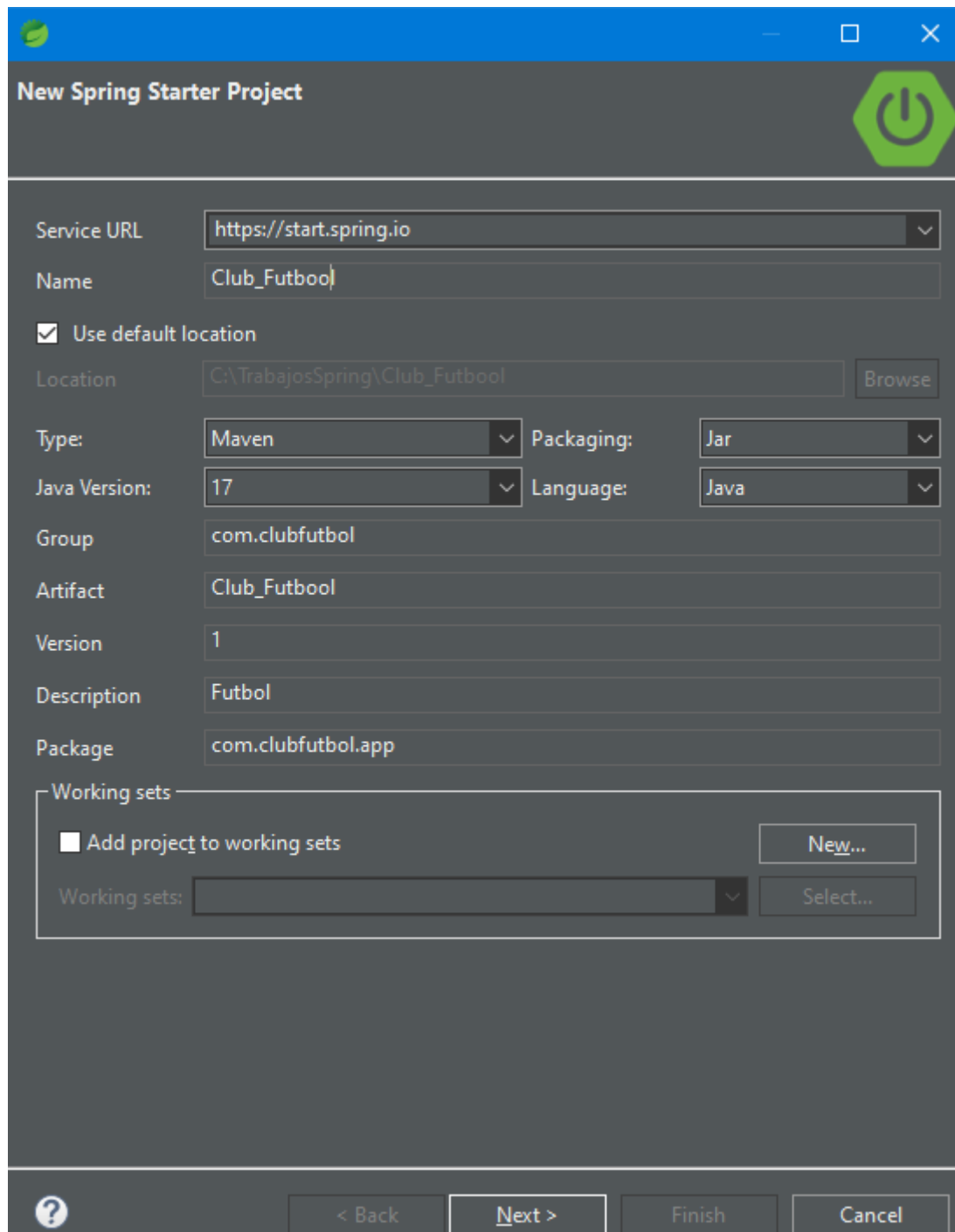


MANUAL PROYECTO CLUB DE FUTBOL

Creación del proyecto



The screenshot shows the 'New Spring Starter Project' dialog box in the Eclipse IDE. The dialog has a blue title bar with the Eclipse logo and standard window controls. The main area is dark gray with a green power button icon in the top right corner. The fields are as follows:

- Service URL:** A dropdown menu showing 'https://start.spring.io'.
- Name:** A text field containing 'Club_Futbol'.
- Use default location:** A checked checkbox.
- Location:** A text field showing 'C:\TrabajosSpring\Club_Futbol' with a 'Browse' button to its right.
- Type:** A dropdown menu showing 'Maven'.
- Packaging:** A dropdown menu showing 'Jar'.
- Java Version:** A dropdown menu showing '17'.
- Language:** A dropdown menu showing 'Java'.
- Group:** A text field containing 'com.clubfutbol'.
- Artifact:** A text field containing 'Club_Futbol'.
- Version:** A text field containing '1'.
- Description:** A text field containing 'Futbol'.
- Package:** A text field containing 'com.clubfutbol.app'.

Below these fields is a section titled 'Working sets' with a collapsed arrow. It contains:

- Add project to working sets:** An unchecked checkbox.
- New...:** A button to the right of the checkbox.
- Working sets:** A dropdown menu showing no selection.
- Select...:** A button to the right of the dropdown menu.

At the bottom of the dialog is a footer bar with a help icon (question mark) on the left and four buttons: '< Back', 'Next >', 'Finish', and 'Cancel'.

Seleccionamiento de dependencias



The image shows a 'New Spring Starter Project Dependencies' dialog box. At the top, it has a title bar with a green icon and standard window controls. Below the title bar, the title 'New Spring Starter Project Dependencies' is displayed next to a green power button icon. The main area is divided into sections. The first section, 'Spring Boot Version:', has a dropdown menu set to '3.0.0'. Below this is a 'Frequently Used:' section containing a grid of nine checkboxes, all of which are checked: H2 Database, MySQL Driver, Spring Boot DevTools, Spring Data JPA, Spring Data MongoDB, Spring Security, Spring Web, Thymeleaf, and Validation. The next section is split into two panes. The left pane, titled 'Available:', contains a search bar with the placeholder text 'Type to search dependencies' and a list of dependency categories with expandable arrows: Developer Tools, Google Cloud Platform, I/O, Messaging, Microsoft Azure, NoSQL, Observability, Ops, SQL, Security, Spring Cloud, Spring Cloud Circuit Breaker, Spring Cloud Config, and Spring Cloud Discovery. The right pane, titled 'Selected:', contains a list of the selected dependencies, each preceded by an 'X' mark: Spring Boot DevTools, Validation, Spring Data MongoDB, Spring Data JPA, H2 Database, MySQL Driver, Spring Security, Thymeleaf, and Spring Web. At the bottom of the 'Selected:' list are two buttons: 'Make Default' and 'Clear Selection'. The bottom of the dialog box features a footer with a help icon (a question mark in a circle) on the left and four buttons: '< Back', 'Next >', 'Finish', and 'Cancel'.

Spring Boot Version: 3.0.0

Frequently Used:

- ☒ H2 Database
- ☒ MySQL Driver
- ☒ Spring Boot DevTools
- ☒ Spring Data JPA
- ☒ Spring Data MongoDB
- ☒ Spring Security
- ☒ Spring Web
- ☒ Thymeleaf
- ☒ Validation

Available:

Type to search dependencies

- ▶ Developer Tools
- ▶ Google Cloud Platform
- ▶ I/O
- ▶ Messaging
- ▶ Microsoft Azure
- ▶ NoSQL
- ▶ Observability
- ▶ Ops
- ▶ SQL
- ▶ Security
- ▶ Spring Cloud
- ▶ Spring Cloud Circuit Breaker
- ▶ Spring Cloud Config
- ▶ Spring Cloud Discovery

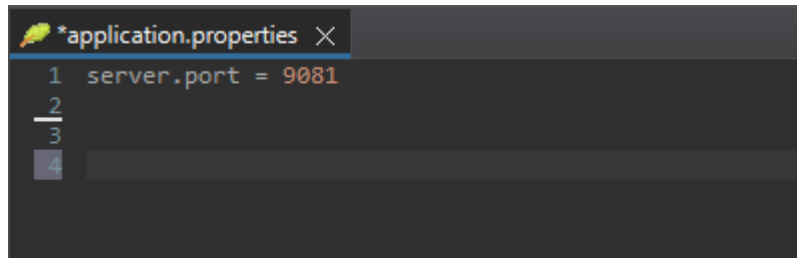
Selected:

- X Spring Boot DevTools
- X Validation
- X Spring Data MongoDB
- X Spring Data JPA
- X H2 Database
- X MySQL Driver
- X Spring Security
- X Thymeleaf
- X Spring Web

Make Default Clear Selection

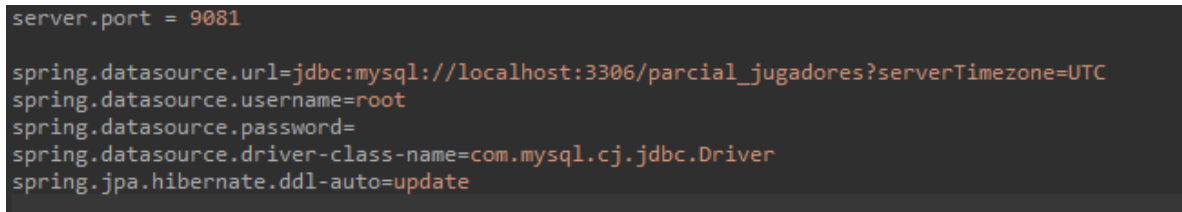
< Back Next > Finish Cancel

Configuración del application.properties (puerto)

A screenshot of an IDE window titled '*application.properties'. The file contains the following content:

```
1 server.port = 9081
2
3
4
```

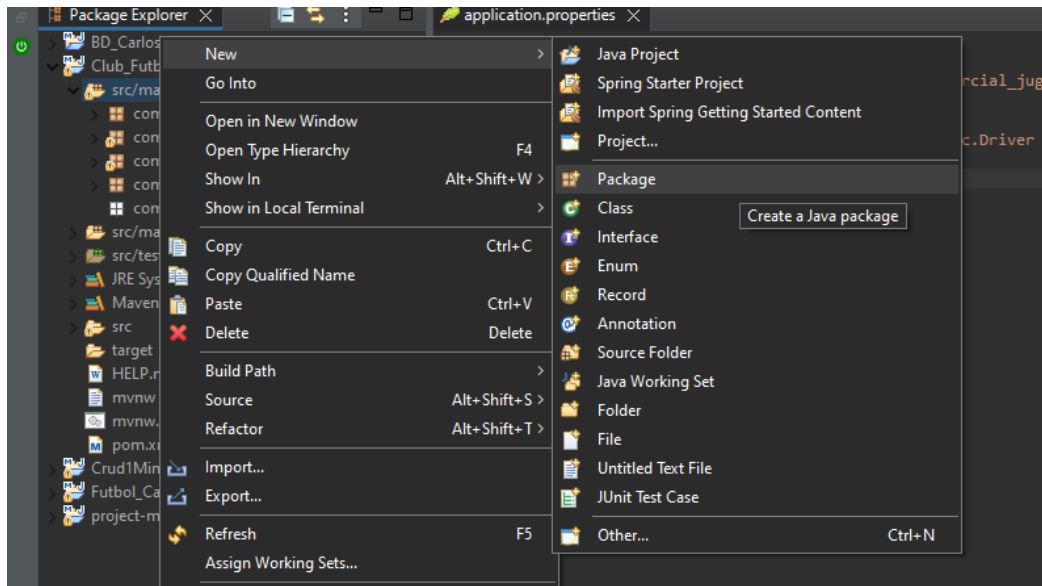
Configuración del application.properties (mySQL)

A screenshot of an IDE window showing the configuration for a MySQL database in application.properties. The content is as follows:

```
server.port = 9081

spring.datasource.url=jdbc:mysql://localhost:3306/parcial_jugadores?serverTimezone=UTC
spring.datasource.username=root
spring.datasource.password=
spring.datasource.driver-class-name=com.mysql.cj.jdbc.Driver
spring.jpa.hibernate.ddl-auto=update
```

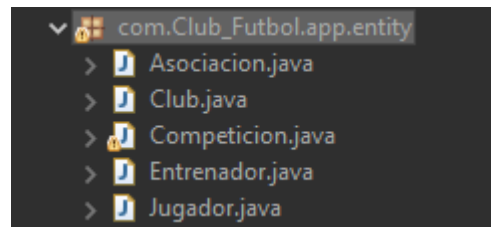
Creación de packages



Nuestro proyecto cuenta con 3 packages muy importantes los cuales son: entity, controller y repository



Creacion de clases (entity)



New Java Class

Java Class

Type already exists.

Source folder:

Package:

☐ Enclosing type:

Name:

Modifiers: ☐ public ☐ package ☐ private ☐ protected
☐ abstract ☐ final ☐ static
☐ none ☐ sealed ☐ non-sealed ☐ final

Superclass:

Interfaces:

Which method stubs would you like to create?

☐ public static void main(String[] args)

☐ Constructors from superclass

☒ Inherited abstract methods

Do you want to add comments? (Configure templates and default value [here](#))

☐ Generate comments

En el package entity encontraremos las clases para cada entidad o para cada objeto, un ejemplo de esto seria la clase de Jugador:

```
package com.Club_Futbol.app.entity;

import javax.persistence.*;

@Entity
public class Jugador {

    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private Integer id;
    private String nombre;
    private String apellido;
    private String numero;
    private String posicion;

    @ManyToOne
    @JoinColumn(name = "club_id")
    private Club club;

    public Jugador() {
        super();
        // TODO Auto-generated constructor stub
    }

    public Jugador(Integer id, String nombre, String apellido, String numero, String posicion, Club club) {
        super();
        this.id = id;
        this.nombre = nombre;
        this.apellido = apellido;
        this.numero = numero;
        this.posicion = posicion;
        this.club = club;
    }

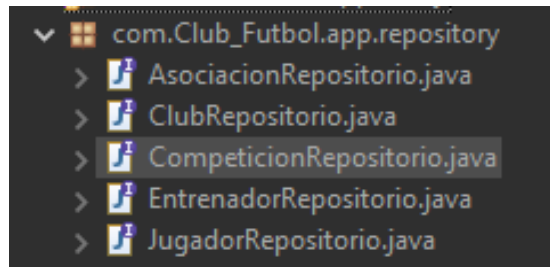
    public Jugador(String nombre, String apellido, String numero, String posicion, Club club) {
        super();
        this.nombre = nombre;
        this.apellido = apellido;
        this.numero = numero;
        this.posicion = posicion;
        this.club = club;
    }

    public Jugador(String nombre) {
        super();
        this.nombre = nombre;
    }

    public Club getClub() {
        return club;
    }

    public void setClub(Club club) {
        this.club = club;
    }
}
```

Creacion de clases (repository)



New Java Class

Java Class
Create a new Java class.

Source folder:

Package:

☐ Enclosing type:

Name:

Modifiers:
☐ public ☐ package ☐ private ☐ protected
☐ abstract ☐ final ☐ static
☐ none ☐ sealed ☐ non-sealed ☐ final

Superclass:

Interfaces:

Which method stubs would you like to create?

☐ public static void main(String[] args)

☐ Constructors from superclass

☒ Inherited abstract methods

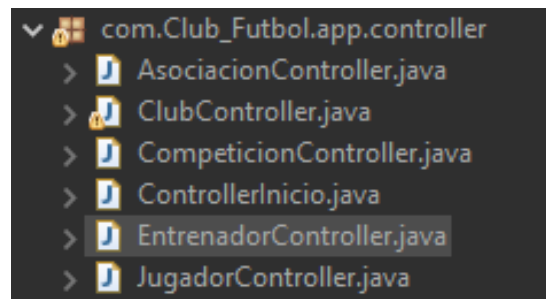
Do you want to add comments? (Configure templates and default value [here](#))

☐ Generate comments

En el package Repository estarán las clases que hacen el funcionamiento de interfaces de cada una de las clases de entity, implementamos los métodos de JPA Repository y se le asigna un objeto de Jugador y de ID

```
Jugador.java  JugadorRepositorio.java X
1 package com.Club_Futbol.app.repository;
2
3 import org.springframework.data.jpa.repository.JpaRepository;
4
5
6
7 public interface JugadorRepositorio extends JpaRepository<Jugador, Integer>{
8
9 }
10
```


Creacion de clases (controller)



New Java Class

Java Class

Create a new Java class.



Source folder:

Club_Futbol/src/main/java

Browse...

Package:

com.Club_Futbol.app.controller

Browse...

☐ Enclosing type:

Browse...

Name:

JugadorCoontroller

Modifiers:

☐ public

☐ package

☐ private

☐ protected

☐ abstract

☐ final

☐ static

☐ none

☐ sealed

☐ non-sealed

☐ final

Superclass:

java.lang.Object

Browse...

Interfaces:

Add...

Remove

Which method stubs would you like to create?

☐ public static void main(String[] args)

☐ Constructors from superclass

☒ Inherited abstract methods

Do you want to add comments? (Configure templates and default value [here](#))

☐ Generate comments



Finish

Cancel

En el package controller estarán las clases que básicamente llama los métodos de la interface y por ende le pasa el objeto de jugador, ya sea para eliminarlo, para listarlo, para editarlo o para guardar el nuevo objeto en la base de datos

```
package com.Club_Futbol.app.controller;

import java.util.List;

@Controller
public class JugadorController {

    @Autowired
    private JugadorRepositorio jugadorRepositorio;

    @Autowired
    private ClubRepositorio clubRepositorio;

    @GetMapping("/verJugador/formJugador")
    public String insertarJugador (Model model) {

        List<Club> listaClub = clubRepositorio.findAll();

        model.addAttribute("jugador", new Jugador());
        model.addAttribute("listaClub", listaClub);

        return "formJugador";
    }

    @PostMapping("/guardarJugador")
    public String guardarJugador (Jugador jugador) {
        jugadorRepositorio.save(jugador);
        return "redirect:/verJugador";
    }

    @GetMapping("/verJugador")
    public String listarJugador (Model model) {
        List<Jugador> listaJugador = jugadorRepositorio.findAll();
        model.addAttribute("listaJugador", listaJugador);
        return "verJugador";
    }
}
```

Insertamos las APIS que usaremos en el proyecto en el archivo llamado pom.xml, las APIS nos ayudan para facilitarnos ciertas cosas, como por ejemplo el Login

```
<dependencies>
    <dependency>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-starter-data-jpa</artifactId>
    </dependency>
```

```
<dependency>
    <groupId>com.github.librepdf</groupId>
    <artifactId>openpdf</artifactId>
    <version>1.3.28</version>
</dependency>
```

```
<dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-thymeleaf</artifactId>
</dependency>
```

```
<dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-validation</artifactId>
</dependency>
```

```
<dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-web</artifactId>
</dependency>
```

```
<dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-devtools</artifactId>
    <scope>runtime</scope>
    <optional>true</optional>
</dependency>
```

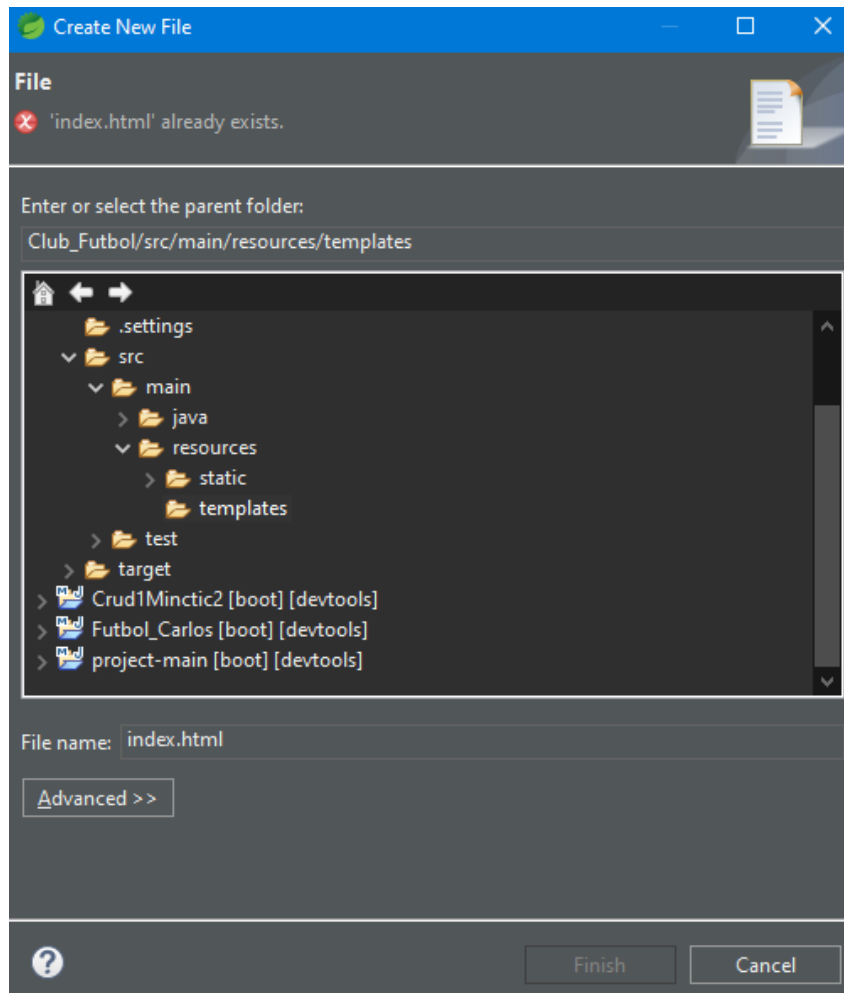
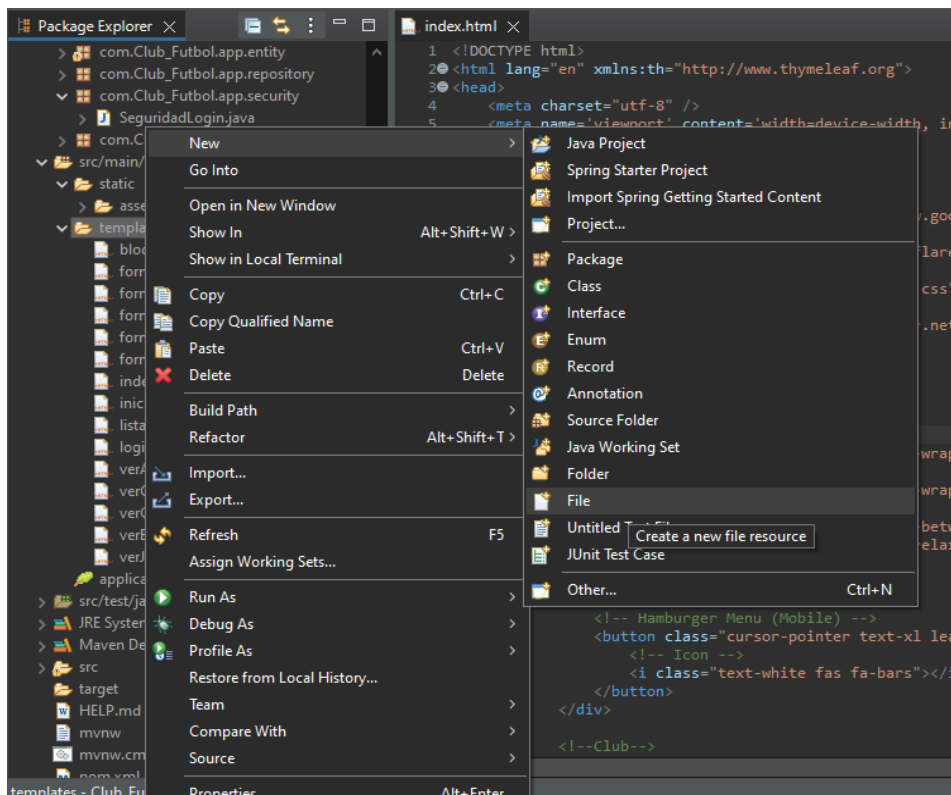
```
<dependency>
    <groupId>com.mysql</groupId>
    <artifactId>mysql-connector-j</artifactId>
    <scope>runtime</scope>
</dependency>
```

```
<dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-test</artifactId>
    <scope>test</scope>
</dependency>
```

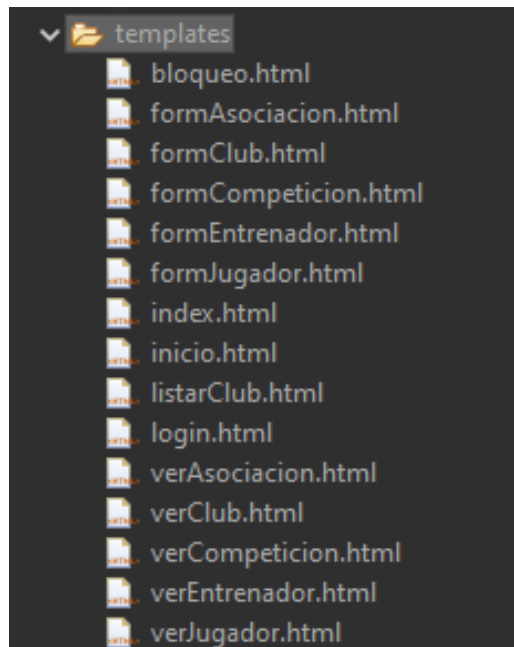
Configuración del application.properties (pdf)

```
#PDF
spring.mvc.contentnegotiation.favor-parameter=true
spring.mvc.contentnegotiation.media-types.pdf=application/pdf
```

Creación de las páginas html del proyecto



En la carpeta llamada templates estarán todas las paginas html, para este proyecto necesitamos de 15 paginas



Las paginas html son las que el usuario interactúa en el momento en el que esta navegando en la pagina del Club de futbol, cabe aclarar que cada página contiene Bootstrap para así hacer un diseño más estético para el publico

```
index.html X
1 <!DOCTYPE html>
2 <html lang="en" xmlns:th="http://www.thymeleaf.org">
3 <head>
4   <meta charset="utf-8" />
5   <meta name="viewport" content="width=device-width, initial-scale=1">
6   <meta name="theme-color" content="#000000" />
7   <title>Club de Futbol</title>
8
9   <!-- Favicon -->
10  <link rel="icon" type="img/png" href="https://www.google.com/s2/favicons?domain=lostpenart.com">
11  <!-- Font Awesome -->
12  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.10.2/css/all.min.css"/>
13  <!-- AOS CSS -->
14  <link href="https://unpkg.com/aos@2.3.1/dist/aos.css" rel="stylesheet">
15  <!-- Tailwind Library -->
16  <link rel="stylesheet" href="https://cdn.jsdelivr.net/gh/creativetimofficial/tailwind-starter-kit/compiled-tailwind.min.css" />
17
18 </head>
19
20 <body class="text-gray-800 antialiased bg-black">
21
22   <!-- Navigation -->
23   <nav class="top-0 absolute z-50 w-full flex flex-wrap items-center justify-between px-2 py-3 navbar-expand-lg">
24
25     <div class="container px-4 mx-auto flex flex-wrap items-center justify-between">
26       <!-- Inicio -->
27       <div class="w-full relative flex justify-between lg:w-auto lg:static lg:block lg:justify-start">
28         <a class="text-sm font-bold leading-relaxed inline-block mr-4 py-2 whitespace-no-wrap uppercase text-gray-300 href="#">
29           <!-- Nav Title -->
30           Inicio
31         </a>
32         <!-- Hamburger Menu (Mobile) -->
33         <button class="cursor-pointer text-xl leading-none px-3 py-1 border border-solid border-transparent rounded bg-transparent">
34           <!-- Icon -->
35           <i class="text-white fas fa-bars"></i>
36         </button>
37       </div>
38
39     <!-- Club -->
```

Página principal (index)

The screenshot shows the main page of a website for 'Academia de Futbol'. The background is a dark image of soccer players on a field. At the top left, there are links for 'INICIO' and 'INICIA SESION'. At the top right, there are social media icons for Facebook and Instagram. The main heading is 'Academia de Futbol' in a large, white, sans-serif font. Below the heading, there is a paragraph of text in Spanish: 'Contamos con alternativas dinámicas y entretenidas que propenden por la formación del deportista Integral a través de prácticas lúdicas, formativas y competitivas que contribuyen a mejorar las capacidades físicas, mentales, emocionales y socio afectivas de los deportistas.' Below this text, there are three white rectangular boxes with rounded corners, each containing an icon and a title. The first box has a red icon with a crown and the title 'Programa'. The second box has a blue icon with a gear and the title 'Objetivo'. The third box has a green icon with a lightning bolt and the title 'Grupo'. Each box contains a short paragraph of text in Spanish. The 'Programa' box mentions 'Se viene desarrollando desde el año 2008 la enseñanza del fútbol como instrumento forjador de paz, habilidades, destrezas y competencias en'. The 'Objetivo' box mentions 'Contribuir en la formación integral de niños y jóvenes de población vulnerable a través de la práctica del fútbol'. The 'Grupo' box mentions 'Niños, niñas y jóvenes que estén matriculados en algún grado de nivel de educación básica y media en un establecimiento educativo en convenio con la escuela.'

INICIO INICIA SESION

Academia de Futbol

Contamos con alternativas dinámicas y entretenidas que propenden por la formación del deportista Integral a través de prácticas lúdicas, formativas y competitivas que contribuyen a mejorar las capacidades físicas, mentales, emocionales y socio afectivas de los deportistas.

Programa

Se viene desarrollando desde el año 2008 la enseñanza del fútbol como instrumento forjador de paz, habilidades, destrezas y competencias en

Objetivo

Contribuir en la formación integral de niños y jóvenes de población vulnerable a través de la práctica del fútbol

Grupo

Niños, niñas y jóvenes que estén matriculados en algún grado de nivel de educación básica y media en un establecimiento educativo en convenio con la escuela.

Página de ingresar (Login)

The screenshot shows a login page with a dark background. In the center, there is a white rectangular box with rounded corners. Inside the box, the word 'Login' is written in a bold, black, sans-serif font. Below 'Login', there are two input fields. The first field is labeled 'Username' and the second field is labeled 'Password'. Both labels are in a small, black, sans-serif font. Below the input fields, there is a button labeled 'SUBMIT' in a small, black, sans-serif font. The button has a thin black border and a small shadow effect.

Login

Username

Password

SUBMIT

Página principal luego de logearse

[INICIO](#) [CLUB](#) [ASOCIACIONES](#) [JUGADORES](#) [COMPETICIONES](#) [ENTRENADOR](#) [CERRAR SESION](#)

Academia de Futbol

Contamos con alternativas dinámicas y entretenidas que propenden por la formación del deportista Integral a través de prácticas lúdicas, formativas y competitivas que contribuyen a mejorar las capacidades físicas, mentales, emocionales y socio afectivas de los deportistas.



Programa

Se viene desarrollando desde el año 2008 la enseñanza del fútbol como instrumento forjador de paz, habilidades, destrezas y competencias en



Objetivo

Contribuir en la formación integral de niños y jóvenes de población vulnerable a través de la práctica del fútbol



Grupo

Niños, niñas y jóvenes que estén matriculados en algún grado de nivel de educación básica y media en un establecimiento educativo en convenio con la escuela.

Listado de entrenadores

Entrenadores Registrados

Id	Nombre	Apellido	Edad	Nacionalidad	Acciones
1	Carlos	Espinosa	29	Colombiana	editar eliminar
2	Lessly	Barba	36	Colombiana	editar eliminar
3	Jose	García	30	Colombiana	editar eliminar

[Insertar Entrenador](#) [Inicio](#)

Listado de organización

Asociaciones Registradas

Id	Nombre	Pais	Presidente	Acciones
2	Asociacion de Futbol Argentino	Argentina	Claudio Fabian	editar eliminar
3	Asociacion de Futbol Colombiano	Colombia	Gionni Fantino	editar eliminar

[Insertar Asociacion](#) [Inicio](#)

Listado de club

Clubes Registrados				
Id	Nombre	Entrenador	Asociacion	Acciones
1	Santa Fe	Carlos	Asociacion de Futbol Colombiano	editar eliminar

[Insertar Club](#) [Inicio](#)

Listado de jugadores

Solicitud de datos:

Nombre:
Kevin

Apellidos:
Castellanos

Numero:
1

Posicion:
Portero

Club:
Santa Fe



Enviar

Restablecer

Cancelar

Jugadores Registrados

Id	Nombre	Apellido	Numero	Posicion	Club	Acciones	
1	Wilson	Morelo	15	Delantero	Santa Fe	editar	eliminar
2	Sebastian	Gomez	3	Defensa	Santa Fe	editar	eliminar
3	Kevin	Castellanos	1	Portero	Santa Fe	editar	eliminar

Insertar Jugador

Inicio

Listado PDF



1 / 1 | - 100% + |



Id	Nombre	Apellido	Numero	Posicion
1	Wilson	Morelo	15	Delantero
2	Sebastian	Gomez	3	Defensa
3	Kevin	Castellanos	1	Portero