

ARIADNA REDONDO MARTÍN
2 DAM



valenciaApp

Memoria
valenciaApp

ÍNDICE



1. INTRODUCCIÓN	1
<i>Resumen del proyecto</i>	<u>1</u>
<i>Idea</i>	<u>2</u>
<i>¿Que se propone?</i>	
2. COMPONENTES DEL PROYECTO	
<i>Frontend</i>	<u>3-11</u>
- Pantallas y diseño de la app	
<i>Backend</i>	<u>12-13</u>
- Almacenamiento de todos los datos	
<i>Tecnologías utilizadas para el desarrollo de la app</i>	<u>15</u>
3. JUSTIFICACIÓN Y OBJETIVOS	<u>16</u>
<i>¿Para que se propone el proyecto?</i>	<u>16</u>
<i>¿A quien va dirigido?</i>	<u>16</u>
<i>¿Que se pretende alcanzar?</i>	<u>16</u>
4. DESARROLLO DEL PROYECTO	
<i>Metodologías utilizadas</i>	<u>17</u>
<i>Problemas/Dificultades</i>	<u>17</u>
<i>Resultados obtenidos</i>	<u>18</u>
5. CONCLUSIONES	
<i>Conclusiones tras la realización del PFC</i>	<u>20</u>
6. LÍNEAS FUTURAS DE TRABAJO	<u>21</u>
7. BIBLIOGRAFIA	<u>22</u>

1. INTRODUCCIÓN Y RESUMEN DEL PROYECTO

El propósito de este informe es documentar y analizar en detalle el desarrollo y los resultados alcanzados por el proyecto "valenciaApp".

Este proyecto ha sido implementado durante el periodo de la FCT del grado superior de Desarrollo de Aplicaciones Multiplataforma.

La aplicación móvil presentada está destinada a proporcionar a los valencianos una experiencia única, permitiéndoles descubrir restaurantes y actividades en la ciudad. En este documento, describo las fases clave del proyecto, desde el inicio hasta la implementación, junto con los desafíos enfrentados y las soluciones aplicadas.

valenciaApp es una aplicación que se basa en dos pilares fundamentales. En primer lugar, encontramos la sección dedicada a los **restaurantes**. Esta sección proporciona a los usuarios un listado de fotografías de diversos restaurantes, permitiéndoles tener una visión general y rápida de las opciones disponibles. El objetivo es facilitar la selección de un restaurante de interés con una simple ojeada.

Una vez que se ha seleccionado uno de los restaurantes que captó la atención del usuario, la aplicación mostrará toda la información relevante sobre el mismo. Esto permitirá a los usuarios acceder, de manera eficiente, a una amplia variedad de opciones gastronómicas y obtener información detallada sin tener que realizar múltiples búsquedas.

En segundo lugar, encontramos la sección dedicada a los **eventos**. Al igual que en la sección anterior, esta área ofrece una lista de imágenes junto con breves descripciones de los eventos que tendrán lugar en la ciudad de Valencia en las próximas fechas. Una vez que el usuario encuentra un evento de su interés, puede hacer clic en él para acceder a una información mucho más detallada y completa.

En resumen, valenciaApp proporciona a los usuarios una plataforma centralizada que reúne información sobre restaurantes y eventos en la ciudad de Valencia. A través de una visualización intuitiva de imágenes y descripciones concisas, la aplicación facilita la selección rápida y eficiente de opciones gastronómicas y eventos culturales. Esto elimina la necesidad de realizar múltiples búsquedas y permite a los usuarios acceder a la información necesaria de manera conveniente y eficaz.

La idea

La idea de este proyecto surgió de una necesidad personal como residente de Valencia. Como alguien que vive en esta maravillosa ciudad, he experimentado la limitación de siempre ir a los mismos lugares a comer y no descubrir nuevas opciones.

Aunque existen redes sociales que muestran planes para hacer en Valencia, me resulta poco práctico pasar horas buscando y filtrando información que en muchas ocasiones parece dispersa.

Motivada por esta situación y dados mis estudios, surgió la idea de desarrollar valenciaApp. Mi objetivo principal era crear una solución que simplificara y agilizara el proceso de descubrimiento de restaurantes en la ciudad. Quería brindar a los residentes de Valencia, como yo, una herramienta que les permitiera acceder fácilmente a una variedad de opciones gastronómicas interesantes y orientadas de manera menos turística.

Con esta motivación en mente, decidí desarrollar una aplicación móvil que reuniera en un solo lugar un listado completo de restaurantes con información relevante y fotografías para ayudar a los usuarios a seleccionar de manera rápida y eficiente opciones de comida en Valencia. Además, también añadí una sección dedicada a eventos que tienen lugar en la ciudad. Ya que también ocurre que hay muchos planes para realizar en la ciudad pero no hay una plataforma que englobe ambas cosas y simplifique la información.

En resumen, la idea de valenciaApp surge de mi propia experiencia como residente de Valencia, cansada de ir siempre a los mismos lugares pero sin tiempo ni paciencia para explorar opciones por redes sociales.

¿Qué se propone?

La aplicación móvil busca mejorar la experiencia de búsqueda y selección de restaurantes y eventos, ofreciendo información relevante y de calidad en un solo lugar.

valenciaApp también tiene como objetivo promover y apoyar el comercio local. Al destacar una variedad de restaurantes y eventos en la ciudad, ofreciendo alternativas más allá de los restaurantes turísticos típicos, apoyando así a la comunidad y la economía valenciana.

Además, la sección de eventos de valenciaApp incluirá actividades culturales y festivales locales, brindando una plataforma para que los residentes de Valencia puedan descubrir y participar en la vida cultural de la ciudad.

2. COMPONENTES DEL PROYECTO

¿DE QUE ESTA FORMADO?

La arquitectura de valenciaApp se divide en dos componentes fundamentales: el **frontend** y el **backend**. Cada uno desempeña un papel crucial en el funcionamiento y la experiencia general de la aplicación.

FRONTEND

El **frontend** de valenciaApp se encarga de la interfaz de usuario y la interacción directa con los usuarios. Está diseñado para ser intuitivo y atractivo, brindando una experiencia de usuario fluida y agradable. El frontend se desarrolla utilizando el framework de Ionic, que combina tecnologías web como HTML, SCSS y TypeScript, que más adelante explicaré en profundidad y se centra en la presentación visual de los restaurantes y eventos, así como en la navegación y la interacción del usuario con la aplicación.

PANTALLAS DE LA APP Y SU FUNCIONAMIENTO

Pantalla de inicio:



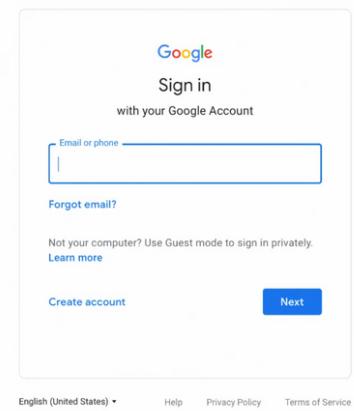
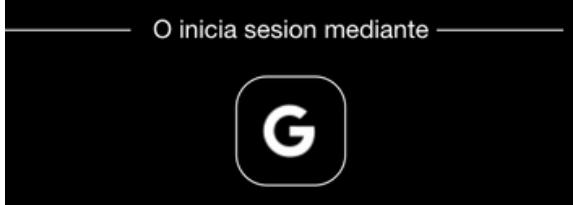
En la captura de pantalla anterior, se puede observar que la pantalla de inicio de sesión solicitará dos parámetros principales: el **nombre de usuario** y la **contraseña**. Una vez que el usuario introduzca estos datos, la aplicación se encargará de realizar una validación para asegurarse de que se hayan introducido correctamente.

Para llevar a cabo esta validación, se utilizará una expresión regular (regex) que permitirá verificar que los datos cumplan con un formato específico.

```
regexPassword = new RegExp('^(?=.*[A-Z])[a-zA-Z0-9_]{8,}$');
```

```
username: ['', [Validators.required, Validators.minLength(2)]],  
password: ['', [Validators.required, Validators.pattern(this.regexPassword)]]
```

Además de la opción de registro tradicional, valenciaApp también ofrece la posibilidad de iniciar sesión utilizando su cuenta de Google. Esta opción de inicio de sesión con Google ofrece comodidad y facilidad de acceso, ya que los usuarios pueden usar sus credenciales existentes de Google para acceder a la aplicación.



Como se muestra en la captura anterior, nos saldrá un desplegable que nos permitirá elegir la cuenta de google (en el caso de que tengamos varias) con la que nos queremos logear,

Una vez la hayamos seleccionado e introducido nuestra contraseña, accederemos a la pagina de inicio, con nuestra sesión iniciada y nuestra información de cuenta cargada.

Para implementar este inicio de sesión con Google, se ha utilizado Firebase Authentication, que es parte de la plataforma Firebase de la cual hablaré más adelante en la en apartado de backend.

Al utilizar Firebase Authentication, se puede aprovechar la infraestructura de autenticación segura y confiable de Google, lo que garantiza que los datos de inicio de sesión del usuario estén protegidos.

Toda la documentación necesaria para implementar dicha funcionalidad, la podremos encontrar en el siguiente enlace:

<https://firebase.google.com/docs/auth/web/start?hl=es-419>

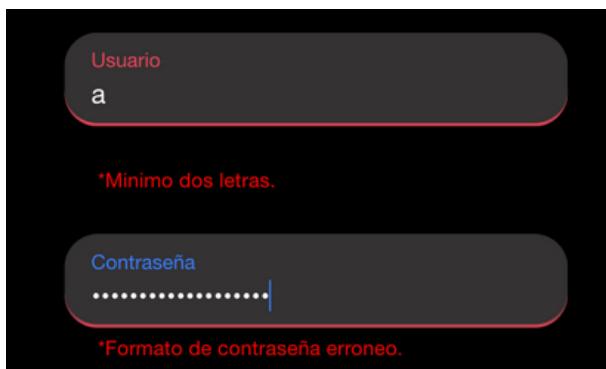
Pantalla de registro:

Una vez verificada la información de inicio de sesión y en el caso de que el usuario esté registrado, se podrá acceder a la pantalla **principal** de la aplicación. Sin embargo, en caso de que el usuario no esté registrado, deberá registrarse.

Para acceder a esta **página de registro**, el usuario deberá pulsar sobre el botón correspondiente, el cual estará disponible en la pantalla de inicio de sesión. Al hacerlo, se redirigirá al usuario a la página de registro, tal como se muestra en la imagen adjunta.

Una vez ya en la página de registro, el usuario deberá introducir los campos solicitados, y podrá completar su registro.

Estos campos también serán verificados previamente mediante un regex, en el caso de que el formato sea incorrecto, el usuario este ya conectado etc. Se mostrará un pop up o un error en el input indicando el problema.



En el caso de que todos los datos sean correctos, el usuario será redirigido a la pantalla principal de restaurantes.

Pantalla principal (home):

Como mencioné anteriormente, la pantalla principal de valenciaApp estará dedicada a proporcionar información completa sobre los restaurantes disponibles.

Cuando el usuario entre con su sesión ya iniciada, podrá ver lo que se muestra en la captura de nuestra derecha,

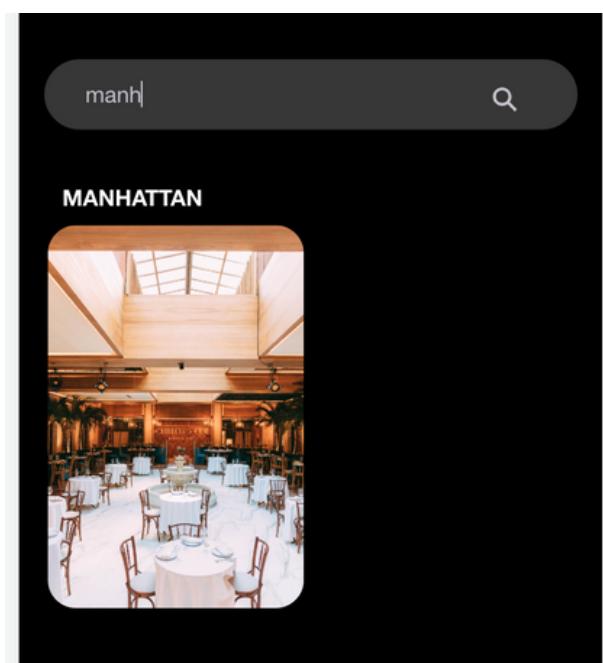
Un listado de imágenes de los restaurantes, junto a sus correspondientes nombres

De esta forma, logramos tener, de forma rápida e intuitiva, todos los restaurantes disponibles.

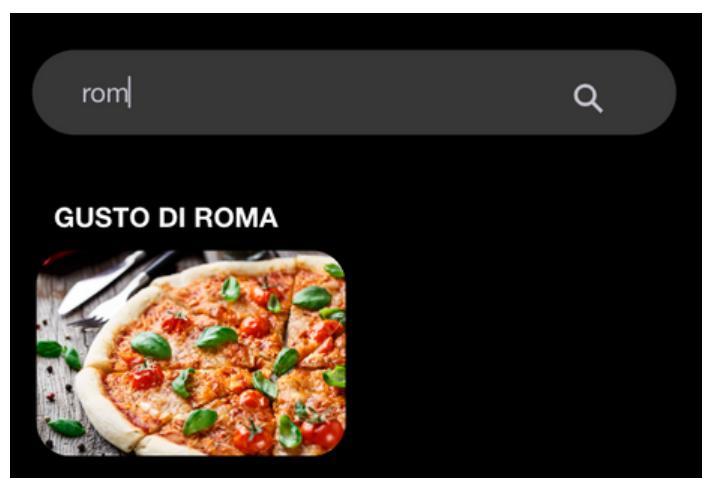
Además, esta pantalla, también contará con un elemento propio de la documentación de Ionic, un Ion-searchbar, que será un buscador, Con este, también tendremos la opción de buscar rápidamente cualquier restaurante que queramos, introduciendo su nombre o parte de él.



Cuando hayamos realizado nuestra búsqueda, nuestro listado de imágenes de restaurantes, filtrará aquellas coincidencias, y mostrará únicamente aquellos restaurantes cuyo nombre coincide con el introducido, en las siguientes capturas de pantalla, se puede ver un ejemplo de búsqueda;



Búsqueda por el nombre



Búsqueda por elementos del nombre

Información de los restaurantes:

Una vez el usuario haya elegido el restaurante del cual se quiere informar, bastará con clicar sobre su imagen, para pasar a la pantalla que muestre toda la información sobre este:

En esta pantalla se mostrará la información detallada del restaurante, incluyendo;

- Nombre
- Descripción
- Ubicaciones
- Fotografías
- Link a para realizar una reserva en la web oficial del restaurante

Esta pantalla va a incluir otro componente propio de Ionic, que será un ion-modal,

Como se puede apreciar en la captura adjuntada, es un modal, que según vayamos desplegando, irá mostrando más información sobre el restaurante.

Una vez este modal este completamente desplegado, nos encontraremos con la siguiente vista:



Y ahora si, tendremos toda la información disponible sobre el restaurante, Incluyendo, fotografías de los platos más destacados, las cuales podremos ir pasando pulsando sobre los botones > <

Por último el modal contendrá un botón que nos redirigirá a la web oficial de restaurante, al apartado de reservas.

Para así, en caso de estar interesado en reservar en el, tener de forma rápida y eficaz, acceso a realizar una reserva.

Navegación:

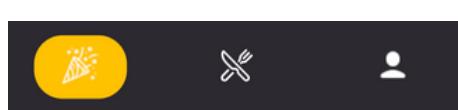
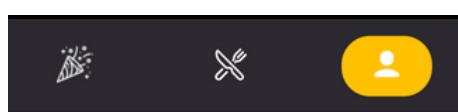
Como se puede observar en la anterior captura, la navegación por la app, va a ser mediante un componente propio también de Ionic, un **ion-tab**,

Como se muestra en las imágenes adjuntas, este tab, nos permitirá navegar entre las tres diferentes pantallas de la app.

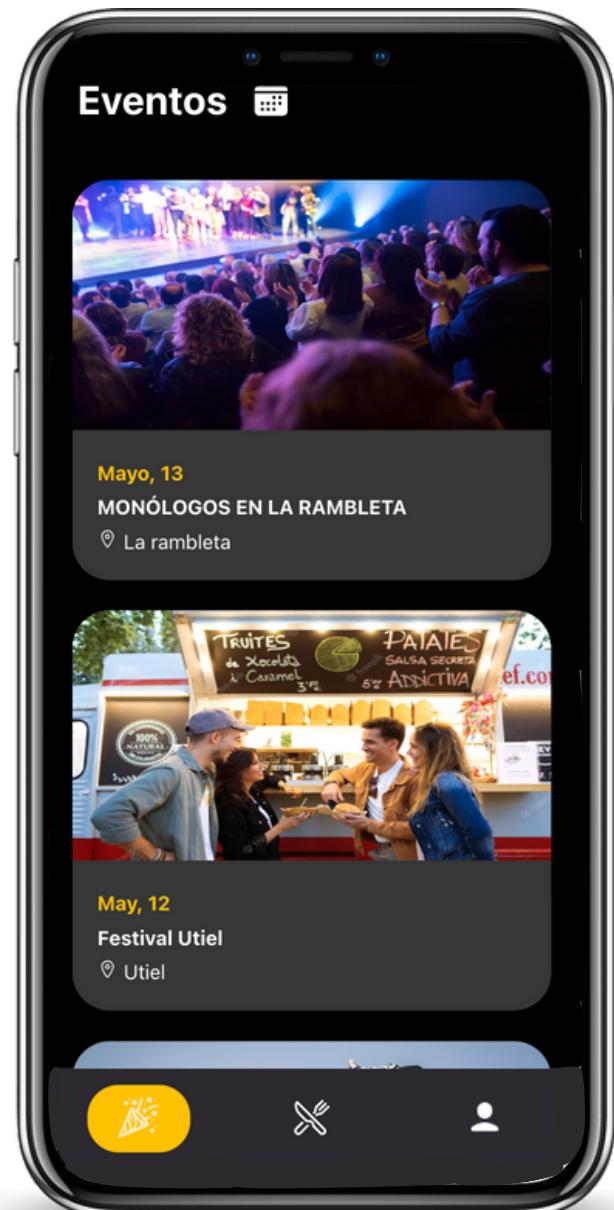
-Eventos

-Restaurantes

-Información de la cuenta



Pantalla de eventos:



En esta pantalla, encontraremos también, un listado con todos los próximos **eventos** de la ciudad,

En este caso a demás de incluir una **fotografía** y un nombre del evento como visión general, también esta incluida, **La fecha de inicio** del evento, y la **ubicación** donde se realiza.

La decisión de hacerlo así, es más bien, porque hay muchas zonas donde se realizan estos eventos, y a diferencia de los restaurantes, que suelen estar ubicados en partes más céntricas y fácilmente accesibles, hay algunos eventos que se encuentran en pueblos o zonas más alejadas que tal vez los usuarios no estén dispuestos a desplazarse.

Asimismo, resaltar también la **fecha** del evento en la vista principal es importante también ya que si no coincide con las fechas que el usuario busca, o tiene disponibilidad, no tendría sentido que entrase en el evento,

Entonces en base a esto, está planteado el listado inicial de los eventos.

Componentes:

Para crear la visión general de los eventos en ValenciaApp, se ha utilizado un componente propio de Ionic llamado ion-card. El ion-card es un componente versátil que permite mostrar información de manera estructurada y visualmente atractiva en forma de tarjetas. Al aprovechar las capacidades y estilos del ion-card, se logra una experiencia visual coherente y agradable para los usuarios.

```
<ion-card class="background">
  <img [src]="event.photo" class="image"/>
  <ion-card-content class="background">
    <div class="card-content">
      <div class="shortDate">
        <ion-text>{{event.shortDate}}</ion-text>
      </div>
      <div class="title">
        <ion-text>{{event.title}}</ion-text>
      </div>
      <div class="place">
        <ion-icon name="location-outline"></ion-icon> <ion-text>{{event.place}}</ion-text>
      </div>
    </div>
  </ion-card-content>
</ion-card>
```

Para cargar todas las listas que contiene la app de manera dinámica, ha sido utilizada la función "***ng for**" que es un directiva de Angular y también compatible con Ionic.

Esta función permite recorrer un array de objetos y generar dinámicamente elementos visuales para cada objeto en el array.

```
<div class="principal-content" *ngFor="let event of eventsList" routerLink="events-details" (click)="showEventDetails(event)">
```

Como se puede apreciar en la anterior imagen, estamos recorriendo nuestro objeto "**events-list**" que en este caso tiene todos nuestros eventos, y mostrándolos uno a uno.

Efectivamente, la estructura de ambas pantallas en valenciaApp sigue un patrón similar, con una pantalla inicial que muestra una vista rápida de eventos o restaurantes, y una pantalla secundaria que proporciona información más detallada sobre el evento o restaurante seleccionado por el usuario.

Esta estructura de navegación entre pantallas se ha implementado utilizando el enrutamiento propio de Angular. Cada tarjeta en la pantalla principal está asociada a una función "**(click)**" que permite la navegación hacia la pantalla secundaria correspondiente al evento o restaurante seleccionado.

Para gestionar la navegación entre pantallas, se utiliza el componente "**navController**", que es una característica de Angular compatible con Ionic.

El "navController" permite controlar la navegación entre páginas y realizar transiciones entre ellas.

A continuación, se presenta un ejemplo básico de cómo funciona:

```

showEventDetails(event: Event ) {
  this.navCntrlr.navigateForward('events-details', {
    queryParams: {
      object: event,
    }
  })
}

```

En este caso, aparte de la navegación entre pantallas, también lo vamos a utilizar para compartir objetos entre pantallas, en este caso el objeto va a ser el evento seleccionado, el cual le pasamos desde el fichero html.

```
<div class="principal-content" *ngFor="let event of eventsList" routerLink="events-details" (click)="showEventDetails(event)">
```

Y por ultimo, en cuando al apartado de eventos, tendríamos lo que es la vista secundaria una vez seleccionamos uno;

Detalles del evento:



En la pantalla de detalles del evento vamos a encontrar, de forma muy parecida a la de detalles del restaurante, Un modal que nos va a mostrar toda la información de dicho evento, de manera resumida y visual.

Encontrando:

- Titulo del evento.**
- Ubicación.**
- Fechas y horas.**
- Descripción.**

El uso del modal para mostrar los detalles del evento permite que los usuarios accedan a la información esencial de manera clara y ordenada, sin abrumarlos con una cantidad excesiva de texto o elementos visuales. Además, el diseño visual del modal garantiza una experiencia de usuario atractiva y enfocada en los detalles importantes del evento.

Por último, el modal en la pantalla de detalles del evento incluye una breve descripción del evento. Si el usuario desea obtener más información, puede hacer clic en la opción "**Leer más**" para acceder a una descripción más detallada del evento. Además, si el usuario desea volver a la descripción corta, puede hacer clic en la opción "**Leer menos**" para regresar a la versión resumida.

Para implementar esta funcionalidad, se ha utilizado la función "***ng_if**" que nos permitirá poner un condicional, y en base a su cumplimiento, se mostrará un formato de la descripción u otro, adjunto a continuación una foto de su funcionamiento;

```
<ng-container *ngIf="isReadMore; else elseTemplate">
  {{ event.description | slice:0:event.description.length }}
</ng-container>
<ng-template #elseTemplate>
  {{ event.description | slice:0:100 }}
</ng-template>
<ion-text class="readMoreText" (click)="readMore()"> {{isReadMore ? '...leer menos' : '...leer mas'}} </ion-text>
```

En la imagen, también se puede apreciar, como gastamos el método "**slice**" para decir el numero de caracteres que queremos mostrar dependiendo de en que estado este la variable que controla si han sido pulsadas o no las opciones de leer más o menos.

Pantalla de información personal:

Por último, valenciaApp incluirá un apartado dedicado a la gestión de la cuenta del usuario, donde se mostrará toda la **información** relevante relacionada con su **perfil**. Esta sección permitirá al usuario acceder y gestionar los siguientes datos:

- Nombre**
- Cuenta de valenciaApp**
- Contraseña**

Lamentablemente, debido a limitaciones de tiempo y dificultades técnicas para diferenciar el proceso de inicio de sesión con Google y el inicio de sesión interno de la aplicación, la pantalla dedicada a la gestión de la cuenta no pudo ser desarrollada completamente.

Entendemos que esta funcionalidad es importante para los usuarios y lamentamos no poder proporcionarla en su totalidad. Sin embargo, se ha implementado el inicio de sesión con Google como una alternativa para que los usuarios puedan acceder a la aplicación de forma segura y rápida.

A pesar de las dificultades encontradas, se seguirá trabajando en mejorar y completar esta pantalla en futuras actualizaciones de valenciaApp.

Con esto, hemos concluido el apartado dedicado al frontend de valenciaApp. En el próximo apartado, nos centraremos en explorar y profundizar en el desarrollo del backend de valenciaApp.

2. COMPONENTES DEL PROYECTO

¿DE QUE ESTA FORMADO?



BACKEND

El **backend** de ValenciaApp, desarrollado con **Firebase Firestore**, se encargará de;

- La gestión de usuarios: registro y autenticación de usuarios utilizando Firebase Authentication. Almacenará y validará la información de los usuarios, como nombres, direcciones de correo electrónico y contraseñas. También gestionará las sesiones de usuario y la seguridad de la autenticación.
- Almacenamiento de datos: se guardarán y recuperarán los datos relacionados con restaurantes, eventos y perfiles de usuario. El backend realizará operaciones de lectura y escritura en Firestore para mantener los datos actualizados y accesibles.

Además, integrar servicios como es el caso del inicio de sesión. con Google mediante Firebase Authentication.

La gestión de usuarios y almacenamiento de datos:

Como se menciona anteriormente, el registro y la autenticación de los usuarios, ha sido realizada mediante firestore,

Inicialmente se ha creado un servicio de firestore, en el cual, hemos creado los diferentes métodos utilizados para interactuar con la base de datos.

En el caso de la autenticación y agregación de usuarios se han implementado los siguientes métodos:

1. Conexión con la base de datos.

```
app = initializeApp(environment.firebaseioConfig);
db = getFirestore(this.app);
```

2. Extracción de la colección de usuarios:

```
async getAllUsers() {
  let usersList: User | any = [];
  const querySnapshot = await getDocs(collection(this.db, "users"));
  querySnapshot.forEach((doc) => {

    usersList.push(doc.data());
  });
  return usersList;
}
```

3. Método de agregación de un nuevo usuario:

```
async addNewUser(userAdd : User){
  await addDoc(collection(this.db, "users"), userAdd);
}
```

Así se ve la base de datos desde firestore;

The screenshot shows the Google Cloud Firestore interface. On the left, there's a sidebar with collections: events, explore, restaurants, and users (which is selected). The main area shows the 'users' collection with several documents listed. One document is expanded, showing its fields: name, password, and username. The document ID is 2ymV0RdchbPmCkvz8LKZ.

Document ID	Fields
2ymV0RdchbPmCkvz8LKZ	name: "pepe", password: "hola123A", username: "user2"
EuUCdotSxqzkU6c1SATU	
GB61LhagzZkaBSP3DT18	
cj1fZem50la35sjP5K3	
nvEMPLANVkhHQV8XpEmF	
owwodCubvEdcwFnyri6T	

Se puede observar, en la primera columna, el listado de colecciones, en la segunda, ya dentro de la colección de usuarios, todos los usuarios existentes y en la tercera, como se ve la información del primer usuario.

Login con google:

Para implementar el inicio de sesión con Google en nuestra aplicación Android, se ha llevado a cabo la configuración correspondiente en Firebase. Esto ha implicado crear un proyecto específico de tipo Android nativo en Firebase, donde hemos generado las SDK necesarias para la integración. Es importante destacar que la configuración para Android es diferente de la configuración web.

A continuación, muestro como quedaría la configuración desde Firebase;

The screenshot shows the Firebase console interface under the 'Tus apps' section. On the left, there's a list of apps: 'Apps para Android' containing 'com.valenciaApp.app' (selected) and 'Apps web' containing 'valencia-app'. The main panel is titled 'Configuración del SDK' for 'com.valenciaApp.app'. It includes instructions to configure the SDK, a 'Ver las instrucciones del SDK' button, and a download link for 'google-services.json'. Below this, it shows the app ID (1:422692884929:android:888a7a0508bb87dc7a9f40), the package name (com.valenciaApp.app), and SHA-1 fingerprints (86:b5:96:ed:aa:12:1a:4a:83:e1:8a:d7:c5:00:eb:ae:dc:5b:63:15). A 'Agregar huella digital' button is also present. At the bottom right is a 'Quitar esta app' button.

Y esta sería la función que implementa el login;

```
async loginGoogleAndroid() {
  const res = await this.googlePlus.login({
    'webClientId': '422692884929-ug8m3k72srvquddgi0om4at1ggoki2os.apps.googleusercontent.com',
    'offline': true
  });

  console.log(res);
}
```

2. COMPONENTES DEL PROYECTO

TECNOLOGÍAS UTILIZADAS

A lo largo de este documento, se han mencionado diversas tecnologías que han sido utilizadas en el desarrollo de este proyecto. A continuación, se presenta un resumen de cada una de ellas:

1. **Ionic**: Un framework de desarrollo de aplicaciones móviles híbridas que utiliza tecnologías web como **HTML**, **SCSS** y **TypeScript** para crear aplicaciones multiplataforma.
2. **HTML**: Lenguaje de marcado utilizado para estructurar el contenido de las páginas web.
3. **SCSS**: Lenguaje de estilos utilizado para darle diseño y apariencia visual a las páginas web.
4. **TypeScript**: Un superset de JavaScript que añade características adicionales y tipado estático al lenguaje.
5. **Angular**: Un framework de desarrollo web basado en TypeScript que permite crear aplicaciones web de una manera estructurada y modular.
6. **Firebase**: Una plataforma de desarrollo de aplicaciones móviles y web que proporciona una variedad de servicios en la nube, incluyendo almacenamiento de datos en tiempo real, autenticación de usuarios, notificaciones push, entre otros.
7. **Firestore**: Un servicio de base de datos no relacional proporcionado por Firebase, que permite almacenar y sincronizar datos en tiempo real entre diferentes dispositivos y plataformas.
8. **Google Sign-In**: Un servicio de autenticación proporcionado por Firebase que permite a los usuarios iniciar sesión en la aplicación utilizando sus cuentas de Google.



3 . JUSTIFICACIÓN Y OBJETIVOS

Para que se propone el proyecto?

El presente proyecto tiene como objetivo principal **mejorar la experiencia de búsqueda** y descubrimiento de planes y eventos en la comunidad valenciana, al tiempo que **promueve y apoya el comercio local** de la ciudad.

A través de la aplicación, se incentiva a los usuarios a **explorar** nuevas opciones gastronómicas y participar en eventos locales, fomentando así el apoyo y crecimiento de la economía local.

En resumen, ValenciaApp se presenta como una solución integral para aquellos que desean descubrir y disfrutar de una amplia gama de opciones gastronómicas y eventos en Valencia. Al ofrecer una plataforma fácil de usar y con información concisa.

¿A quien va dirigido?

El proyecto ValenciaApp está dirigido principalmente a los **residentes** de la ciudad de **Valencia**, con un enfoque especial en el público **joven**.

Además, el proyecto también puede ser de interés para los visitantes y turistas que deseen alejarse de los **destinos turísticos convencionales** y descubrir lugares auténticos y eventos únicos en Valencia.

En resumen, el proyecto está dirigido a personas que buscan una **alternativa** a los restaurantes turísticos tradicionales, así como a aquellos que desean estar **informados** sobre los eventos y actividades que tienen lugar en la ciudad.

¿Que se pretende alcanzar?

A niveles globales, podría decir, que se pretende alcanzar el fomento de aplicaciones móviles para realizar actividades en ciudades de interés, ya que si hablamos de una forma más global, se podría aplicar en muchas otras ciudades y esto ayudaría mucho a que los comercios tuvieran mayor visibilidad tanto para gente de la ciudad como para turistas.

4 . DESARROLLO DEL PROYECTO

Metodologías utilizadas

En el desarrollo de este proyecto, he intentado ceñirme a los conocimientos que tengo sobre la gestión de proyectos, en base a la experiencia que he ido ganando tanto a nivel académico como profesional en la FCT, y teniendo siempre como base la metodología ágil, he intentado, en la medida de lo posible ajustarme a esta.

Al ser un trabajo totalmente individual, he adaptado estas bases que plantea la metodología ágil, a una persona individual, ya que inicialmente están planteadas para trabajar en equipo,

Con esto quiero decir que he intentado organizarme en ciertos sprints de tiempo, y fijar ciertos objetivos con fechas predefinidas.

Es decir, priorizando las diferentes competencias demandadas en la entrega del proyecto, y fijando objetivos sobre cada una de ellas, tanto objetivos de tiempo como de nivel de evolución de trabajo,

Cabe decir que se nota mucho al trabajar una única persona y tener que auto gestionar tú mismo el nivel de exigencia y de objetivos demandadas.

Porque al no tener diversas opiniones como seria en el caso de un trabajo en equipo, cuesta más diferenciar hasta que punto lo estas haciendo bien o si tal vez deberías exigir más.

Resumiendo, la metodología utilizada como digo anteriormente , ha sido una metodología ágil adaptada a una persona individual.



4 . DESARROLLO DEL PROYECTO

Problemas/Dificultades

En todo proyecto es común enfrentarse a diversos desafíos y obstáculos que pueden surgir a lo largo del desarrollo. En el caso de este proyecto, se han identificado varios problemas significativos que han impactado su desarrollo.

El principal problema ha sido la limitación de **tiempo**. El hecho de querer utilizar una tecnología nueva y aprenderla en el proceso ha requerido un tiempo considerable. Esto ha supuesto un desafío para establecer una continuidad en el desarrollo del proyecto, ya que fue necesario adquirir conocimientos previos antes de poder comenzar a trabajar de manera más seria y comprometida.

Además, a nivel técnico, han surgido diversos problemas como errores de compilación, planteamiento inicial de la lógica y errores en la programación en general de la aplicación.

Estos problemas son frecuentes en el desarrollo de software y han requerido dedicación adicional para su resolución.

Es importante destacar que la aparición de estos problemas no ha sido inesperada, ya que es normal enfrentarse a desafíos técnicos durante el proceso de desarrollo.

A pesar de estos inconvenientes, se ha trabajado para superarlos y avanzar en el desarrollo del proyecto.



Resultados obtenidos

A nivel **personal**, el desarrollo de este proyecto ha sido una experiencia enriquecedora y de aprendizaje significativo. Durante el proceso, he adquirido habilidades y conocimientos que van más allá del aspecto técnico de la aplicación.

En primer lugar, he aprendido a desarrollar una aplicación desde cero, lo cual ha implicado comprender los diferentes aspectos involucrados en el proceso de desarrollo de software. Esto incluye la planificación, la organización de tareas, la gestión del tiempo y la resolución de problemas. He tenido la oportunidad de enfrentar desafíos y superar obstáculos, lo que ha contribuido a mi crecimiento personal y profesional.

Además, he adquirido habilidades en la gestión del estrés y la presión. El desarrollo de un proyecto individualmente requiere una buena gestión de las emociones y la capacidad de manejar situaciones adversas. A lo largo del proceso, he aprendido a mantener la calma y a encontrar soluciones efectivas para los problemas que han surgido.

En términos más **profesionales**, he obtenido un conocimiento profundo de la tecnología Ionic. A través de este proyecto, he explorado las funcionalidades y características de Ionic, comprendiendo su estructura y su integración con otras tecnologías. Esto me ha permitido adquirir una base sólida en el desarrollo de aplicaciones móviles utilizando Ionic y estar preparado para enfrentar futuros desafíos en este campo.

El desarrollo de este proyecto ha sido una valiosa experiencia personal y profesional.



CONCLUSIONES

Al concluir este proyecto, se pueden extraer varias conclusiones importantes. A continuación, destacaré algunas de las principales conclusiones a raíz de esta experiencia:

1. Aprendizaje:

El desarrollo de este proyecto ha sido una oportunidad para adquirir **conocimientos prácticos** en el desarrollo de aplicaciones móviles. He tenido la posibilidad de aplicar los conceptos teóricos aprendidos y enfrentarme a desafíos reales, lo que ha contribuido a un gran aprendizaje.

2. Gestión del tiempo:

La realización de un proyecto individual requiere una buena gestión del tiempo y una adecuada organización de las tareas. He aprendido a establecer prioridades, planificar adecuadamente y administrar mi tiempo de manera eficiente para cumplir con los objetivos establecidos.

3. Resolución de problemas

Durante el desarrollo del proyecto, me he enfrentado a diversos problemas y obstáculos. Esta experiencia me ha enseñado a ser proactiva en la búsqueda de soluciones, a investigar, experimentar y aprender de los errores. La resolución de problemas se ha convertido en una habilidad clave que se desarrolla a base de resolverlos continuamente.

4. Mejora de habilidades técnicas:

A través de este proyecto, he mejorado mis habilidades técnicas en el desarrollo de aplicaciones móviles, especialmente en el uso de tecnologías como Ionic y Firebase. He ampliado mi conocimiento en estas herramientas y he adquirido experiencia práctica que me será útil en futuros proyectos.

En conclusión, el desarrollo de este proyecto ha sido una experiencia enriquecedora que ha contribuido a mi crecimiento personal y profesional.

5. LÍNEAS FUTURAS DE TRABAJO

A partir de este proyecto y considerando su desarrollo y resultados, se pueden identificar varias líneas futuras de trabajo para mejorar y expandir la aplicación ValenciaApp.

Algunas de estas líneas podrían ser:

1. **Modo claro/oscuro:** Implementar la opción de cambiar entre un tema claro y oscuro en la aplicación para adaptarse a las preferencias de los usuarios y mejorar la experiencia visual.
2. **Reservar directamente en los restaurantes:** Integrar una función de reserva en la aplicación, permitiendo a los usuarios realizar reservas directamente desde la app, agilizando el proceso y facilitando la gestión de reservas para los restaurantes.
3. **Apartado de opiniones de otros usuarios:** Permitir a los usuarios dejar reseñas y opiniones sobre los restaurantes y eventos que han visitado, brindando a otros usuarios información adicional y ayudando en la toma de decisiones.
4. **Filtros de búsqueda:** Ampliar las opciones de filtrado en la búsqueda de restaurantes y eventos. Esto puede incluir filtros por precio, tipo de comida, ubicación, categoría de evento, entre otros, para facilitar la búsqueda de acuerdo a las preferencias individuales de los usuarios.
5. **Apartado de favoritos:** Agregar una sección de favoritos donde los usuarios puedan guardar y acceder fácilmente a los restaurantes y eventos que les interesen, permitiendo una experiencia personalizada y la posibilidad de seguir de cerca sus lugares favoritos.
6. **Opción de compartir:** Habilitar la opción de compartir restaurantes o eventos a través de diversas plataformas, como redes sociales o mensajería, para que los usuarios puedan recomendar y compartir descubrimientos con sus amigos y contactos.
7. **Enlace a la compra de entradas:** Para eventos que requieran la compra de entradas, incluir un enlace directo a la plataforma de compra de entradas correspondiente, facilitando a los usuarios acceder y adquirir sus boletos de manera rápida.
8. **Ampliación de datos del usuario:** Permitir a los usuarios agregar más información a su perfil, como una foto de perfil, intereses adicionales u otros detalles relevantes, para personalizar aún más la experiencia y crear un sentido de comunidad en la aplicación.

Estas mejoras sin duda enriquecerán la funcionalidad y experiencia de la aplicación ValenciaApp, proporcionando más opciones y comodidades para los usuarios.

5. BIBLIOGRAFIA / WEBGRAFIA

A continuación, se mencionan algunas de las fuentes web utilizadas durante el desarrollo del proyecto:

- Página oficial de Ionic: <https://ionicframework.com/>
- Documentación de Ionic: <https://ionicframework.com/docs>
- Página oficial de Angular: <https://angular.io/>
- Documentación de Angular: <https://angular.io/docs>
- Documentación de Firebase: <https://firebase.google.com/docs>
- Stack Overflow: <https://stackoverflow.com/>

Estas fuentes proporcionaron información detallada sobre las tecnologías utilizadas, ejemplos de código, soluciones a problemas comunes y mejores prácticas. Además, se consultaron otros recursos en línea, tutoriales y blogs relevantes para obtener información adicional y ampliar el conocimiento sobre el desarrollo de aplicaciones móviles y el uso de Firebase. Cabe mencionar que, a lo largo del proceso, también se realizaron búsquedas en línea específicas para resolver problemas técnicos y obtener orientación sobre implementaciones particulares.