

1. Introducción y Resumen del Proyecto

Elemento	Descripción
Nombre del Proyecto	El Sazon
Propósito	Documentar la arquitectura, el desarrollo, la configuración y el mantenimiento de la aplicación móvil.
Plataforma Objetivo	iOS (generado vía EAS).
Funcionalidades Clave	Login, Registro, Visualización de Recetas, Favoritos, Creación de Recetas, Perfil de Usuario.

2. Pila Tecnológica

Esta sección es el centro del manual técnico, ya que define todas las herramientas y versiones utilizadas.

2.1. Frontend (Aplicación Móvil)

Componente	Tecnología	Versión Específica
Framework Base	React Native	0.73.6
Plataforma de Desarrollo	Expo	SDK 54.0.0
Navegación	Tab bars	
Lenguaje	JavaScript / TypeScript (indicar cuál) N/A	

2.2. Backend y Servicios

Componente	Tecnología	Versión Específica
Backend as a Service (BaaS)	Firebase	v8.

Componente	Tecnología	Versión Específica
Base de Datos	Firestore Database	N/A
Autenticación	Firebase Authentication	N/A
Almacenamiento	Firestore Database	N/A

3. Entorno de Desarrollo y Flujo de Trabajo

3.1. Requisitos de Configuración

- **IDE:** Visual Studio Code.
- **Gestor de Paquetes:** npm
- **Node.js:** 16.20.2 o más recientes
- **CLI de Expo:** EAS CLI (npm install -g eas-cli).

3.2. Proceso de Creación y Despliegue (Build Pipeline)

1. **Inicialización:** El proyecto se inició usando **Expo Snack** y luego se descargó el ZIP.
2. **Preparación Local:** Instalación de dependencias con npm install.
3. **Configuración de Firebase:** Se requiere el archivo de configuración de Firebase (firebaseConfig.js)

4. Generación de Builds (iOS Preview):

- Se utiliza **EAS** (Expo Application Services).
 - Comando: `eas build -p ios --profile preview`
 - **Resultado:** Generación de un archivo **.ipa** o un enlace de *preview* que se sube a la plataforma de Expo.
-

4. Arquitectura de la Aplicación (Código y Navegación)

4.1. Estructura de Carpetas

Describir la organización típica de React Native:

- `/src/screens`: Contiene los componentes de pantalla principales.
- `/src/components`: Componentes reutilizables (Botones, Tarjetas de Receta).
- `/src/navigation`: Configuración de **React Navigation**.
- `/src/database`: Lógica de comunicación con **Firebase**.

4.2. Flujo de Navegación (React Navigation)

El flujo de navegación se gestiona con una arquitectura de *Stack Navigator* y/o *Tab Navigator*.

- **Flujo de Autenticación (Auth Stack):**
 1. LoginScreen
 2. RegisterScreen
- **Flujo Principal (App Tabs o Main Stack):**
 1. InicioScreen (Lista de Recetas)
 2. DetalleScreen
 3. FavoritosScreen
 4. AgregarScreen
 5. UsuarioScreen
- **Lógica de Redirección:** Se usa un **Listener** de **Firestore Authentication** para redirigir al usuario entre el Auth Stack y el App Tabs al iniciar/cerrar sesión.

5. Modelo de Datos y Seguridad (Firebase)

5.1. Firebase Authentication

- **Mecanismo:** Autenticación por **Email y Contraseña**.
- **Funciones Principales:** `signUp`, `signIn`, `signOut`.
- **Datos en el Registro:** El proceso de registro crea la cuenta en Firebase Auth y, **simultáneamente**, un documento inicial en la colección `users` de Firestore.

5.2. Modelo de Datos de Firestore

Colección		Propósito	Campos Clave (Ejemplo)
<code>users</code>	Almacena la información de perfil.		<code>uid</code> (ID de Firebase Auth), <code>nombre</code> , <code>email</code> , <code>imagenURL</code> (el link de la imagen).
<code>recipes</code>	Almacena todas las recetas de la aplicación.		<code>recipeId</code> , <code>title</code> , <code>ingredients</code> , <code>steps</code> , <code>authorId</code> , <code>createdAt</code> .
<code>favorites</code>	Almacena las recetas marcadas por el usuario.		<code>favoriteId</code> , <code>userId</code> , <code>recipeId</code> .

5.3. Reglas de Seguridad de Firestore

Es fundamental incluir una descripción de las reglas de Firestore que garantizan:

1. **Acceso a users:** Un usuario solo puede leer o modificar su propio documento (donde `request.auth.uid == resource.data.uid`).
2. **Acceso a recipes:** Lectura pública (`read`) y escritura solo para usuarios autenticados (`write` para agregar nuevas recetas).
3. **Acceso a favorites:** Solo el usuario autenticado puede agregar o eliminar favoritos.

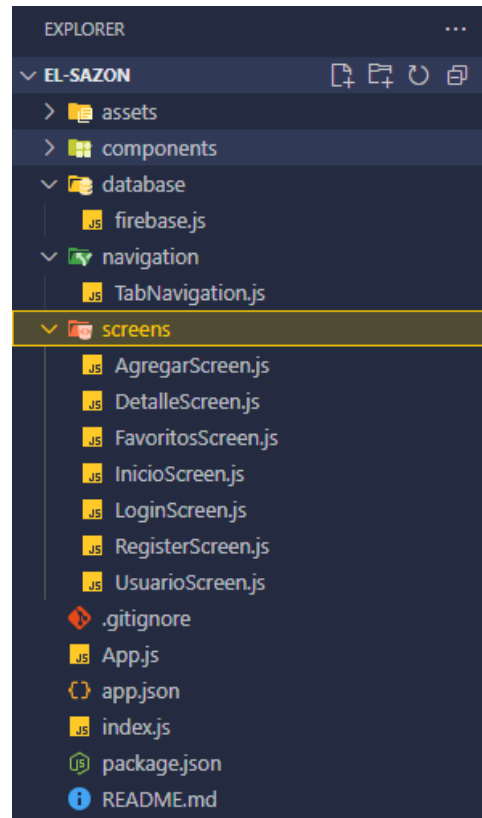
Aqui tenemos las licencias del package.json

Aqui tenemos los screens

```

package.json X
package.json > ...
1  {
2    "license": "0BSD",
3    "main": "index.js",
4    "scripts": {
5      "start": "expo start",
6      "android": "expo start --android",
7      "ios": "expo start --ios",
8      "web": "expo start --web"
9    },
10   "dependencies": {
11     "expo": "~54.0.13",
12     "expo-status-bar": "~3.0.8",
13     "react": "19.1.0",
14     "react-native": "0.81.4",
15     "firebase": "8.2.3",
16     "expo-constants": "~18.0.9",
17     "@expo/vector-icons": "^15.0.2",
18     "react-native-paper": "3.6.0",
19     "react-native-screens": "~4.16.0",
20     "react-native-elements": "*",
21     "@react-navigation/stack": "*",
22     "@react-navigation/native": "*",
23     "react-native-gesture-handler": "~2.28.0",
24     "@react-navigation/bottom-tabs": "*",
25     "react-native-safe-area-context": "~5.6.0",
26     "@react-native-community/masked-view": "0.1.10"
27   },
28   "private": true
29 }

```





```

firebase.js X
database > firebase.js
1  import firebase from "firebase";
2  import "firebase/firestore";
3
4  const firebaseConfig = {
5    apiKey: "AIzaSyCdyHOLS8rRP0jP493ISE-fh00hEM375jw",
6    authDomain: "appiosrecetas.firebaseio.com",
7    projectId: "appiosrecetas",
8    storageBucket: "appiosrecetas.firebaseio.com",
9    messagingSenderId: "782273903015",
10   appId: "1:782273903015:web:7c2d3182ff04449c481e82",
11   measurementId: "G-PWTF4542W6"
12 };
13
14 firebase.initializeApp(firebaseConfig);
15
16 const db = firebase.firestore();
17
18 export default {
19   firebase,
20   db
21 };

```

Aquí tenemos la base de datos firebase

Build	Commit	Profile	Runtime	Channel
  iOS simulator build 1.0.0 (1)	003698c*	preview	None	None

Aqui armamos el APK para simularlo en los

Builds

>

a723f242

EAS Workflows

Automate your builds, submissions, updates, and more with a CI/CD built for app developers.

Learn More

ios simulator build

003698c · Initial commit

Profile	Environment	SDK version	Version	Fingerprint	Commit	Created by
preview	production	54.0.0	1.0.0 (1)	7e521e0	003698c*	Jr Montoya jrmontoya490

Build artifact

APP

Download

Open with Orbit

Status	Start time	Wait time	Queue time	Build time	Total time	Availability
Finished	Oct 13, 2025 5:04 PM	0s	49s	3m 53s	4m 42s	29 days

Logs

>

Waiting to start

5s

>

Spin up build environment

43s

>

Read package.json

1s

>

Install dependencies

18s

>

Read app config

1s

>

Run expo doctor

1s

>

Prebuild

2s

>

Install pods

21s

>

Bundle JavaScript

9s

>

Run fastlane

2m 51s

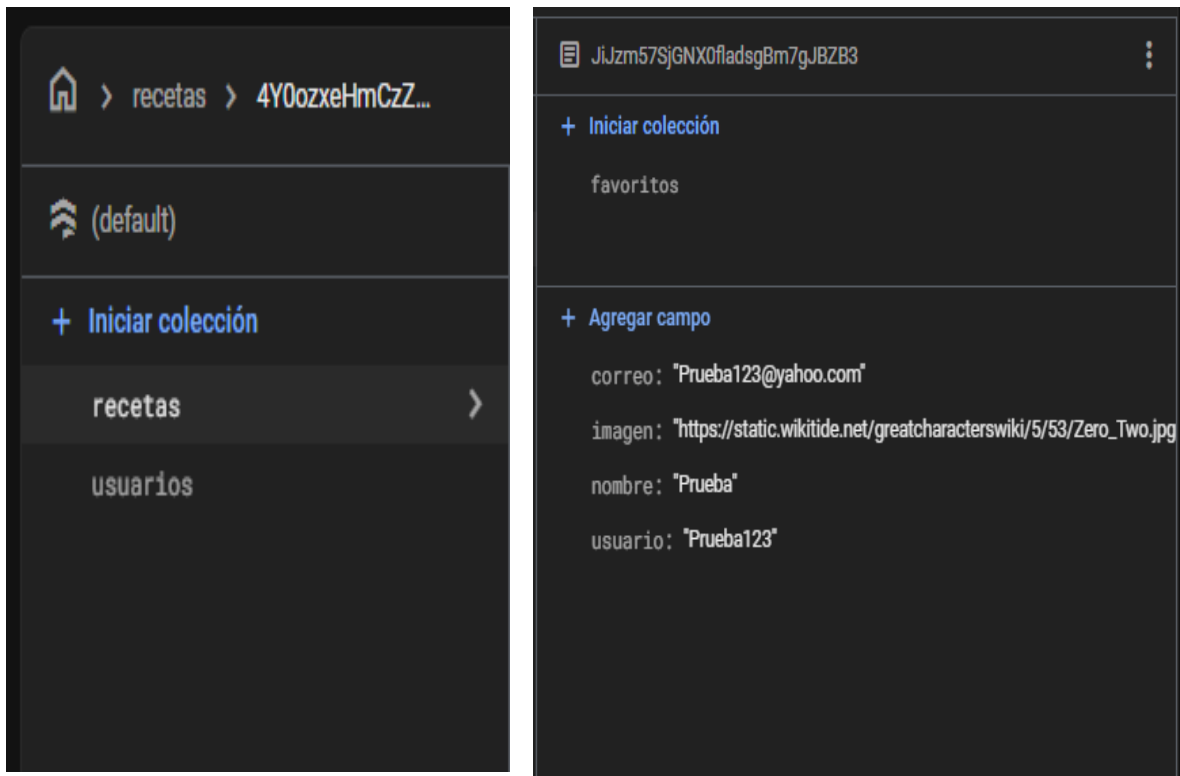
>

Upload application archive

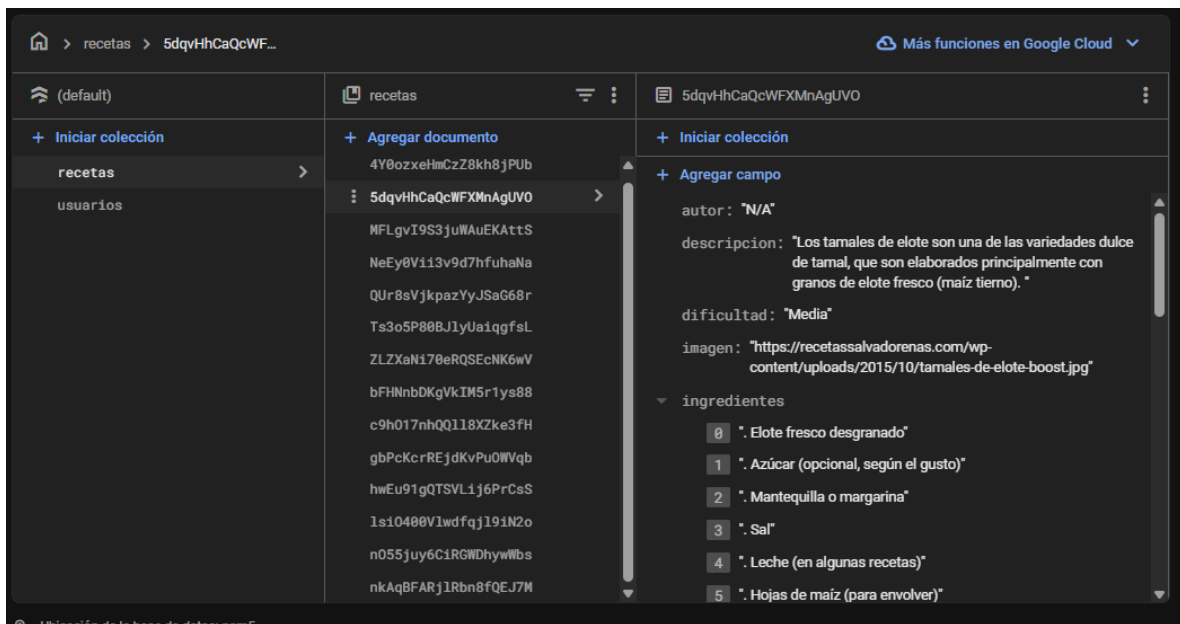
2s

>



Xcode Logs



Aqui tenemos las colecciones de la base de datos con los datos ingresado



Tenemos el apartado de las recetas ya ingresadas a la base de datos

Proveedores de acceso	
<div>Agregar proveedor nuevo</div>	
Proveedor	Estado
 Correo electrónico/contraseña	 Habilitada

Habilitamos el permiso de correo como contraseña para poder ingresar nuestro correo personal

```
$ npm install firebase
```

Luego, inicializa Firebase y comienza a usar los SDK de los productos que quieres utilizar.

```
// Import the functions you need from the SDKs you need
import { initializeApp } from "firebase/app";
import { getAnalytics } from "firebase/analytics";
// TODO: Add SDKs for Firebase products that you want to use
// https://firebase.google.com/docs/web/setup#available-libraries

// Your web app's Firebase configuration
// For Firebase JS SDK v7.20.0 and later, measurementId is optional
const firebaseConfig = {
  apiKey: "AIzaSyCdyHOLS8rRP0jP493ISE-fh00hEM375jw",
  authDomain: "appiosrecetas.firebaseio.com",
  projectId: "appiosrecetas",
  storageBucket: "appiosrecetas.firebaseio.com",
  messagingSenderId: "782273903015",
  appId: "1:782273903015:web:7c2d3182ff04449c481e82",
  measurementId: "G-PWTF4542W6"
};

// Initialize Firebase
const app = initializeApp(firebaseConfig);
const analytics = getAnalytics(app);
```

Tenemos el apartado de lo que se generó de la base de datos

```
eas.json X
eas.json > ...
1 {
2   "build": {
3     "preview": {
4       "ios": {
5         "simulator": true
6       }
7     },
8     "production": {}
9   }
10 }
11
```

Y esto es lo que se genera al loguearse con eas lofin
nos da un eas.json para simularlo