Manual Técnico - HealthyLife

Índice - Manual Técnico HealthyLife

1.Introducción	2
2.Tecnologías Utilizadas	
3.Estructura del Proyecto	2
4.Instalación y Configuración	4
4.1 Requisitos Previos	4
4.2 Instalación	5
4.3 Configuración de Firebase	5
5.Funcionalidades y Pantallas	
5.1 Pantalla de Inicio	5
5.2.1 Pantalla de Login	5
5.2.2 Pantalla de Registro	5
5.3 Menú Principal (Drawer)	5
5.4 Pantalla de Hidratación	5
5.5 Pantalla de Ejercicio	5
5.6 Pantalla de Sueño	6
5.7 Pantalla de Dietas	6
5.8 Pantalla de Configuraciones	
5.9 Pantalla de Perfil	6
5.10 Pantalla de Firebase	6
6.Uso de Firebase y Base de Datos	6
7. Manejo de Errores y Depuración	7
8.Despliegue y Mantenimiento	7
9.Problemas y mejoras	7
10.APK Ubicación y videos de prueba	8
11.Android	8
12.Bibliografía	12

1. Introducción

HealthyLife es una aplicación desarrollada en Flutter que permite a los usuarios registrar y monitorear hábitos saludables, como la hidratación, el ejercicio, el sueño y la alimentación. Se apoya en Firebase para la autenticación de usuarios y almacenamiento de datos, además de integrar una API externa para la búsqueda de dietas.

2. Tecnologías Utilizadas

- Flutter (Framework principal para la UI)
- Firebase Authentication (Gestón de usuarios)
- Cloud Firestore (Almacenamiento en la nube)
- fl chart (Visualización de datos con gráficos)
- Provider (Gestor de estado)
- HTTP & URL Launcher (Conexión a API externas y apertura de enlaces)

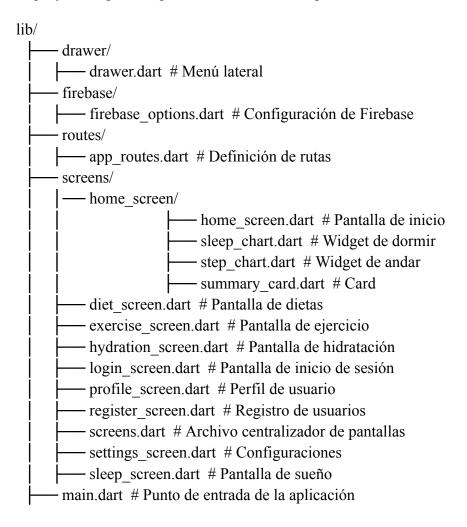
Adjunto código usado en el pubspec.yaml:

```
name: flutter healthylife
description: "A new Flutter project."
publish_to: 'none'
version: 1.0.0+1
 sdk: ^3.5.3
dependencies:
 flutter:
  sdk: flutter
 provider: ^6.0.5
 google fonts: ^6.2.1
 cupertino_icons: ^1.0.8
```

```
intl: ^0.20.1
 cloud_firestore: ^5.6.2
 timezone: ^0.10.0
 http: ^1.3.0
 url_launcher_web: ^2.0.11
 fl_chart: ^0.70.2
dev_dependencies:
  sdk: flutter
flutter:
 uses-material-design: true
 # Aquí se añaden los assets, incluyendo la ruta de la imagen si fuese necesario
```

3. Estructura del Proyecto

El proyecto sigue la siguiente estructura de carpeta lib en Flutter:



4. Instalación y Configuración

4.1 Requisitos Previos

- Tener Flutter SDK instalado
- Tener Android Studio y VS Code configurado
- Tener Firebase configurado en el proyecto

4.2 Instalación

Ejecutar en la terminal:

flutter pub get # Instalar dependencias flutter run # Ejecutar la aplicación Si se requiere compilar un APK:

flutter build apk

4.3 Configuración de Firebase

- 1. Crear un proyecto en Firebase Console.
- 2. Descargar y colocar el archivo google-services.json en android/app/.
- 3. Habilitar Authentication y Firestore Database.

5. Funcionalidades y Pantallas

5.1 Pantalla de Inicio

- Resumen de los hábitos diarios.
- Estadísticas semanales.
- Consejos de salud.
- Retos
- La pantalla se ha dividido en varios codigos dentro de la carpeta home screen

5.2.1 Pantalla de Login

• Autenticación de login con Firebase Authentication.

5.2.2 Pantalla de Registro

• Registro con Firebase Authentication

5.3 Menú Principal (Drawer)

• Navegación a distintas secciones.

5.4 Pantalla de Hidratación

- Registro de vasos de agua consumidos diarios.
- Consejos

5.5 Pantalla de Ejercicio

- Registro de actividad física.
- Registro de pasos diarios

5.6 Pantalla de Sueño

- Registro de horas dormidas.
- Consejos de descanso.

5.7 Pantalla de Dietas

- Búsqueda de dietas saludables desde una API externa.
- Filtros por tipo de dieta.
- Registro de comidas y registro de comidas favoritas.
- Nos permite visualizar la receta en un enlace externo.

5.8 Pantalla de Configuraciones

- Tema claro/oscuro.
- Ajuste de tamaño de fuente y tipo de letra.
- Se guarda en firestore los cambios.

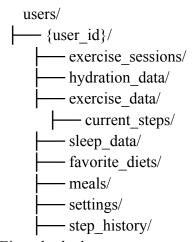
5.9 Pantalla de Perfil

• Tuve que crear una clase perfil donde podemos cambiar la contraseña, hacer log out y añadir los creditos de la app.

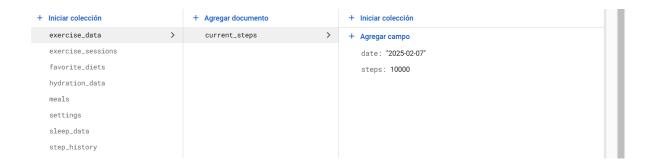
5.10 Pantalla de FireBase

• Es una pantalla generada por FlutterFire CLI.

6. Uso de Firebase y Base de Datos



Ejemplo de documento en current steps:



7. Manejo de Errores y Depuración

Para depurar el código:

flutter analyze # Detecta errores en el código flutter logs # Muestra logs en tiempo real

Se recomienda usar try-catch en las llamadas a Firebase para manejar errores correctamente.

8. Despliegue y Mantenimiento

Para generar una versión lista para producción:

flutter build apk --release

Para actualizar dependencias:

flutter pub upgrade

9. Problemas y mejoras

- Problemas:
- He tenido que usar "const" en textos.
- He tenido un problema para implementar el icono y el nombre, debido a ios, a pesar de que tengo completamente habilitados los permisos.

```
PS C:\Users\AlvaroDocuments\flutter\flutter_HealthyLife> flutter pub run flutter_launcher_icons
Deprecated. Use 'dart run' instead.

FLUTTER LAUNCHER ICONS (v0.13.1)

AWarning: flutter_icons has been deprecated please use flutter_launcher_icons instead in your yaml files

• Creating default icons Android

• Overwriting the default Android launcher icon with a new icon

• Overwriting the default Android launcher icon with new icon

• Could not generate launcher icons

**PathAccessException: Cannot open file, path = 'ios/Runner/Assets.xcassets/AppIcon.appiconset/Icon-App-20x20@1x.png' (OS Error: Acceso denegado., erron = 5)

**FathaccessException: Cannot open file, path = 'ios/Runner/Assets.xcassets/AppIcon.appiconset/Icon-App-20x20@1x.png' (OS Error: Acceso denegado., erron = 5)
```

• Mejoras:

- Añadir en un futuro mejores métodos de registro y de eliminar cuenta.
- Mejorar aún más la pantalla de dietas
- Poder incorporar una pulsera inteligente para suministrar mejor los datos.

10. APK Ubicación y videos de prueba

El APK está ubicado en la carpeta build "C:\Users\Alvaro\Documents\flutter\flutter_HealthyLife\build\app\outputs\flutter-apk" donde se generá habitualmente, también he subido dos videos a una carpeta llamada video-prueba donde se me muestra usando la app con el emulador y abriendo el apk desde el emulador (son cortos).

11.Android

He tenido que cambiar permisos en android en:

Este es el código:

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android">
<uses-permission android:name="android.permission.RECEIVE_BOOT_COMPLETED" />
 <uses-permission android:name="android.permission.FOREGROUND SERVICE" />
 <uses-permission android:name="android.permission.POST NOTIFICATIONS" />
   android:label="flutter_application_1"
   android:name="${applicationName}"
   android:icon="@mipmap/ic launcher">
     android:name=".MainActivity"
     android:exported="true"
     android:launchMode="singleTop"
     android:taskAffinity=""
     android:theme="@style/LaunchTheme"
```

```
android:configChanges="orientation|keyboardHidden|keyboard|screenSize|smallestScreenSize|locale|layou
tDirection|fontScale|screenLayout|density|uiMode"
       android:hardwareAccelerated="true"
       android:windowSoftInputMode="adjustResize">
       <meta-data
         android:name="io.flutter.embedding.android.NormalTheme"
         android:resource="@style/NormalTheme"
      <intent-filter>
         <action android:name="android.intent.action.MAIN" />
         <category android:name="android.intent.category.LAUNCHER" />
       </intent-filter>
    <!-- Requerido para la generación de plugins -->
    <meta-data
       android:name="flutterEmbedding"
       android:value="2"/>
    <!-- Configuración de notificaciones programadas -->
       android:name="com.dexterous.flutterlocalnotifications.ScheduledNotificationReceiver"
       android:exported="false" />
```

```
android:name="com.dexterous.flutterlocalnotifications.ActionBroadcastReceiver"
    android:exported="false" />
    android:name="com.dexterous.flutterlocalnotifications.BootReceiver"
    android:exported="false">
    <intent-filter>
      <action android:name="android.intent.action.BOOT_COMPLETED" />
      <action android:name="android.intent.action.MY_PACKAGE_REPLACED" />
    </intent-filter>
 <service
    android:name="com.dexterous.flutterlocalnotifications.ForegroundService"
    and roid: permission = "and roid.permission.FOREGROUND\_SERVICE"
    android:exported="false">
</application>
    <action android:name="android.intent.action.PROCESS_TEXT" />
    <data android:mimeType="text/plain" />
    <action android:name="android.intent.action.VIEW" />
```

12. Bibliografía

Flutter. (2024). Documentación oficial de Flutter. Recuperado de https://flutter.dev/docs

Firebase. (2024). Documentación de Firebase. Recuperado de https://firebase.google.com/docs

Dart. (2024). Guía de programación en Dart. Recuperado de https://dart.dev/guides

fl_chart. (2024). Documentación de fl_chart. Recuperado de https://pub.dev/packages/fl_chart

Provider. (2024). Manejo de estado con Provider. Recuperado de https://pub.dev/packages/provider

Google. (2024). Material Design Guidelines. Recuperado de https://material.io/design

Android Developers. (2024). Documentación sobre permisos y configuración en AndroidManifest.xml. Recuperado de https://developer.android.com/guide/topics/manifest

Stack Overflow. (2024). Solución de errores comunes en Flutter y Firebase. Recuperado de https://stackoverflow.com/questions/tagged/flutter+firebase