

SISTEMA GESTOR DE PLANTILLAS DE UNA LIGA DE BALONCESTO (SGPLB)

Martín Gómez, Javier 47231977M javi040898@gmail.com-Ayala Urbanos, Christian 51196082Z
christianayala.saramago@gmail.com

UNIVERSIDAD DE ALCALÁ GII PATRONES SOFTWARE GRUPO 5

Contenido

INTRODUCCIÓN.....	3
REQUISITOS.....	3
Requisitos Funcionales.....	3
Requisitos No Funcionales	4
Excepciones.....	4
Distinción de Usuarios	5
Usuarios de la aplicación.....	5
Jugadores	5
Cuerpo técnico	6
Directiva	6
Socio.....	6
Dueño.....	6
DIAGRAMA DE CASOS DE USO.....	7
PATRONES UTILIZADOS.....	9
Patrones de creación	9
Patrón Factory Method.....	9
Patrón Singleton.....	10
Patrones estructurales.....	11
Patrón Fachada	11
Patrón Proxy.....	13
Patrón Adapter.....	13
Patrón DAO	14
Patrones de comportamiento.....	15
Patrón Iterator	15
Patrón State	16
Patrón Strategy	17
MANUAL DE INSTALACIÓN	18
Base de datos.....	20
Inserción de datos.....	20
MANUAL DE USUARIO	20
Login.....	21
Interfaz Dueño	25
Interfaz Directiva.....	28
Interfaz Jugador	33
Interfaz Miembro del cuerpo técnico	34

Interfaz Socio	34
Interfaz Registro Socio	35
Botones volver y cerrar sesión.....	36
Iconos	36

INTRODUCCIÓN

Desde una Liga de Baloncesto nos han solicitado el desarrollo de una aplicación informática para la gestión de las plantillas de los equipos.

Se va a desarrollar un sistema software para la gestión de plantillas de una liga de baloncesto.

El sistema almacenará los datos en una base de datos PostgreSQL con todos los datos de las plantillas de la liga, incluidos datos de directiva, contratos, socios... Dependiendo del tipo de usuario que acceda al sistema, podrá realizar unas operaciones u otras.

El sistema contará con una interfaz gráfica sencilla para con esto facilitar su uso de cara a los usuarios. Esto se debe a que se usará el lenguaje de programación Java para el desarrollo de dicha interfaz gráfica. De cara a una implementación venidera, se desarrollará una aplicación web y/o aplicación móvil para con ello facilitar más aún el acceso de los usuarios.

Esta interfaz gráfica será diferente para cada tipo de usuario. Por ejemplo, un socio no podrá realizar las mismas operaciones que un directivo del equipo, esto se basará en los distintos roles, como se ha explicado anteriormente.

El sistema se desarrollará aplicando patrones software, incluyendo de creación, estructurales, comportamiento, etc. También, se usará el patrón DAO (Data Access Object) para separar la lógica de negocio con la lógica de los datos, ya que usaremos una base de datos PostgreSQL.

Se incluirá un manual de usuario con la explicación del funcionamiento del sistema y, además, los pasos detallados de la instalación.

REQUISITOS

Requisitos Funcionales

Identificativo	Descripción
RF1	Pedir autenticarse al usuario para acceder al sistema
RF2	Comprobar que el usuario y la contraseña coinciden con lo registrado en la base de datos
RF3	Dar de alta, baja o modificar contratos de jugadores
RF4	Distinguir los diferentes tipos de usuarios
RF5	Registrar las plantillas de los equipos
RF6	Actualizar la información de las plantillas
RF7	Permitir a los usuarios consultar la diferente información sobre equipos, jugadores, etc. (Dependiendo del rol esta información será diferente)
RF8	Limitar el número de jugadores por plantilla a 16 jugadores como máximo

RF9	Limitar el número de jugadores por plantilla a 11 jugadores como mínimo
-----	---

Requisitos No Funcionales

Identificativo	Descripción
RNF1	Los datos se almacenarán en una base de datos PostgreSQL
RNF2	Se desarrollará el proyecto usando el lenguaje de programación Java a través del entorno de desarrollo NetBeans IDE.
RNF3	Los nombres de usuarios será el DNI, por lo que será único
RNF4	El sistema tendrá una interfaz gráfica utilizando la librería Swing
RNF5	Se entregará un manual de usuario que detallará el uso de la aplicación
RNF6	La aplicación se desarrollará siguiendo unos patrones software
RNF7	El sistema debe ser accesible a cualquier hora del día
RNF8	El sistema se podrá ejecutar en el sistema operativo Windows
RNF9	El sistema tendrá que ser escalable, permitiendo la implementación de futuras funcionalidades
RNF10	El sistema deberá proporcionar advertencias de error informativos y orientados a los usuarios.
RNF11	El sistema permite las operaciones de selección, copiado y pegado de texto

Excepciones

Además de las excepciones predefinidas, la aplicación deberá controlar al menos las siguientes excepciones:

- Alta de un directivo, jugador o miembro de cuerpo técnico existente.
- Baja de un directivo, jugador o miembro de cuerpo técnico no existente.
- Solicitud de renovación de un jugador cedido.
- Alta de un socio que ya está dado de alta en la aplicación.
- Los equipos no pueden tener más de 16 jugadores en plantilla (RF8).
- Los equipos no pueden tener menos de 11 jugadores en plantilla (RF9).
- Solo se puede ser socio de un equipo.
- Alta de un jugador con dorsal ya existente en el equipo.
- Ausencia de campos en el alta de un usuario.

Distinción de Usuarios

Usuarios de la aplicación

En el sistema se verán representados distintos tipos de usuarios, los cuales se pueden representar por roles. Por un lado, se encuentra el personal de cada club y por otro lado los usuarios socios de cada equipo, los cuales podrán consultar los distintos tipos de datos o estadísticas de los jugadores, clasificación, etc. Los usuarios deberán identificarse para así dar unos permisos u otros, para ello dispondrán de un usuario diferenciador. Estos nombres de usuario vendrán acompañados obviamente de una clave personal

Por su parte los socios entrarán en la aplicación mediante su correo y clave y tendrán posibilidad de ver las plantillas de cada equipo.

La aplicación tendrá diferentes tipos de usuarios:

1. Jugadores.
2. Cuerpo técnico.
3. Directiva.
4. Socio.
5. Dueño

Los datos a considerar para cada usuario son comunes sin tener distinciones de roles:

- o DNI
- o Nombre
- o Nacionalidad
- o Correo electrónico
- o Contraseña

También, habrá roles comunes a todos los usuarios:

- o Consulta de plantillas
- o Ver líderes de la liga
- o Consulta de clasificación
- o Modificación datos personales

Jugadores

Para todos los jugadores, se deben añadir los siguientes datos:

- o Posición
- o Dorsal
- o Fecha fin de contrato
- o Estado
- o Salario
- o Media de puntos
- o Media de rebotes
- o Media de asistencias

Roles:

- Solicitar renovación
- Solicitar traspaso
- Renuncia de contrato

Cuerpo técnico

Para todos los entrenadores, se deben añadir los siguientes datos:

- Puesto dentro del cuerpo técnico
- Fecha fin de contrato
- Táctica favorita
- Salario

Roles:

- Solicitar renovación
- Renuncia de contrato

Directiva

Para todo el personal, se deben añadir:

- Puesto
- Fecha fin de contrato
- Salario

Roles:

- Solicitar renovación
- Renuncia de contrato
- Alta jugador o miembro del cuerpo técnico
- Baja jugador o miembro del cuerpo técnico
- Modificación contrato jugador o miembro del cuerpo técnico

Socio

Para todos los socios, se deben añadir:

- Equipo

Roles:

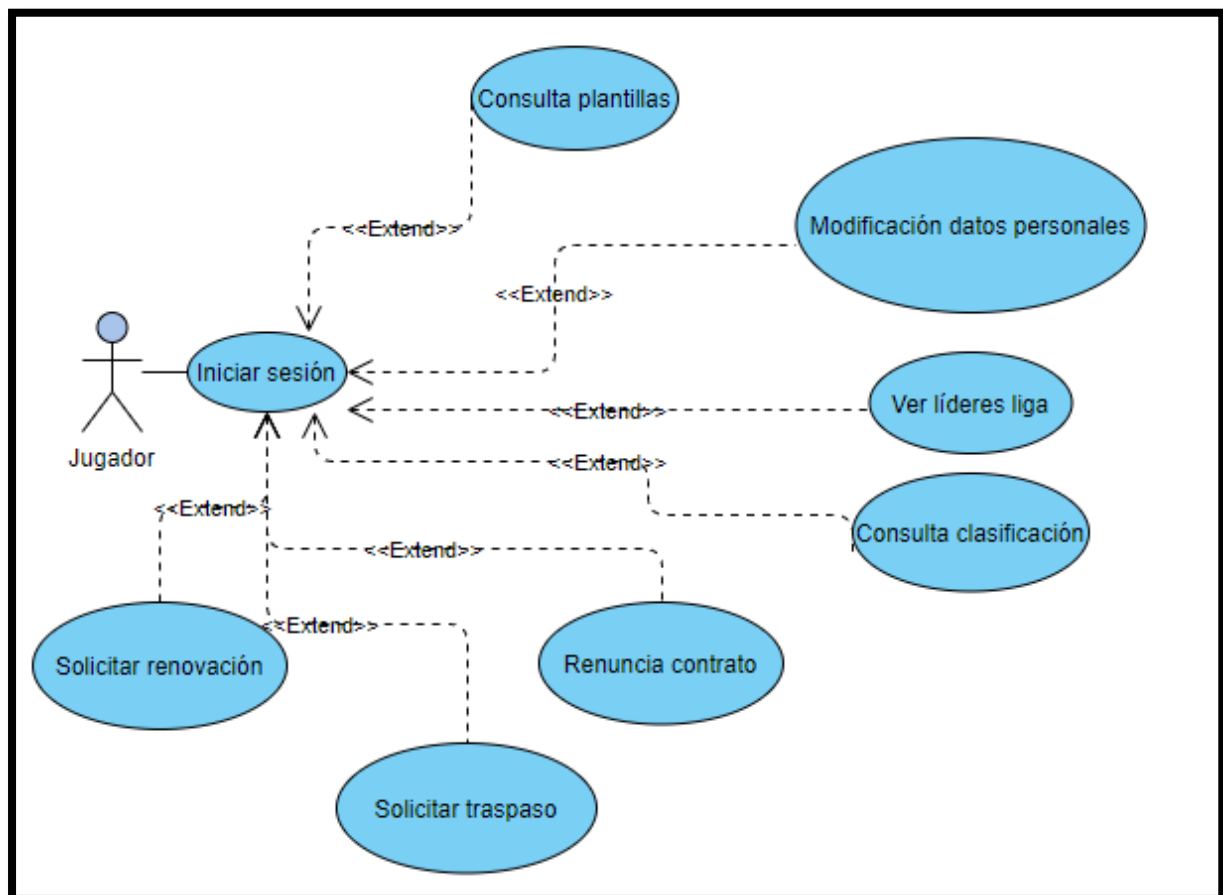
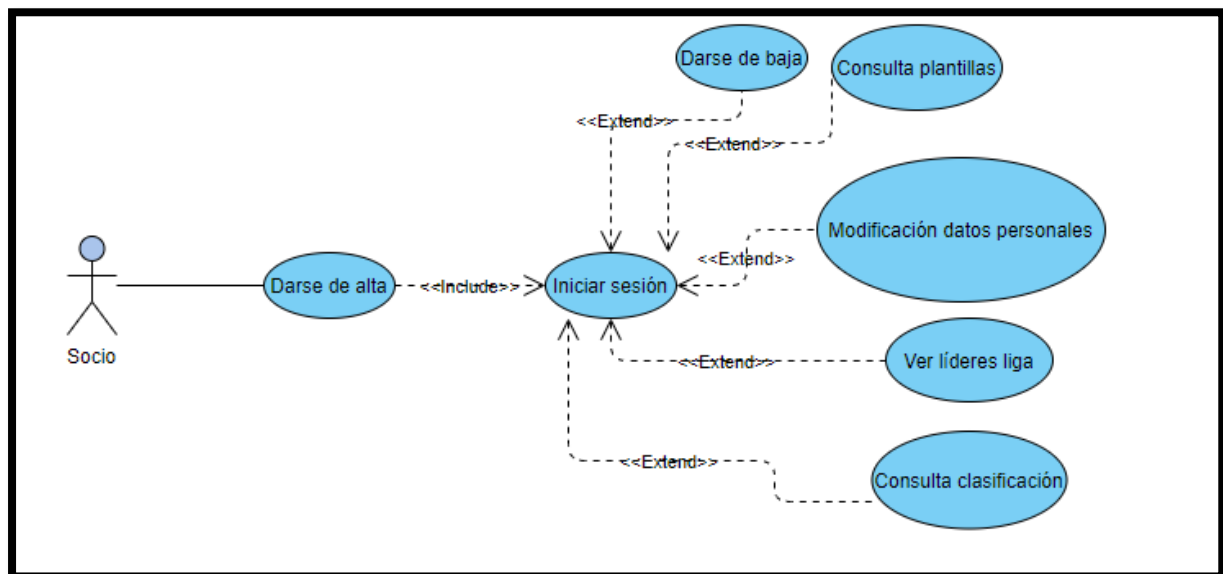
- Darse de baja del equipo

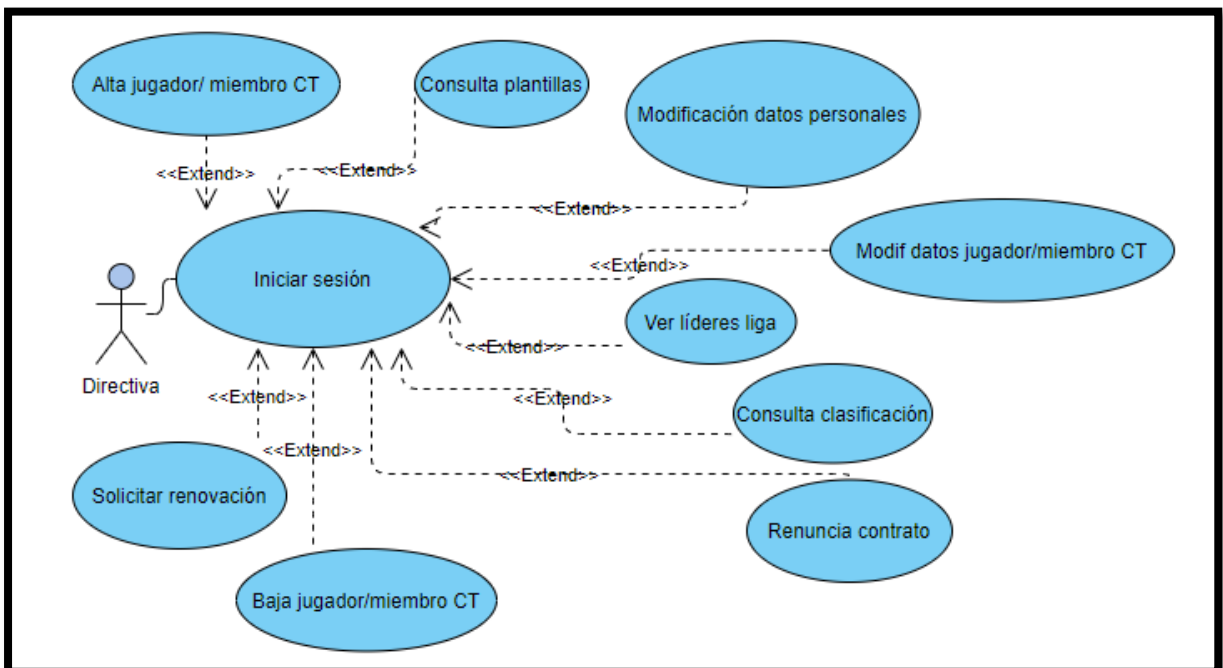
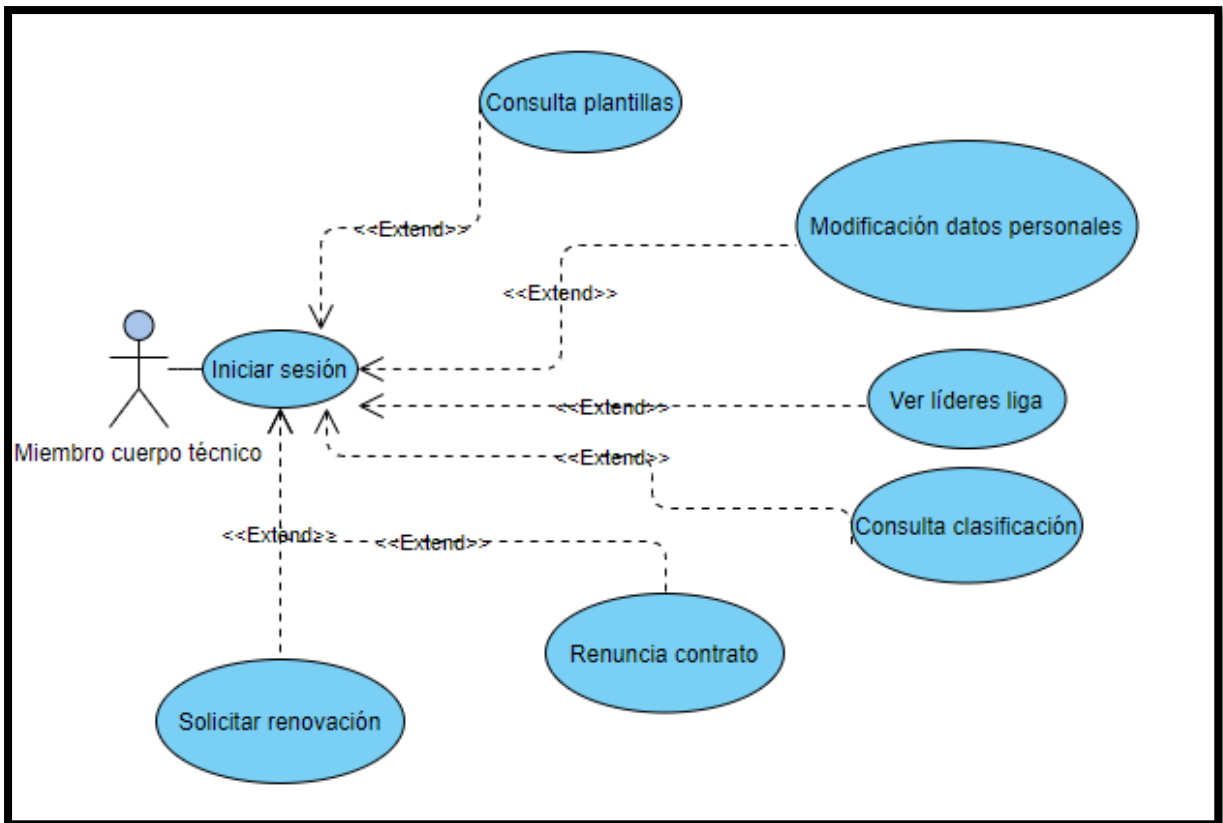
Dueño

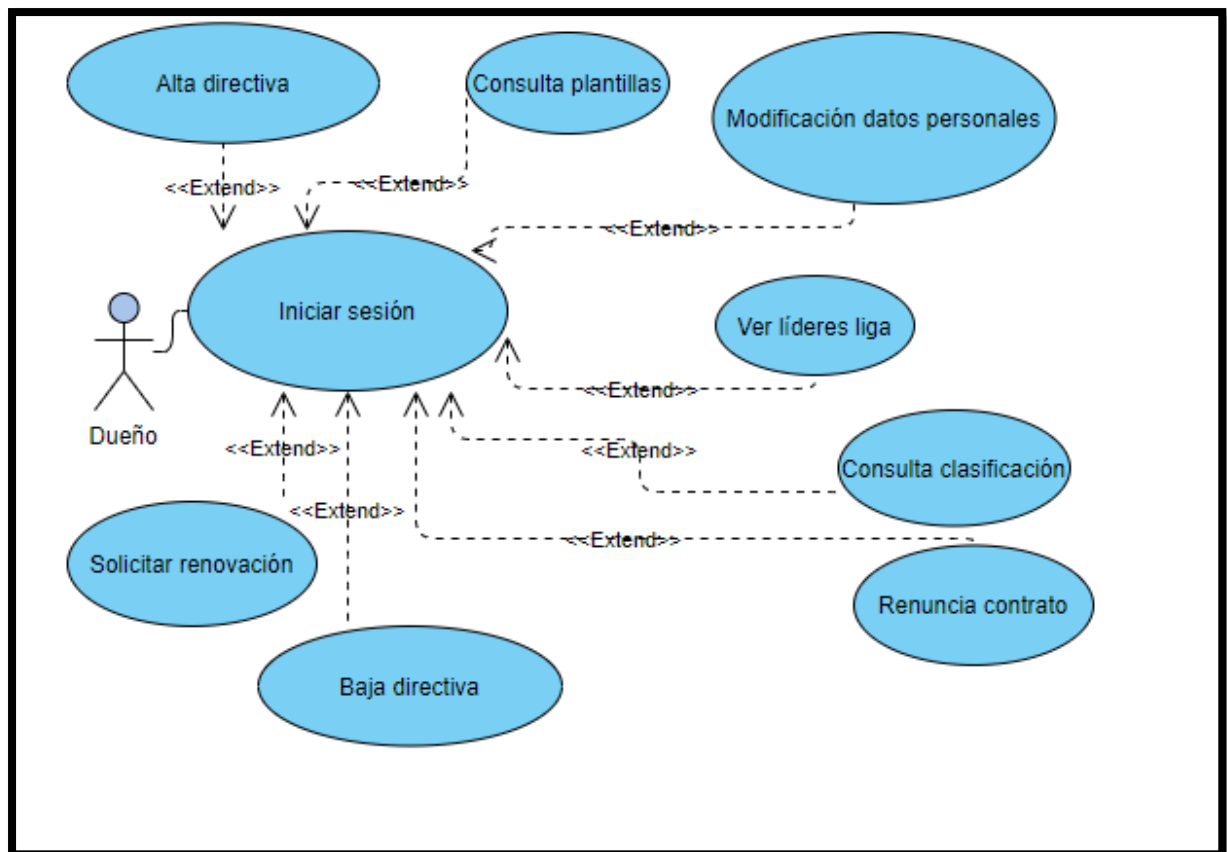
Para todos los dueños, se deben añadir los siguientes roles:

- Alta directivos
- Baja directivos

DIAGRAMA DE CASOS DE USO







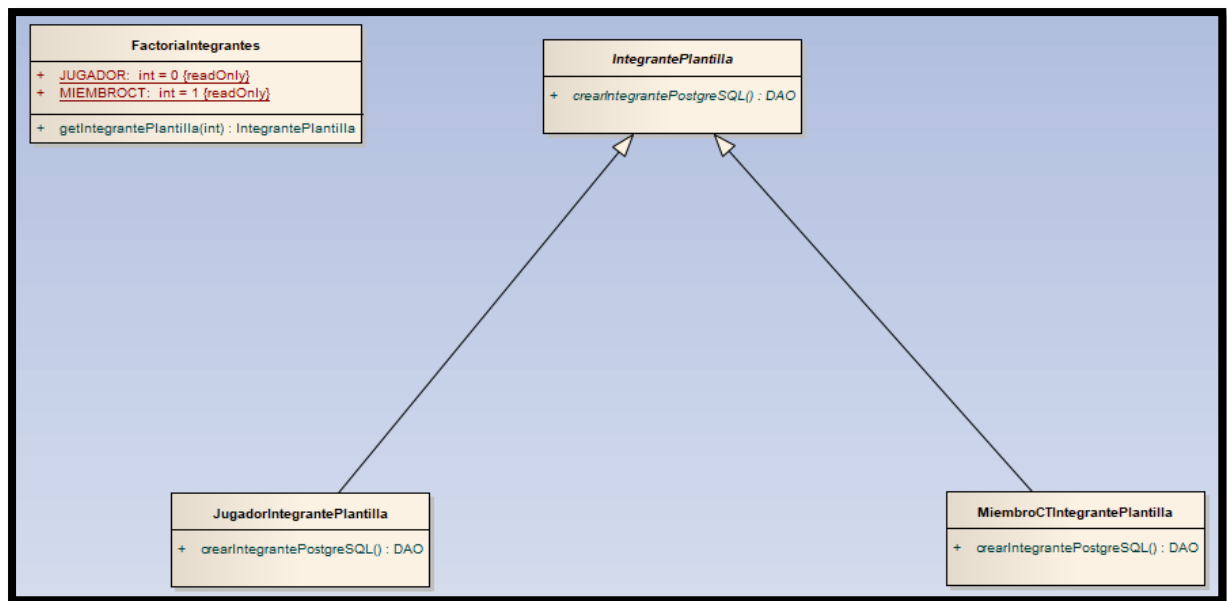
PATRONES UTILIZADOS

Patrones de creación

Patrón Factory Method

Este patrón se utiliza definir una interfaz que cree un objeto, pero delega a las subclases que decidan qué método instanciar. En este caso queremos crear integrantes de plantilla (Jugador y Miembro del Cuerpo Técnico).

El UML es el siguiente:



Producto: IntegrantePlantilla

ProductoConcreto: JugadorIntegrantePlantilla y MiembroCTIntegrantePlantilla

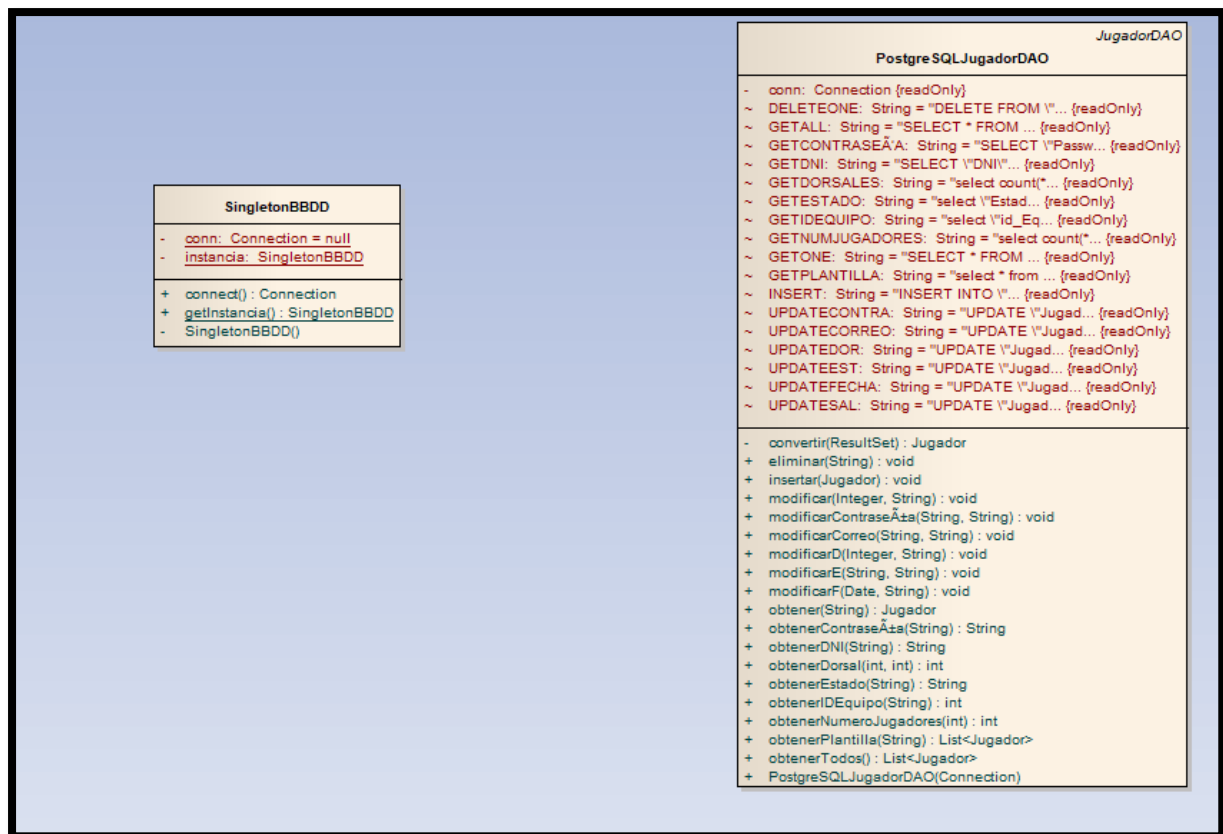
Creador: FactorialIntegrantes.

Las subclases de la fachada (explicada posteriormente) serán quienes utilicen estas clases, haciendo de cliente.

Patrón Singleton

Este patrón se utiliza para garantizar que las clases relacionadas con la base de datos solo tengan una instancia y así proporcionar un acceso global.

El UML es el siguiente:



Cuando se instancien objetos de las clases del paquete SQL, se usará la única instancia del singleton a través del método getInstancia().

Patrones estructurales

Patrón Fachada

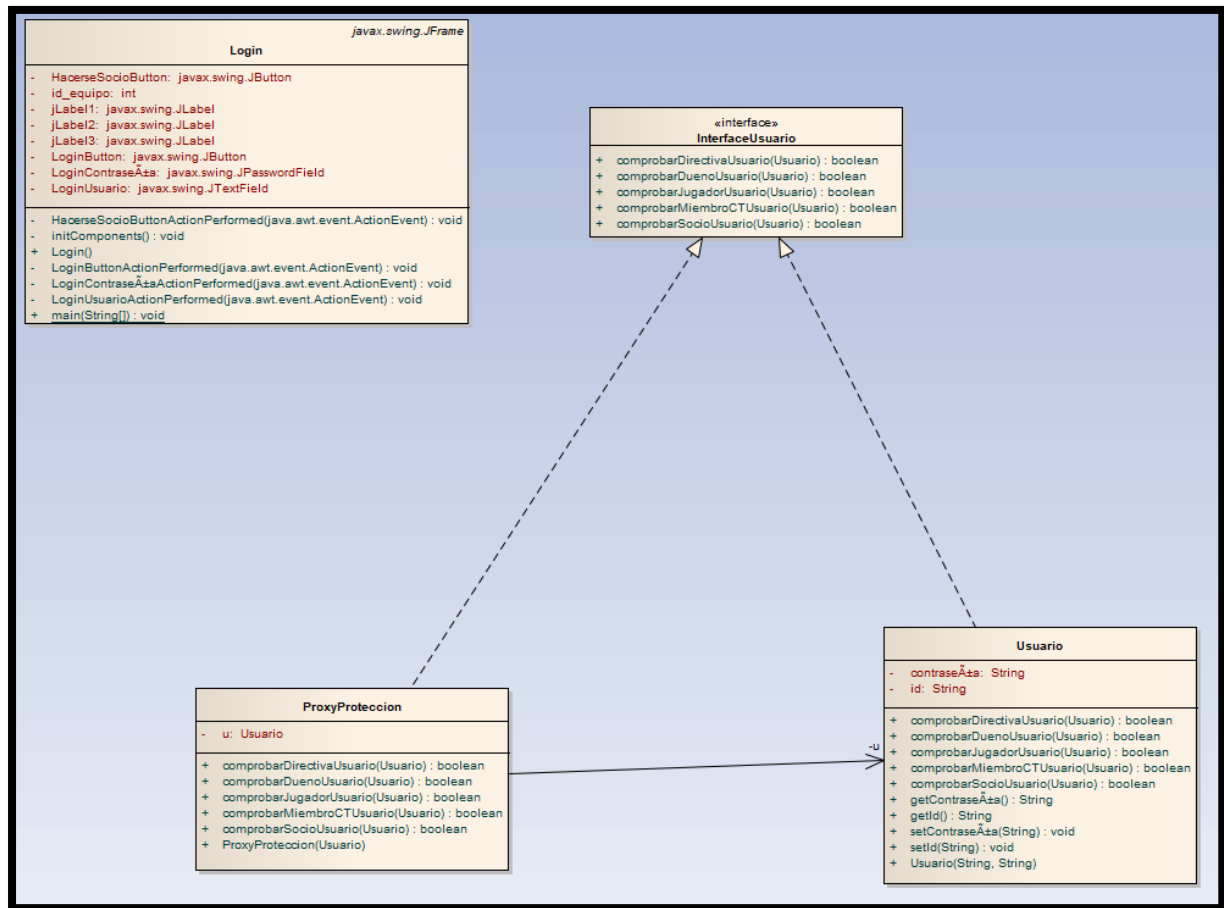
Se usa el patrón fachada en la interfaz FachadaIntegrante. Se utiliza para tener un acceso más sencillo a otras interfaces más complejas. Estas interfaces son AltaJugador, BajaJugador, BajaMiembroCT, AltaMiembroCT, ModificarJugador y ModificarMiembroCT.

La interfaz de la fachada es la siguiente:

Patrón Proxy

Se utiliza un proxy de protección para controlar los accesos a la aplicación únicamente a los usuarios a través de su usuario (DNI) y contraseña.

El UML es el siguiente:



Sujeto: InterfaceUsuario. Es la interfaz común entre el Proxy y el SujetoReal.

Proxy: ProxyProtección. Redirige las llamadas de los métodos al objeto real.

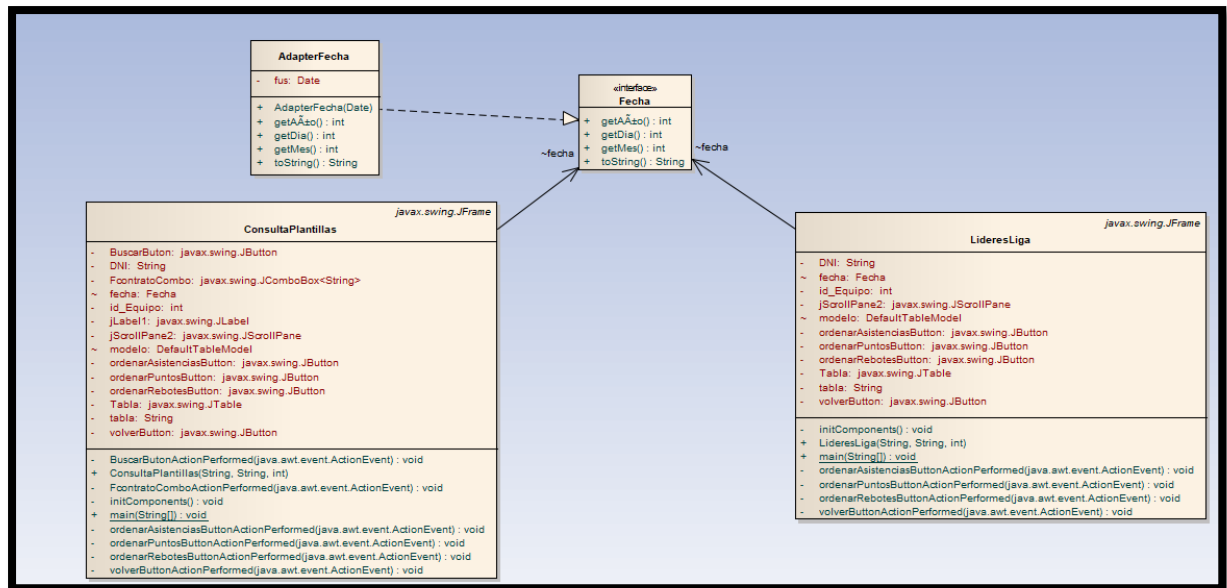
SujetoReal: Usuario. Implementa los servicios reales ofrecidos.

Cliente: Login. Utiliza la interfaz del proxy para usar la clase real (Usuario).

Patrón Adapter

Este patrón se utiliza para cambiar el formato de fecha US a nuestro formato, es decir, se adapta la fecha que tiene como formato el de la clase Date de Java al formato utilizado en España.

El UML es el siguiente:



Objetivo: en nuestro caso sería la interfaz Fecha que especifica el dominio a usar.

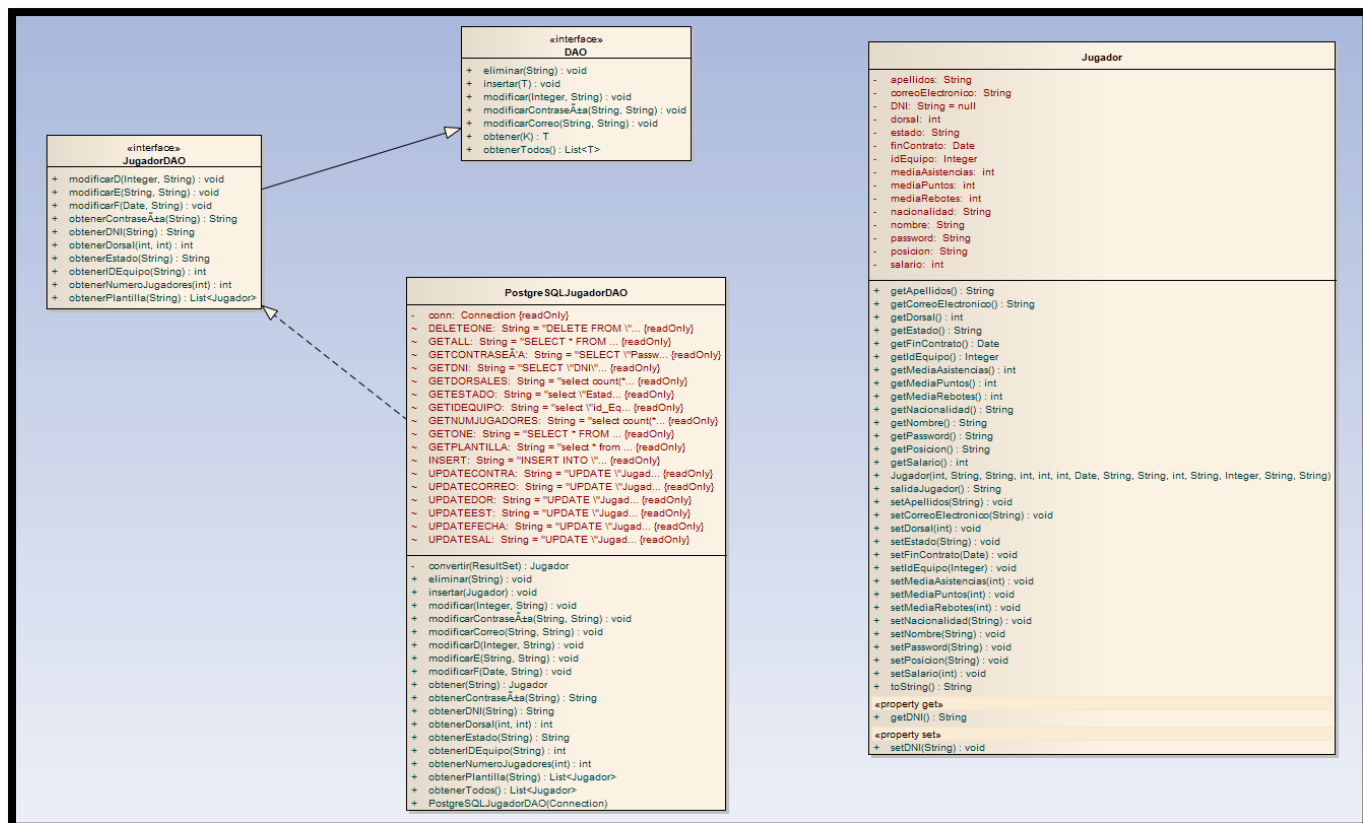
Adaptable: clase Date de Java que necesita ser adaptada y cambiar su formato

Adaptador: clase AdapterFecha que adapta el Adaptable a la interfaz Objetivo.

Patrón DAO

Este patrón se utiliza para abstraer todos los accesos a la base de datos. En el diagrama UML que mostramos a continuación, únicamente mostramos el acceso a la tabla Jugador, pero lo hacemos con todas las tablas (MiembroCuerpoTecnico, Socio, Equipo, Directiva, Dueño y Jugador).

El UML es el siguiente:

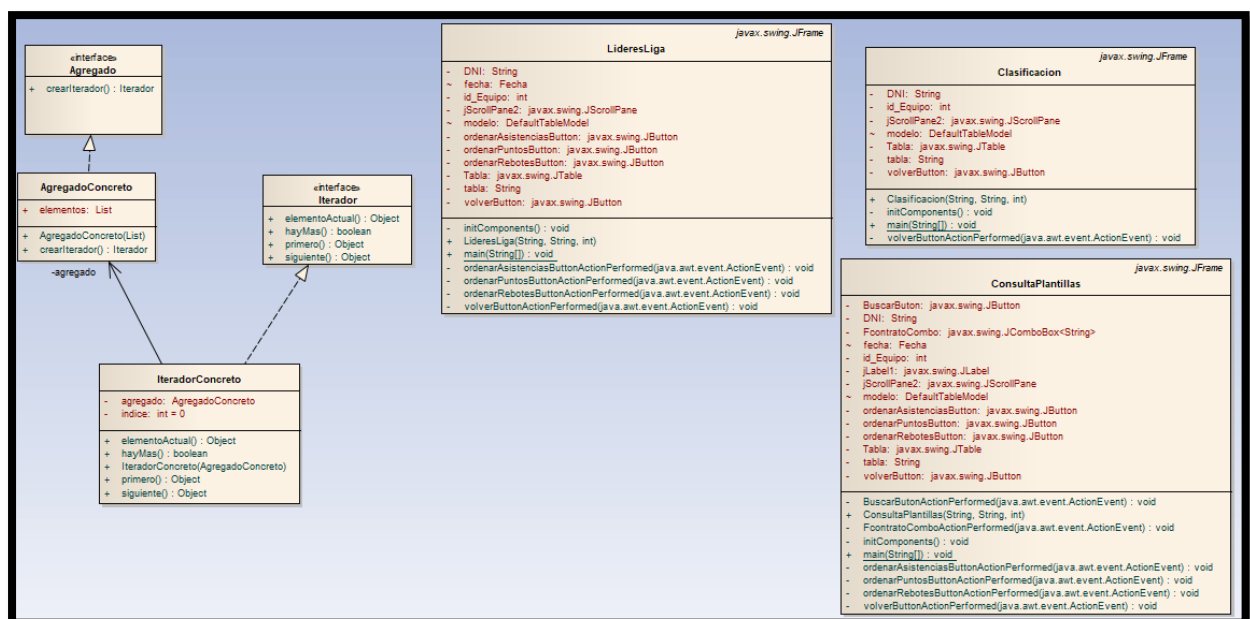


Patrones de comportamiento

Patrón Iterator

Se utiliza el patrón Iterator para recorrerse una lista de jugadores y equipos e ir añadiéndolos en la tabla que le corresponda, ya sea la de consulta plantillas, líderes de la liga y la clasificación general de equipos.

El diagrama UML es el siguiente:



Iterador: interfaz en la que se definen los métodos de iteración (primero(), siguiente(), hayMas(), elementoActual()).

Iterador Concreto: Implementación de los métodos de la interfaz Iterador que se han nombrado anteriormente y la creación de instancias se realiza por medio del Agregado Concreto

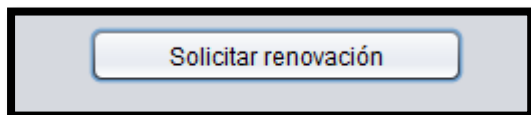
Agregado: se define el método para la creación del Iterador.

Agregado Concreto: implementa la interfaz Agregado, crea el Iterador y se define la lista que se recorrerá.

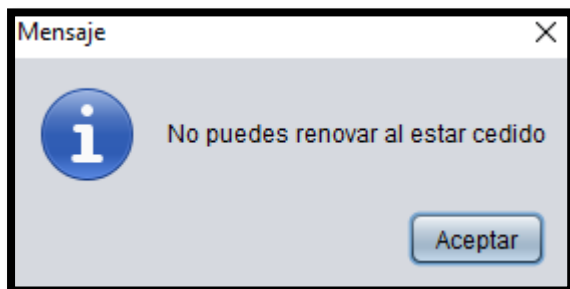
Patrón State

Se utiliza el patrón Estado para cambiar el comportamiento de un objeto (Jugador) en tiempo de ejecución.

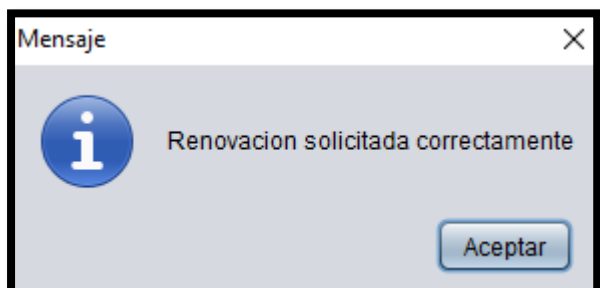
El Jugador podrá tener dos estados, Cedido y En propiedad y podrá solicitar renovación a través de este botón:



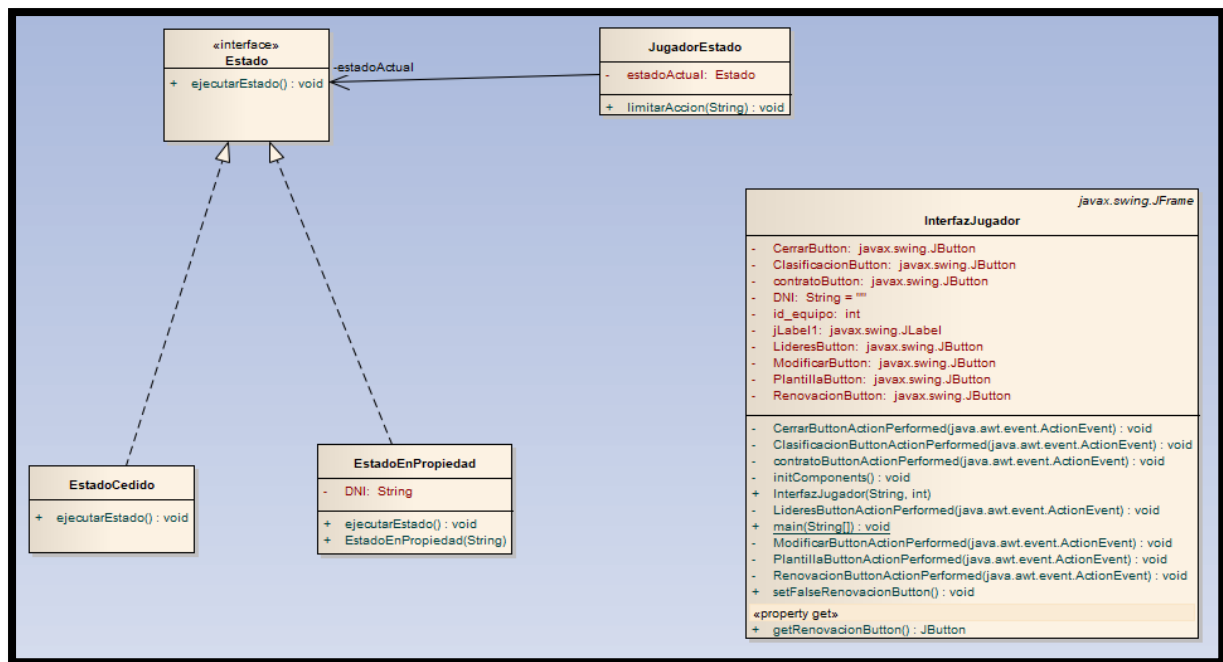
Si el estado es Cedido, no podrá solicitar renovación, ya que el Jugador no pertenece al club. Aparecerá esta advertencia:



Si el estado es En propiedad, el Jugador podrá solicitar la renovación al club y se enviará un correo al dueño del club:



El UML es el siguiente:



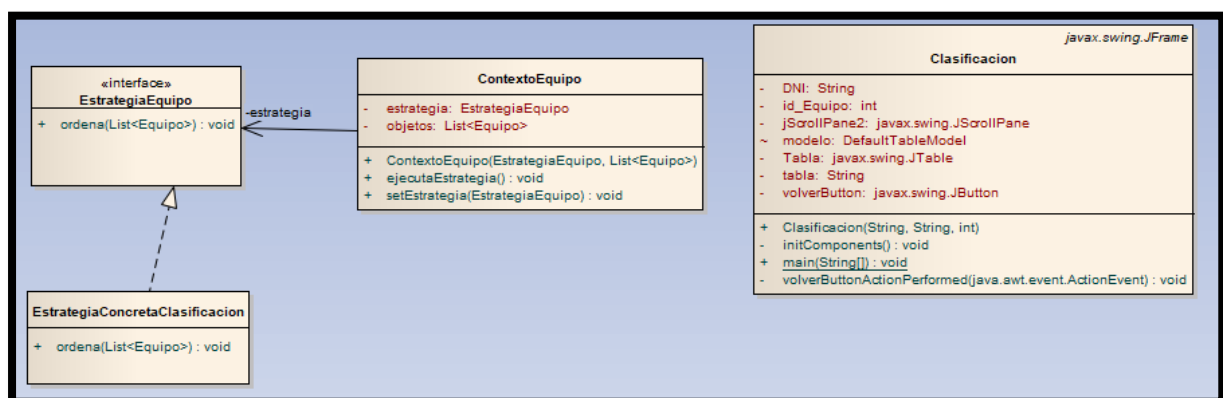
Contexto: JugadorEstado. Mantiene la referencia al estado actual.

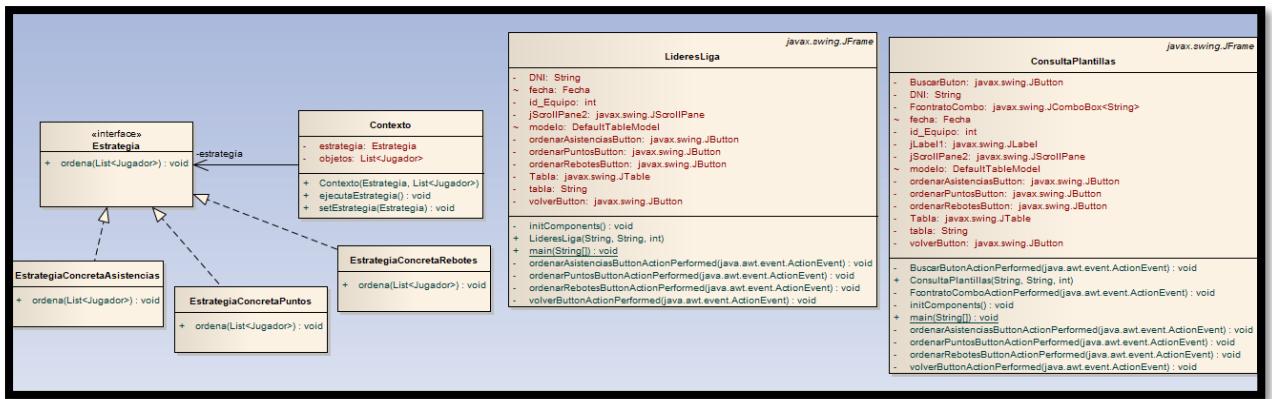
Estado: Estado. Define los métodos que dependen del estado del objeto.

EstadoConcreto: EstadoCedido y EstadoEnPropiedad. Implementan el comportamiento específico de cada estado, explicados previamente.

Patrón Strategy

Este patrón se utiliza para realizar la ordenación de la lista de jugadores según sus rebotes, puntos y asistencias de orden mayor a menor. Por otro lado, también se usa para ordenar de mayor a menor según sus puntos, la lista de equipos de la liga.





Contexto: Contexto, ContextoEquipo, ambas clases se utilizan para ver que estrategia de ordenación se va a utilizar y ejecutarla a la lista de jugadores y lista de equipos respectivamente.

Estrategia: Estrategia y EstrategiaEquipos, son las interfaces en las que se definen los métodos de ordenación.

Estrategia Concreta: EstrategiaConcretaAsistencias, EstrategiaConcretaRebotes, EstrategiaConcretaPuntos, EstrategiaConcretaClasificacion, estas clases se utilizan para ordenar las listas de jugadores y equipos.

MANUAL DE INSTALACIÓN

Para el correcto funcionamiento es necesario tener instalado en el equipo, en primer lugar java JDK y Netbeans. Para instalar el JDK habrá que introducirse en la siguiente dirección:

<https://www.oracle.com/java/technologies/javase/javase9-archive-downloads.html>

Aparecerá lo siguiente, donde se instalará dependiendo del sistema operativo de cada uno.

Java SE Development Kit 9.0.4		
This software is licensed under the Oracle Binary Code License Agreement for the Java SE Platform Products		
Product / File Description	File Size	Download
Linux	305.07 MB	jdk-9.0.4_linux-x64_bin.rpm
Linux	338.21 MB	jdk-9.0.4_linux-x64_bin.tar.gz
OS X	382.11 MB	jdk-9.0.4_osx-x64_bin.dmg
Solaris SPARC	206.97 MB	jdk-9.0.4_solaris-sparcv9_bin.tar.gz
Windows	375.56 MB	jdk-9.0.4_windows-x64_bin.exe

Después, para instalar NetBeans habrá que introducirse en la dirección:

<https://netbeans.org/downloads/8.2/rc/>

Aparecerá lo siguiente y habrá que instalarse lo remarcado:

NetBeans IDE 8.2 RC Download

8.1 | 8.2 RC | Development | Archive

Email address (optional):

Subscribe to newsletters: ☒ Monthly ☐ Weekly

☒ NetBeans can contact me at this address

IDE Language: English Platform: Windows

Note: Greyed out technologies are not supported for this platform

NetBeans IDE Download Bundles

Supported technologies *	Java SE	Java EE	HTML5/JavaScript	PHP	C/C++	All
NetBeans Platform SDK	•	•				•
Java SE	•	•				•
Java FX	•	•				•
Java EE		•				•
Java ME						•
HTML5/JavaScript		•	•	•		•
PHP			•	•		•
C/C++					•	•
Groovy						•
Java Card™ 3 Connected						•
Bundled servers						
GlassFish Server Open Source Edition 4.1.1		•				•
Apache Tomcat 8.0.27		•				•

Ahora, para instalar la base de datos (PostgreSQL), habrá que introducirse en el siguiente enlace:

<https://www.postgresql.org/download/windows/>

Aparecerá lo siguiente, donde seleccionamos la versión de PostgreSQL que queremos instalar (en nuestro caso hemos instalado la 12):

Windows installers

Interactive installer by EDB

Download the installer certified by EDB for all supported PostgreSQL versions.

This installer includes the PostgreSQL server, pgAdmin; a graphical tool for managing and developing your databases, and StackBuilder; a package manager that can be used to download and install additional PostgreSQL tools and drivers. Stackbuilder includes management, integration, migration, replication, geospatial, connectors and other tools.

This installer can run in graphical or silent install modes.

The installer is designed to be a straightforward, fast way to get up and running with PostgreSQL on Windows.

Advanced users can also download a **zip archive** of the binaries, without the installer. This download is intended for users who wish to include PostgreSQL as part of another application installer.

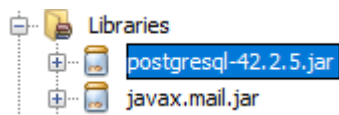
Platform support

The installers are tested by EDB on the following platforms. They can generally be expected to run on other comparable versions:

PostgreSQL Version	64 Bit Windows Platforms	32 Bit Windows Platforms
13	2019, 2016	
12	2019, 2016, 2012 R2	
11	2019, 2016, 2012 R2	
10	2016, 2012 R2 & R1, 7, 8, 10	2008 R1, 7, 8, 10

Durante la instalación, también se instalará PgAdmin.

Para la conexión de PostgreSQL con NetBeans y poder enviar correos desde NetBeans, se utilizarán las siguientes librerías:

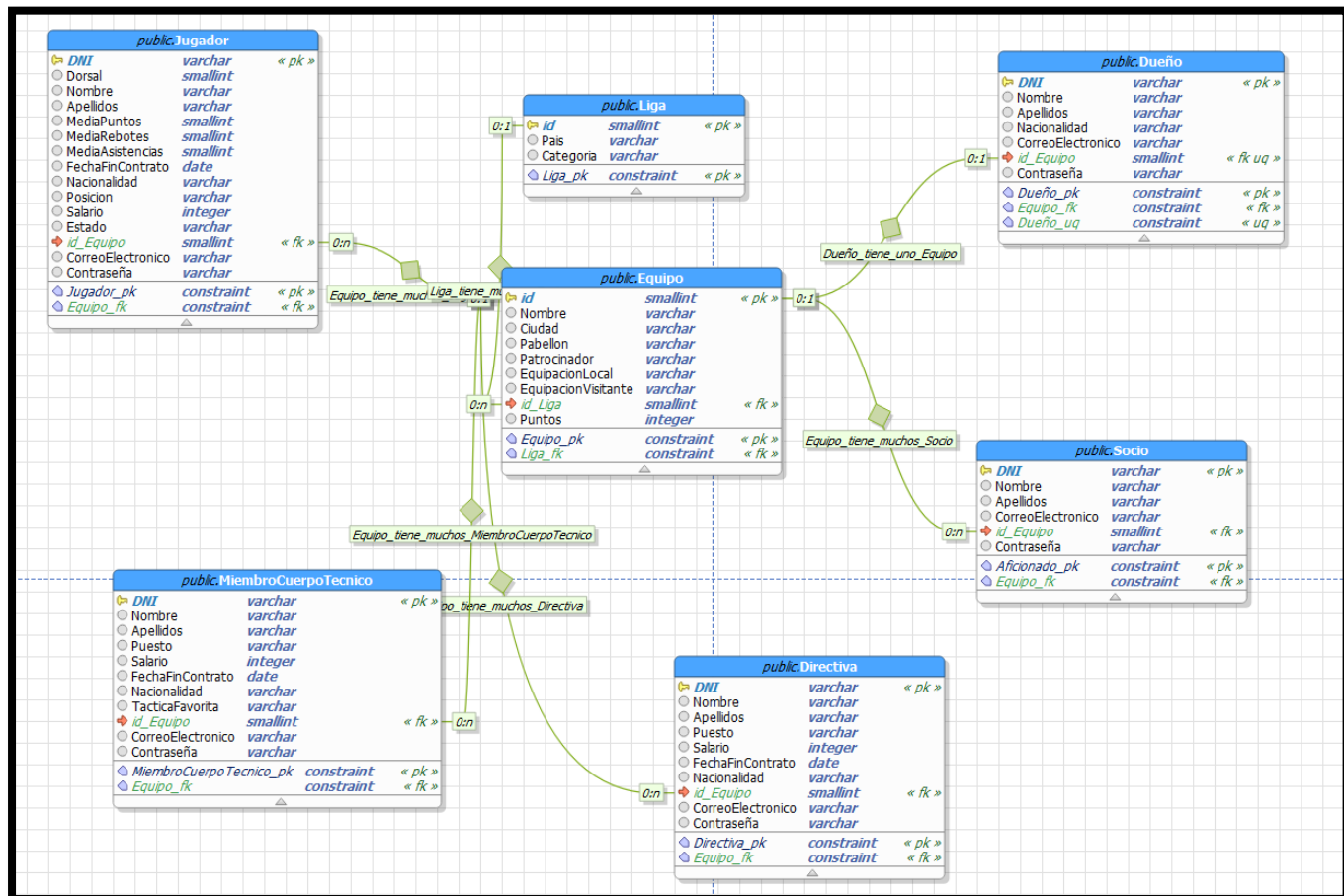


Con estas herramientas, se podrá ejecutar la aplicación.

Base de datos

Para poner en funcionamiento la base de datos, habrá que ejecutar el script proporcionado llamado "sqlBD.sql". Donde se crean las tablas.

El modelo e/r de la base de datos, realizado en PgModeler es el siguiente:



Inserción de datos

Para crear registros en las tablas, habrá que abrir el script llamado "insertLigaDueño.sql" y ejecutar la línea donde se encuentra el insert de Liga.

Ahora, habrá que importar en la tabla "Equipo" el archivo "equipos.txt". Después, se ejecutarán las líneas de los inserts de "Dueño" del script "insertLigaDueño.sql".

Por último, se importarán en cualquier orden los archivos "cuerpotecnico.txt", "directiva.txt", "jugadores.txt" y "socios.txt" en sus respectivas tablas. Estos archivos contienen registros aleatorios que se usarán de modelo para probar la aplicación correctamente.

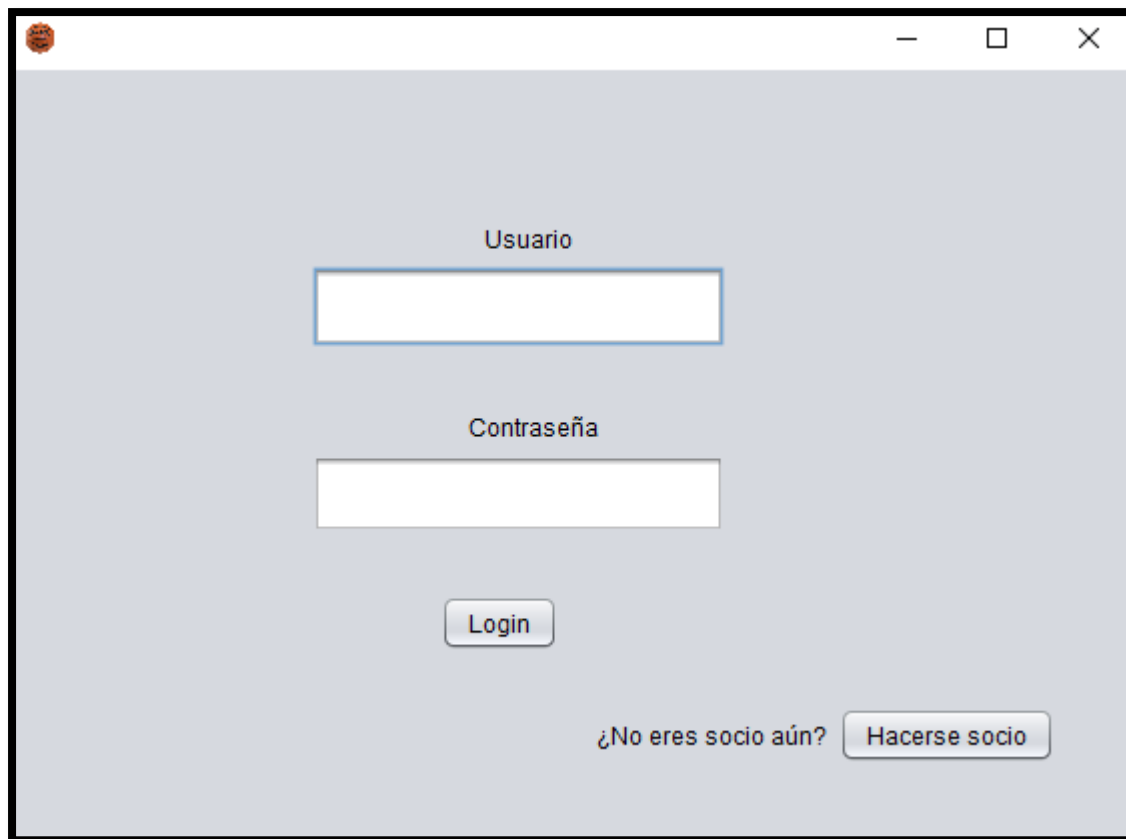
MANUAL DE USUARIO

La aplicación se pondrá en funcionamiento arrancando el proyecto denominado PLPatrones.

Para el correcto funcionamiento de la aplicación, la clase Login (Login.java dentro del paquete Interfaces) debe ser la primera en ejecutarse.

Login

Al iniciar la aplicación, aparecerá la ventana de login, donde cada usuario introducirá su usuario (DNI) y su contraseña. También, habrá un apartado para registrarte como socio:



The login window has a light gray background. At the top left is a small circular logo. In the center, there are two white rectangular input fields. The first field is labeled 'Usuario' and the second is labeled 'Contraseña'. Below these fields is a button labeled 'Login'. At the bottom right, there is a link '¿No eres socio aún?' followed by a button labeled 'Hacerse socio'.

Dependiendo del usuario que se introduzca, aparecerá su interfaz correspondiente. Recordemos que los usuarios son: dueño, directivo, jugador, miembro del cuerpo técnico y socio. Cada usuario tendrá su propia interfaz dependiendo de su tipo, sin embargo, habrá funciones comunes a todos los usuarios:

- **Consultar plantillas:** aquí se podrán observar los jugadores de cada equipo, pudiendo seleccionar cada equipo y, además, ordenar mediante botones a dichos jugadores por puntos, asistencias y rebotes. Vemos la plantilla de los lakers:

Selección el equipo del que quiere consultar la plantilla										
Lakers			Buscar	Ordenar por puntos	Ordenar por asistencias	Ordenar por rebotes				
Nombre	Apellidos	Dorsal	Media Puntos	Media Asistencias	Media Rebotes	Nacionalidad	Posicion	Salario (\$)	Estado	Fecha fin Contrato
Gianis	Kent	21	16	5	6	Argentina	Alero	14236222	Cedido	08/06/2023
Christian	Wallace	78	27	3	3	Argentina	Pivot	8705210	En propiedad	29/07/2022
Gianis	Taylor	82	5	4	8	Italia	Ala Pivot	17429139	En propiedad	16/07/2029
Jordan	García	37	21	7	5	Canada	Escolta	28728970	En propiedad	24/11/2023
Alexander	White	70	20	7	8	Francia	Pivot	12865337	En propiedad	04/11/2026
Lucas	Griffin	23	26	5	8	Mexico	Ala Pivot	5568944	En propiedad	19/02/2025
Isaac	Durant	24	10	7	4	Canada	Escolta	24892184	En propiedad	15/06/2023
Ethan	Cole	77	26	3	5	Grecia	Escolta	26536205	En propiedad	24/08/2025
Marc	Thomas	25	27	9	6	Ecuador	Pivot	21591937	En propiedad	11/08/2024
Samuel	White	46	25	9	4	España	Alero	14087607	En propiedad	25/01/2021
Marc	Gasol	14	26	4	4	España	Pivot	3000000	En propiedad	31/07/2030
Mario	Wallace	2	14	8	2	Grecia	Escolta	29788913	En propiedad	05/09/2021
Alex	Caruso	0	20	1	5	Estadounidense	Base	500000	En propiedad	31/07/2028

Se ordenan por puntos:

Seleccione el equipo del que quiere consultar la plantilla

Lakers

Nombre	Apellidos	Dorsal	Media Puntos	Media Asistencias	Media Rebotes
Christian	Wallace	78	27	3	3
Marc	Thomas	25	27	9	6
Lucas	Griffin	23	26	5	8
Ethan	Cole	77	26	3	5
Marc	Gasol	14	26	4	4
Samuel	White	46	25	9	4
Jordan	García	37	21	7	5
Alexander	White	70	20	7	8
Alex	Caruso	0	20	1	5
Gianis	Kent	21	16	5	6
Mario	Wallace	2	14	8	2
Isaac	Durant	24	10	7	4
Gianis	Taylor	82	5	4	8

Ordenamos por asistencias la plantilla de los clippers:

Seleccione el equipo del que quiere consultar la plantilla

Clippers

Nombre	Apellidos	Dorsal	Media Puntos	Media Asistencias	Media Rebotes
Carter	Lee	0	27	9	8
Kevin	Miller	57	26	9	1
Elijah	Trump	94	8	9	5
Sebastian	Brown	20	23	8	7
Stephen	Cole	27	9	7	8
Russell	Wallace	1	21	7	1
James	Ramos	55	8	7	3
Kevin	Young	29	19	4	1
James	Kent	43	6	4	2
Benjamin	Brown	66	14	3	4
Logan	Brown	28	10	3	1
Andrew	Williams	15	17	2	5
Logan	Jackson	2	29	2	5

Ordenamos por rebotes a la plantilla de los bucks:

Seleccione el equipo del que quiere consultar la plantilla

Bucks

Nombre	Apellidos	Dorsal	Media Puntos	Media Asistencias	Media Rebotes	Nacionalidad
Christian	Ford	53	27	7	9	Mexico
Lucas	Williams	63	12	7	9	España
William	Murray	81	23	3	8	USA
Geronimo	Brown	57	27	9	6	España
Stefano	Davies	26	16	8	6	Colombia
Oliver	Cole	28	22	1	5	España
Andrew	Harris	55	11	9	4	Canada
Oliver	Durant	52	23	9	4	Canada
Daniel	Morales	60	25	4	2	Italia
Liam	Ramos	17	19	1	2	Ecuador
David	White	71	15	1	1	Italia

- **Modificar mis datos personales:** aquí podrás modificar tu correo electrónico y tu contraseña:

Volver

E-mail

Contraseña


Modificamos el e-mail y la contraseña:

E-mail

Contraseña

Aparecen mensajes de confirmación:

Mensaje

 E-mail modificado correctamente

Comprobamos en la base de datos que se ha modificado correctamente:

	DNI [PK] character varying	Nombre character varying	Apellidos character varying	Nacionalidad character varying	CorreoElectronico character varying	Id_Equipo smallint	Password character varying
1	91741234F	Joseph	Tsai	Estados Unidos	josephtsai@gmail.com	3	5450
2	13568315R	Wes	Edens	Estados Unidos	wesedens@gmail.com	4	6457
3	57896455D	Logan	Williams	Estados Unidos	LoganWilliams@gmail...	5	7455
4	14789655L	Jerry	Reinsdorf	Estados Unidos	JerryReinsdorf@gmail...	6	8455
5	94567315R	Jerry	Buss	Estados Unidos	jerrybuss@gmail.com	1	1234
6	45648315S	Steve	Ballmer	Estados Unidos	stevemaquina@hotmail...	2	123456

- **Ver líderes de la liga:** aquí se podrán observar las estadísticas (puntos, asistencias y rebotes) de todos los jugadores de la liga, pudiendo ordenarlos por puntos, asistencias o rebotes. Ordenados por puntos:

Ordenar por puntos Ordenar por asistencias Ordenar por rebotes				
Nombre	Apellidos	Media Puntos	Media Asistencias	Media Rebotes
Logan	Gordon	29	7	8
Logan	Jackson	29	2	5
Liam	White	28	3	7
Theodore	Westbrook	28	8	7
Lebron	Gibson	28	8	4
Christian	Wallace	27	3	3
Geronimo	Brown	27	9	6
Christian	Ford	27	7	9
Mario	Jackson	27	4	5
Russell	Ramos	27	7	3
Carter	Lee	27	9	8
Marc	Thomas	27	9	6
Samuel	Torres	26	6	2

Ordenados por asistencias:

Ordenar por puntos Ordenar por asistencias Ordenar por rebotes				
Nombre	Apellidos	Media Puntos	Media Asistencias	Media Rebotes
Andrew	Harris	11	9	4
Geronimo	Brown	27	9	6
Liam	Kent	24	9	1
Carter	Lee	27	9	8
Oliver	Durant	23	9	4
Kevin	Miller	26	9	1
Elijah	Trump	8	9	5
Marc	Thomas	27	9	6
Samuel	White	25	9	4
Sebastian	Brown	23	8	7
Russell	Cole	24	8	7
Isaiah	Bryant	25	8	8
Christian	Marquez	9	8	9

Ordenados por rebotes:

Ordenar por puntos Ordenar por asistencias Ordenar por rebotes				
Nombre	Apellidos	Media Puntos	Media Asistencias	Media Rebotes
Christian	Ford	27	7	9
Lucas	Williams	12	7	9
Christian	Marquez	9	8	9
Jacob	Thomas	18	3	9
Marc	Bryant	13	5	9
James	West	8	4	9
Stephen	Cole	9	7	8
Theodore	Manuteball	16	1	8
Gianis	Taylor	5	4	8
Alexander	White	20	7	8
Lucas	Griffin	26	5	8
Isaiah	Bryant	25	8	8
Logan	Gordon	29	7	8
Carter	Lee	27	9	8
Nacho	Taylor	6	3	8

- **Ver clasificación:** se podrán observar los equipos de la liga, con algunas de sus características, ordenados por puntos:

Nombre	Ciudad	Pabellon	Patrocinador	Equipacion Local	Equipacion Visitante	Puntos
Bucks	Milwaukee	Fiserv Forum	Heinekein	Verde	Blanco	55
Clippers	Los Angeles	Staples Center	Coca cola	Blanco	Rosa	52
Lakers	Los Angeles	Staples Center	Wish	Amarillo	Morado	50
Nets	Brooklyn	Bardays Center	Amazon	Negro	Azul	47
Celtics	Boston	Garden	Bwin	Verde	Negro	45
Bulls	Chicago	United center	American Airlines	Rojo	Blanco	45

Interfaz Dueño

La interfaz del usuario Dueño será la siguiente:



Aparte de las funciones comunes, se puede observar que también puede dar de alta y dar de baja a directivos:

- **Alta directivo:** podrá dar de alta a directivos en su equipo

Volver

Nombre

Javier

Apellidos

Martín

DNI

6546565J

Fecha fin contrato

2024

Nacionalidad

Española

Puesto

Secretario Técnico

Salario

112222

Mail

javier@gmail.com

Contraseña

Aceptar

Puesto

Salario

Mail

Director Deportivo

Gerente

Secretario Técnico

Asesor

Nacionalidad

Puesto

Salario

Mail

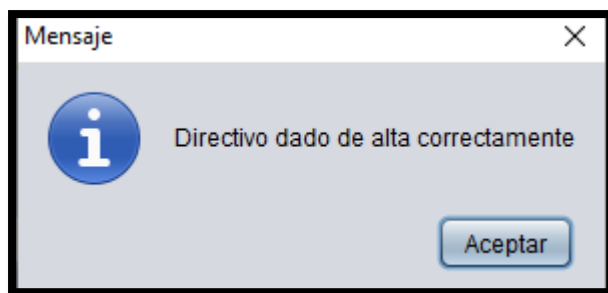
Argentina

Española

Estadounidense

Francesa

Italiana



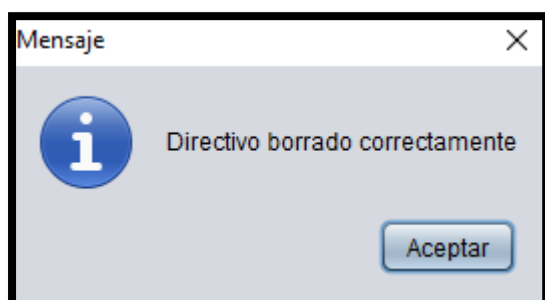
Vemos que en la base de datos se introduce correctamente:

24	40497346Y	Elijah	Johnson	Asesor	2468969	2027-03-22	Mexico	6	ElijahAsesor@gmail.co...	1875
25	6546565J	Javier	Marín	Secretario Técnico	112222	2024-06-30	Española	2	javier@gmail.com	1234

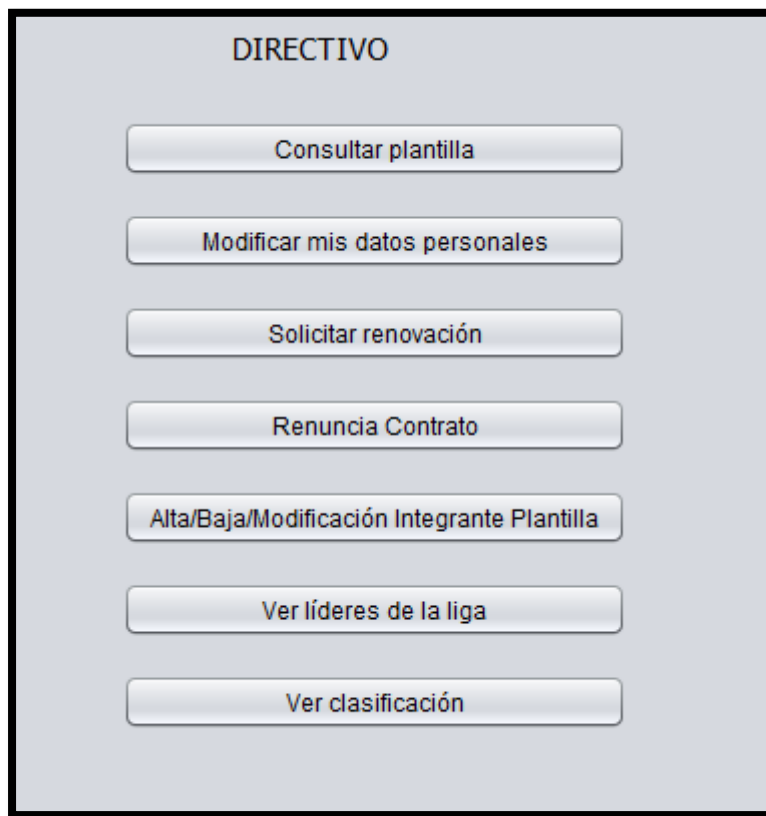
- **Baja directivo:** se da de baja a un directivo introduciendo su DNI. Borramos el directivo creado previamente.

Introduzca el DNI del jugador que desea dar de baja

Aceptar

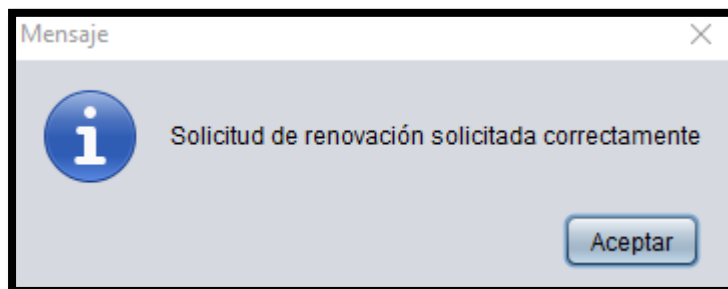


Interfaz Directiva



Se añaden diferentes funcionalidades como se puede comprobar:

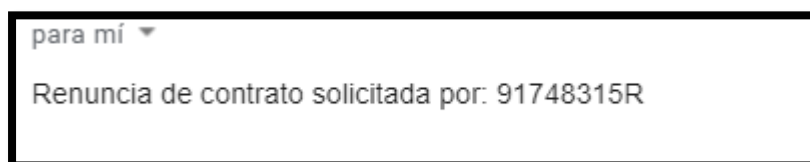
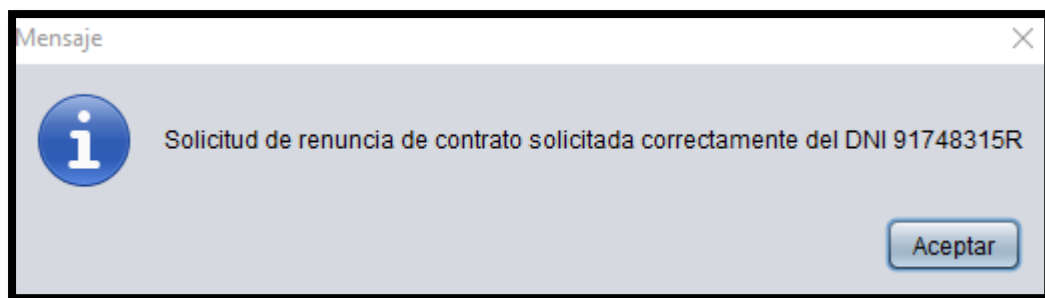
- Solicitar renovación: envía un e-mail al dueño solicitando su renovación:



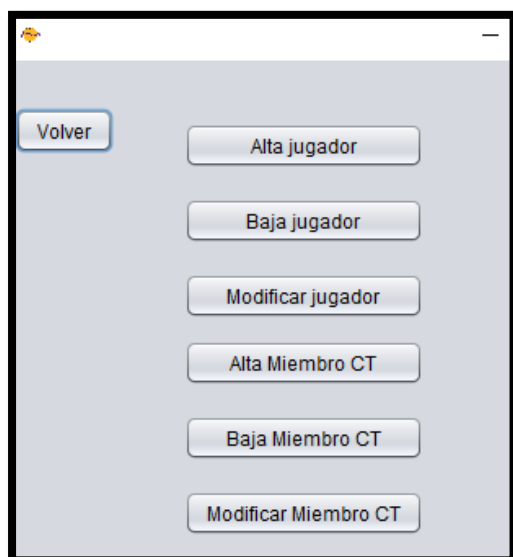
para mí ▼

Renovación de contrato solicitada por: 91748315R


- Renuncia de contrato: también envía un correo, pero esta vez solicita renuncia de su contrato:



- Alta/baja/modificación integrante plantilla: esta función permite al directivo dar de alta, de baja o modificar a un integrante de la plantilla (jugador o miembro del cuerpo técnico):



- Alta jugador: da de alta a un jugador:



Volver

Nombre

Alex

Apellidos

Caruso

DNI

11155544K

Fecha fin contrato

2027

Nacionalidad

Estadounidense

Posición

Base

Salario

444444

Estado

En propiedad

Mail

caruso@gmail.com


Dorsal

52

Contraseña

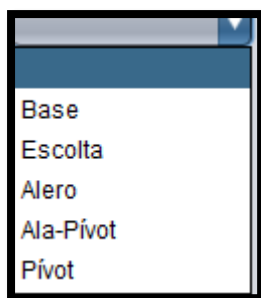
Aceptar

Mensaje



Jugador dado de alta correctamente

Aceptar



Controla excepciones como dar de alta a un jugador con el mismo DNI o mismo dorsal.ç

- Dar de baja jugador:

A screenshot of a web form for removing a player. It has a light gray background. At the top, it says 'Introduzca el DNI del jugador que desea dar de baja'. Below this is a text input field containing the value '11155544K'. At the bottom, there is a button labeled 'Aceptar'.

- Modificar jugador: se podrá modificar el dorsal, salario, fecha del fin del contrato o el estado:

A screenshot of a web form for modifying a player. It has a light gray background. At the top left is a button labeled 'Volver'. To its right is the text 'Introduzca el DNI del jugador que desea modificar'. Below this is a text input field. Further down, there are four rows of fields and buttons: 1. 'Dorsal' with a dropdown menu and a 'Modificar dorsal' button. 2. 'Salario' with a text input field and a 'Modificar salario' button. 3. 'Fecha fin de contrato' with a dropdown menu and a 'Modificar fecha fin de contrato' button. 4. 'Estado' with a dropdown menu and a 'Modificar estado' button.

Del miembro del cuerpo técnico será prácticamente lo mismo, exceptuando pequeños matices como el puesto en vez de la posición o la táctica favorita:

olver

Nombre

Apellidos

DNI

Fecha fin contrato

Nacionalidad

Puesto

Salario

Mail

Tactica Favorita

Contraseña

Aceptar

Tactica Favorita

Contraseña

Defensa en zona

Pick and roll

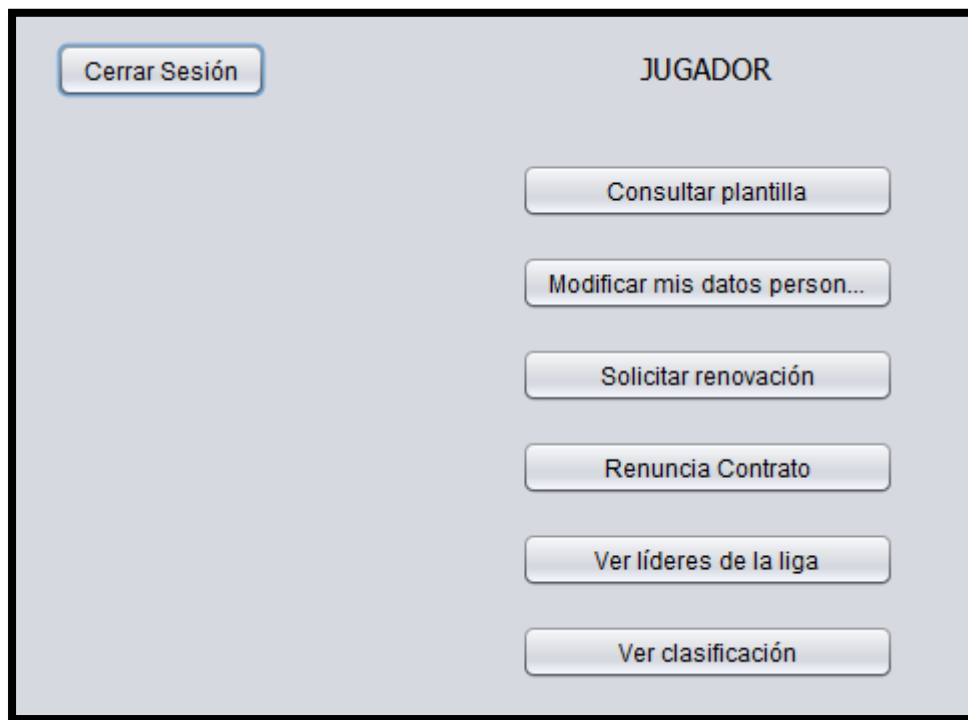
Triple

Small ball

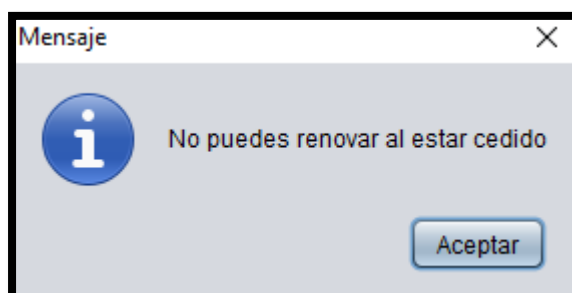
Defensa al hombre

Otras

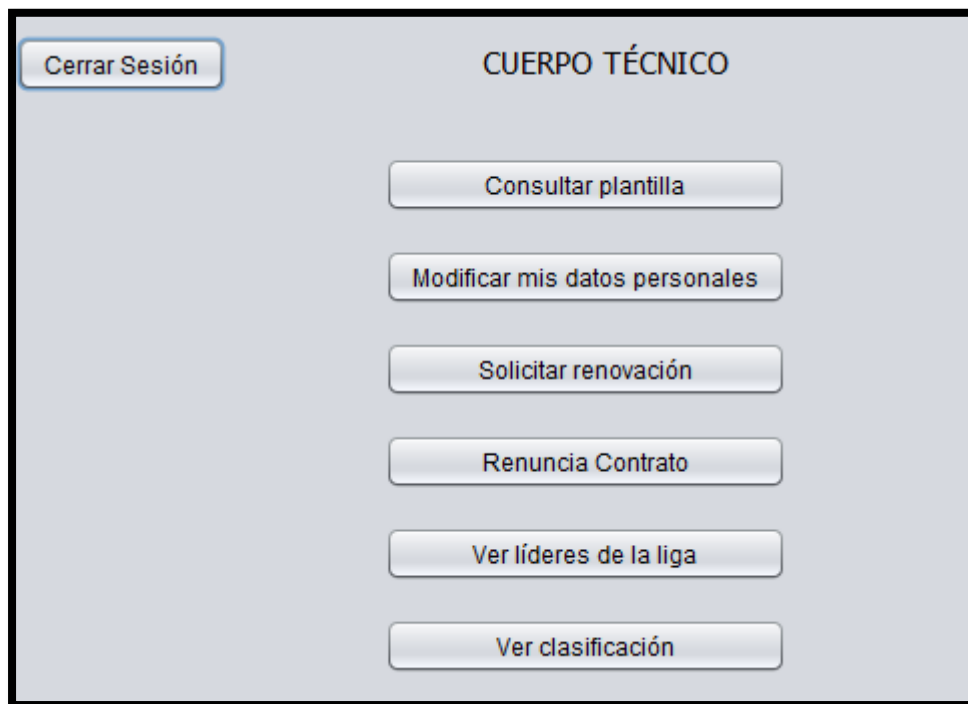
Interfaz Jugador



Tendrá las funciones comunes más renuncia de contrato y solicitar renovación. Esta última se podrá solicitar solo si es un jugador en propiedad (los jugadores cedidos no pueden solicitar renovación):



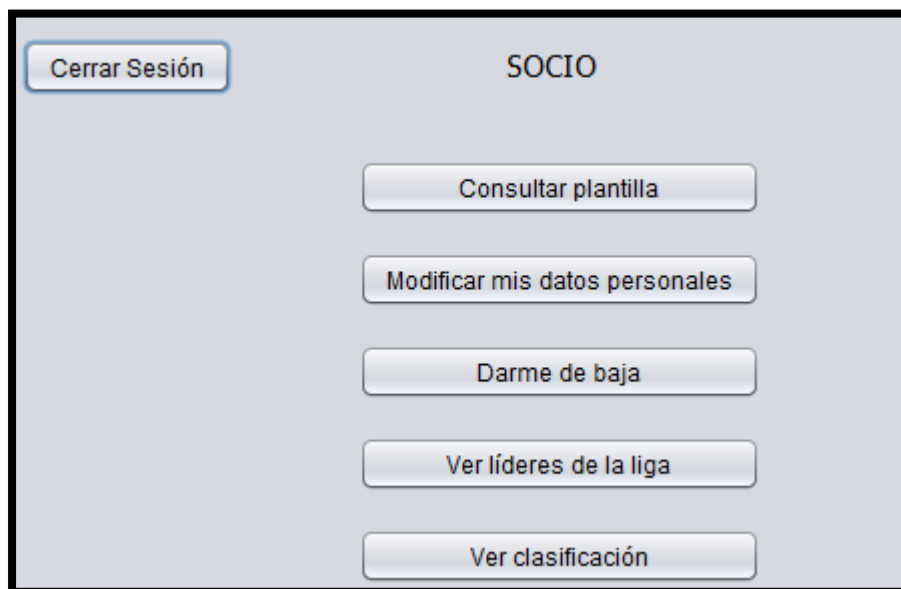
Interfaz Miembro del cuerpo técnico



Interfaz de usuario para el Miembro del cuerpo técnico. El panel tiene un fondo gris claro y una barra superior con el título "CUERPO TÉCNICO" a la derecha y un botón "Cerrar Sesión" a la izquierda. En el centro del panel, hay una columna vertical de seis botones rectangulares con bordes redondeados y un efecto de sombra. Los botones, desde arriba hacia abajo, son: "Consultar plantilla", "Modificar mis datos personales", "Solicitar renovación", "Renuncia Contrato", "Ver líderes de la liga" y "Ver clasificación".

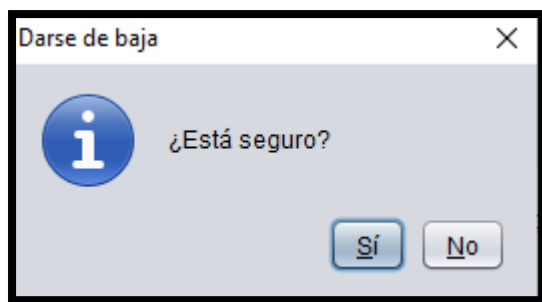
Igual que la de jugador, aunque ahora cualquier miembro del cuerpo técnico puede solicitar renovación.

Interfaz Socio



Interfaz de usuario para el Socio. El panel tiene un fondo gris claro y una barra superior con el título "SOCIO" a la derecha y un botón "Cerrar Sesión" a la izquierda. En el centro del panel, hay una columna vertical de cinco botones rectangulares con bordes redondeados y un efecto de sombra. Los botones, desde arriba hacia abajo, son: "Consultar plantilla", "Modificar mis datos personales", "Darme de baja", "Ver líderes de la liga" y "Ver clasificación".

Tendrá las funciones comunes más darse de baja. Esta función elimina al socio directamente de la base de datos y cierra la sesión:



Interfaz Registro Socio

A registration form with a light gray background. At the top, the text "Usuario" is above a white text input field. Below that, the text "Contraseña" is above another white text input field. Under the password field is a button labeled "Login". At the bottom right, there is a link "¿No eres socio aún?" followed by a button labeled "Hacerse socio".

Pulsando este botón aparece la siguiente interfaz:

A form for club subscription with a light gray background. It contains several input fields: "DNI", "Nombre", "Apellidos", "E-mail", and "Contraseña", each followed by a white text input field. Below these is a dropdown menu for "Equipo" with a blue arrow icon. At the bottom center is a button labeled "Abonarse al club".

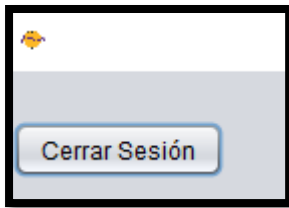
Introducirás tus datos personales y el equipo al que te quieres abonar.

Botones volver y cerrar sesión

Todas las interfaces contendrán botón de volver a la interfaz anterior:



Cada interfaz principal, tendrá un botón de cerrar sesión que redirige a la interfaz del Login:



Iconos

Habrà un icono genérico en la venta del Login:



Cada equipo dispondrá de su propio icono. Cada vez que un usuario de un equipo inicie sesión, las ventanas tendrán el icono del equipo del usuario registrado. Algunos de los iconos de los equipos son los siguientes:

