haya sido marcada entre la fecha y hora establecida anteriormente y la fecha y hora en la que se generó el reporte.

Pantallas de información institucional y localización geográfica.

Para ingresar la información institucional se lo puede hacer por la pantalla **Empresa y Oficinas** que se encuentra en la categoría **Empresa** del menú. Esta pantalla contiene algunas pestañas en las cuales se debe ingresar información básica de la institución.



Pantalla 7: Ingreso y mantenimiento de información institucional y localización geográfica.
Fuente: Sistema SOCCER MANAGER.

a) Grilla de datos: Contiene datos referentes a la institución como el nombre, ruc, etc.

b) Ventana modal: Es una ventana que se despliega sobre la pantalla desde la cual es llamada y permite ingresar nuevos datos o modificar los datos existentes. Todas las pantallas de sistema tienen una ventana modal por defecto.

c) Componente para imágenes: Este componente permite cargar una imagen. Para hacerlo solo se debe hacer clic sobre el componente y este desplegará un cuadro de diálogo en el cual debemos indicar la ruta de la imagen. El campo sello que es de tipo imagen debe ser una imagen marca de agua.

Todos los datos que son obligatorios y relevantes de llenar están en color **Negrita**.

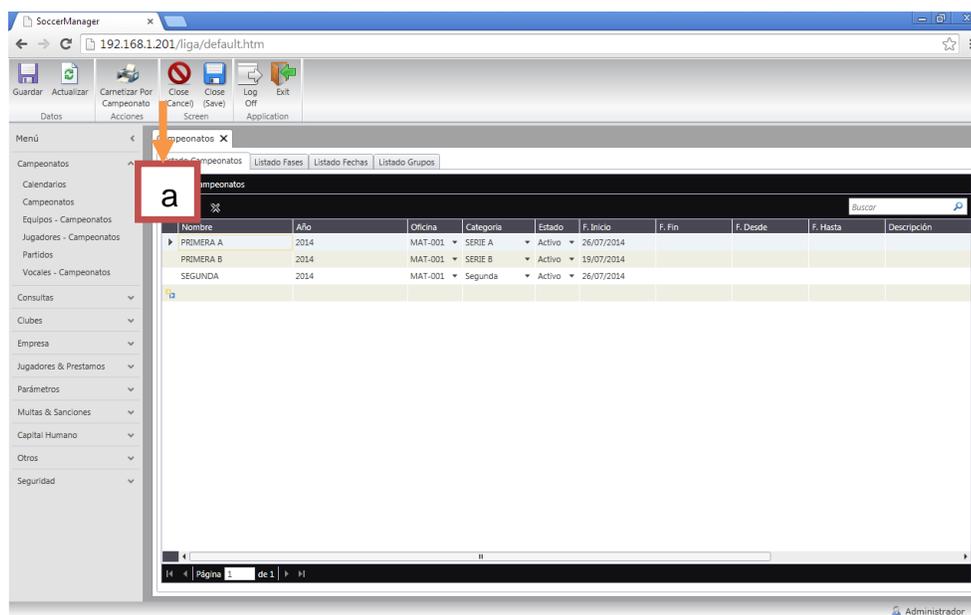
La pantalla de información geográfica o localizaciones es **Geografía** se encuentra dentro de la categoría **Empresa**. En esta pantalla podemos ingresar un listado de países, provincias,

ciudades, parroquias y un listado de estadios. Todas estas pestañas están relacionadas jerárquicamente entre sí.

Pantallas de categorías y campeonatos.

Estas dos pantallas nos permiten la administración de las diferentes categorías que puede tener la institución así como también los campeonatos que se organizan para las diferentes categorías cada año. La pantalla para ingresar o modificar información referente a las categorías la encontramos dentro del menú en la opción **Empresa**, y escogemos **Categorías**.

Una vez ingresada la información de las categorías, podemos crear los campeonatos y asociarlos a las categorías antes creadas. También podemos crear las fases, fechas y grupos que va a tener un campeonato. Es obligatorio crear una fase y un grupo por campeonato. La pantalla para hacer esto es **Campeonatos** que está dentro del menú en la opción **Campeonatos**.



Pantalla 8: Mantenimiento de campeonatos.

Fuente: Sistema SOCCER MANAGER.

a) Carnetizar por campeonato: Genera un reporte de todos los jugadores **Habilitados** que se encuentre participando en el campeonato seleccionado.

Cuando una fase se haya terminado debemos quitarle el visto de la casilla **vigente**, esto pondrá en un estado **cerrado** y desmarcara la casilla **vigente** de todo lo que tenga que ver con esa fase, es decir equipos, jugadores, grupos, fechas y calendarios. Al guardar los datos mostrar un mensaje de confirmación.

Si usted desea terminar un campeonato lo podrá hacer siempre y cuando la fecha finalización del campeonato sea menor a la fecha actual. Para finalizar deberá cambiar el estado ha **cerrado**. Antes de guardar los datos aparecerá un mensaje de confirmación, tenga en cuenta que al terminar un campeonato también se terminara todo lo relacionado con el mismo. Similar al cerrar una fase.

Pantallas de procesos de habilitación de equipos y jugadores.

Para empezar con el proceso de habilitación de equipos, tenemos que ingresar la información del club ya que un club puede tener varios equipos en las diferentes categorías. La información del club la ingresamos en la pantalla **Clubes** que está ubicada dentro del menú en la opción **Clubes**.

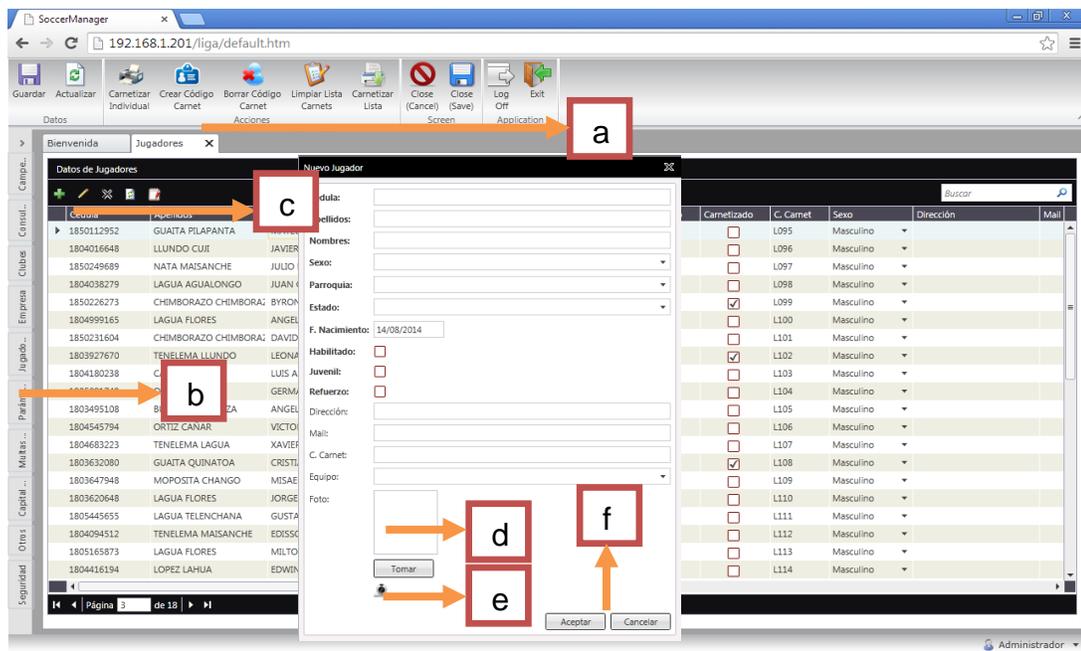
Una vez lleno los datos más importantes de los clubes procedemos a ingresar los equipos, para ello procedemos de la siguiente manera. Ingresamos a la pantalla **Equipos** que está ubicada en la opción **Clubes**, se nos desplegara una pantalla en la que tenemos que seleccionar de la lista de valores una categoría, una vez seleccionada la categoría a la cual pertenecerá el equipo procedemos a ingresar la información requerida del equipo.

En esta pantalla también podemos modificar la información, cabe mencionar que el equipo obligatoriamente debe pertenecer a un grupo, es decir la columna **Grupo** debe tener un valor. También hay campos que no pueden ser modificados como los puntos, etc.

| Nombre | Género | Club | Estado | Grupo | Campeonato | Puntos | G. P. |
|-------------------------|-----------|-------------------------|--------|--------|--------------|--------|-------|
| 24 DE MAYO II | Masculino | 24 DE MAYO II | Activo | FASE 1 | SEGUNDA 2014 | 0 | 0 |
| ALIANZA JUVENIL II | Masculino | ALIANZA JUVENIL | Activo | FASE 1 | SEGUNDA 2014 | 0 | 0 |
| ATLETICO ALTAMIRA | Masculino | ATLETICO ALTAMIRA | Activo | FASE 1 | SEGUNDA 2014 | 0 | 0 |
| ATLETICO ALTAMIRA II | Masculino | ATLETICO ALTAMIRA II | Activo | FASE 1 | SEGUNDA 2014 | 0 | 0 |
| ATLETICO ESTUDIANTIL II | Masculino | ATLETICO ESTUDIANTIL II | Activo | FASE 1 | SEGUNDA 2014 | 0 | 0 |
| ATLETICO MARTINEZ II | Masculino | ATLETICO MARTINEZ | Activo | FASE 1 | SEGUNDA 2014 | 0 | 0 |
| BORUSSIA DORMUNT | Masculino | BORUSSIA DORTMUND II | Activo | FASE 1 | SEGUNDA 2014 | 0 | 0 |
| BUENOS MUCHACHOS II | Masculino | BUENOS MUCHACHOS | Activo | FASE 1 | SEGUNDA 2014 | 0 | 0 |
| CHELSEA II | Masculino | CHELSEA | Activo | FASE 1 | SEGUNDA 2014 | 0 | 0 |
| CRUZEIRO A II | Masculino | CRUZEIRO A | Activo | FASE 1 | SEGUNDA 2014 | 0 | 0 |
| CRUZEIRO B II | Masculino | CRUZEIRO B | Activo | FASE 1 | SEGUNDA 2014 | 0 | 0 |
| CRYSTAL PALACE II | Masculino | CRYSTAL PALACE II | Activo | FASE 1 | SEGUNDA 2014 | 0 | 0 |
| EMPREENDEDORES FC II | Masculino | EMPREENDEDORES F.C. II | Activo | FASE 1 | SEGUNDA 2014 | 0 | 0 |
| ESTUDIANTES II | Masculino | ESTUDIANTES | Activo | FASE 1 | SEGUNDA 2014 | 0 | 0 |
| FIORENTINA A II | Masculino | FIORENTINA A | Activo | FASE 1 | SEGUNDA 2014 | 0 | 0 |
| FIORENTINA B II | Masculino | FIORENTINA B | Activo | FASE 1 | SEGUNDA 2014 | 0 | 0 |
| FUTBOL CLUB TIGRES II | Masculino | FUTBOL CLUB TIGRES | Activo | FASE 1 | SEGUNDA 2014 | 0 | 0 |
| INDEPENDIENTE II | Masculino | INDEPENDIENTE | Activo | FASE 1 | SEGUNDA 2014 | 0 | 0 |

Pantalla 9: Mantenimiento de Equipos.
Fuente: Sistema SOCCER MANAGER.

Para el ingreso de jugadores lo hacemos en la pantalla **Jugadores** ubicada en la opción **Clubes** del menú. En esta pantalla podemos ingresar y modificar los datos de los jugadores. Es necesario que el jugador tenga asignado un equipo. Es decir hay que seleccionar un equipo en la columna del mismo nombre.



Pantalla 10: Mantenimiento de Jugadores.
Fuente: Sistema SOCCER MANAGER.

a) **Botones de acciones:** Estos botones permiten realizar varias acciones sobre los datos de los jugadores.

- ✓ **Carnetizar Individual:** Genera un reporte de todos los jugadores **Habilitados** que tengan marcada la columna **Carnetizado**, la cual haya sido marcada entre la fecha y hora establecida en el parámetro **CarnetIndividual** y la fecha y hora en la que se generó el reporte.
- ✓ **Crear Código Carnet:** Crea un código para el jugador en la columna **C. Carnet** este código es generado con el parámetro **CodigoCarnet** + número secuencial. (Ejm. L045). Si este parámetro no existe solo se genera con el número secuencial.
- ✓ **Borrar Código Carnet:** Borra los códigos de los jugadores, esta acción y la anterior se lo hace por cada página de datos.
- ✓ **Limpiar Lista Carnets:** Quita el marcado de la columna **C. Carnet** de todos los jugadores existentes.

✓ **Carnetizar Lista:** Genera un reporte de todos los jugadores **Habilitados** que tengan marcada la columna **Carnetizado**, sin importar la fecha en la esta fue marcada.

b) **Grilla de datos:** Contiene datos de los jugadores, contienen algunos campos de son modificados automáticamente y estos no pueden ser modificados por el usuario.

c) **Ventana Modal:** Permite ingresar o modificar datos de los jugadores, esta pantalla se lo utiliza para cargar la foto o tomar una foto a través de una cámara web.

d) **Componente para imagen:** Permite cargar la fotografía del jugador. El archivo debe estar en el computador o en algún medio de almacenamiento.

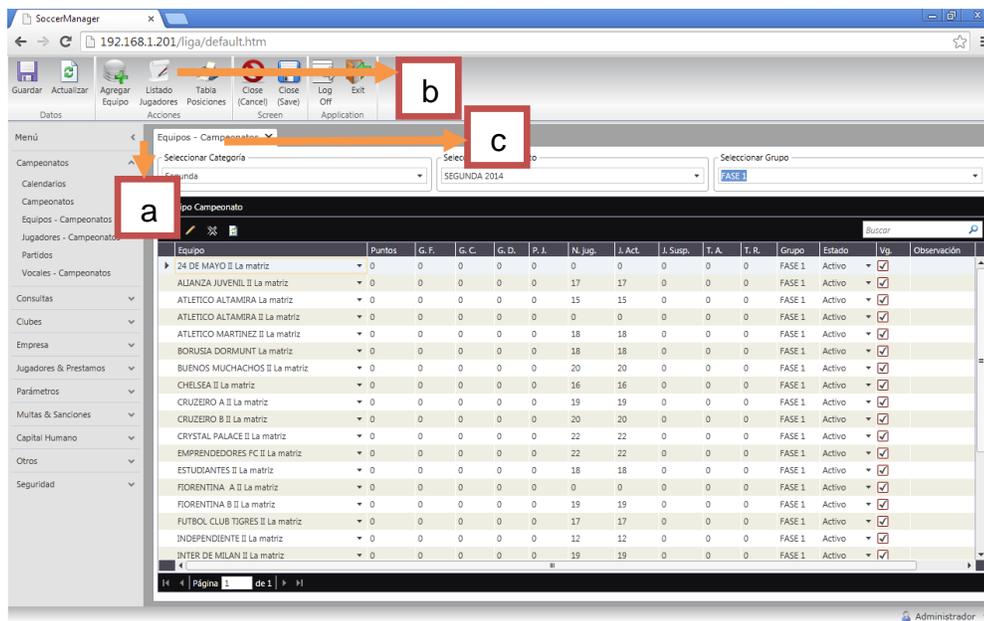
e) **Componente para cámara web:** Permite tomar una fotografía al jugador a través de una cámara web. Este componente funcionará si el equipo del usuario tiene instalado una cámara.

f) **Botones:** El botón **Aceptar** guarda en la grilla el nuevo dato ingresado o modificado y el botón **Cancelar** como su nombre lo indica cancela la operación actual.

Pantallas de procesos de habilitación de equipos y jugadores en un campeonato.

Cuando se hayan ingresado los equipos y los jugadores y estos están habilitados, esto no significa que los equipos ni los jugadores están participando en un campeonato. Es decir aun no podemos carnetizarlos por ningún método mencionado anteriormente.

Para habilitar los equipos en un campeonato lo hacemos en la pantalla **Equipos – Campeonatos** la cual se encuentra en la opción **Campeonatos**. En esta pantalla debemos seleccionar de las listas de valores una categoría, un campeonato y un grupo al cual pertenecerán los equipos. Este proceso de debe hacer para cada campeonato y grupo.



Pantalla 11: Mantenimiento de Equipos en Campeonatos.

Fuente: Sistema SOCCER MANAGER.

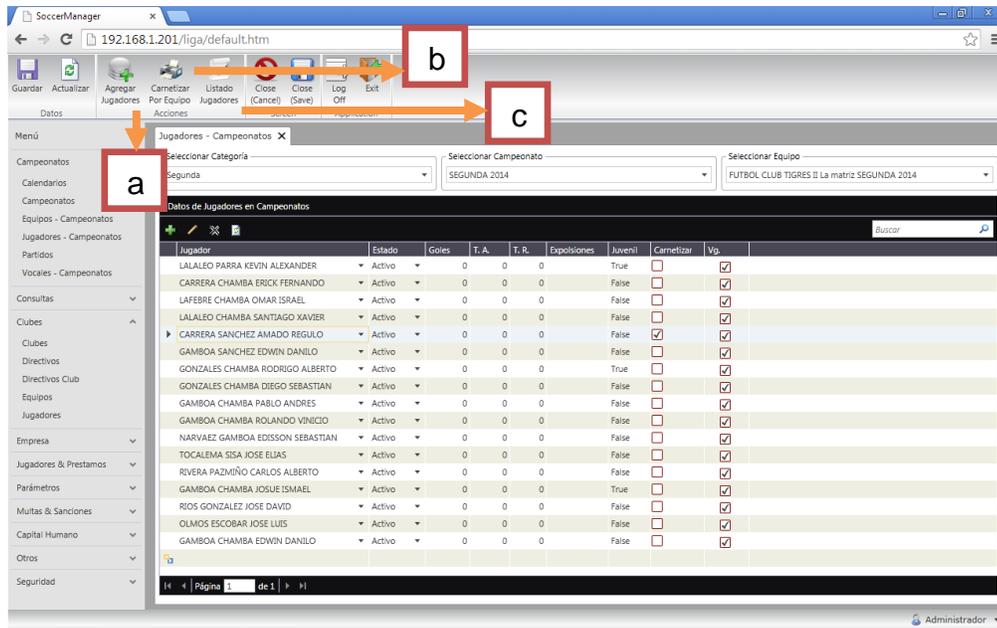
a) Agregar equipo: Permite agregar los equipos de la pantalla **Equipos**, que pertenezcan a la categoría y grupo seleccionados, al campeonato seleccionado en la lista. Esto también se lo puede realizar de forma manual agregando uno por uno los equipos. Lo que hace este botón es agregar a todos los equipos o a los equipos faltantes a un campeonato.

b) Listado jugadores: Genera un reporte con el listado de jugadores que están participando en un campeonato agrupado por equipos.

c) Tabla posiciones: Genera un reporte con la tabla de posiciones, este reporte se genera agrupado por todos los **Grupos** vigentes que tenga el campeonato seleccionado.

Cuando se hayan agregado los equipos en un campeonato, podemos agregar los jugadores a esos equipos que ya están inscritos en un campeonato. Lo hacemos en la pantalla **Jugadores – Campeonato**, que se encuentra en la opción **Campeonatos**. Esta pantalla

contiene información del listado de jugadores que presento un equipo para un campeonato específico.



Pantalla 12: Mantenimiento de Jugadores en Campeonatos.

Fuente: Sistema SOCCER MANAGER.

a) Agregar jugadores: Agrega los jugadores que en la pantalla **Jugadores** tengan asignado el mismo equipo que el de la lista de valores de esta pantalla. Para esto se deben seleccionar valores en las listas.

b) Carnetizar Por Equipo: Genera un reporte con los carnets de todos los jugadores que estén habilitados y pertenezcan al equipo seleccionado en la lista. Es decir de los jugadores que se ve en la grilla de datos.

c) Listado Jugadores: Genera un reporte con el listado de jugadores que están en el campeonato y equipo seleccionados en la lista.

Estas pantallas permiten manipular la información de equipos y jugadores que estén participando en un campeonato y que los mismos estén en un estado activo y vigente. Si

usted desea consultar los históricos de las particiones de equipos y jugadores en campeonato lo puede hacer por la opción **Consultas** del menú.

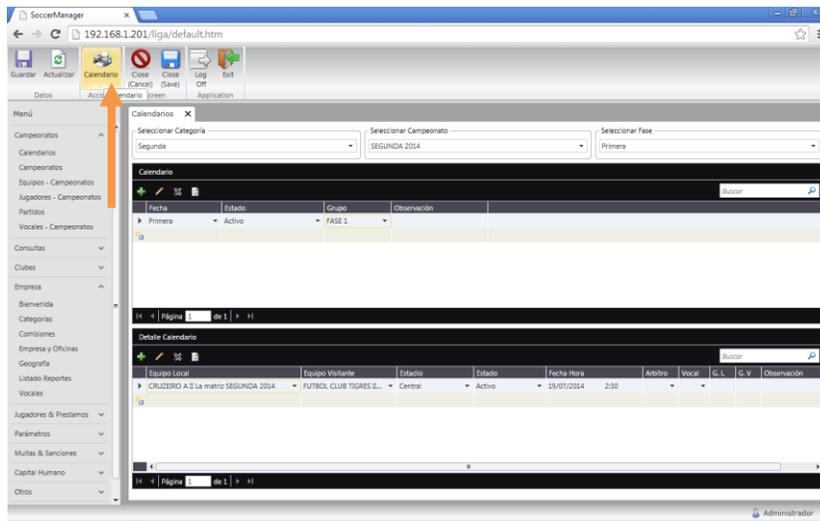
Proceso de carnetización.

La carnetización de los jugadores se lo puede realizar de cuatro formas en diferentes pantallas, que ya se explicó anteriormente. Para esto se debe poblar los datos y seguir los pasos anteriores y también se debe obligatoriamente crear una comisión esto se lo puede hacer en la opción **Empresa** del menú en la pantalla del mismo nombre.

- ✓ Carnetizar Individual → Pantalla Jugadores.
- ✓ Carnetizar Lista → Pantalla Jugadores.
- ✓ Carnetizar Por Equipo → Pantalla Jugadores – Campeonatos.
- ✓ Carnetizar Por Campeonato → Pantalla Campeonato.

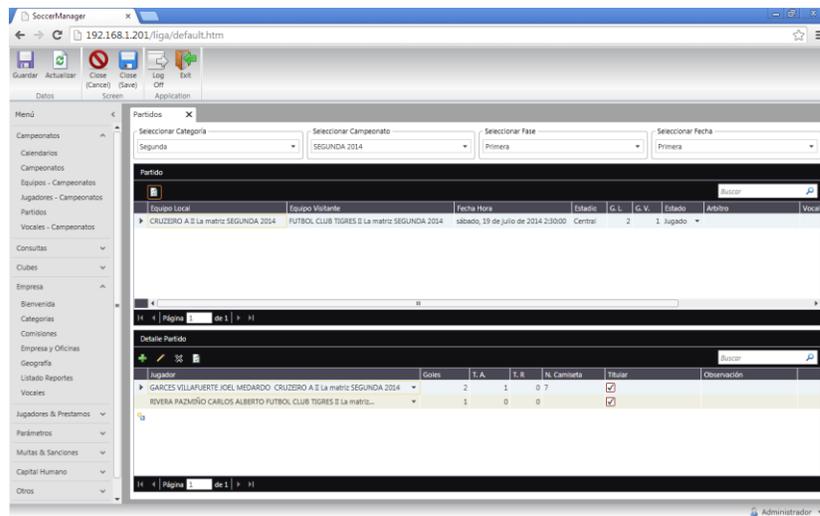
Pantallas de administración de calendarios y partidos.

La pantalla **Calendarios** que se encuentra en la opción **Campeonatos**, permite gestionar los calendarios. Para ellos debemos seleccionar datos en las listas de valores, y agregar o modificar datos tanto de las cabeceras de los calendarios como del detalle de los mismos. Para generar el reporte los hacemos dando clic en el botón **Calendario**.



Pantalla 13: Mantenimiento de Calendarios.
Fuente: Sistema SOCCER MANAGER.

La pantalla **Partidos** que se encuentra en la opción **Campeonatos**, permite gestionar los partidos y sus incidencias. Para ello se deben seleccionar datos de las listas de valores, en el detalle del partido podemos registrar goles con los cuales se suman puntos a los equipos y se suman los goles que tiene un jugador en un campeonato, así como también se registran las tarjetas tanto amarillas como rojas, con las cuales se generan multas o sanciones en dependencia de los parámetros establecidos.

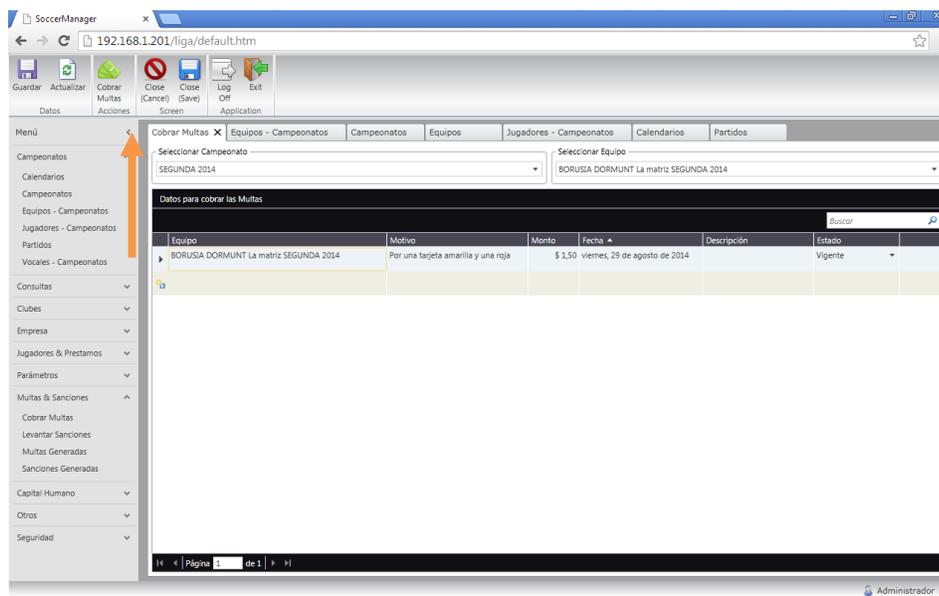


Pantalla 14: Mantenimiento de Partidos.
Fuente: Sistema SOCCER MANAGER.

Pantallas de procesos de multas y sanciones.

Para ver las multas y sanciones generadas automáticamente por el sistema al ingresar las actas de los partidos, así como también para cobrar las multas y levantar las sanciones lo hacemos por la opción **Multas & Sanciones** del menú. Las pantallas **Multas Generadas** y **Sanciones Generadas** son vistas en las cuales podemos consultar.

Si usted desea cobrar las multas lo puede hacer a través de la pantalla **Cobrar Multas**, deberá seleccionar la categoría y el equipo de las listas de valores, luego deberá seleccionar la(s) multa(s) que desea cobrar y cambiarlas a un estado de canceladas.

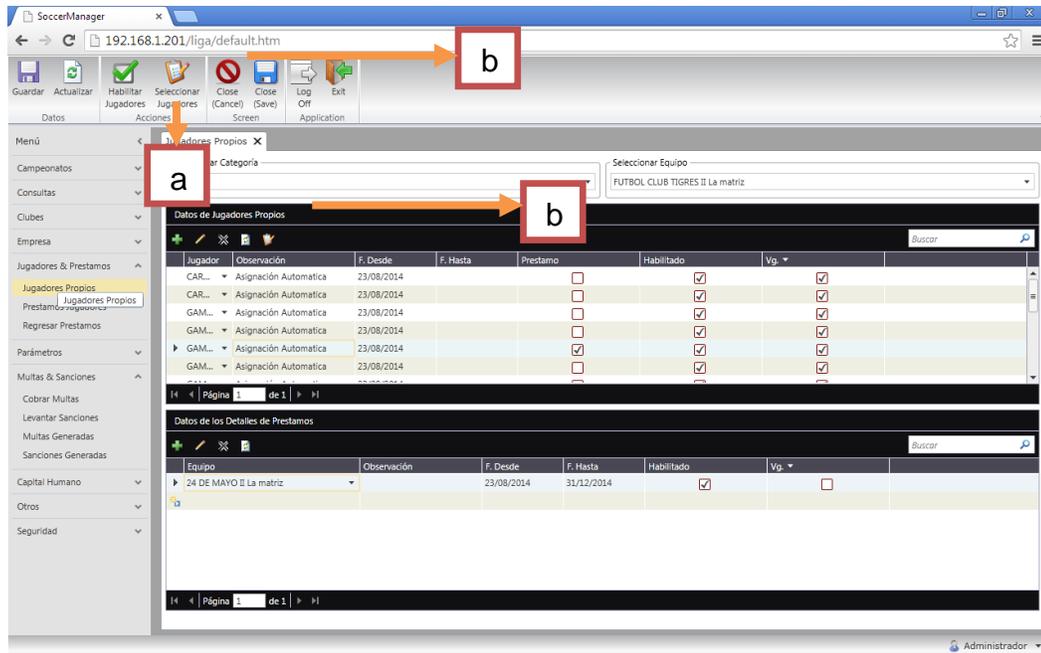


Pantalla 15: Cobro de multas.
Fuente: Sistema SOCCER MANAGER.

Las sanciones las podemos levantar en la pantalla **Levantar Sanciones** haciendo clic en el botón **Procesar** que se encuentra en la parte superior izquierda de la pantalla, estas se levantarán siempre y cuando cumplan las condiciones establecidas en los parámetros, en cuanto al número de partidos que un jugador fue sancionado.

Pantallas de pases de jugadores.

Estas pantallas permiten gestión los pases de los jugadores, cuales son los equipos propietarios de sus derechos deportivos y en que equipos se encuentran jugando a modo de préstamo. Esto lo podemos hacer en las pantallas que se encuentran en la opción **Jugadores & Prestamos** del menú.



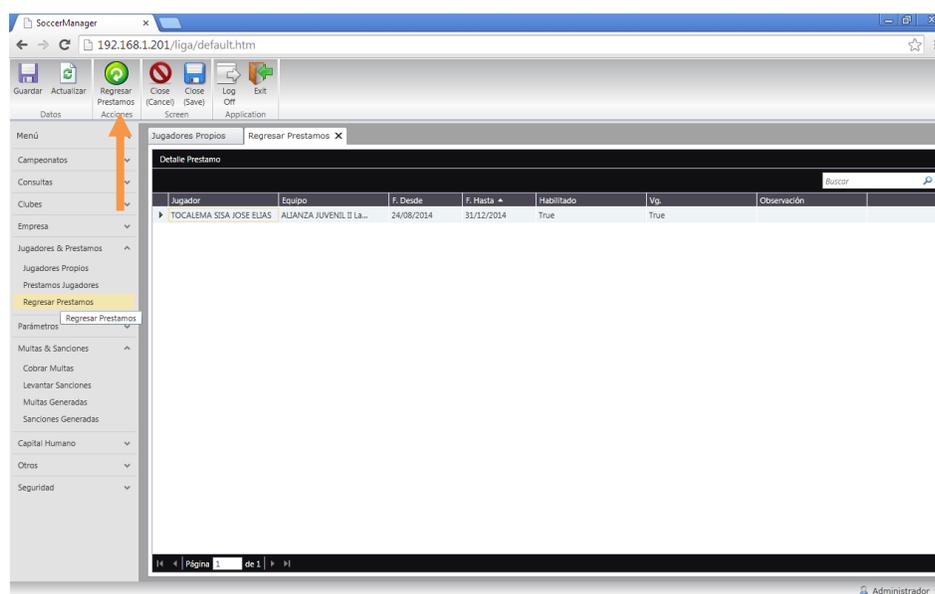
Pantalla 16: Jugadores Propios y Pases.
Fuente: Sistema SOCCER MANAGER.

a) Habilitar Jugadores: Este botón pone en un estado de habilitado y asigna un equipo al jugador sea este propio o a préstamo, para hacer eso se basa en las fechas desde y hasta tanto de jugadores propio como de prestamos y en las casillas habilitado, vigente y préstamo.

b) Seleccionar jugadores: Obtiene una lista de jugadores que actualmente están jugando en el equipo seleccionado en la lista de valores, de los cuales se puede seleccionar quienes son jugadores propios.

Si se desea generar un préstamo a un equipo que no está en la lista de valores se lo debe de hacer por la pantalla **Prestamos Jugadores**.

Para regresar los préstamos de los jugadores lo hacemos a través de la pantalla **Regresar Prestamos** dando clic en el botón **Regresar** esto permitirá regresar todos los préstamos siempre y cuando las fechas de los préstamos se hayan cumplido.



Pantalla 17: Prestamos activos de Jugadores.
Fuente: Sistema SOCCER MANAGER.

En el sistema también se puede ingresar información de Directivos de la Institución o de los clubes, periodos de los directivos, vocales, árbitros. Para ingresar estos datos se lo puede hacer en la opciones **Empresa, Clubes, Campeonato** y árbitros como son externos a la institución se lo puede hacer a través de la opción **Otros**. El proceso de ingreso y asignación de directivos de clubes y vocales se hará de forma similar a la asignación de clubes a un campeonato, seguir los mismos pasos.

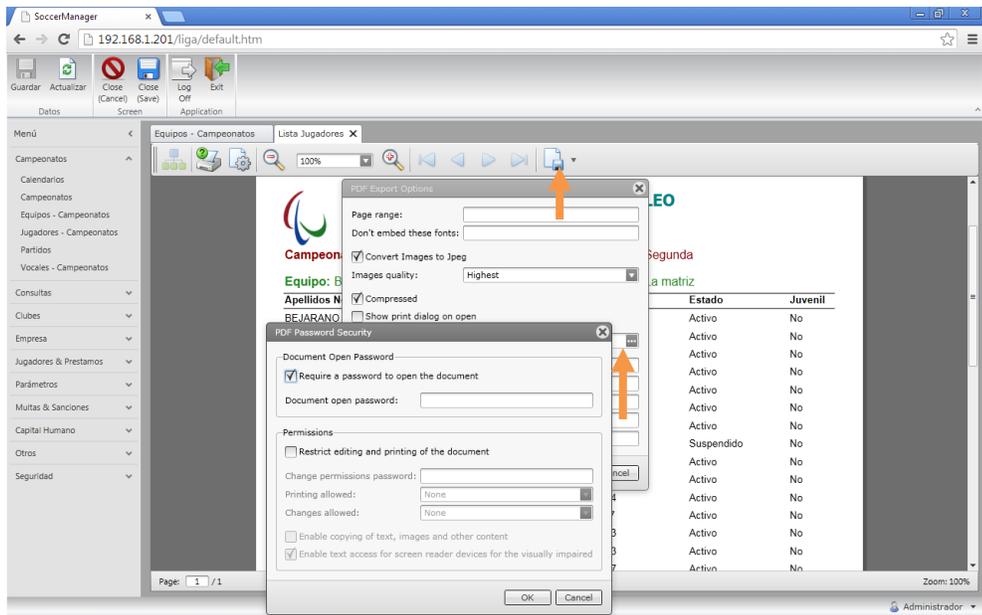
Generar reportes.

Para poder generar cualquier reporte en el sistema debemos ingresar la siguiente información a través de la pantalla **Listado Reportes** que se encuentra en la opción **Empresa** del menú. Los reportes que puede generar el sistema son los siguientes.

| Nombre | Descripción | Opciones | Ubicación (Pantalla) |
|-----------------------------|---|---|-----------------------|
| Carnets | Permite generar los carnets de los jugadores, existen dos modelos de carnets. | Carnetizar Individual | Jugadores |
| | | Carnetizar Lista | |
| | | Carnetizar Por Equipo | Jugadores-Campeonatos |
| | | Carnetizar por Campeonato | Campeonato |
| Tabla de Posiciones | Genera la tabla de posiciones de los equipos que estén vigentes en un campeonato | Por cada campeonato se genera de todos los grupos vigentes. | Equipos-Campeonato |
| | | | Equipo-Campeonato |
| Tabla de Goleadores | Genera la tabla de goleadores de los jugadores que estén vigentes en un campeonato. | Por campeonato | Tabla Goleadores |
| Calendarios | Listado del calendario con su respectivo detalle | Un calendario de toda la fecha. | Calendarios |
| Listado de Jugadores | Listado de jugadores que tiene un equipo en un campeonato | Jugadores en un equipo | Equipos-Campeonato |
| | | Jugadores de una fase | Jugadores-Campeonatos |

Matriz 2: Opciones de carnetización de jugadores.
Fuente: Sistema SOCCER MANAGER.

Todas las pantallas de reportes permiten exportar los mismos en diferentes formatos, los cuales pueden ser almacenados en cualquier dispositivo de almacenamiento. Los archivos de tipo PDF pueden ser exportados con una clave de seguridad.



Pantalla 18: Reportes del sistema.

Fuente: Sistema SOCCER MANAGER.

Anexo 4: Manual Técnico

En el presente manual se detallan las configuraciones que se deben hacer para implementar la aplicación así como también se incluye el SQL de las vistas a través de las cuales la aplicación genera los reportes, esto con el objetivo de proporcionar una guía al administrador de la aplicación.

Para configurar la aplicación se deben seguir los siguientes pasos. Los cuales se describen de forma global y general.

1. Instalar el Internet Information Server siguiendo los pasos correspondientes.
2. Instalar Framework 4.0.
3. Registrar el Framework 4.0, para poder hacer esto debemos ir al directorio C:\Windows\Microsoft.NET\Framework64 o C:\Windows\Microsoft.NET\Framework en dependencia del sistema operativo. Y ejecutar como administrador en una terminal **aspnet_regiis -i**, como esto registramos el framework.
4. Instalar una base de datos SqlServer Express 2008 puede ser la siguiente versión SQLEXPRESS_x86_ESN_2008.
5. Instalar el administrador de gráfico de la base de datos SQLManagementStudio_x86_ESN.
6. Habilitar los protocolos TCP/IP de la base de datos, se lo puede hacer con la herramienta Sql Server Configuration Manager, en la opción Configuración de SQL Native Client => Protocolos del cliente=> TCP/IP debe estar en modo habilitado si no lo está hay que habilitarlo.

7. Adjuntar la base de datos o el respaldo de la base de datos.
8. Modificar las cadenas de conexión del archivo de configuración de la aplicación web.config, en este archivo solo se cambiaran los valores marcados, el primero es el nombre del equipo donde se haya instalado la base de datos y el segundo valor a modificar es la contraseña con la cual hayamos instado la base de datos.

```
<connectionStrings>
  <add name=" IntrinsicData" connectionString="Data
Source=SRVVISUAL\SQLEXPRESS;Initial Catalog=liga_pelileo;Persist
Security Info=True;User ID=sa;Password=liga" />
  <add name="742c7313-e149-4500-b667-2a69087cfe51"
connectionString="" />
  <add
name="DevExpress.XtraReports.LightSwitch.Server.XtraReportsServi
ce" connectionString="" />
  <add name="ConsultasData" connectionString="Data
Source=SRVVISUAL\SQLEXPRESS;Initial Catalog=liga_pelileo;Persist
Security Info=True;User ID=sa;Password=liga" />
</connectionStrings>
```

9. Copiar el proyecto en el siguiente directorio C:\inetpub\wwwroot\.
10. Convertir el proyecto que hemos copiado en una aplicación esto a través del administrador del IIS.
11. Instalar el Plugin de Silverlight en el servidor y en los clientes para que puedan acceder a la aplicación.

Listado de tablas que contiene la aplicación.

Estas son las tablas que tiene el sistema en las cuales no se incluye las tablas del esquema de seguridades que maneja la aplicación ya que este es un esquema propio de asp.net para Sql Server. Por tal motivo solo se enlistan las que se crearon en el proyecto.

Módulo General

- | | | |
|----------------|----------------|----------------|
| ✓ GEN_CARGOS | ✓ GEN_ESTACIVI | ✓ GEN_PARROQUI |
| ✓ GEN_CIUDADES | ✓ GEN_ESTADOS | ✓ GEN_PERIODOS |
| ✓ GEN_DEPARTAM | ✓ GEN_MODULOS | ✓ GEN_PROVINCI |
| ✓ GEN_DIRELIGA | ✓ GEN_OFICINAS | ✓ GEN_REPORTES |
| ✓ GEN_DIREOFIC | ✓ GEN_OPERADOR | ✓ GEN_TELEFONO |
| ✓ GEN_EMPRESAS | ✓ GEN_PAISES | ✓ GEN_TIPOCONT |
| ✓ GEN_ERRORES | ✓ GEN_PARAMETR | ✓ GEN_TIPOPARA |

Módulo Ligas.

- | | | |
|----------------|----------------|----------------|
| ✓ LIG_ARBITRO | ✓ LIG_DIOFCOMI | ✓ LIG_JUGADORE |
| ✓ LIG_CALEDARI | ✓ LIG_DIRECLUB | ✓ LIG_JUGAEQCA |
| ✓ LIG_CAMPEONA | ✓ LIG_DIRECTIV | ✓ LIG_JUGAPROP |
| ✓ LIG_CATEGORI | ✓ LIG_EQUICAMP | ✓ LIG_SANCIONE |
| ✓ LIG_CLUBES | ✓ LIG_EQUIPOS | ✓ LIG_TARJETAS |
| ✓ LIG_COMISION | ✓ LIG_ESTADIOS | ✓ LIG_TIPOSANC |
| ✓ LIG_DETACALE | ✓ LIG_FASES | ✓ LIG_VOCACAMP |
| ✓ LIG_DETAPART | ✓ LIG_FECHAS | ✓ LIG_VOCAL |
| ✓ LIG_DETPREJU | ✓ LIG_GRUPOS | |

Modulo Recursos Humanos.

- ✓ RHH_EMPLCARG
- ✓ RHH_EMPLEADO

Vistas del sistema utilizadas para generar reportes.

Las siguientes vistas son utilizadas para la generación de reportes, a continuación tenemos el respectivo SQL de cada una de ellas.

1. Vista para la generación de calendarios.

```
CREATE VIEW vw_lig_Calendario as
SELECT    equipoLocal.NOMB AS NombreEquipoLocal, parroquiaLocal.NOMB AS
NombreParroquiaLocal, equipoVisitante.NOMB AS NombreEquipoVisitante,
        parroquiaVisitante.NOMB AS NombreParroquiaVisitante, estadio.NOMB
AS NombreEstadio, parroquia.NOMB AS NombreParroquia, categoria.Id AS IdCategoria,
        categoria.NOMB AS NombreCategoria, campeonato.Id AS IdCampeonato,
campeonato.NOMB + '' + campeonato.ANIO AS NombreCampeonato, fase.Id AS IdFase,
        fase.NOMB AS NombreFase, fecha.Id AS IdFecha, fecha.NOMB AS
NombreFecha, grupo.Id AS IdGrupo, grupo.NOMB AS NombreGrupo,
        estado.NOMB AS NombreEstado, detalleCalendario.FECHOR AS
FechaHora, empresa.NOMCOM AS NombreEmpresa, empresa.LOGO AS LogoEmpresa
FROM      dbo.LIG_DETACALE AS detalleCalendario INNER JOIN
        dbo.LIG_CALEDARI AS calendario ON calendario.Id =
detalleCalendario.Calendario_DetalleCalendario INNER JOIN
        dbo.LIG_CAMPEONA AS campeonato ON campeonato.Id =
calendario.Campeonato_Calendario INNER JOIN
        dbo.LIG_FASES AS fase ON fase.Id = calendario.Fase_Calendario INNER
JOIN
        dbo.LIG_FECHAS AS fecha ON fecha.Id = calendario.Fecha_Calendario
INNER JOIN
        dbo.LIG_GRUPOS AS grupo ON grupo.Id = calendario.Grupo_Calendario
INNER JOIN
        dbo.LIG_EQUICAMP AS equipoCampeonatoLocal ON
equipoCampeonatoLocal.Id = detalleCalendario.EquipoCampeonato_DetalleCalendario
INNER JOIN
        dbo.LIG_EQUICAMP AS equipoCampeonatoVisitante ON
equipoCampeonatoVisitante.Id =
detalleCalendario.EquipoCampeonato_DetalleCalendario1 INNER JOIN
        dbo.LIG_EQUIPOS AS equipoLocal ON equipoLocal.Id =
equipoCampeonatoLocal.Equipo_EquipoCampeonato INNER JOIN
        dbo.LIG_EQUIPOS AS equipoVisitante ON equipoVisitante.Id =
equipoCampeonatoVisitante.Equipo_EquipoCampeonato INNER JOIN
        dbo.LIG_CLUBES AS clubLocal ON clubLocal.Id =
equipoLocal.Club_Equipo INNER JOIN
        dbo.LIG_CLUBES AS clubVisitante ON clubVisitante.Id =
equipoVisitante.Club_Equipo INNER JOIN
        dbo.GEN_PARROQUI AS parroquiaLocal ON parroquiaLocal.Id =
clubLocal.Parroquia_Club INNER JOIN
        dbo.GEN_PARROQUI AS parroquiaVisitante ON parroquiaVisitante.Id =
clubVisitante.Parroquia_Club INNER JOIN
        dbo.LIG_ESTADIOS AS estadio ON estadio.Id =
detalleCalendario.Estadio_DetalleCalendario INNER JOIN
```

```

        dbo.GEN_PARROQUI    AS    parroquia    ON    parroquia.Id    =
estadio.Parroquia_Estadio INNER JOIN
        dbo.GEN_ESTADOS    AS    estado        ON    estado.Id    =
detalleCalendario.Estado_DetalleCalendario INNER JOIN
        dbo.GEN_OFICINAS    AS    oficina      ON    oficina.Id    =
campeonato.Oficina_Campeonato INNER JOIN
        dbo.GEN_EMPRESAS    AS    empresa     ON    empresa.Id    =
oficina.Empresa_Oficina INNER JOIN
        dbo.LIG_CATEGORI    AS    categoria   ON    categoria.Id    =
campeonato.Categoria_Campeonato
WHERE (oficina.Oficina_Oficina IS NULL)

```

2. Vista para la generación de carnets.

```

CREATE VIEW vw_lig_CarnetGeneral as
SELECT    empresa.NOMCOM AS NombreEmpresa, empresa.LOGO AS LogoEmpresa,
comision.NOMB AS NombreComision, jugador.Id AS IdJugador, jugador.CEDU AS
Cedula,
        jugador.APEL AS Apellidos, jugador.NOMB AS Nombres, jugador.FOTO
AS FotoJugador, jugador.FECNAC AS FechaNacimiento, jugador.NUMCAR AS
NumeroCarnet,
        jugador.FECARN AS FechaCarnetiza, CASE WHEN jugador.HABI = 1
THEN 'Si' ELSE 'No' END AS Habilitado,
        CASE WHEN jugador.CARN = 1 THEN 'Si' ELSE 'No' END AS
Carnetizado, CASE WHEN jugadorEquiCamp.JUVE = 1 THEN 'Si' ELSE 'No' END AS
Juvenil,
        CASE WHEN jugador.REFU = 1 THEN 'Si' ELSE 'No' END AS Refuerzo,
CASE WHEN jugador.SEXO = 'M' THEN 'Masculino' ELSE 'Femenino' END AS Genero,
        equipo.Id AS IdEquipo, equipo.NOMB AS NombreEquipo, equipoCamp.Id
AS IdEquipoCampeonato, categoria.Id AS IdCategoria, categoria.NOMB AS
NombreCategoria,
        campeonato.Id AS IdCampeonato, campeonato.NOMB + ' ' +
campeonato.ANIO AS NombreCampeonato, estado.NOMB AS NombreEstado,
        parroquia.NOMB AS NombreParroquia,
empresa.sello AS Sello
FROM      dbo.LIG_JUGAEQCA AS jugadorEquiCamp INNER JOIN
        dbo.LIG_JUGADORE    AS    jugador    ON    jugador.Id    =
jugadorEquiCamp.Jugador_JugadorEquipoCampeonato INNER JOIN
        dbo.LIG_EQUICAMP    AS    equipoCamp ON    equipoCamp.Id    =
jugadorEquiCamp.EquipoCampeonato_JugadorEquipoCampeonato INNER JOIN
        dbo.LIG_EQUIPOS    AS    equipo      ON    equipo.Id    =
equipoCamp.Equipo_EquipoCampeonato INNER JOIN
        dbo.LIG_CAMPEONA    AS    campeonato ON    campeonato.Id    =
jugadorEquiCamp.Campeonato_JugadorEquipoCampeonato INNER JOIN

```

```

        dbo.LIG_CATEGORI AS categoria ON categoria.Id =
equipo.Categoria_Equipo INNER JOIN
        dbo.GEN_ESTADOS AS estado ON estado.Id =
jugadorEquiCamp.Estado_JugadorEquipoCampeonato INNER JOIN
        dbo.GEN_OFICINAS AS oficina ON oficina.Id =
campeonato.Oficina_Campeonato INNER JOIN
        dbo.GEN_EMPRESAS AS empresa ON empresa.Id =
oficina.Empresa_Oficina INNER JOIN
        dbo.LIG_COMISION AS comision ON oficina.Id =
comision.Oficina_Comision INNER JOIN
        dbo.LIG_CLUBES AS club ON club.Id = equipo.Club_Equipo INNER JOIN
        dbo.GEN_PARROQUI AS parroquia ON parroquia.Id = club.Parroquia_Club
WHERE (estado.NOMB = 'Activo') AND (jugadorEquiCamp.VIGE = 1) AND
(oficina.Oficina_Oficina IS NULL) AND (comision.DSCR = 'Carnet')

```

3. Vista para la generación de la tabla de goleadores.

```

CREATE VIEW vw_lig_TablaGoleadores as
SELECT jugador.CEDU AS CedulaJugador, jugador.APEL + ' ' + jugador.NOMB AS
NombreJugador, jugador.GOLE AS Goles, equipo.NOMB AS NombreEquipo,
        parroquia.NOMB AS NombreParroquia, campeonato.Id AS IdCampeonato,
campeonato.NOMB + ' ' + campeonato.ANIO AS NombreCampeonato,
        categoria.Id AS IdCategoria, categoria.NOMB AS NombreCategoria,
estado.NOMB AS NombreEstado, empresa.NOMCOM AS NombreEmpresa,
        empresa.LOGO AS LogoEmpresa, CASE WHEN jugador.JUVE = 1 THEN
'Si' ELSE 'No' END AS Juvenil, equipo.Id AS IdEquipoCampeonato
FROM    dbo.LIG_JUGADORE AS jugador INNER JOIN
        dbo.LIG_EQUIPOS AS equipo ON equipo.Id = jugador.Equipo_Jugador
INNER JOIN
        dbo.LIG_CAMPEONA AS campeonato ON campeonato.Id =
equipo.Campeonato_Equipo INNER JOIN
        dbo.LIG_CATEGORI AS categoria ON categoria.Id =
campeonato.Categoria_Campeonato INNER JOIN
        dbo.LIG_CLUBES AS club ON club.Id = equipo.Club_Equipo INNER JOIN
        dbo.GEN_PARROQUI AS parroquia ON parroquia.Id = club.Parroquia_Club
INNER JOIN
        dbo.GEN_OFICINAS AS oficina ON oficina.Id =
campeonato.Oficina_Campeonato INNER JOIN
        dbo.GEN_EMPRESAS AS empresa ON empresa.Id =
oficina.Empresa_Oficina INNER JOIN
        dbo.GEN_ESTADOS AS estado ON estado.Id = jugador.Estado_Jugador
WHERE (oficina.Oficina_Oficina IS NULL) AND (campeonato.Estado_Campeonato =
        (SELECT Id
        FROM    dbo.GEN_ESTADOS AS est

```

WHERE (NOMB = 'Activo'))

4. Vista para la generación de la tabla de posiciones.

```

CREATE VIEW vw_lig_TablaPosicionesGeneral as
SELECT      equipoCampeonato.PUNT AS Punto, equipoCampeonato.GOLAFA AS
GolAFavor, equipoCampeonato.GOLENC AS GolEnContra,
           equipoCampeonato.GOLAFA - equipo.GOLENC AS GolDiferencia,
equipoCampeonato.PARJUG AS PartidosJugador,
           CASE WHEN equipoCampeonato.[VIGE] = 1 THEN 'S' ELSE 'N' END AS
VgEquipoCampeonato, campeonato.Id AS IdCampeonato,
           campeonato.NOMB + ' ' + campeonato.ANIO AS NombreCampeonato,
campeonato.ANIO AS AnioCampeonato, equipo.NOMB AS NombreEquipo,
           estado.NOMB AS NombreEstado, grupo.Id AS IdGrupo, grupo.NOMB AS
NombreGrupo, parroquia.NOMB AS NombreParroquia, categoria.Id AS IdCategoria,
           categoria.NOMB AS NombreCategoria, empresa.NOMCOM AS
NombreEmpresa, empresa.LOGO AS LogoEmpresa, oficina.NOMCOR AS
NombreOficina
FROM      dbo.LIG_EQUICAMP AS equipoCampeonato INNER JOIN
           dbo.LIG_CAMPEONA AS campeonato ON campeonato.Id =
equipoCampeonato.Campeonato_EquipoCampeonato INNER JOIN
           dbo.LIG_EQUIPOS AS equipo ON equipo.Id =
equipoCampeonato.Equipo_EquipoCampeonato INNER JOIN
           dbo.GEN_ESTADOS AS estado ON estado.Id =
equipoCampeonato.Estado_EquipoCampeonato INNER JOIN
           dbo.LIG_GRUPOS AS grupo ON grupo.Id =
equipoCampeonato.Grupo_EquipoCampeonato INNER JOIN
           dbo.LIG_CLUBES AS club ON club.Id = equipo.Club_Equipo INNER JOIN
           dbo.GEN_PARROQUI AS parroquia ON parroquia.Id = club.Parroquia_Club
INNER JOIN
           dbo.LIG_CATEGORI AS categoria ON categoria.Id =
campeonato.Categoria_Campeonato INNER JOIN
           dbo.GEN_OFICINAS AS oficina ON oficina.Id =
campeonato.Oficina_Campeonato INNER JOIN
           dbo.GEN_EMPRESAS AS empresa ON empresa.Id =
oficina.Empresa_Oficina
WHERE     (oficina.Oficina_Oficina IS NULL) AND (grupo.VIGE = 1)

```

Anexo 5: Acta de configuración.

| Acta de configuración de la aplicación SoccerManager | |
|---|--|
| Descripción | Valor |
| Sistema Operativo | Windows 7 |
| Usuario del Sistema Operativo | EST2 |
| Contraseña del Usuario del Sistema Operativo | ***** |
| Base de Datos | Sql Server Express 2008 |
| Versión de Base de datos | SQLEXPR_x86_ESN_2008 |
| Nombre de la Base de datos | liga_pelileo |
| Usuario de la Base de datos | sa |
| Contraseña del usuario de base de datos | ***** |
| Archivos a respaldar de la base de datos | liga_pelileo.dmf liga_pelileo_log.ldf |
| Servidor Web | Internet Information Server 7 |
| URL aplicación | http://192.168.1.5/liga |
| Usuario Administrador de la Aplicación | Admin |
| Contraseña del Usuario de la Aplicación | ***** |
| Directorio de Ubicación de la Aplicación | C:\inetpub\wwwroot\liga |
| Material entregable | Manuales del aplicativo Instaladores del aplicativo |

Pelileo Septiembre 25, 2014

Investigador

Byron Rodrigo Moreno Moreno

C.C. 1803980844