

Universidad de la sierra sur

LEAGUE CONTROL

Bustamante Cortés Iván Guadalupe
Hernandez Ramírez Heber Esaú
Hernandez Vite Jesús Fernando

Control league

Heber Esaú Hernández Ramírez
Iván Guadalupe Bustamante Cortés
Jesús Fernando Hernández Vite

Noviembre 2021

Índice

1. Introducción.	4
2. Propósitos y alcance	5
3. Estudio de viabilidad y factibilidad	5
4. Mapa de navegación	6
5. Interfaces	6
6. Requerimientos funcionales	9
7. Requerimientos no funcionales	14
8. Estándares de codificación	18
8.1. Normas de codificación	18
8.1.1. Generales	18
8.1.2. Nombrar variable y clases	18
8.1.3. Sangría	18
8.1.4. Comentarios	19
8.1.5. Espacios	19
8.1.6. Nombre de interfaces	20
9. Colaboradores	21

1. Introducción.

League control es un software el cual será diseñado para crear y mantener el orden de una liga de futbol, en ella se puede llevar el control de los jugadores o del equipo en general, en este programa se podrá registrar a los jugadores y a los equipos, pero al mismo tiempo se podrá consultar dichos datos. Este programa facilitará a las personas poder conocer los resultados del equipo de su preferencia.

El software está inspirado en las problemáticas vistas en una pequeña liga de futbol de la ciudad de Miahuatlán, este proyecto tiene como fin el facilitar el trabajo a los administradores de la liga. El programa ofrecido aparte de facilitar eliminará errores de las personas, porque ya no será necesario llevar un control a lápiz y papel.

2. Propósitos y alcance

La creación de este proyecto tiene la finalidad de facilitar a las organizaciones encargadas de administrar una liga de futbol, por lo que se busca tener una base de datos en la cual el propietario o presidente de dicha liga de futbol pueda registrar los datos y estadísticas de cada equipo y jugador. Dichas estadísticas son: goles, asistencias, tarjetas amarillas y tarjetas rojas.

Otra característica de este software es que se busca que tanto el presidente de la liga como los capitanes tengan acceso a la información en cualquier momento, esto con el fin de poder saber la posición de algún equipo en la tabla o de algún jugador en la pelea por ser el máximo anotador. Es necesario destacar que dicha información solo podrá ser manipulada por el presidente de la liga o personas autorizadas, por lo que se implementara una interfaz de invitado para toda aquella persona que desee consultar los resultados (los invitados no podrán modificar ningún dato dentro de la tabla).

3. Estudio de viabilidad y factibilidad

El problema planteado sobre facilitar el trabajo a los administradores de una liga de futbol es algo de lo cual muy pocas personas se han preocupado, es por esto mismo que se decidió darle resolución a la problemática, pues al haber en el mercado muy pocos o casi ningún software capaz de llevar un control de una liga, se llegó a la idea de crear un software con las principales necesidades de una liga de futbol.

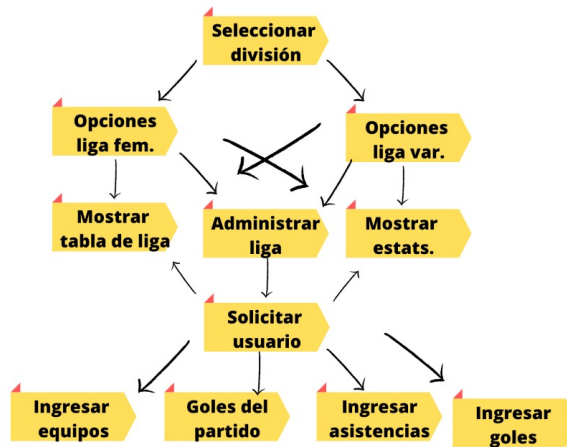
Creemos que el producto ofrecido será capaz de resolver las problemáticas de cualquier liga ya que se podrá reemplazar las libretas y cualquier error humano a la hora de ordenar puntos, con nuestro software ofrecido. Esto ya que no requerirá de más que una simple computadora con conexión parcial a internet.

El proyecto no requiere de un presupuesto económico para llegar a ser elaborado, pues solo requerirá de nuestros conocimientos en el área de la informática para la elaboración del mismo, ya que, no se plantea utilizar ningún software de suscripción o paga, mucho menos la compra de algún hardware o material de elaboración.

El plazo de finalización del producto tiene fecha estimada de finales de enero a principios de febrero, pues se plantea trabajar en el a partir de la fecha actual hasta tener el software en su versión funcional y sin fallas.

4. Mapa de navegación

En la siguiente imagen podemos ver el diagrama de lo qué es el mapa de navegación.



5. Interfaces

La siguiente imagen nos muestra el diseño de la ventana *interfaz* de usuario principal, en ella el usuario seleccionará la división que desee entre femenino y varonil.



La siguiente imagen nos muestra el diseño de la ventana *interfaz* de selección de tabla, en ella se puede acceder a la tabla general, las estadísticas de jugadores y a la administración del torneo en caso de contar con usuario y contraseña.



La siguiente imagen nos muestra el diseño de la ventana *interfaz* de la tabla general, en ella podremos observar la posición que tienen los equipos, los puntos y demás estadísticas de equipo.

The image shows the same 'LEAGUE CONTROL' interface, but with a table displayed on the green field background. The table has 10 columns: Posición, Equipo, PJ, G, E, P, GF, GC, DG, and Puntos. It has 18 rows for team positions, numbered 1 to 18 on the left.

Posición	Equipo	PJ	G	E	P	GF	GC	DG	Puntos
1									
2									
3									
4									
5									
6									
7									
8									
9									
10									
11									
12									
13									
14									
15									
16									
17									
18									

La siguiente imagen nos muestra el diseño de la ventana *interfaz* de las estadísticas de goleo y asistencias por jugadores

The screenshot shows a window titled 'LEAGUE CONTROL' with a soccer ball icon and a close button (X). The main area is a green field with two white tables overlaid. The top table has columns: 'Jugador', 'No. Dorsal', and 'Goles'. The bottom table has columns: 'Jugador', 'No. Dorsal', and 'Asistencias'. Both tables have 5 empty rows for data entry.

La siguiente imagen nos muestra el diseño de la ventana *interfaz* del DBA, en ella el Administrador de Bases de Datos *coordinadordeltorneo* registrará cada una de las estadísticas del torneo

The screenshot shows a window titled 'LEAGUE CONTROL' with a soccer ball icon and a close button (X). The main area is a green field with several gray buttons on the left and a white table on the right. The buttons are: 'Registrar equipos', 'Registro de resultados', 'Registro de tarjetas', 'Registro de goles', 'Registro de asistencias', 'Maximizar tabla gral', and 'Estadísticas de jugadores'. The table has columns: 'Posición', 'Equipo', 'PJ', 'G', 'E', 'P', 'GF', 'GC', 'DG', and 'Puntos'. The table has 18 rows, with the first row containing numbers 1 through 18.

6. Requerimientos funcionales

Después de una fase de análisis del desarrollo de nuestro proyecto, encontramos requerimientos funcionales los cuales nos servirán para realizar un buen proyecto y definir un modelo a seguir.

Los requerimientos funcionales están bajo código en un formato RF#; donde R significa Requisito, F significa Funcional y # es el número asignado.

Código de requerimiento: RF1	
Nombre	Registro de equipo
Propósito	Crea un usuario capaz de registrar el nombre de cada uno de los equipos.
Descripción	Una vez ubicados en la página del inicio abra un botón con el cual se podrá registrar el equipo, en esa interfaz se pedirá el nombre del equipo y número de jugadores.
Entrada	Se le insertara el nombre del equipo y el número de jugadores que lo conforman.
Salida	Un mensaje el cual dirá ¡Su registro ha sido todo un éxito.
Prioridad	Alta

Código de requerimiento: RF2	
Nombre	Registro de resultados
Propósito	Crea una interfaz capaz de registrar los resultados de cada uno de los partidos.
Descripción	Una vez iniciada sesión, se ubicaran en el botón registro de resultados, en esta interfaz se pedirá mencionar el nombre de los equipos que jugaron y el resultado que se obtuvo.
Entrada	Se les insertaran los nombre de los dos equipos los cuales se enfrentan y se insertara el resultado obtenido del partido
Salida	Se le sumaran tres puntos al equipo ganador, en caso de empate se le sumara solo un punto a cada equipo
Prioridad	Alta

Código de requerimiento: RF3	
Nombre	Registro de jugadores
Propósito	Crea una interfaz que será capaz de ordenar por equipos y guardar los nombres de ellos.
Descripción	Una vez iniciada sesión, se ubicarán en el botón insertar jugador y se registrara su nombre, edad y numero de camiseta que portara.
Entrada	Se le insertaran los nombres y apellidos de cada uno de los jugadores así como su edad y numero de camiseta.
Salida	Un mensaje de éxito el cual dirá ¡el jugador a sido registrado con éxito!
Prioridad	Alta

Código de requerimiento: RF4	
Nombre	Estadísticas de jugador
Propósito	Crea una interfaz donde se muestra las estadísticas (goles, faltas, tarjetas de amonestación) de cada uno de los jugadores
Descripción	Una vez iniciada sesión, se ubicarán en el botón de insertar estadísticas. En este se registraran los datos del jugador y de las estadísticas de cada partida.
Entrada	Se le insertara el nombre y apellidos de cada jugador, así como el equipo al que pertenecen, y las estadísticas que se hace en cada partido.
Salida	Se dará un mensaje de éxito ¡felicidades sus estadísticas han sido guardadas con éxito!
Prioridad	Alta

Código de requerimiento: RF5	
Nombre	Estadísticas de equipo
Propósito	Crear una interfaz donde se mostraran las estadísticas por equipo.
Descripción	Una vez iniciada sesión y en la página de inicio se ubicarán en el botón de estadísticas de equipo. Y en este se registrara el nombre del equipo y las estadísticas que tiene.
Entrada	Se le insertara el nombre del equipo, y se registraran las estadísticas cada que se termine un partido.
Salida	Se mandara un mensaje de éxito, el cual nos dirá ¡felicidades sus estadísticas de equipo han sido registrado con éxito!
Prioridad	Alta

7. Requerimientos no funcionales

Los requerimientos no funcionales también surgieron postanálisis y con ayuda de ellos sabremos que cosas no debemos priorizar.

Los requerimientos no funcionales están bajo código en un formato RNF#; donde R significa Requisito, N significa No, F significa Funcional y # es el número asignado.

Código de requerimiento: RNF1	
Nombre	Conexión a internet
Propósito	El propósito es que se pueda tener conexión a los datos desde cualquier dispositivo.
Descripción	La base de datos tendrá conexión a internet, esto permitirá que los datos se puedan descargar desde cualquier computadora, y se puedan consultar los resultados y las estadísticas.
Entrada	Solo se necesitara el nombre del usuario y la contraseña.
Salida	De salida se tendrán los datos que uno desee consultar, este ellos pueden estar los resultados y los nombres de los jugadores así como sus estadísticas.
Prioridad	Baja

Código de requerimiento: RNF2	
Nombre	Seguridad
Propósito	Tener seguridad y que no cualquiera pueda tener acceso a esto, esto por medio de una contraseña y de un usuario.
Descripción	Se creara una interfaz gráfica en la cual se registre un usuario y una contraseña, esto con el fin de tener mayor seguridad.
Entrada	Se pedirá de entrada un nombre de usuario y una contraseña.
Salida	Se mandara un mensaje de registro éxito ¡el usuario ha sido registrado con éxito!
Prioridad	Alta

Código de requerimiento: RNF3	
Nombre	Rapidez del programa
Propósito	Se hace con el propósito de aceptar los mayores números de usuarios y de datos.
Descripción	Se verá el tipo de almacenamiento y los datos que soportara el programa, esto con el fin de que no sature.
Entrada	Datos de entrada así como los registros.
Salida	Una mayor rapidez en el programa, esto asegura que no se esté trabando el programa a cada rato.
Prioridad	Baja

Código de requerimiento: RNF4	
Nombre	confidencialidad
Propósito	Que la información que brinde el usuario del programa, se mantenga con una discreción y solo con fines corporativos.
Descripción	Los datos que proporcione el usuario serán resguardados con confidencialidad, esto quiere decir que nada puede salir de los programadores.
Entrada	Datos de los partidos equipos y jugadores
Salida	Protección y confidencialidad de los datos
Prioridad	Alta

Código de requerimiento: RNF5	
Nombre	<u>Look and feel</u>
Propósito	El propósito es que la interfaz sea amigable para el usuario y no tenga problemas para usarlo.
Descripción	El aspecto y la interfaz que se le muestra al usuario debe de ser de una manera amigable e intuitiva.
Entrada	El cliente
Salida	Un aspecto amigable y fácil de utilizar para cualquier usuario.
Prioridad	Alta

8. Estándares de codificación

8.1. Normas de codificación

8.1.1. Generales

Todo el código para el desarrollo de este software se realizará en lenguaje java, utilizando este mismo lenguaje como entorno gráfico. Se utilizarán las clases JFrame como ventana principal y como ventanas secundarias JDialog.

8.1.2. Nombrar variable y clases

El estilo utilizado para nombrar las variables y clases es el CamelCase en su tipo UpperCamelCase, donde la primera letra de cada una de las palabras es mayúscula.

8.1.3. Sangría

Todo el código desarrollado tendrá una Sangría de 8 espacios. Esta Sangría es la utilizada en el kernel de linux.

8.1.4. Comentarios

Los comentarios se realizarán como se observa en el siguiente código:

```
|  
/**  
 * nombre_funcion:  
 * @par1: descripción del parámetro 1  
 * @par2: descripción del parámetro 2  
 *  
 * Descripción de la función. Puede incluir referencias a los parámetros  
 * como @par1 y su fuente será ajustada correctamente en la salida.  
 * Las constantes pueden ser referidas como %constante  
 *  
 * valor de retorno: un entero y su descripción  
 */
```

8.1.5. Espacios

Es muy importante que a la hora de escribir código, se dejen los espacios de separación entre los elementos para poder leer con soltura. Para separar secciones de código se utilizara un renglón vacío, un buen ejemplo sería el que se observa en el siguiente fragmento de código:

```

// Creamos el usuario con los datos obtenidos

usuario = g_new(TipoEspacioUsuario, 1);

usuario->nombre = g_string_new ("Ficheros de sistema");

usuario->particion = g_string_new (cadena_aux);

usuario->espacio = espacio_acumulado;

|

lista_part = g_list_append (lista_part, usuario);

```

8.1.6. Nombre de interfaces

Para llamar a cada una de las interfaces y sus elementos, se les deberá colocar una pequeña abreviación de 3 letras en minúsculas antes de utilizar el CamelCase en su tipo UpperCamelCase.

9. Colaboradores

Colaboradores	Cargo/Función	Descripción del cargo
Bustamante Cortés Iván Guadalupe	Project Manager.	Es el encargado de dirigir al equipo durante el desarrollo del proyecto.
Hernández Vite Jesús Fernando	Desarrollador de Software y Base de datos.	Es el encargado de programar y supervisar a programadores en la codificación del software producto, delega responsabilidades.
Hernández Ramírez Heber Esaú	Programador (Junior).	Es el encargado de traducir a código (lenguaje java) las especificaciones del sistema dadas por el cliente.
Ruiz Guerra Lirio	Analista y Arquitecto de Base de Datos.	Es la encargada de explicar al equipo de desarrollo las características que debe incluir la Base de Datos.
Pacheco Antonio Everardo de Jesús	Analista y Arquitecto de Software.	Es el encargado de explicar al equipo de desarrollo las necesidades que el software producto indicadas por el usuario. Tiene en cuenta los RF y RNF.
Pedro Gabriel Rolando	Usuario final.	Es el encargado de dar el uso al producto final, dar el visto bueno al software producto y aceptar el proyecto.