|  |
| --- |
| **CODE HUNTERS** |
| The MedAide Project |
| The text file contains the documentation and description of our project, MedAide, along with its contents and ways to host our project on a localhost. |
|  |

**INDEX**

|  |  |  |
| --- | --- | --- |
| **SL.NO** | **SUB-TITLE** | **PAGE NUMBER** |
| **1** | **Description** | 2 |
| **2** | **Contents** | 3 |
| **3** | **Methods to Host** | 5 |

Description

Our project was developed with an aim to ease the task of searching and fixing appointments with leading doctors. It allows users to enter a particular location and search for hospitals near that location. In addition, it keeps track of the user’s medical history which is necessary for future appointments. The feature of an emergency dial makes initial respondence to a crisis easier. It is also equipped with the feature of providing messages to motivate users to take care of their health.

When a user opens the application for the first time, it prompts the user to create a profile so that it can keep track of his activities. This prompt never repeats once the user creates his profile.

In the homepage, the user is greeted by motivational quotes and inspirational messages to strengthen his hope. He can search for doctors and fix an appointment with the chosen doctor. Due to constraints, the appointment page is still under construction.

The user will also find an emergency dial in the home page. By default, the number which is to be dialed is set to 102.

The application has a feature of storing all medical history of the user to make the consulting easier. Once the profile is created, the user can upload all his previous medical reports. Having all medical reports together under the same application, makes it easier and more efficient for the user to keep track of his medical progress.

Contents

**Project Folder**

**Lib**

* main.dart
* models
* user.dart – defines the class User
* screens
* authenticate
* authenticate.dart – directing to the sign in page
* register.dart – shows the login screen and validates the details
* home
* home.dart – facilitates the display of the home screen
* home\_screen.dart – navigates between different features provided by our app
* profile.dart – allows users to enter profile details like name, location, phone no., etc.
* pages
* fixAppointment.dart – That feature is currently not available
* loading.dart
* results.dart – shows the details of hospitals near a particular location
* search.dart – allows the user to enter a location of his choice and filter the results based on the specialisation
* pdf
* modal.dart – defines a class Modal
* pdffirst.dart – used to upload pdf files to the firebase
* pdfsecond.dart – to access the pdf files from the firebase and display the data it contains
* wrapper.dart – checks whether it is a first time user or not
* services
* auth.dart – collects the user id from the firebase
* clinicsData.dart – class obtained after parsing the json data from the api
* location.dart - retrieves data from the api

**pubspec.yaml**: All plugins and dependencies are stated in this file.

Methods to host MedAide on Local System

The file which is hosted on GitHub should be cloned to the remote system and run using the Flutter Application.

Accordingly, the prerequisites are as follows:

* Flutter Application should be pre-installed in the system
* Andriod Studio or VS code also to be installed. If VS code is being used, then Genymotion Android Emulator is also required.
* The folder is opened in android studio and all dependencies are downloaded using pub get.
* Then the emulator/physical device is connected, run command “flutter run” in the terminal.

Instead of a virtual device , users can also connect a Physical device to run the MedAide app on local phone