



PROYECTO GRADO

PLAZAFIT

Documentación del proyecto del grado
superior de Desarrollo de Aplicaciones
Web.

Alberto Romero Rubiales

I.E.S. Francisco Romero Vargas
Desarrollo de Aplicaciones Web
Curso: 2023/2024

ÍNDICE

2.Explicación:	3
3.Finalidad	3
4.Objetivos	4
5.Medios necesarios	5
6.Planificación	5
7.Trabajos realizados	7
8.Realización del Proyecto.....	8
9.Instalación de paquetes.....	9
10.Explicación manual usuario	13
11.Problemas encontrados	16
12.Modificaciones sobre el proyecto planteado inicialmente	17
13.Posibles mejoras al proyecto	17
14.Bibliografía	18

1.Introducción

En este apartado explicaré de qué va a ir el proyecto. La idea de esta introducción es que cualquier persona que la lea sea capaz de entender el proyecto que se va a realizar.

2.Explicación:

En este anteproyecto damos a explicar el proyecto que se quiere realizar, el cual, es el siguiente. Queremos realizar una aplicación donde se pueda coger y agendar citas de un lugar específico, donde cada usuario se registre y una vez registrado poder coger cita para un día y hora en concreto, a la vez que modificar o eliminar la cita. Por otra parte, el administrador de la empresa puede ver, modificar, eliminar y añadir citas también. Para la realización de esta aplicación necesitaremos utilizar página HTML, página CSS y JavaScript con biblioteca React.

3.Finalidad

La finalidad principal de este proyecto basado en React no es otro que proporcionar a las empresas una plataforma donde sus usuarios puedan coger una cita en un día y hora en concreto, al igual que el administrador pueda también realizar esas consultas.

- Optimización síncrona: Para implementar las vistas del calendario simples y rápidas, así como la navegación entre las diferentes vistas del calendario.

- Escalabilidad: El proyecto se desarrollará teniendo en cuenta la escalabilidad futura. En un futuro se podrán agregar opciones como recordatorios.

4.Objetivos

Los objetivos que queremos cumplir en la realización de este proyecto son los siguientes:

- Diseñar la interfaz de usuario: Crear un diseño que sea fácil de entender y navegar para los usuarios. Esto incluye la visualización del calendario y los controles para seleccionar fechas y horas disponibles.
- Mostrar la disponibilidad de citas: Mostrar la disponibilidad de citas y poder marcar los horarios disponibles con colores diferentes o agregar etiquetas para indicar la disponibilidad.
- Permitir a los usuarios reservar citas: Implementar la funcionalidad para que los usuarios puedan hacer clic en un horario disponible en el calendario y reservar una cita.
- Validar las citas: Agregar validaciones para evitar que los usuarios reserven citas en horarios que ya están ocupados o que no están dentro del horario de trabajo de la empresa.
- Almacenar las citas: Guardar la información de las citas reservadas en una base de datos o en algún otro sistema de almacenamiento para que la empresa pueda acceder a ella más tarde.
- Administrar las citas: Crear una interfaz de administración para que la empresa pueda ver todas las citas reservadas, cancelar citas si es necesario y realizar otras acciones relacionadas con la gestión de citas.

5. Medios necesarios

En este apartado indicaremos qué medios físicos (hardware) y los medios lógicos (software) vamos a necesitar, siendo estos los siguientes:

1. Hardware y Software:

- Computadora: Una computadora personal o portátil que sea adecuada al desarrollo de software, que esté lo más actualizada posible y con suficiente potencia.
- Sistema Operativo: El sistema operativo que utilizaremos será Windows.
- Entorno de Desarrollo: El entorno utilizado será Visual Studio Code con extensiones necesarias para el trabajo necesario.
- Recursos Gráficos: Imágenes o emoticonos relacionados con la relación del calendario y agenda de citas.
- Base de Datos: Para almacenar los datos de los usuarios, así como las citas de cada usuario e información. Para ello utilizaremos una base de datos, como puede ser SQLite.

6. Planificación

En este apartado vamos a poner el orden de realización del proyecto y el tiempo más o menos estimado que vamos a tardar en cada tarea. Estos tiempos son aproximados y pueden ser diferentes mientras se va realizando el proyecto.

- Semana 1:

Planificación e investigación sobre diferentes tipos de aplicaciones de citas.

Investigación de las tecnologías a utilizar.

Creación de las páginas del proyecto dentro del visual studio.

- Semana 2:

Descarga de la aplicación utilizada de la base de datos.

Diseño inicial de la estructura de la aplicación.

Creación del repositorio Proyecto para ir subiendo los diferentes cambios.

- Semana 3:

Inicio de creación de las páginas HTML.

Inicio de la creación de formularios para la inserción de la cita.

- Semana 4:

Inicio de la creación de las páginas CSS.

Terminación de las páginas HTML.

- Semana 5:

Implementación de la base de datos dentro del proyecto.

Creación de las entidades de la base de datos.

- Semana 6:

Implementación de JavaScript con React para ir añadiendo los usuarios a la base de datos.

Implementación de JavaScript con React dentro de las páginas de HTML.

- Semana 7:

Terminación de la implementación de React.

- Semana 8 :

Pruebas para garantizar el funcionamiento correcto de todas las funciones.

Comienzo de la documentación del proyecto.

- Semana 9:

Terminación de la documentación del proyecto.
Revisión y entrega del proyecto como su documentación.

Horas totales que se planifican para el proyecto: 68.

Estas horas pueden variar, por lo cual pueden ser un poco orientativas.

REALIZACIÓN DEL PROYECTO

7.Trabajos realizados

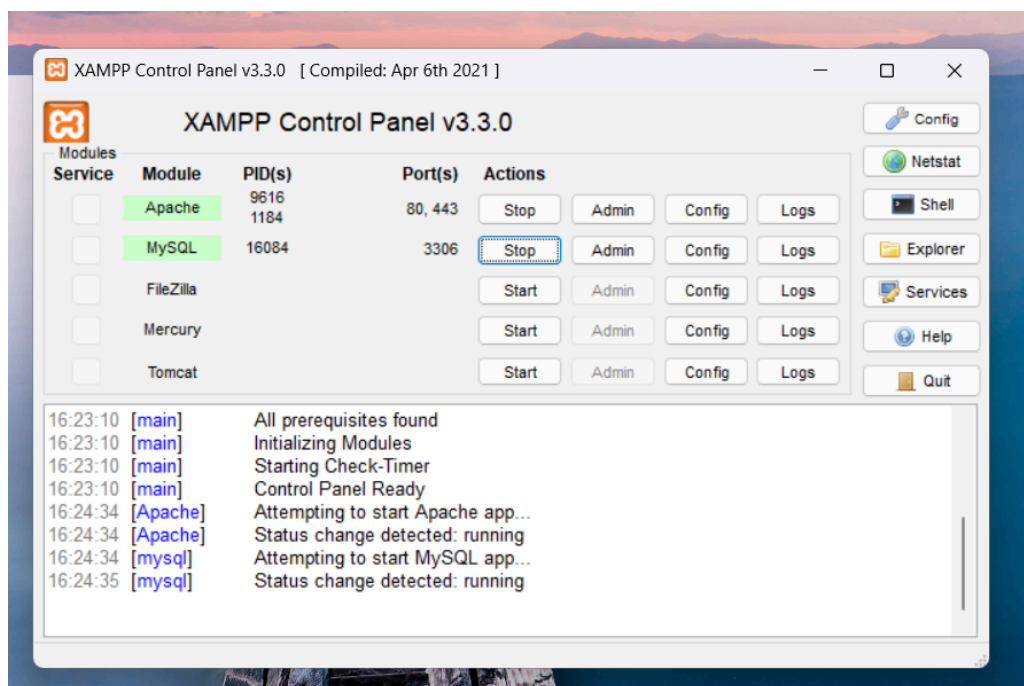
El proyecto que he realizado se basa en una página web de reservas de pistas dentro de un club de pádel. En esta página web se puede observar como es necesario estar registrado e iniciado la sesión para poder reservar una de estas pistas.

Es una página sencilla de descubrir e investigar ya que viene todo muy bien descrito y expuesto. Desde la página home, pasando por la página de reserva, hasta el cierre de sesión. También incluimos una zona de revisión de reservas del usuario en el cual puede cancelar su cita si así lo desea.

Para la realización del proyecto he necesitado utilizar la conjunción de los frameworks de Symfony y React.

8.Realización del Proyecto

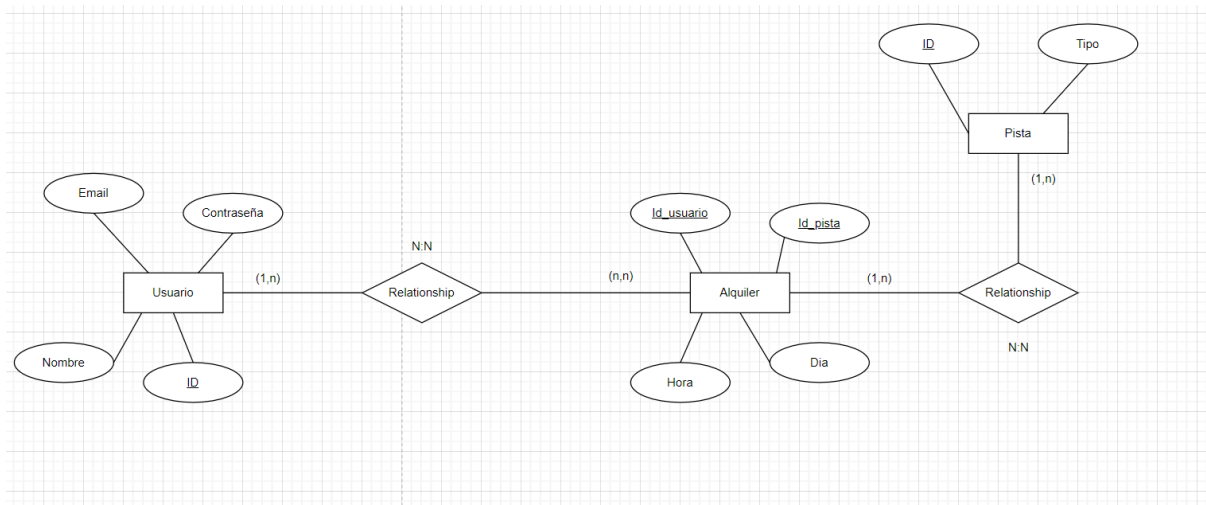
Antes de empezar con la digitalización del proyecto en lo primero que nos hemos centrado ha sido en la creación de una base de datos. Para ello he descargado XAMPP que es un paquete de software libre, que consiste principalmente en el sistema de gestión de bases de datos de MySQL.



Como podemos observar en la imagen debemos tener activo MySQL para que la base de datos tenga su funcionalidad y así no nos de errores a la hora de querer iniciar la aplicación.

Para esta base de datos que se ha creado para el proyecto lo primero que se hizo fue crear un esquema relación-entidad para tener una idea de como realizar la base. En este esquema se utilizaron las entidades **Usuario**, **Pista** y **Alquiler**. Estas tres entidades están relacionadas para que a la hora de que un usuario reserve una pista se guarde automáticamente en la propia base de datos, con los atributos de cada uno.

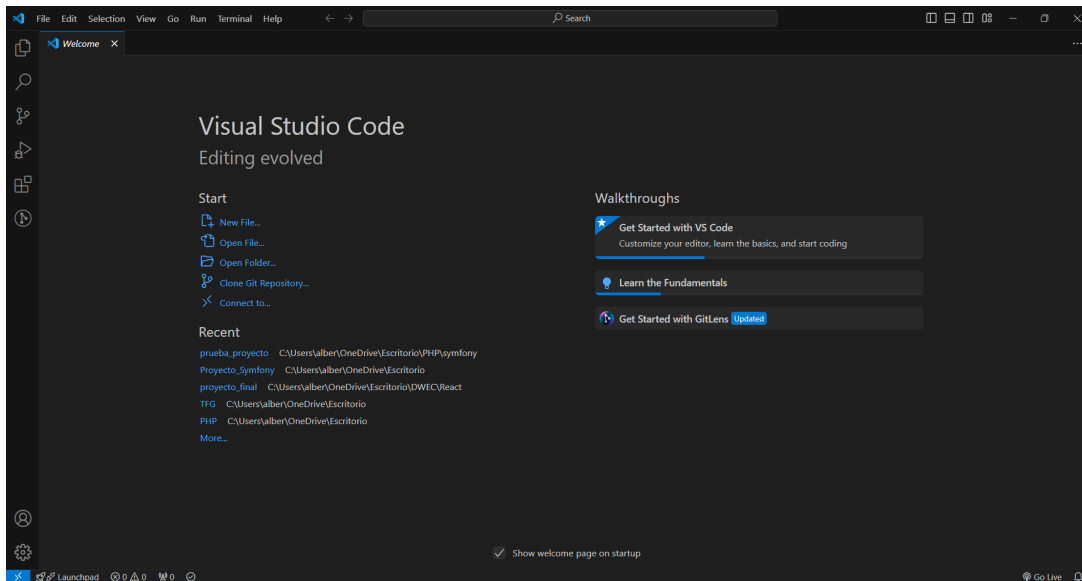
Para ser más claro pongo el esquema de entidad-relación en el cual me he ido guiando.



Como podemos observar en el esquema tenemos dos relaciones N:N entre usuario-alquiler y pista-alquiler, para que este último recoja la información del usuario y la pista que desea alquiler y así no habría problemas de confusiones ni doblajes a la hora de reservar.

9.Instalación de paquetes

Este proyecto ha necesitado de la descarga de diferentes paquetes para que su funcionamiento fuera óptimo y eficiente. Con estos paquetes que se mencionan más adelante hemos sido capaces de construir y elaborar las diferentes zonas que constituyen nuestra página web. Para la edición de código del proyecto se ha utilizado la plataforma **Visual Studio Code**, entorno con el que hemos trabajado durante el año en clase.



Para la utilización de symfony ha sido necesario instalar una serie de paquetes para su funcionamiento, los paquetes instalados son los mencionados aquí:

- composer create-project symfony:
 - Este comando es para generar el proyecto, donde se crean las carpetas y subcarpetas con la información base para el proyecto.
- composer require templates:
 - Los templates son las plantillas de diseño que se necesitan para mostrar la información.
- composer require symfony/debug-pack.
 - Este comando se utiliza para la instalación del depurador.
- composer require symfony/orm-pack.
 - Comando utilizado para los comandos de la base de datos.

Con este comando se nos creará un archivo .env en el que deberemos meter la línea:

```
DATABASE_URL="mysql://root@127.0.0.1:3306/[db_name]"
```

Esta línea es la que nos dejará conectar la base de datos con la aplicación.

- symfony console doctrine:database:create
 - Este comando es el utilizado para la creación de la base de datos, necesario para almacenar nuestros datos.
- composer require maker-bundle.
 - Instalar los comandos de creación de las entidades que tendremos en nuestra base de datos.
- symfony console make:entity
 - Este comando nos crea una entidad, cada vez que tengamos que crear una entidad deberemos meter este comando, al igual que los siguientes que se comentan.
- symfony console make:migration
 - Con este comando nos encargamos de hacer las migraciones en la base de datos.
- symfony console doctrine:migration:migrate
 - Y por último este comando se encarga de hacer la migración, es decir manda la entidad a la base de datos.

- composer require admin
 - Instalamos el backend del CRUD, el cual se encarga de poder crear, eliminar, modificar y leer.

- symfony console make:crud
 - Comando que crea el propio CRUD para su utilización.

- symfony console make:admin:dashboard
 - Crea el panel del administrador, desde el cual puede observar todos los cambios realizados, usuarios registrados y demás opciones que te deja el CRUD.

- composer require symfony/security-bundle
 - Comando para manejar la seguridad del proyecto, este paquete incluye security.yaml.

- symfony console make:user
 - Crea el usuario con su repository.

- composer require symfony/form
 - Genera un nuevo formulario.

- php bin/console make:registration-form
 - Comando que crea el controller de registro

- php bin/console make:security:form-login
 - Comando que crea el controller y el formulario de login

- composer require symfony/webpack-encore-bundle
 - Importante comando para la unión de symfony con react. Este comando crea una unión entre los dos frameworks, creando en el proyecto un archivo llamado *webpack.config.js* y la carpeta *assets* que es donde integraremos nuestra parte de react.

- symfony server:start y symfony server:stop
 - Y por último los comandos que se utilizan tanto para iniciar como cerrar el servidor.

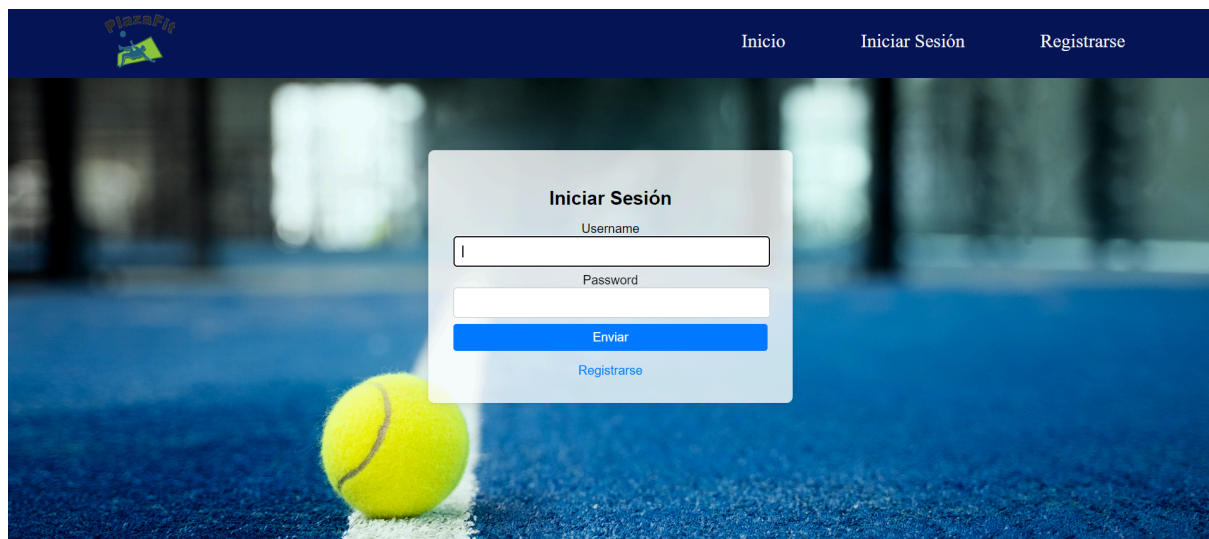
10.Explicación manual usuario

Todo esto que hemos estado mencionando ha sido lo necesario para construir la página web que se distribuye en diferentes partes que pasaremos a explicar lo más brevemente posible.

Al entrar en el inicio de la página lo primero con lo que nos encontramos es con el home, donde podremos ver nuestra navegación para el inicio de sesión o registro por parte del usuario. También dentro del home podemos observar una breve descripción de las funcionalidades que tiene la página web, es decir, te explicaremos que es un sitio de reservas con tan solo un par de click.

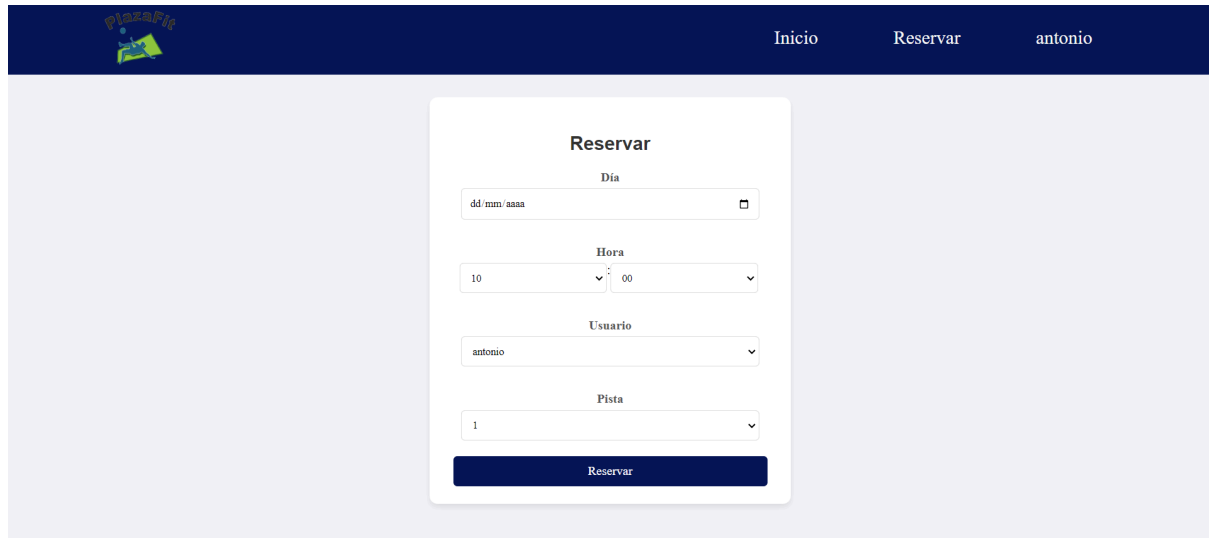


A la hora de la página del registro o inicio de sesión es sencilla con el simple formulario para que puedas ya bien sea registrarte o iniciar tu sesión. Una vez realizado estos pasos se crea tu cuenta y ya puedes reservar las pistas, a la vez que entrar en los diferentes módulos de la página como puede ser ver tu historial de reservas.



Una vez que te hayas logueado volverás a entrar de nuevo en el home pero esta vez ya como un usuario registrado, ya tienes acceso a todo lo que necesitas y es tan sencillo como entrar en reservar y realizar la reserva que necesitas.

Para la página de reserva se mostrará un formulario que deberás rellenar con el día, hora y pista que deseas. Este formulario te dará por válida o no la reserva dependiendo si se encuentra esa hora, ese día y esa pista disponible.

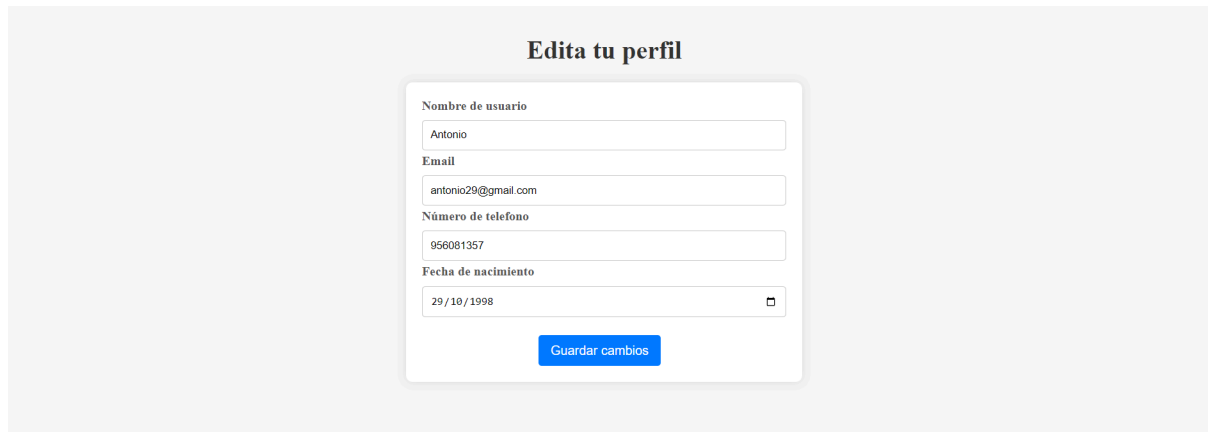


El usuario tendrá la facilidad para ver su reserva en un historial de reservas, desde aquí podrá recordar para qué día y a qué hora tenía su pista reservada. También desde esta misma página podrá realizar la eliminación de su reserva si así lo desea.



Fecha	Hora	Usuario	Pista	Acción
2024-06-10	10:00	antonio	1	<button>Eliminar</button>

Por último dentro de la página web podemos encontrar el enlace para poder editar nuestro perfil. Desde esta página podemos realizar algún cambio de los datos anteriormente introducidos, ya sea por equivocación o por gusto del usuario.



The screenshot shows a web form titled "Edita tu perfil" (Edit your profile). The form contains the following fields:

- Nombre de usuario** (Username): Antonio
- Email**: antonio29@gmail.com
- Número de telefono** (Phone number): 956081357
- Fecha de nacimiento** (Date of birth): 29/10/1998

At the bottom of the form is a blue button labeled "Guardar cambios" (Save changes).

11.Problemas encontrados

Uno de los grandes problemas que he tenido ha sido el poder implementar un calendario para la utilización de la gestión de reservas. Con React encontré la forma de meter el calendario con un solo comando, este comando te creaba un calendario que podrás modificar a tu gusto. El problema vino al intentar coger los datos de symfony ya que no me dejaba realizar la reserva, me daba muchos errores y tuve que usar la forma más sencilla de poder realizar el apartado de reservas.

Otro problema que me surgió al principio fue el de complementar React con Symfony ya que siempre React tiene que encontrar un elemento dentro de la plantilla Twig y al principio no lo sabía. Solo era capaz de realizar la home hasta que investigando encontré la solución.

12.Modificaciones sobre el proyecto planteado inicialmente

Inicialmente ha habido unas cuantas modificaciones de lo planificado desde un principio ya que han ido surgiendo problemas y se han modificado cosas.

- El primer cambio que ha habido ha sido la modificación de solamente utilizar el framework de react. He tenido que meter también Symfony como componente para ayudar a la realización del back-end, para así poder utilizar y manejar el control interno de la página web.
- Otro de los cambios ha sido el del calendario, ya que no he podido implantar como pensaba un calendario y ahí poder realizar las reservas. Para ellos he utilizado un formulario con estilos para que quedará lo más atractivo posible.
- Otro cambio ha sido el de modificar la cita, se ha eliminado ese punto ya que veo innecesario este apartado teniendo en cuenta que se puede eliminar la cita y volver a reservar para otro día.

13.Posibles mejoras al proyecto

Hay cosas que me gustaría mejorar del proyecto que darían una mejor imagen para el usuario al igual que más facilidades a la hora de realizar reservas.

- ☐ Poner la reserva de otra forma, ya bien sea colocando un calendario y que el usuario pulsará sobre el día y le saliera las horas disponibles u otra manera de la que se pudiera realizar.

- ☐ Mandar mensajes al correo electrónico del usuario, con una notificación de que su reserva ha sido realizada con éxito.
- ☐ Colocar foto de perfil en el usuario, en la página web aparece el nombre del usuario, pero estaría bien que apareciera una foto de perfil.
- ☐ Mandar mensaje recordatorio el día que se tiene reservada la pista.
- ☐ Implantar una forma de pago online, para no tener que pagar en la instalación.

14. Bibliografía

Para cualquier duda sobre algún comando de symfony he utilizado la propia página web de ellos que tiene la gran mayoría.

- Symfony Documentation:
<https://symfony.com/doc/current/index.html>

Otra página que me ha ayudado mucho ha sido Stack Overflow ya que hay muchas dudas que la gente resuelve por ahí.

- Stack Overflow: <https://stackoverflow.com/>

Para hacer el carrusel de fotos que se encuentra en la home he utilizado Swiper que es un deslizante de imágenes que nos han enseñado en las prácticas.

- Swiper: <https://swiperjs.com/react>

También he visitado la página de Diego de Lazaro, que habla de Symfony en el cual he cogido mucha información para los formularios.

- <https://diego.com.es/>