Contenido

[1. INTRODUCCIÓN. 3](#_Toc166679795)

[2. ESPECIFICACIÓN DE REQUISITOS. 3](#_Toc166679796)

[3. ANÁLISIS. 3](#_Toc166679797)

[4. DISEÑO 3](#_Toc166679798)

[5. IMPLEMENTACIÓN 4](#_Toc166679799)

[6. EVALUACIÓN 4](#_Toc166679800)

[7. CONCLUSIÓN 4](#_Toc166679801)

# 1. INTRODUCCIÓN.

1.1 Presentación y Objetivos.

1.2 Contexto.

1.3 Planteamiento del Problema (la idea).

Un polideportivo nos ha contactado ya que actualmente tiene un problema de almacenamiento de documentos. Este problema de almacenamiento se debe a que realizaban sobre papel la documentación de reservas, de abonados y trabajadores.

1.4 Análisis de costes.

Costes operativos:

* Sueldos de los trabajadores.
* Alquiler de la oficina.
* Marketing de la empresa.

Costes de desarrollo:

* Costes de hardware.
* Costes de software.

Costes de expansión:

* Costes de marketing.
* Creación de nuevas oficinas.

1.5 Plan de Financiación.

Financiación interna:

* Reinversión de ganancias.
* Inversiones con ahorros personales.

Financiación externa:

* Préstamos bancarios.
* Subvenciones y fondos públicos.

1.6 Plan de recursos humanos.

Reclutamiento y selección de personal:

* Análisis de necesidades a corto y largo plazo.
* Realizar una estrategia de reclutamiento mediante redes sociales y portales especializados.
* Proceso de selección en el que hagamos un filtro de currículos

Bienestar y retención:

* Incentivar el bienestar físico y mental de los trabajadores.
* Tener un ambiente de trabajo positivo.
* Dar oportunidades de crecimiento a nuestros trabajadores.

Gestión de desempeño:

* Realizar un seguimiento del desempeño de los trabajadores.
* Ofrecer bonificaciones o aumentos salariales según su desempeño.

1.7 Plan de prevención de riesgos.

Posibles riesgos laborales:

* Riegos físicos: problemas posturales, fatiga visual y riesgos eléctricos.
* Riesgos psicosociales: estrés laboral, sobrecarga de trabajo, agotamiento emocional y trabajo bajo presión.
* Riesgos ergonómicos: equipos no adecuados.
* Riesgos ambientales: iluminación inadecuada y exceso de ruidos.

Medidas de prevención:

* Riesgos físicos: implementar la regla 20-20-20 (cada 20 minutos, mirar algo a 20 pies de distancia por 20 segundos). Asegurar el uso adecuado de cables y dispositivos eléctricos. Realizar inspecciones regulares de equipos eléctricos para detectar y reparar fallos.
* Riesgos psicosociales: asegurar una carga de trabajo equilibrada y realista. Establecer plazos razonables y realistas para proyectos y tareas. Fomentar un equilibrio saludable entre el trabajo y la vida personal. Proveer acceso a asesoramiento y recursos de salud mental.
* Riesgos ergonómicos: proveer mobiliario ergonómico (sillas ajustables, escritorios a la altura adecuada). Configurar correctamente las estaciones de trabajo (monitores a la altura de los ojos, teclados y ratones ergonómicos). Promover pausas regulares y ejercicios de estiramiento.
* Riesgos ambientales: asegurar una iluminación adecuada y ajustable en todas las áreas de trabajo. Reducir el deslumbramiento y los reflejos en las pantallas. Controlar los niveles de ruido en el entorno de trabajo para que no interfieran con la concentración. Proveer zonas silenciosas para tareas que requieren alta concentración.

Capacitación y concienciación:

* Realización de programas de capacitación regular sobre ergonomía, gestión del estrés, etc.
* Desarrollar campañas de concienciación sobre la importancia de tomar pausas regulares y practicar ejercicios de estiramiento.

# 2. ESPECIFICACIÓN DE REQUISITOS.

2.1 Introducción.

2.2 Descripción general.

Nuestro cliente es el Ayuntamiento de Llanes, que necesita una aplicación de escritorio para el Polideportivo del pueblo. La aplicación debe permitir las diferentes gestiones de abonados, y las pistas con sus respectivas reservas para las horas en las que está abierto el polideportivo (desde las 9 de la mañana hasta las 9 de la noche). Debido a la gran cantidad de reservas y abonados que llega a haber a lo largo del tiempo, se precisa de informatizar la información, ya que anteriormente esta gestión se hacía sobre papel y a largo plazo es inviable. También desean actualizarse y añadir una forma de que sus abonados puedan realizar las reservas online, ya que actualmente la gran mayoría de polideportivos ofrecen este servicio. Además, el cliente quiere tener almacenados los datos de sus empleados.

2.3 Requisitos Específicos.

2.3.1 Requerimientos Funcionales.

Posibilidad de realizar reservas online.

Mostrar los tipos de pistas que se pueden reservar.

Realizar un registro o login.

2.3.2 Requerimientos de Interfaces Externas.

2.3.2.1 Interfaces de los Usuario.

Ha de ser amigable con el usuario.

Ha de ser amigable independientemente de la edad del usuario.

2.3.2.2 Interfaces Hardware.

No se necesita ninguna interfaz hardware.

2.3.2.3 Interfaces Software.

Para desplegar la página web en local, se necesitaría lo siguiente:

* Postgresql para la BBDD.
* jdk22 para Java.
* SpringTools para desplegar el servidor local ya que ofrece muchas facilidades.
* Cualquier navegador para visualizar la página web.

2.3.2.4 Interfaces de Comunicaciones.

Se despliega en local.

2.3.3 Requerimientos de Rendimiento.

2.3.4 Obligaciones del Diseño.

2.3.4.1 Estándares Cumplidos.

2.3.4.2 Limitaciones Hardware.

2.3.5 Atributos.

2.3.5.1 Seguridad.

* Encriptación de la contraseña con Spring Security.
* Gestión del acceso a rutas de la página web con Spring Security.

2.3.5.2 Facilidades de Mantenimiento.

La página web está desarrollada con thymeleaf, ya que tiene una buena integración con Spring Boot. Además la conexión con la base de datos, se realiza con Spring Boot, un framework muy utilizado de Java, al ser tan utilizado esto hace que se implementen nuevas funcionalidades para facilitar el mantenimiento.

2.3.5.3 Portabilidad.

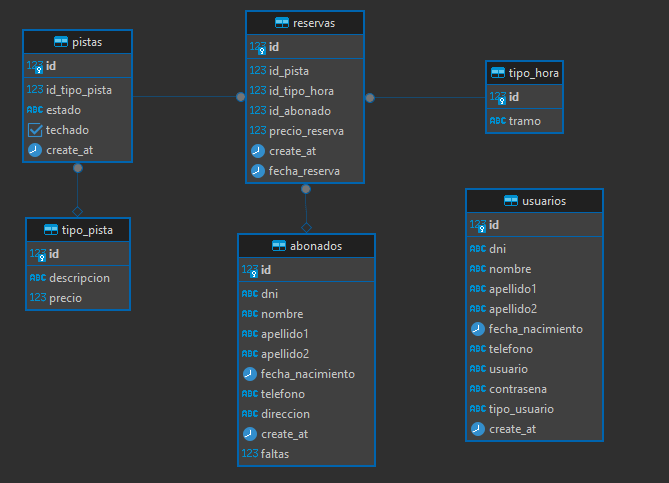
2.3.5.4 Otros Requerimientos.

# 3. ANÁLISIS.

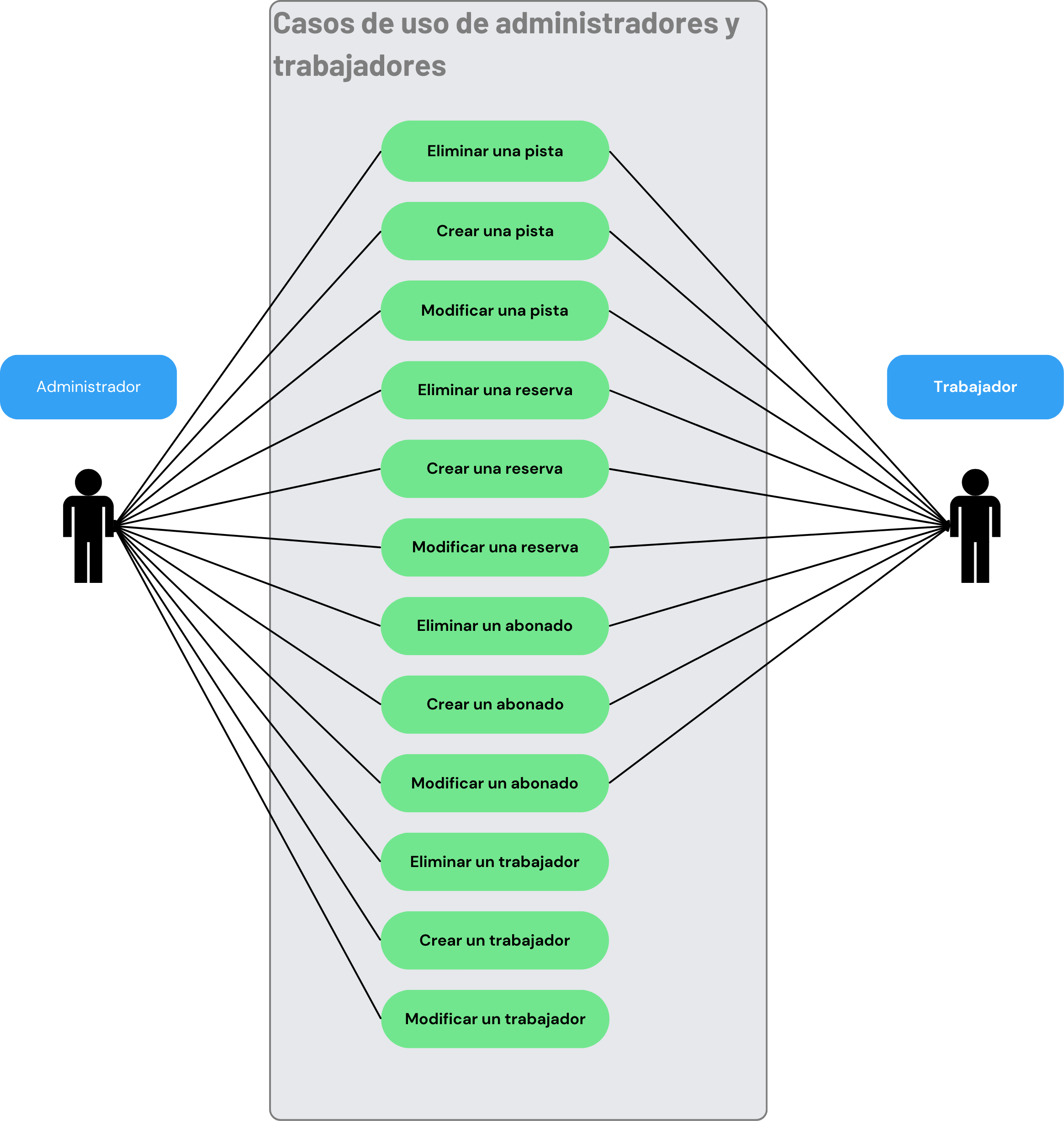
3.1 Introducción.

En los siguientes apartados (3.2 y 3.3), mostraremos un diagrama de clases en el que mostraremos las entidades utilizadas en el desarrollo de la página web. Además de un diagrama de casos de uso en el que mostramos las diferentes operaciones que pueden realizar los tres tipos de usuarios que pueden acceder a la página web que son: abonados, trabajadores y administradores.

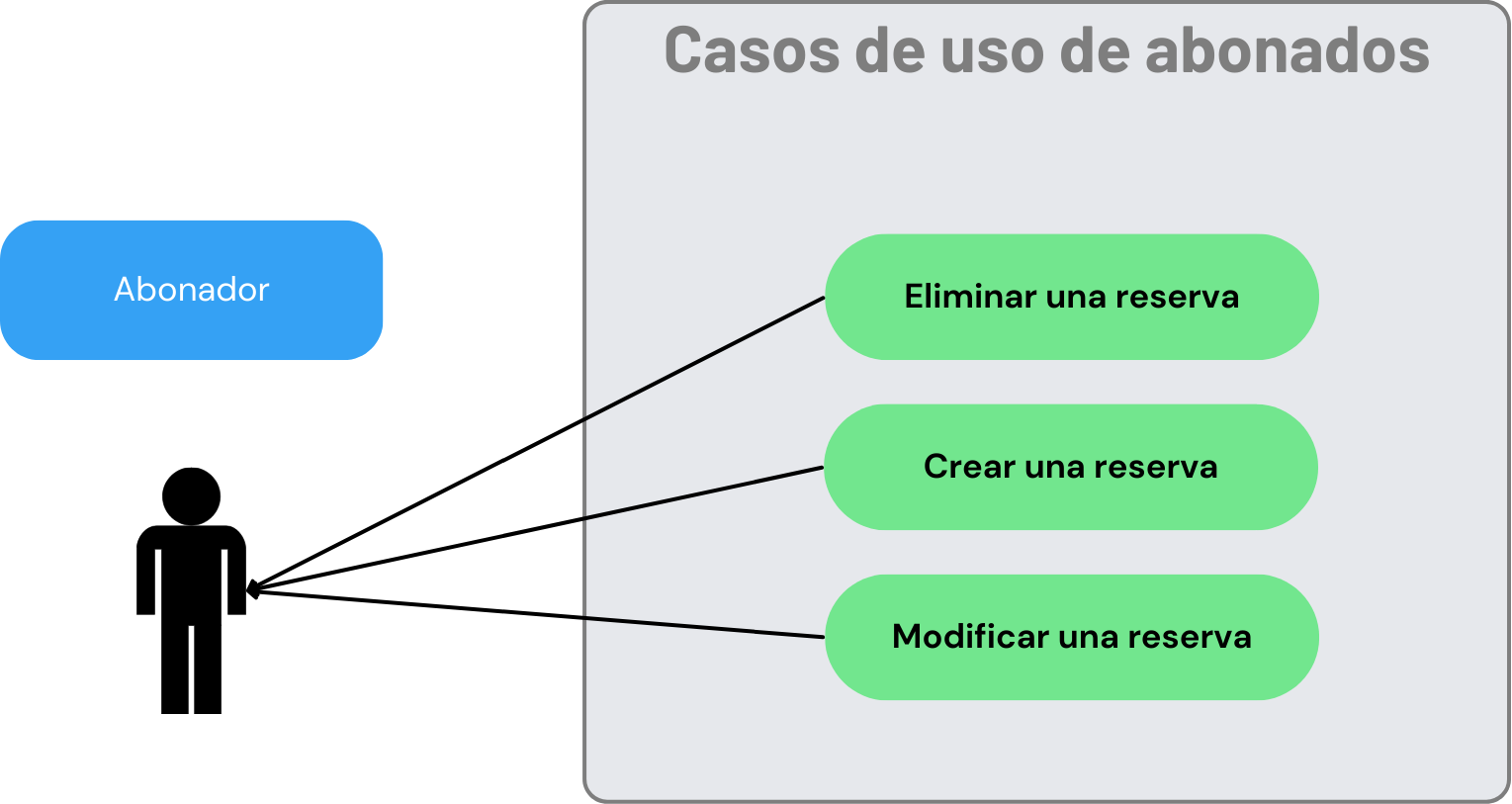
3.2 Diagrama de Clases.



3.3 Diagrama de Casos de Uso.

Casos de uso para administradores y trabajadores:

Casos de uso para abonados:



# 4. DISEÑO

4.1 Introducción.

4.2 Capa de Presentación.

4.3 Capa de Negocio o Lógica de la Aplicación.

4.4 Capa de Persistencia o Datos.

# 5. IMPLEMENTACIÓN

5.1 Tecnologías utilizadas en el desarrollo del proyecto

Para la creación y gestión de la base de datos, he utilizado postgresql, para la conexión con la base de datos y enviar datos al font, he utilizado spring, para el desarrollo del front, he utilizado thymeleaf y para los estilos de la página web, he utilizado bootstrap.

5.2 Descripción del Proyecto

5.2.1 Capa de Presentación.

Para la capa de presentación, he utilizado thymeleaf, ya que tiene una buena integración con Spring Boot y facilita el código html con la posibilidad de realizar comparaciones mediante th:if, bucles con th:each, mapear objetos con formularios mediante th:object y th:field y muchas más opciones.

5.2.2 Capa de Negocio o Lógica de la Aplicación.

Para la lógica de la aplicación, he utilizado Spring Boot, ya que es un framework muy utilizado, tiene mucha documentación y es fácil encontrar la solución a varios errores “típicos” que se pueden dar durante el desarrollo del software, además de que simplifica significativamente la serpación de las diferentes capas de la api, como por ejemplo: el controlador que se comunica con la vista del repositorio que se comunica con la bd. Además, existen una gran variedad de dependencias maven, con las cuales se simplifican la implementación de getter, setter y toString con lombok, el mapeo de clases a objetos dto (Data Transfer Object) con mapstruct, la implementación de un swagger para la documentación de los EndPoints y de seguridad para controlar el acceso a esos EndPoints y cifrado de contraseñas con Spring Security.

5.2.3 Capa de Persistencia o de Datos.

Para el desarrollo del la capa de datos, he utilizado postgresql, ya que contiene un servidor propio con el que no requieres de xampp y al cual puedes realizar la conexión. Además de haber hecho un cursillo con el cual he profundizado en su uso por lo que se me simplifica el poder manejarlo.

# 6. EVALUACIÓN

6.1 Introducción.

6.2 Validaciones de páginas de Estilo.

6.3 Validación de Enlaces.

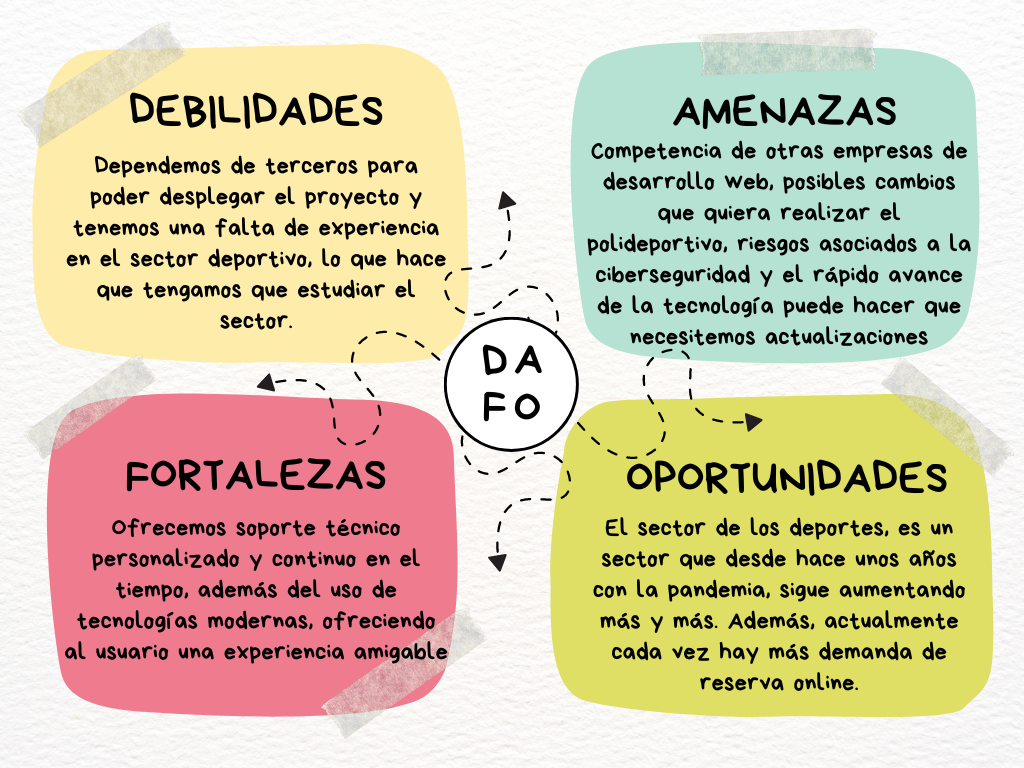
6.4 Validación de la Resolución.

6.5 Validación de Navegadores.

6.6 Otras validaciones.

# 7. CONCLUSIÓN

7.1 Valoración Personal del Trabajo Realizado (análisis DAFO + análisis CAME).

Análisi DAFO:

Análisis CAME:

7.2 Posibles Ampliaciones.

Posibilidades de ampliación:

* Añadir una confirmación mediante correo electrónico de la reserva realizada.
* Añadir un registro de abonados con un método de pago.
* Añadir la posibilidad del pago de la reserva de forma online.