

PRIMAVERA ETERNA
-Por Dragos Teodor Maican-
-Tutor: Jose Antonio Piñero Berbel-

IES Francisco Ayala

INDEX:

1.....	RESUMEN.....	P3
2.....	DESCRIPCIÓN.....	P4
3.....	JUSTIFICACIÓN.....	P4
4.....	REQUERIMIENTOS.....	P5
5.....	ANÁLISIS Y DISEÑO.....	P5
6.....	IMPLEMENTACIÓN.....	P8
7.....	DETALLES DE FICHEROS.....	P9
8.....	PRUEBAS.....	P9
9.....	MANUAL.....	P10
10.....	MANUAL DE ESTILOS.....	P12
11.....	GUÍA DE INSTALACIÓN.....	P12
12.....	SOFTWARE UTILIZADO.....	P12
13.....	MEJORAS.....	P13

1)

RESUMEN (Español):

Este proyecto decidí llamarlo así ya que la idea nació para un gimnasio de artes, en concreto de Wing Chun (que significa primavera eterna). Aunque la idea haya surgido así, la estructura se podrá usar para cualquier tipo de gimnasio o negocio similar.

Se trata de una aplicación no muy compleja pero desde luego tampoco es simple. Esta debe ser capaz de hacer lo siguiente:

1. Discernir entre usuario Administrador y usuario Alumno. Permitiendo solo el registro de nuevos alumnos.
2. ALUMNO debe poder reservar dentro de un listado de fechas, en cada día se mostrarán las clases disponibles
3. ALUMNO también puede ver al resto de alumnos dentro de un listado de los mismos.
4. ALUMNO puede editar su perfil (Foto, Nick, Grado...)
5. ADMINISTRADOR debe poder hacer todo lo que ALUMNO puede.
6. ADMIN debe tener un apartado que solo el pueda ver, en el que se puedan ver y editar los pagos de cada alumno.
7. ADMIN debe poder añadir eventos a las fechas.
8. La aplicación debe asignar automáticamente los eventos (clases) a cada día del año, ya que el horario siempre es el mismo, pudiendo solo ADMIN alterar los eventos de dichas fechas.

RESUME (English):

I decided to call this project that way since the idea was born for an arts gym, specifically Wing Chun (which means eternal spring). Although the idea has arisen this way, the structure can be used for any type of gym or similar business.

It is not a very complex application but it is certainly not simple either. this must be able to do the following:

1. Distinguish between Administrator user and Student user. Allowing only the registration of new students.
2. STUDENT must be able to reserve within a list of dates, on each day show available classes
3. STUDENT can also see the rest of the students within a list of them.
4. STUDENT can edit their profile (Photo, Nick, Degree...)
5. ADMINISTRATOR must be able to do everything STUDENT can.
6. ADMIN must have a section that only he can see, in which they can be seen and edited

payments for each student.

7. ADMIN must be able to add events to dates.

8. The application should automatically assign the events (classes) to each day of the year, since that the schedule is always the same, and only ADMIN can alter the events of those dates.

2)

Descripción General.

Este proyecto decidí llamarlo así ya que la idea nació para un gimnasio de artes, en concreto de Wing Chun (que significa primavera eterna). Aunque la idea haya surgido así, la estructura se podrá usar para cualquier tipo de gimnasio o negocio similar.

Se trata de una aplicación no muy compleja pero desde luego tampoco es simple. Esta debe ser capaz de hacer lo siguiente:

1. Discernir entre usuario Administrador y usuario Alumno. Permitiendo solo el registro de nuevos alumnos.

2. ALUMNO debe poder reservar dentro de un listado de fechas, en cada día se mostrarán las clases disponibles

3. ALUMNO también puede ver al resto de alumnos dentro de un listado de los mismos.

4. ALUMNO puede editar su perfil (Foto, Nick, Grado...)

5. ADMINISTRADOR debe poder hacer todo lo que ALUMNO puede.

6. ADMIN debe tener un apartado que solo el pueda ver, en el que se puedan ver y editar los pagos de cada alumno.

7. ADMIN debe poder añadir eventos a las fechas.

8. La aplicación debe asignar automáticamente los eventos (clases) a cada día del año, ya que el horario siempre es el mismo, pudiendo solo ADMIN alterar los eventos de dichas fechas.

Las tecnologías que tengo pensado utilizar son:

Opción 1:

1. Cliente: Angular, Typescript

2. Servidor: Firebase, con Typescript

3. Diseño Interfaz: CSS, Bootstrap, Angular Material

3)

Me he decantado por Angular por el gustillo que le estoy cogiendo a las tecnologías de Microsoft, por su increíble compatibilidad con todas las plataformas, y su increíble adaptabilidad para los problemas grandes. Además, Angular Material es realmente placentero y fácil de manejar, hace que la cantidad de código orientado al diseño sea mínimo. Y no solo eso, también provee de Form Validators para la seguridad en el frontend.

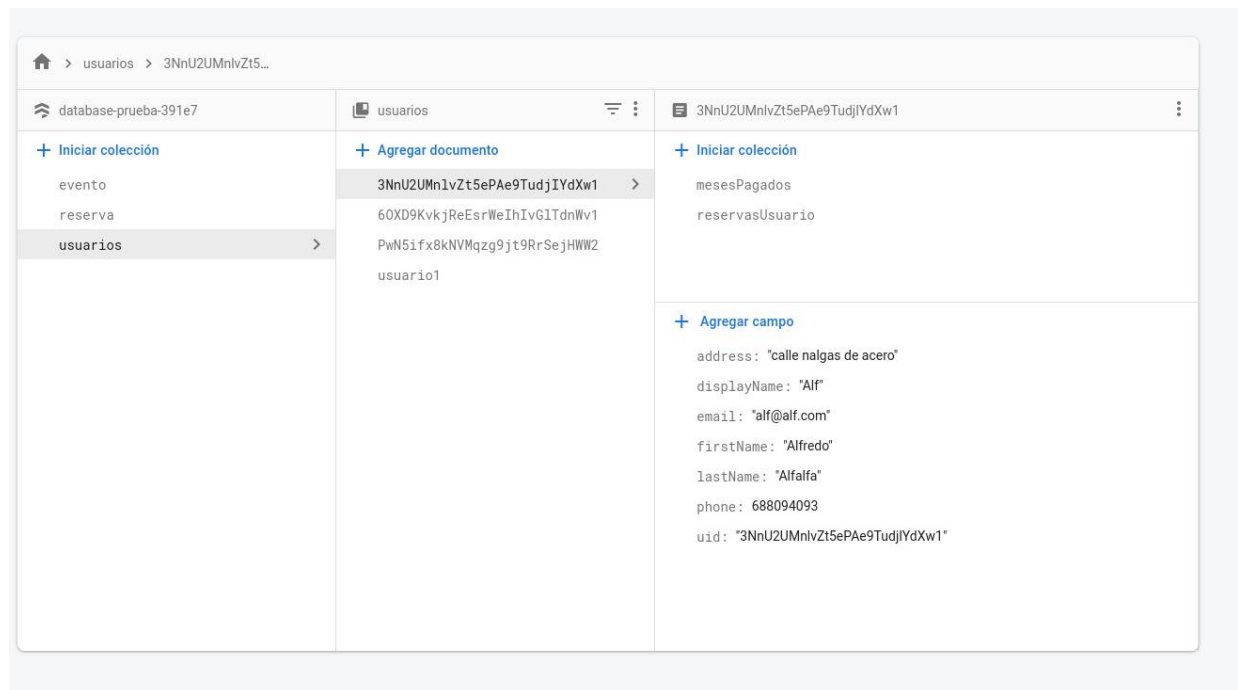
Firestore más de lo mismo, otro hijo de un titán de la tecnología. Con multitud de documentación y de maneras de hacer las cosas, facilidades y gran seguridad en el backend.

4)

Requerimientos: Un ordenador, teléfono o tablet y un navegador.

5)

La estructura de la bse de datos es muy simple.



Hay tres colecciones principales y dentro de usuarios, hay otras.
En usuarios:

<div> <div> <div>🏠</div> <div> <div>usuarios</div> <div>></div> <div>3NnU2UMnlvZt5...</div> <div>></div> <div>mesesPagados</div> <div>></div> <div>2022-6-17 15:33...</div> </div> </div> </div>		
<div> <div>📄</div> <div>3NnU2UMnlvZt5ePAe9TudjIYdXw1</div> <div>⋮</div> </div>	<div> <div>📄</div> <div>mesesPagados</div> <div>☰ ⋮</div> </div>	<div> <div>📅</div> <div>2022-6-17 15:33:24</div> <div>⋮</div> </div>
<div> <div>+ Iniciar colección</div> <div>mesesPagados ></div> <div>reservasUsuario</div> <div>+ Agregar campo</div> <div> <div>address: "calle nalgas de acero"</div> <div>displayName: "Alf"</div> <div>email: "alf@alf.com"</div> <div>firstName: "Alfredo"</div> <div>lastName: "Alfalfa"</div> <div>phone: 688094093</div> <div>uid: "3NnU2UMnlvZt5ePAe9TudjIYdXw1"</div> </div> </div>	<div> <div>+ Agregar documento</div> <div>2022-6-17 15:33:24 ></div> <div>2022-6-17 15:33:38</div> <div>primerMes</div> </div>	<div> <div>+ Iniciar colección</div> <div>+ Agregar campo</div> <div> <div>codigoMes: 5</div> <div>mesPagado: "June"</div> <div>nombre: "Alf"</div> <div>pagadoElDia: "2022/06/17"</div> </div> </div>

<div> <div> <div>🏠</div> <div> <div>usuarios</div> <div>></div> <div>3NnU2UMnlvZt5...</div> <div>></div> <div>reservasUsuario</div> <div>></div> <div>2022-5-19 16:54...</div> </div> </div> </div>		
<div> <div>📄</div> <div>3NnU2UMnlvZt5ePAe9TudjIYdXw1</div> <div>⋮</div> </div>	<div> <div>📄</div> <div>reservasUsuario</div> <div>☰ ⋮</div> </div>	<div> <div>📅</div> <div>2022-5-19 16:54:30</div> <div>⋮</div> </div>
<div> <div>+ Iniciar colección</div> <div>mesesPagados</div> <div>reservasUsuario ></div> <div>+ Agregar campo</div> <div> <div>address: "calle nalgas de acero"</div> <div>displayName: "Alf"</div> <div>email: "alf@alf.com"</div> <div>firstName: "Alfredo"</div> <div>lastName: "Alfalfa"</div> <div>phone: 688094093</div> <div>uid: "3NnU2UMnlvZt5ePAe9TudjIYdXw1"</div> </div> </div>	<div> <div>+ Agregar documento</div> <div>2022-5-19 16:24:24</div> <div>2022-5-19 16:24:27</div> <div>2022-5-19 16:43:5</div> <div>2022-5-19 16:45:55</div> <div>2022-5-19 16:48:43</div> <div>2022-5-19 16:54:30 ></div> <div>2022-5-19 16:55:25</div> <div>2022-5-19 16:56:33</div> <div>2022-5-19 16:56:51</div> <div>2022-5-19 16:57:47</div> <div>2022-5-19 17:12:38</div> <div>2022-5-19 17:54:3</div> <div>2022-5-20 14:30:26</div> <div>2022-5-23 15:35:23</div> <div>2022-5-23 15:38:42</div> <div>2022-5-23 15:48:45</div> </div>	<div> <div>+ Iniciar colección</div> <div>+ Agregar campo</div> <div> <div>diaReserva: "2022/05/19"</div> <div>emailA: "alf@alf.com"</div> <div>fechaReserva: "2022-5-19 16:54:30"</div> <div>horaE: "21:00"</div> <div>nombreA: "USerx"</div> <div>nombreE: "Wing Chun"</div> <div>uidA: "3NnU2UMnlvZt5ePAe9TudjIYdXw1"</div> </div> </div>

Evento:

<div> <div> <div>🏠</div> <div>></div> <div>evento</div> <div>></div> <div>5VpuMOWOEm...</div> </div> </div>		
<div> <div>🔌 database-prueba-391e7</div> <div>📄 evento</div> <div>⌵</div> <div>⋮</div> </div>		<div> <div>📄 5VpuMOWOEmcZ6zHrJ4bL</div> <div>⋮</div> </div>
<div>+ Iniciar colección</div>	<div>+ Agregar documento</div>	<div>+ Iniciar colección</div>
<div>evento ></div>	<div>5VpuMOWOEmcZ6zHrJ4bL ></div>	<div>+ Agregar campo</div>
<div>reserva</div> <div>usuarios</div>	<div>RABjSLLW0rVUv2BBz3kD</div> <div>a1oe063x7I1d40i0sn04</div> <div>nifuWwC7uLwjczbOC7Ws</div> <div>o14UjZgIgQj5A2MwW3Wk</div>	<div>descripcion: "qweqweqwe"</div> <div>fecha: "2022/06/03"</div> <div>hora: "asdf"</div> <div>name: "zzzzzzzzzzzz"</div>

Reserva:

<div> <div> <div>🏠</div> <div>></div> <div>reserva</div> <div>></div> <div>2022-5-19 15:18...</div> </div> </div>		
<div> <div>🔌 database-prueba-391e7</div> <div>📄 reserva</div> <div>⌵</div> <div>⋮</div> </div>		<div> <div>📄 2022-5-19 15:18:11</div> <div>⋮</div> </div>
<div>+ Iniciar colección</div>	<div>+ Agregar documento</div>	<div>+ Iniciar colección</div>
<div>evento</div> <div>reserva ></div> <div>usuarios</div>	<div>2022-5-19 15:18:11 ></div> <div>2022-5-19 15:18:18</div> <div>2022-5-19 15:22:46</div> <div>2022-5-19 15:25:0</div> <div>2022-5-19 15:50:21</div> <div>2022-5-19 15:54:40</div> <div>2022-5-19 15:55:52</div> <div>2022-5-19 15:57:19</div> <div>2022-5-19 16:12:42</div> <div>2022-5-19 16:12:9</div> <div>2022-5-19 16:13:12</div> <div>2022-5-19 16:14:16</div> <div>2022-5-19 16:15:19</div> <div>2022-5-19 16:16:57</div> <div>2022-5-19 16:18:16</div> <div>2022-5-19 16:19:12</div>	<div>+ Agregar campo</div> <div>emailA: "pablo@pablo.com"</div> <div>fechaReserva: "2022-5-19 15:18:11"</div> <div>horaE: "21:00"</div> <div>nombreA: "USerx"</div> <div>nombreE: "Wing Chun"</div>

Diagrama Entidad Relación:

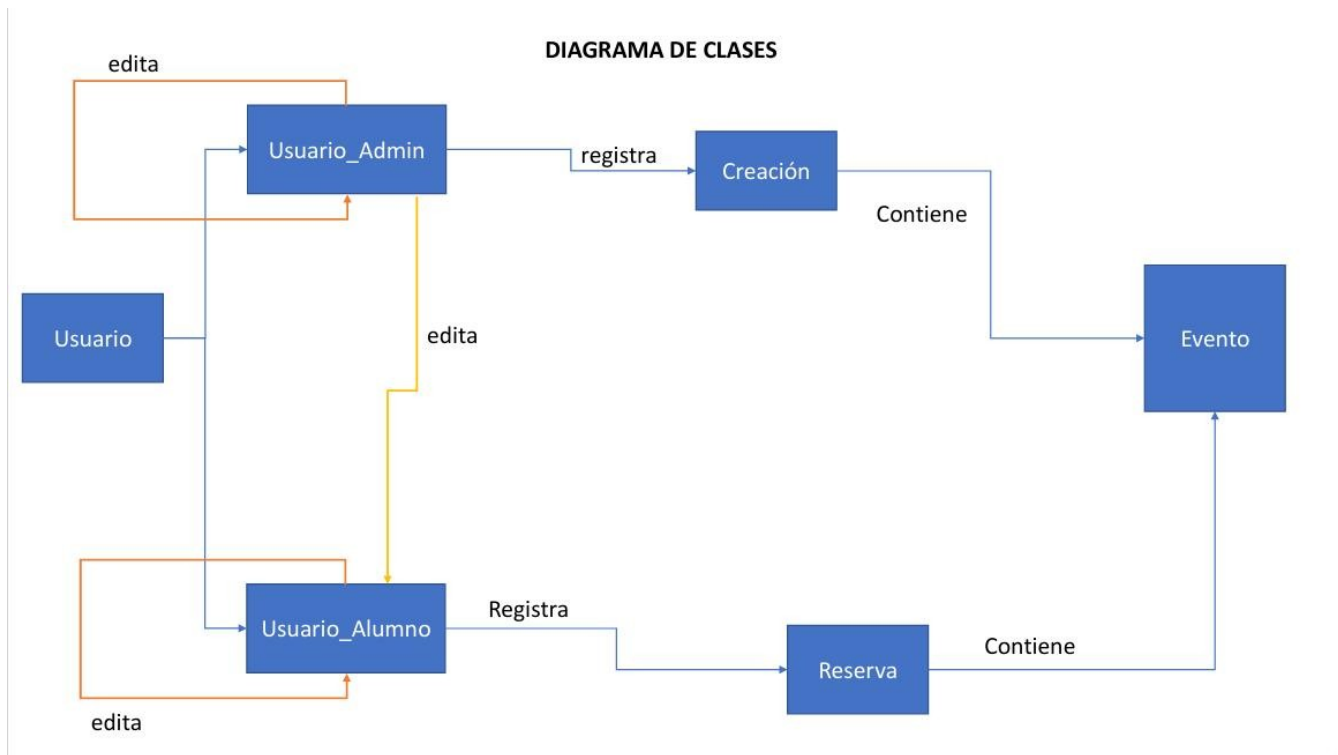
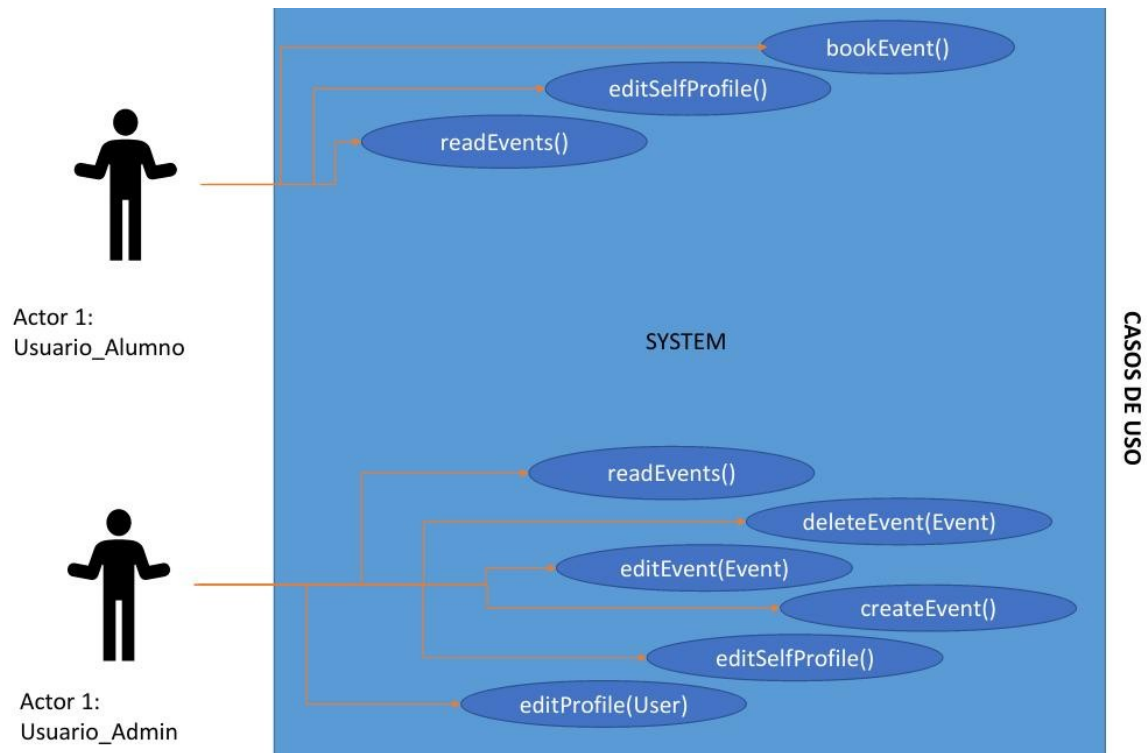


Diagrama de casos de uso:



Para el diseño: Uso angular material y los componentes con los estilos que provee, se encarga de formularios y de animaciones.

Para las consultas: Peticiones asincronas a Firebase.

7)

	FICHERO	FUNCION	ACCESO BD?	LLAMADAS	LLAMADO
ADMIN	Componente Admin	Gestionar Alumnos, Pagos y Eventos	SI	NO	SI, se accede si eres admin desde el sidebar
HOME	Home	Los usuarios reservan y ven sus reservas	SI	NO	SI, desde sidebar
LANDING	Landing	Landing Page	NO	NO	NO
LOGIN	Login	Loguearse	SI	SI, a Sign-Up y a Home	Si, sidebar
SIGN-UP	Sign-Up	Registrarse	SI	SI. a Home	Si, sidebar
MEMBERS	Members	Ver Miembros	SI	NO	SI, sidebar
NOT FOUND	Not-Found	Pagina no encontrada	NO	NO	NO
PROFILE	Profile	Usuario gestiona sus datos	SI	NO	Si, sidebar

8)

El testeo lo realicé yo mismo, haciendome pasar por distintos tipos de usuarios. Existe verificación en frontend y backend.

Frontend: Lo realizan los validadores de la clase FormGroup, además, ninguna de las páginas es accesible para usuarios no logueados. El componente Admin solo carga si el usuario es administrador.

Backend: Firestore Rules. Muy útil y fácil de usar, solo hace peticiones a la BD si el usuario está logueado. Igual con admin.

Siempre que se hace un Cambio en la base de datos, un registro de usuario, o un acceso del mismo, se notifica con la clase hotToast, igual para el caso en el que haya sucedido un error o avisa de cambios ya realizados.

9)

Manual de Usuario.

Aquí dejo unos usuarios ya creados para que podáis acceder:

Supongamos que eres:

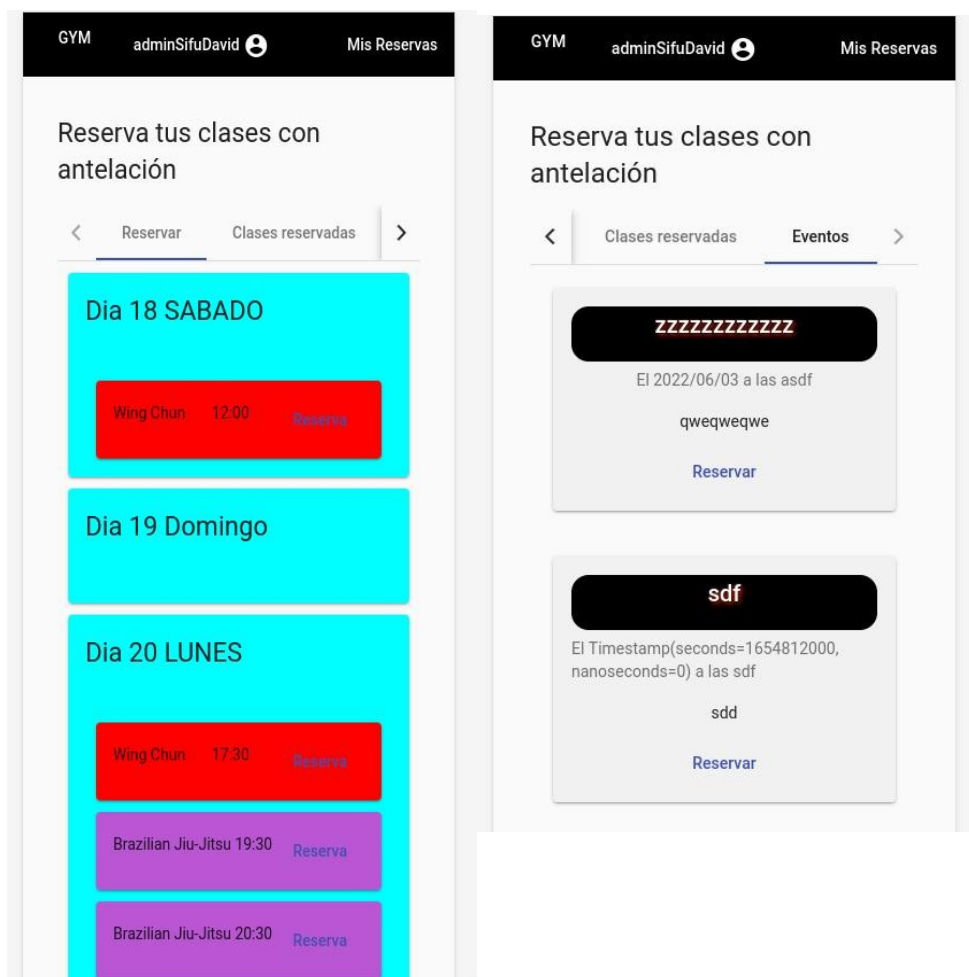
A. Usuario Normal: eli@eli.com , contraseña: elieli

B. Usuario Admin: sifu1@sifu1.com , contraseña: sifusifu

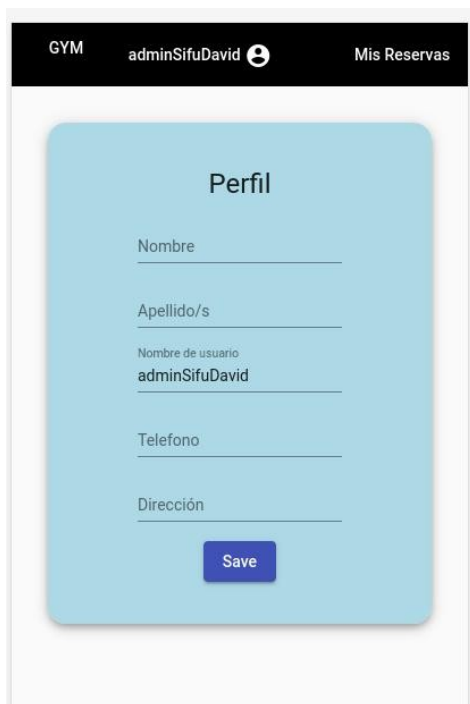
También puedes crear una nueva cuenta de usuario (normal).

En Home Puedes reservar una clase o un evento, y en la pestaña Mis Reservas puedes ver las que ya has reservado.

Si ya reservaste, la aplicación te lo notifica.



En los apartados Profile podemos actualizar nuestros datos, y en Admin/Eventos, el admin puede crear nuevos eventos para los usuarios.



GYM adminSifuDavid Mis Reservas

Perfil

Nombre

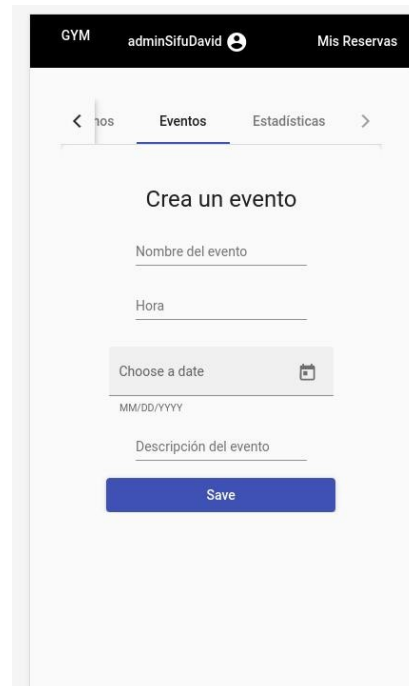
Apellido/s

Nombre de usuario
adminSifuDavid

Telefono

Dirección

Save



GYM adminSifuDavid Mis Reservas

< Mis Eventos Estadísticas >

Crea un evento

Nombre del evento

Hora

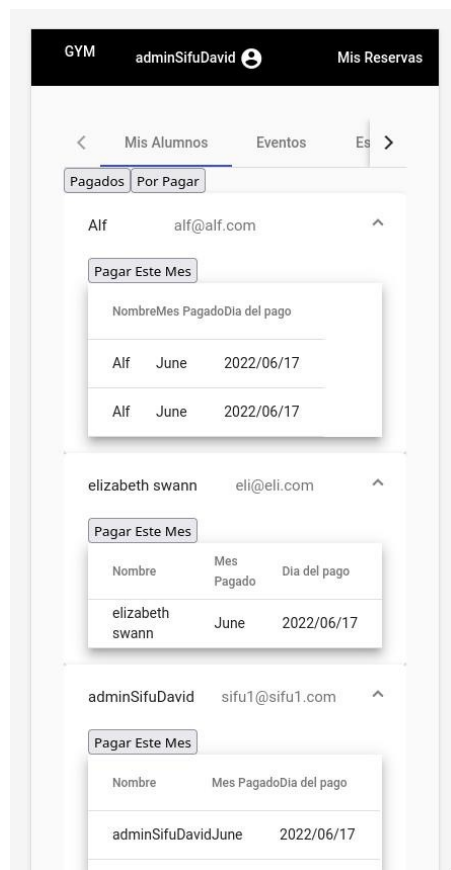
Choose a date

MM/DD/YYYY

Descripción del evento

Save

En Admin/Mis Alumnos, el Administrador puede ver si el usuario ha pagado este mes y actualizar la información.



GYM adminSifuDavid Mis Reservas

< Mis Alumnos Eventos Estadísticas >

Pagados Por Pagar

Alf alf@alf.com

Pagar Este Mes

Nombre	Mes Pagado	Dia del pago
Alf	June	2022/06/17
Alf	June	2022/06/17

elizabeth swann eli@eli.com

Pagar Este Mes

Nombre	Mes Pagado	Dia del pago
elizabeth swann	June	2022/06/17

adminSifuDavid sifu1@sifu1.com

Pagar Este Mes

Nombre	Mes Pagado	Dia del pago
adminSifuDavid	June	2022/06/17

Para desloguearse se clicka el nombre del navbar.

10)

a y b)

Mis consideraciones sobre usabilidad y accesibilidad:

1. **Simplicidad:** hay que estructurar los contenidos de acuerdo a una jerarquía perfectamente definida. El usuario no debe sentirse perdido en ningún momento. Siempre debe saber cómo pasar de un lugar a otro, por lo que todo debe tener una clasificación lógica.
2. **Interacción:** es muy importante que el usuario pueda interactuar de forma fácil e intuitiva. Debe poder desplegar menús, descargar información, ampliar fotos...
3. **Universalidad:** hay que adaptar la web a todo tipo de soportes y navegadores.
4. **Actualización:** para dar credibilidad a una web, el contenido debe actualizarse constantemente.
5. **Rapidez:** la web debe cargarse en una media de 4 segundos. Está comprobado que como máximo los usuarios esperan hasta 10 segundos para que una página se cargue.

c)

Fuentes Principales: Las que proporciona Angular Material

d)

Tamaños principales: También dependen del tipo de componente usado en Angular Material.

e)

Colores: Colores por defecto de Angular material

-#F1F1F1

-#000000

-white

-orangered

f)

La aplicación está pensada para Móviles. Sin embargo puede usarse en cualquier dispositivo. De los puntos de ruptura y el dinamismo de los contenedores se encarga Angular Material.

11)

GUIA DE INSTALACIÓN.

Local: Descargar Repositorio e inputar el comando -- npm install – en la carpeta del proyecto.

Producción: Clickar el link de github.

12)

1. Angular Material: Para diseño, validación y dinamismo de la página.
2. Firebase: Para seguridad en backend, bases de datos, hosting y gestión de usuarios.
3. Rxjs angular, HotToasts, ngNeat para notificar eventos con interacciones.
4. Angular como Framework.

13)

Futuras implementaciones:

- Mensaje motivador diario personalizado de bienvenida a la app (Escrito por sifu).
 - Por cada clase asistida del usuario se incrementará un número (asist*) que representará las clases totales que ha reservado, y el porcentaje de clases reservadas (pct_asist*) que ha cancelado
 - Una de las cosas mas interesantes son las estadísticas de los alumnos.
 - Porcentaje de alumnos que cancelan sus clases.
 - Porcentaje de alumnos de cada GRADO*
 - Velocidad de ascenso en función del DIA_DE_ENTRADA* y del GRADO* actual
 - Porcentaje de alumnos que asisten a cada ARTE* en función del total de los mismos
 - Porcentaje de alumnos que abandonan cada ARTE* y/o la escuela
 - En función de pct_asist* se le mandarán mensajitos al usuario para motivarle dentro de la vista “config”.
 - El usuario podrá ver sus clases reservadas desde HOY* hasta el DIA_DE_ENTRADA a la escuela
- Mensajes por defecto que felicitan por cumpleaños, ascenso. Incluso por abandono de la escuela:

Estas son cosas no muy complejas pero que tal vez sean buena idea. Además deseo hacer el código más limpio y más corto.