

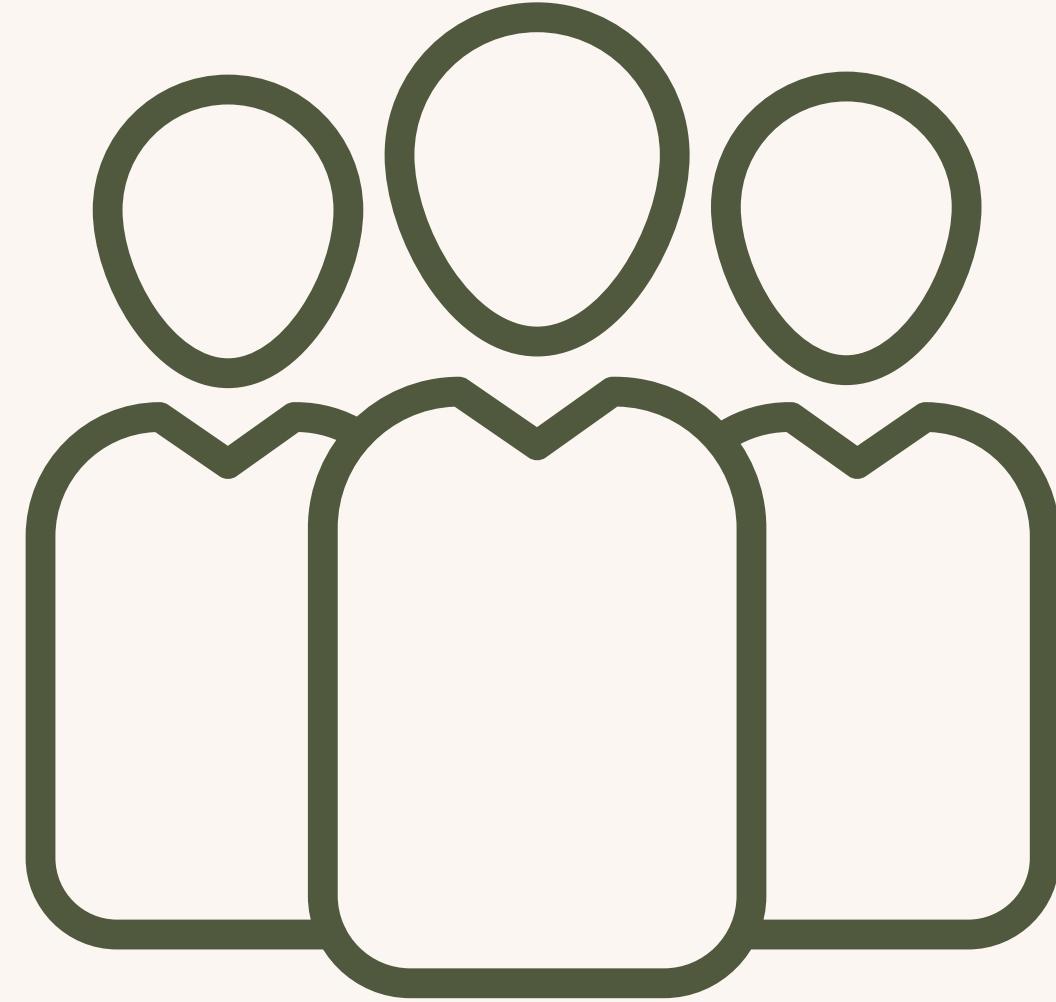


Your health,  
our mission



# About us

- Ahmed Gad
- Hesham Elsayed
- Abdelrahman Adly
- Amira Nasser
- Khlood Ahmed
- Salma Adel
- Sama Ahmed
- Esraa Hamdy



Supervisor

Dr: Amira Idris

# AGENDA

- **Introduction**

- **prolam**
- **solution**
- **motivation**
- **project idea**

- **Requirements**

- **Humen Requirements**
- **Software Requirements**
- **Hardware Requirements**



# AGENDA

- **Analysis and Design**

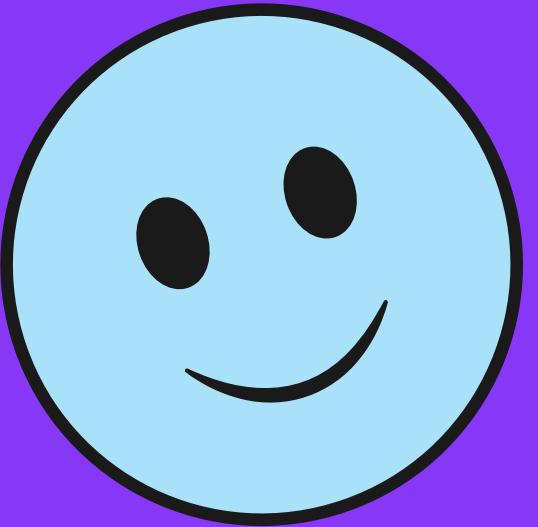
- Time line
- Feasibility Study
- Existing System
- App Design
- Database Design

- **App Development**

- Flutter Implementation
- Database Implementation



INTRO-  
DUCT-  
ION



# PROBLEMS

The number of patients with chronic diseases has increased and speeded among all different age generations.



# PROBLEMS

01.

**PROBLEM NO.1**  
40% of  
Arab have  
Diabetes.

04.

**PROBLEM NO.4**  
Egypt is the 1st county  
in the Arabian Ranking  
of the Diabetes  
Patients Numbers.

02.

**PROBLEM NO.2**  
26% of Arab  
Diabetes  
Patients Are

05.

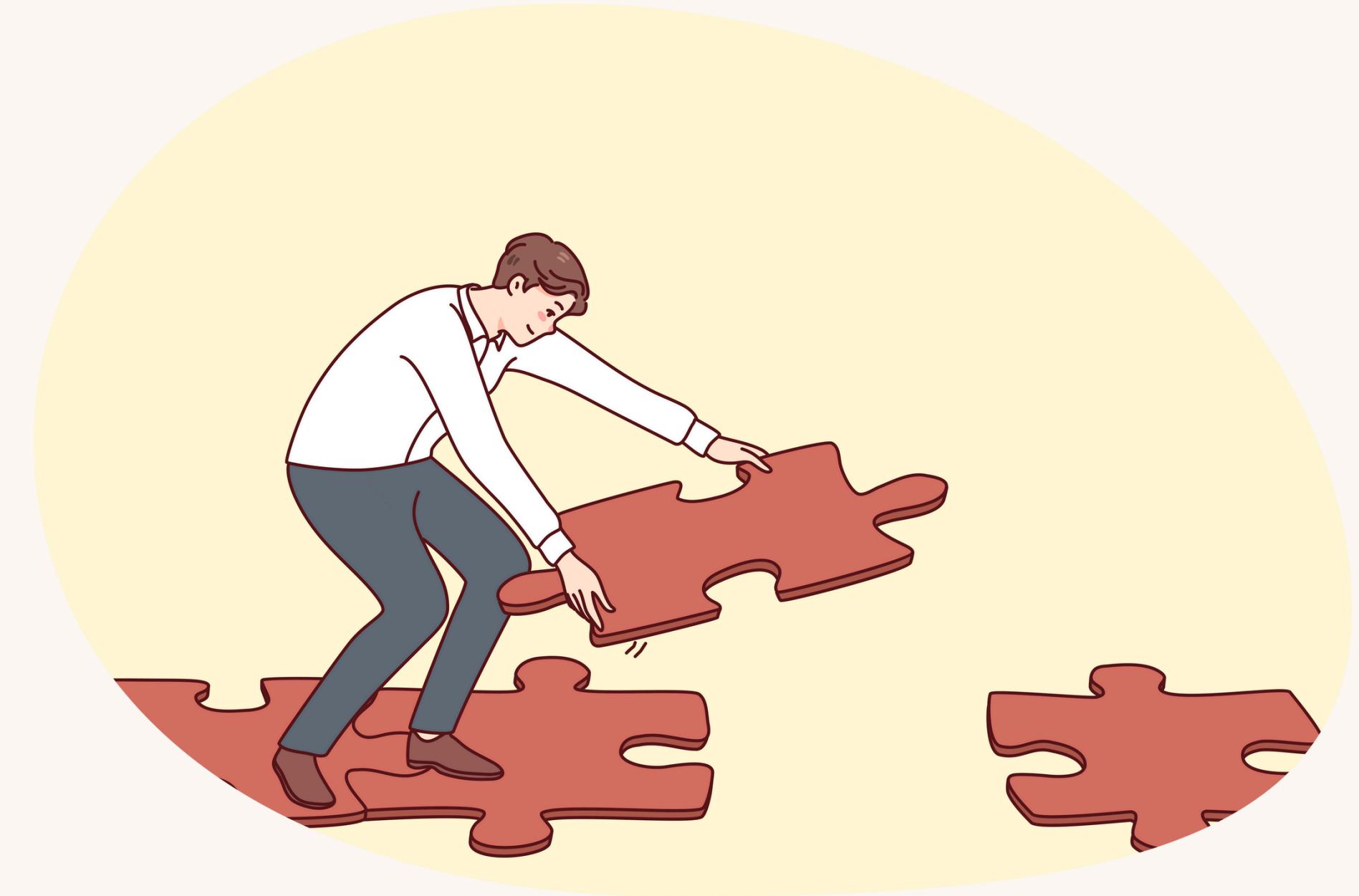
**PROBLEM NO.5**  
Patients are  
Taking medicines  
without medical  
reference

03.

**PROBLEM NO.3**  
8% are the ones who  
follow the treatment.

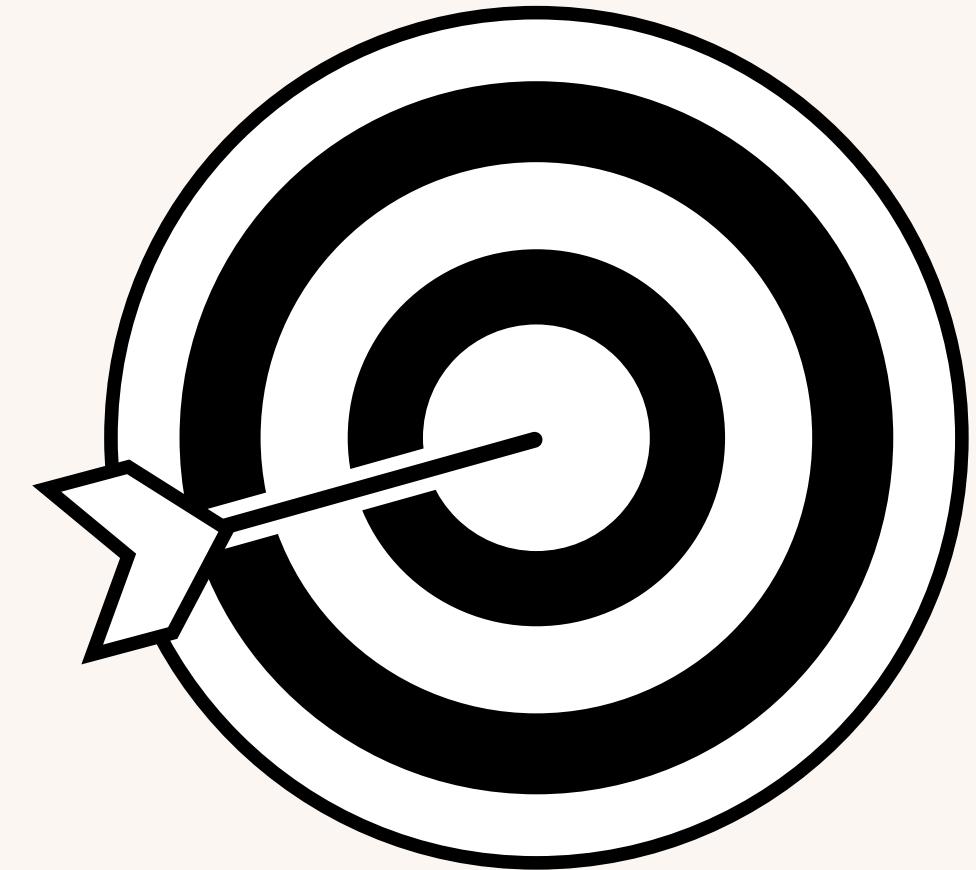
# PROJECT CHALLENGES

- Data Collection Process
- Google ML Kit



# PROJECT MOTIVATION

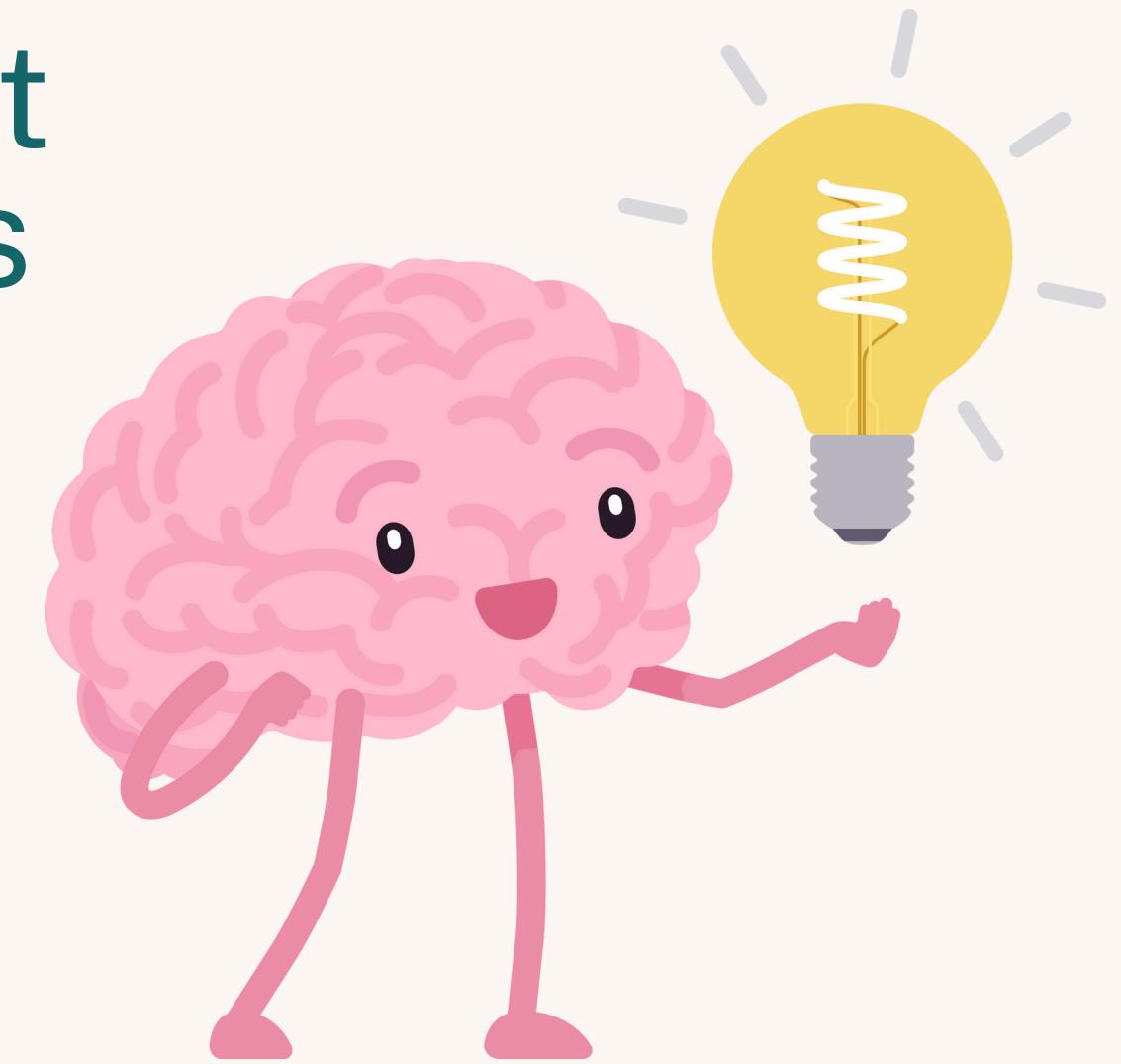
- The idea of Impacting other peoples lives, turn it into a better shape through the simplest means even by changing the lifestyle.
- The increasing numbers of Internet and applications users to the age of 60s.
- The unavailability of a replica to such an idea



# PROJECT IDEA

The idea of Impacting other peoples' lives, turn it into a better shape through the simplest means even by changing the lifestyle.

The increasing numbers of Internet and applications users to the age of 60s. The unavailability of a replica to such an idea Arab 2030 vision.



# SOLUTION

**Developing an instant medical assistant for them. The patient can search for the interaction between the medicine he wishes to take for now and its validation state with the medicine he already has scheduled.**

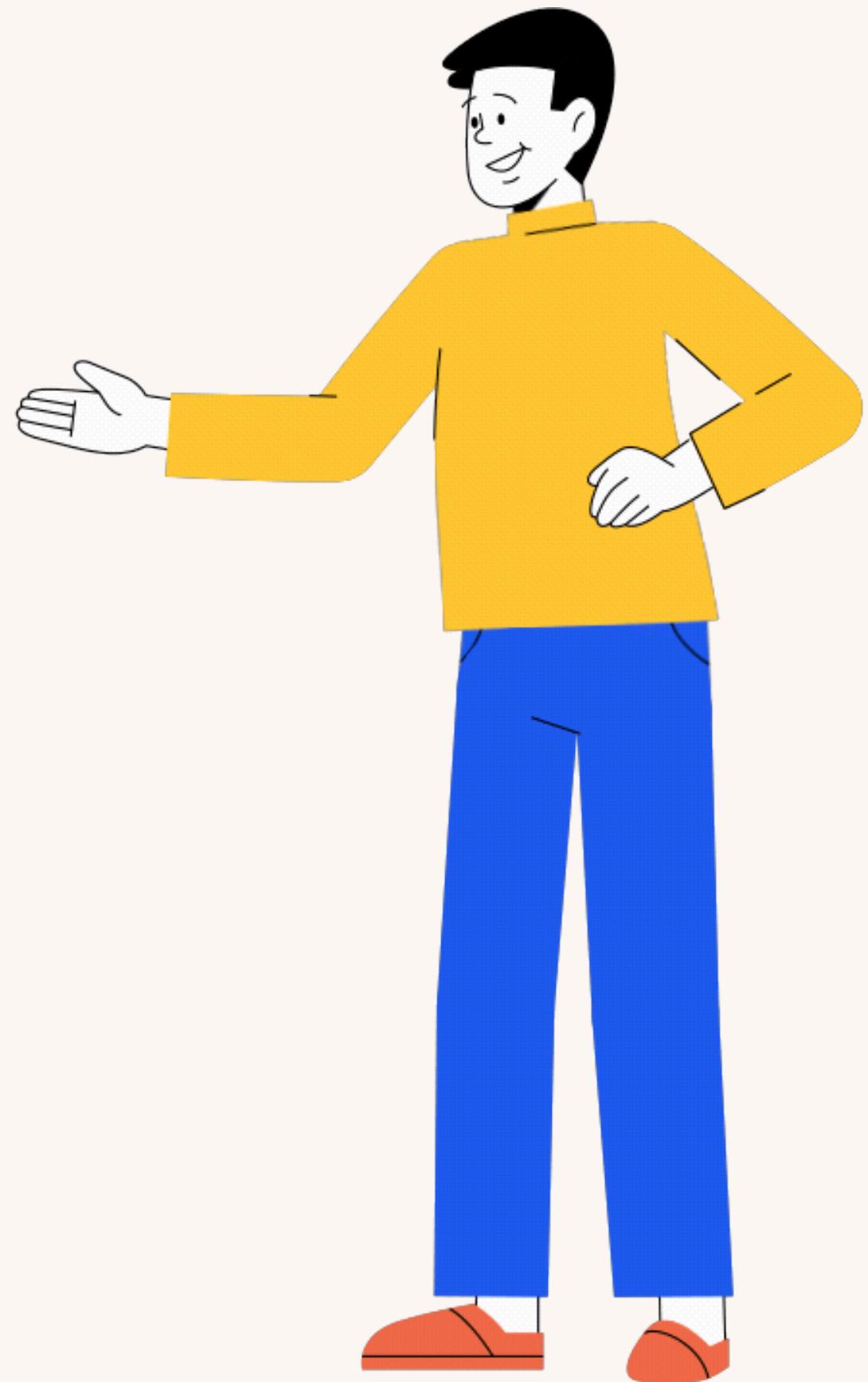


# REQUIREMENTS

## HUMAN REQUIREMENTS

### Team Leader

- UI / UX Developer
- Tester
- Reporter
- Back-end (firebase -dart) Developers
- Database administrator
- Front-end Developers (Dart- Flutter)



# SOFTWARE REQUIREMENTS

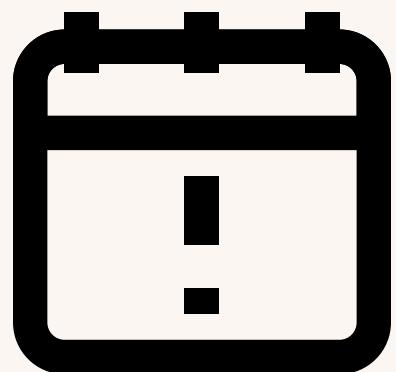
- Allow access to mobile logs for mobile.
- Allow access to camera, Photos, and Locations.

# HARDWARE REQUIREMENTS

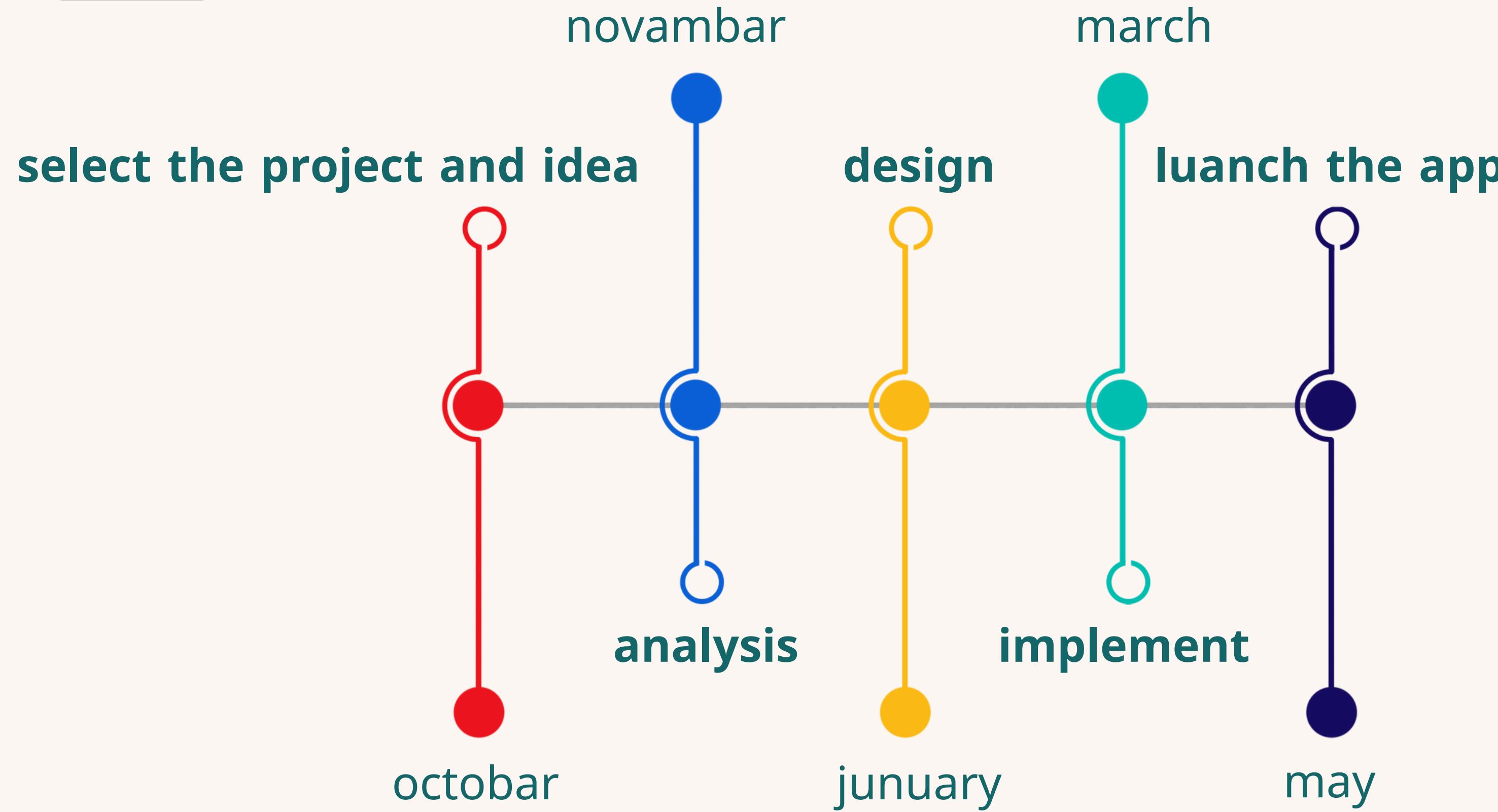
- Pc

# Analysis and Design





# TimeLine



# Feasibility Study

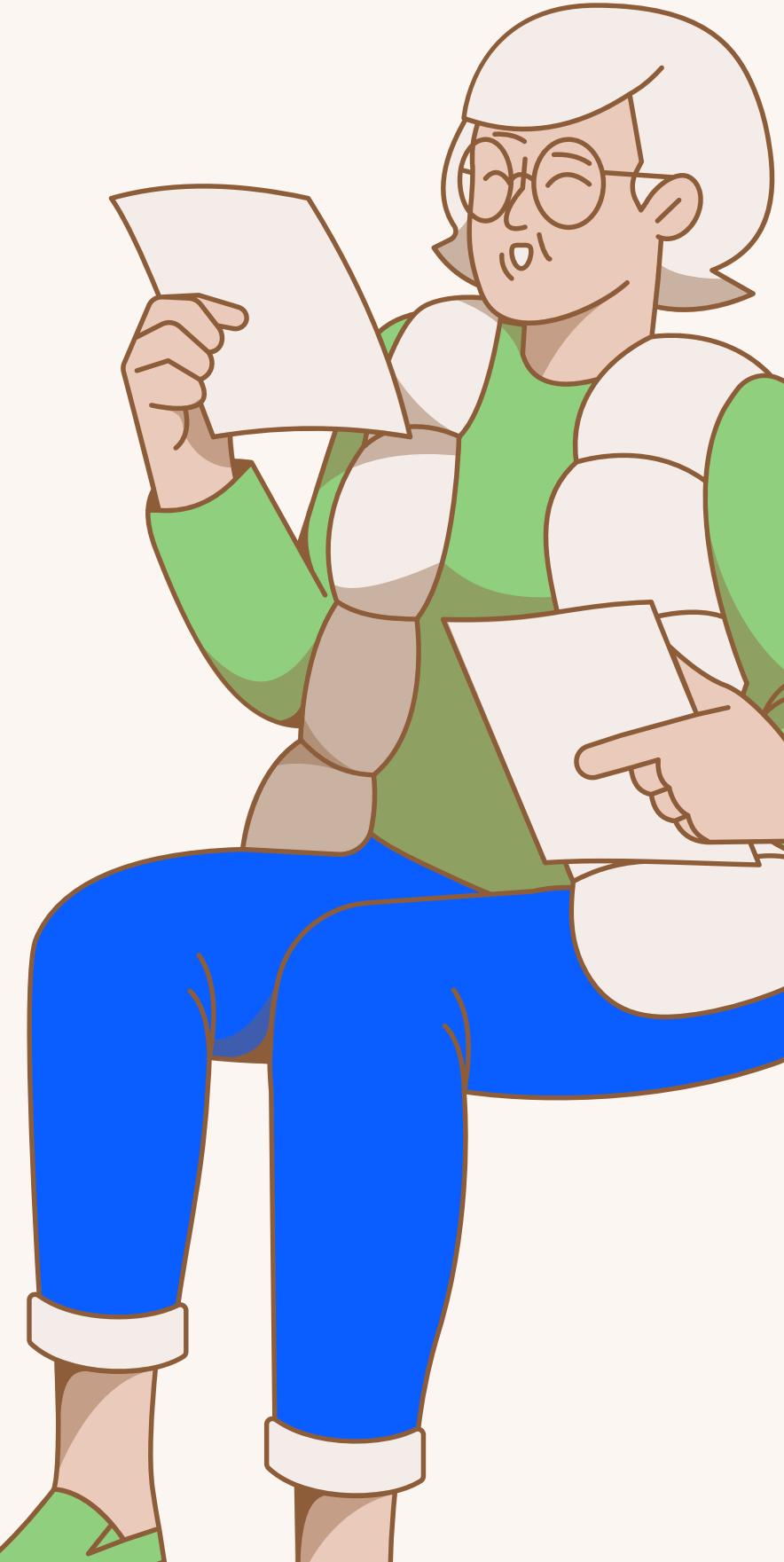
After researching & analyzing the market, there are more than 3.5 million users of only one medical application In Egypt. The society is starving for such a services



# Existing System

**existing systems for drug-drug interaction (DDI) apps provide valuable information but face certain challenges. These include incomplete and outdated information, variability in data sources and quality, lack of contextual information, limited assessment of severity and clinical significance, lack of integration with electronic health records, user interface challenges, and limited patient engagement**

**In the following table, we compare our application called " Shifawuk " with the related applications. In this comparison, we excel in the factors on which the application is measured in terms of free service, ease of use, support for Arabic language and the possibility to use the camera to examine the name of the drug and alert the medicine to take it on time and the partnerships in the applications to ensure quality in the information**



APP	Shifawuk	Medisafe	MyTherapy	CareZone	Sehtay	Dawaee
Free to use						
User-Friendly Interfaces						
Support arabic						
Ability to scan the drug						
Alarm for the drug						
Collaborative Efforts and Partnerships						

# DESIGN OF THE APP

Logo Perspective



Color Perspective



UI / UX Perspective



# Logo Perspective

The image contains a slogan called "Your Healing". It consists of a bowl containing the symbol of a snake and a staff, which indicates health medicine.

Above the bowl are plant leaves, which may symbolize alternative or herbal medicine.

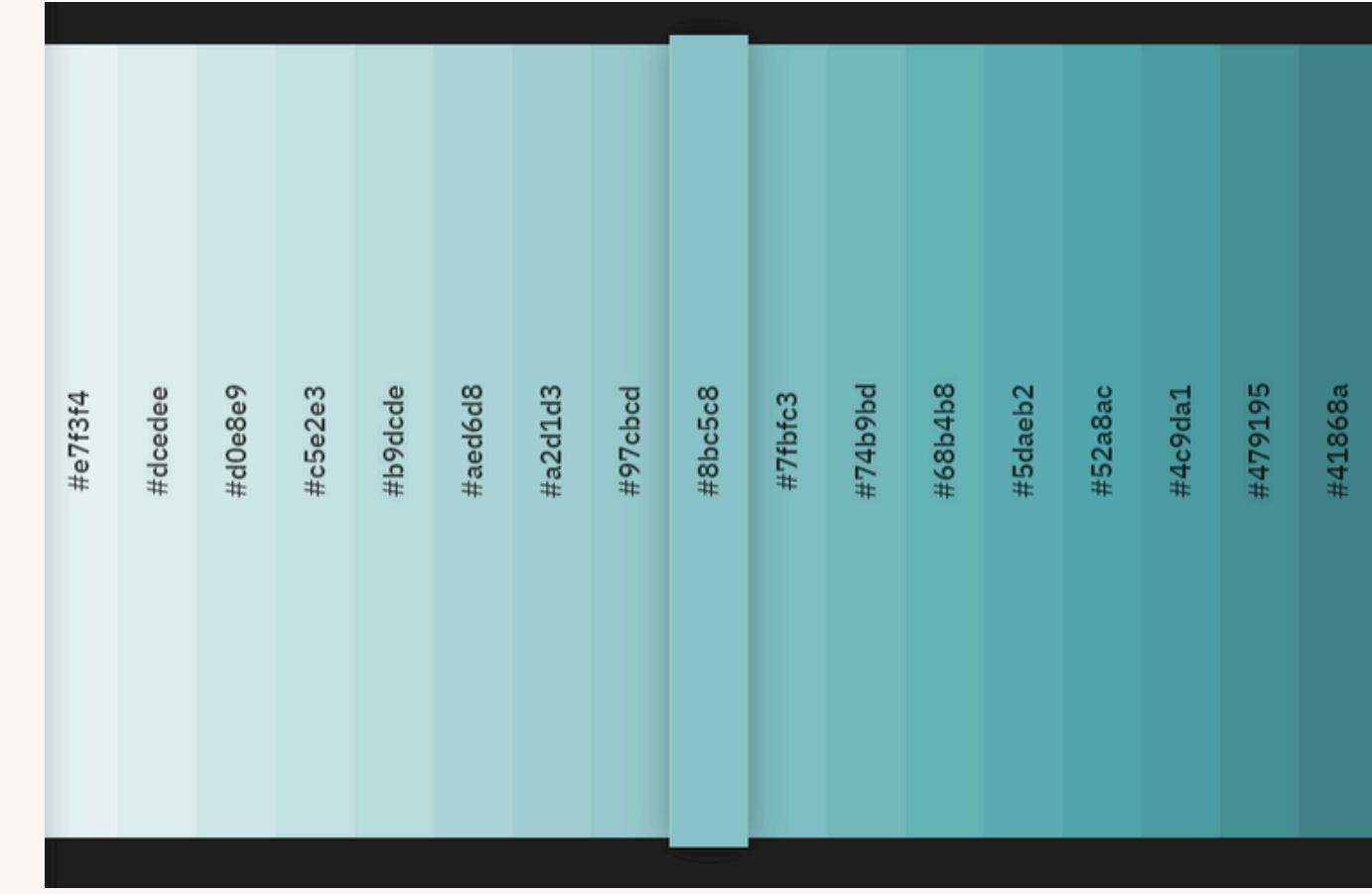
On the side is the year 2024 written, indicating the year the app was created



# COLORS PERSPECTIVE

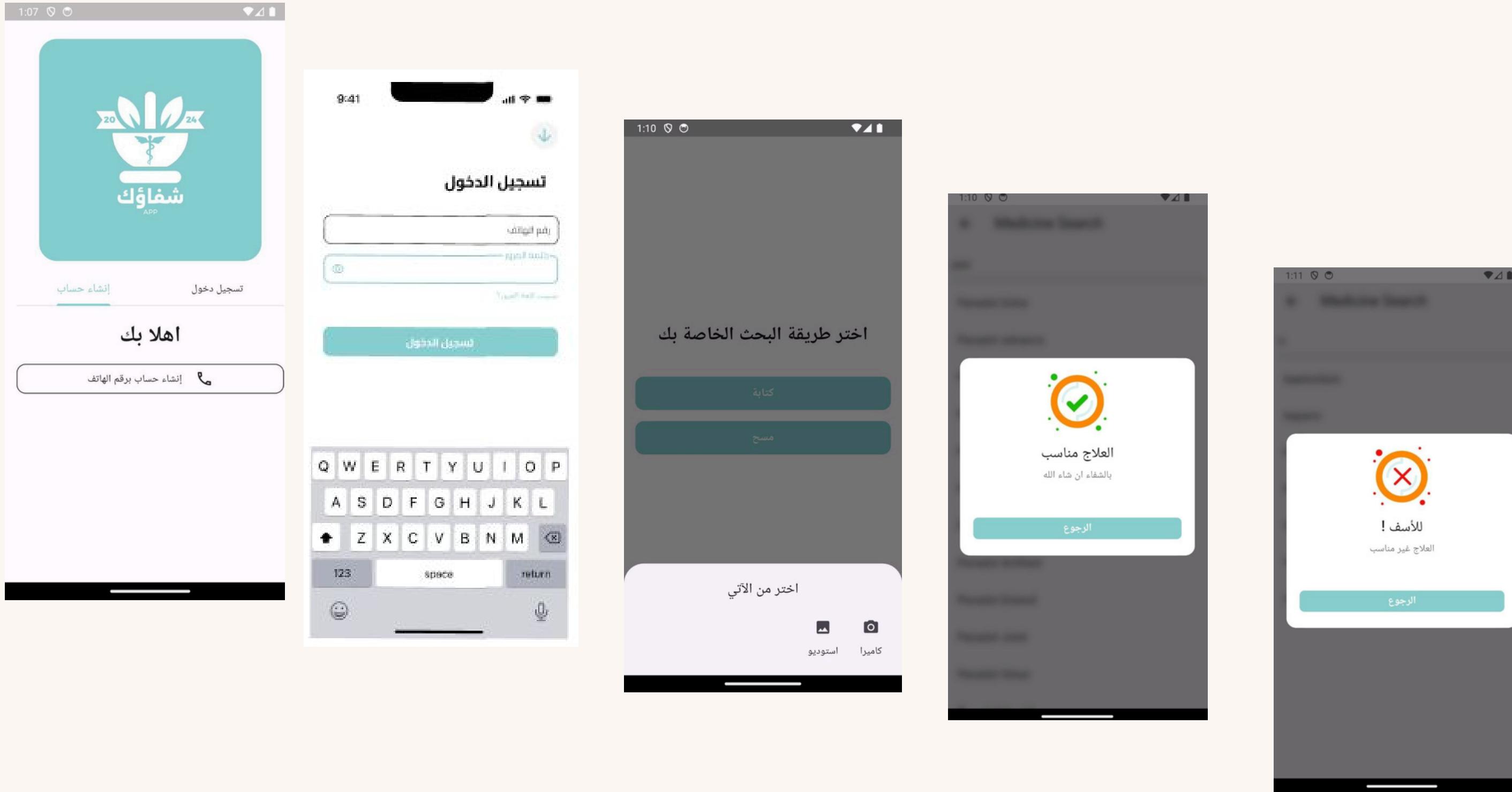


#8bc5c8



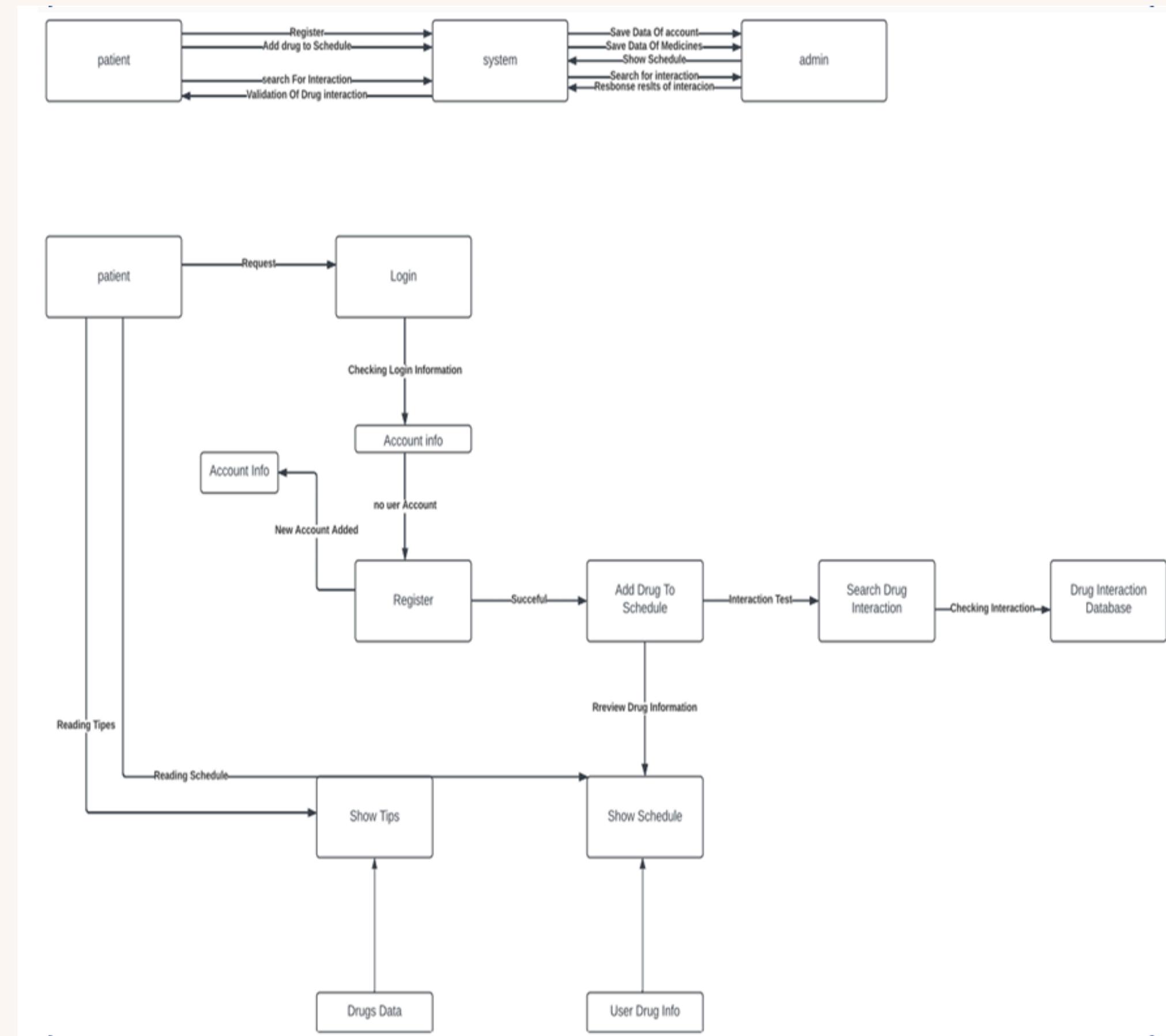
We used poloBlue color palette as our main design palette of the application, as the SkyBlue reflects all meanings of trust and deep profession in investments industry. We seek to implant our routed ideas within our customers minds by the simplest ways as well as the most effective ones as well.

# UI / UX PERSPECTIVE

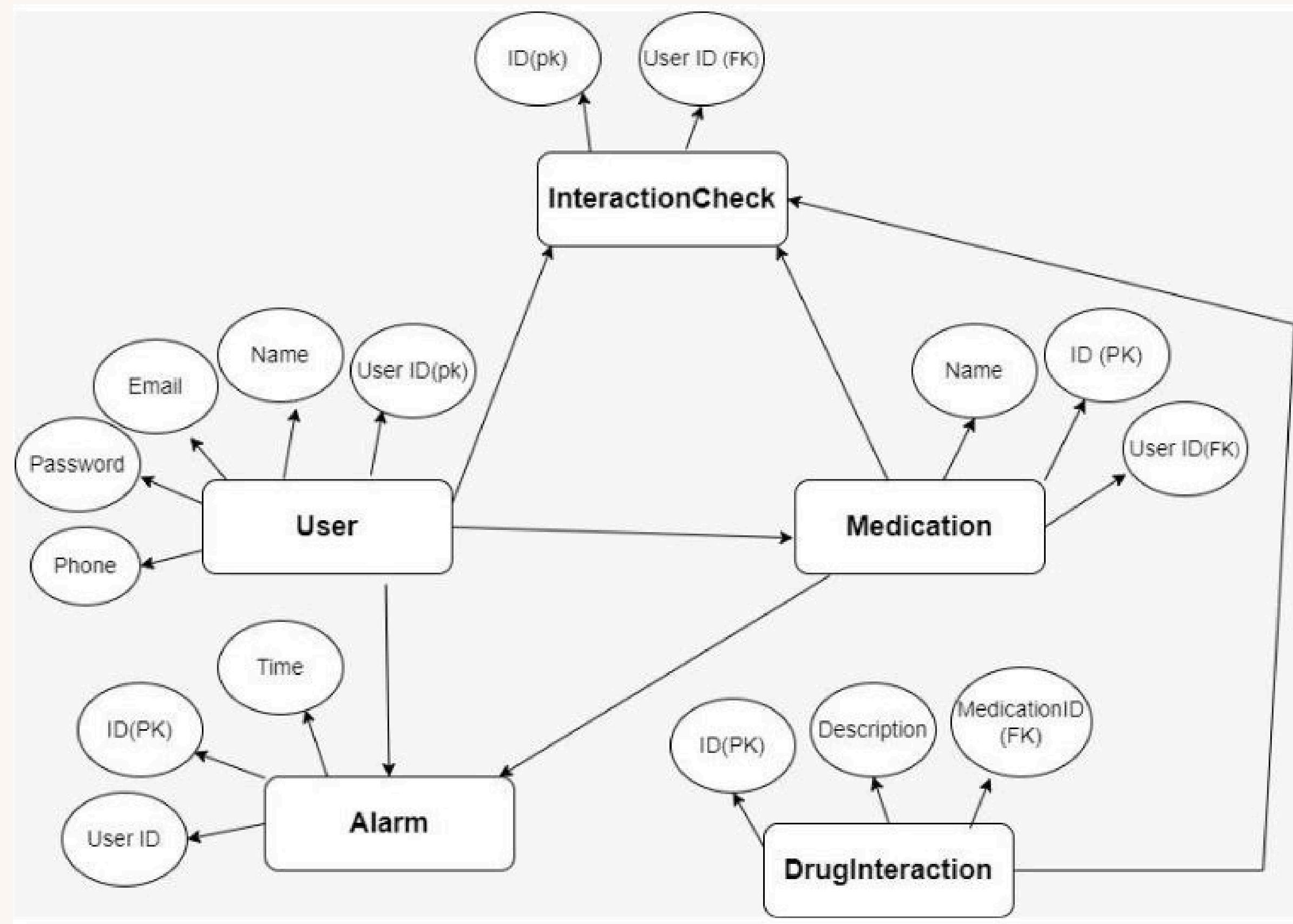


The UI was design and Illustrated from scratch following the ambition of dalvering an easy to use application

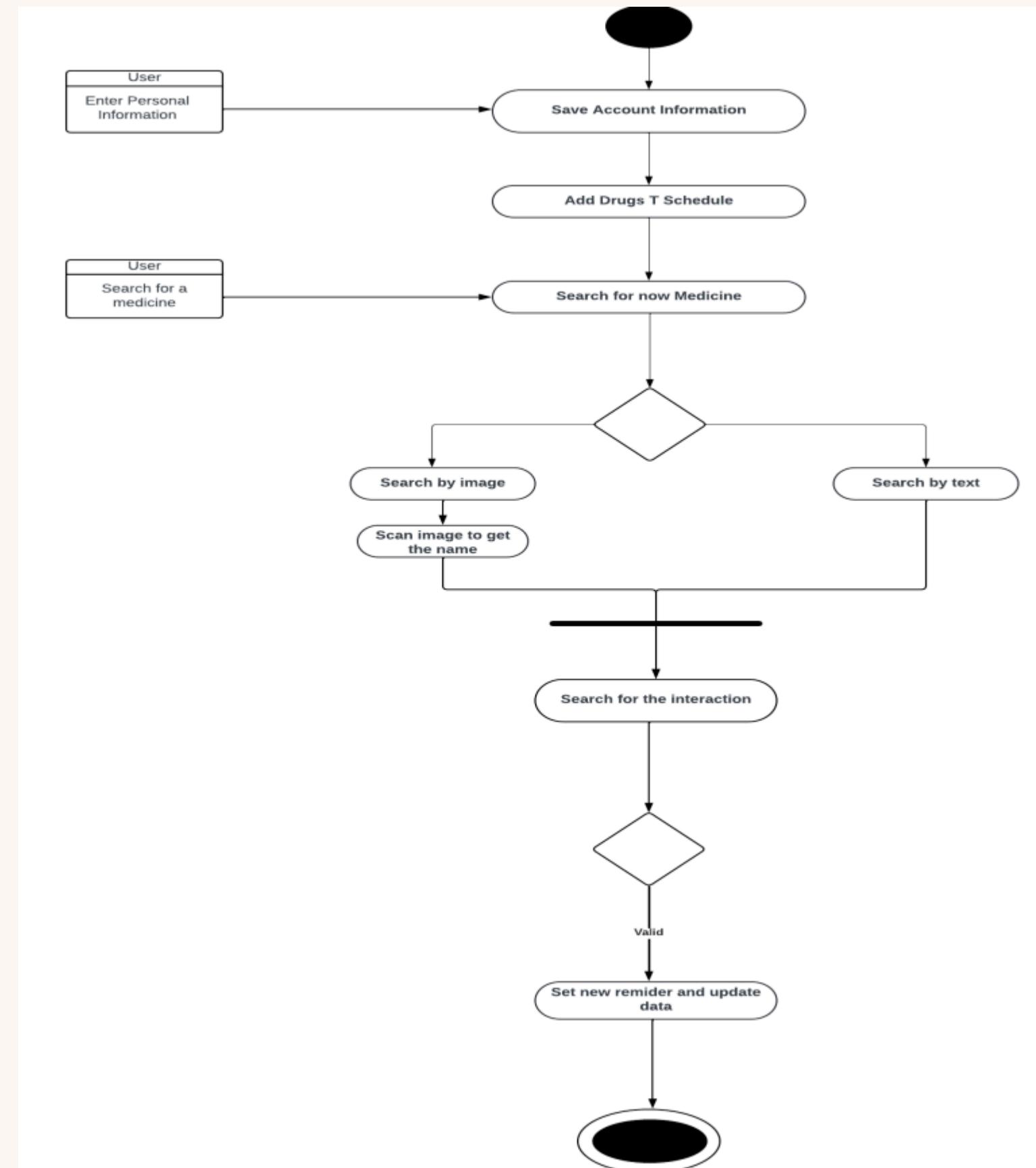
# DFD CONTEXT DIAGRAM



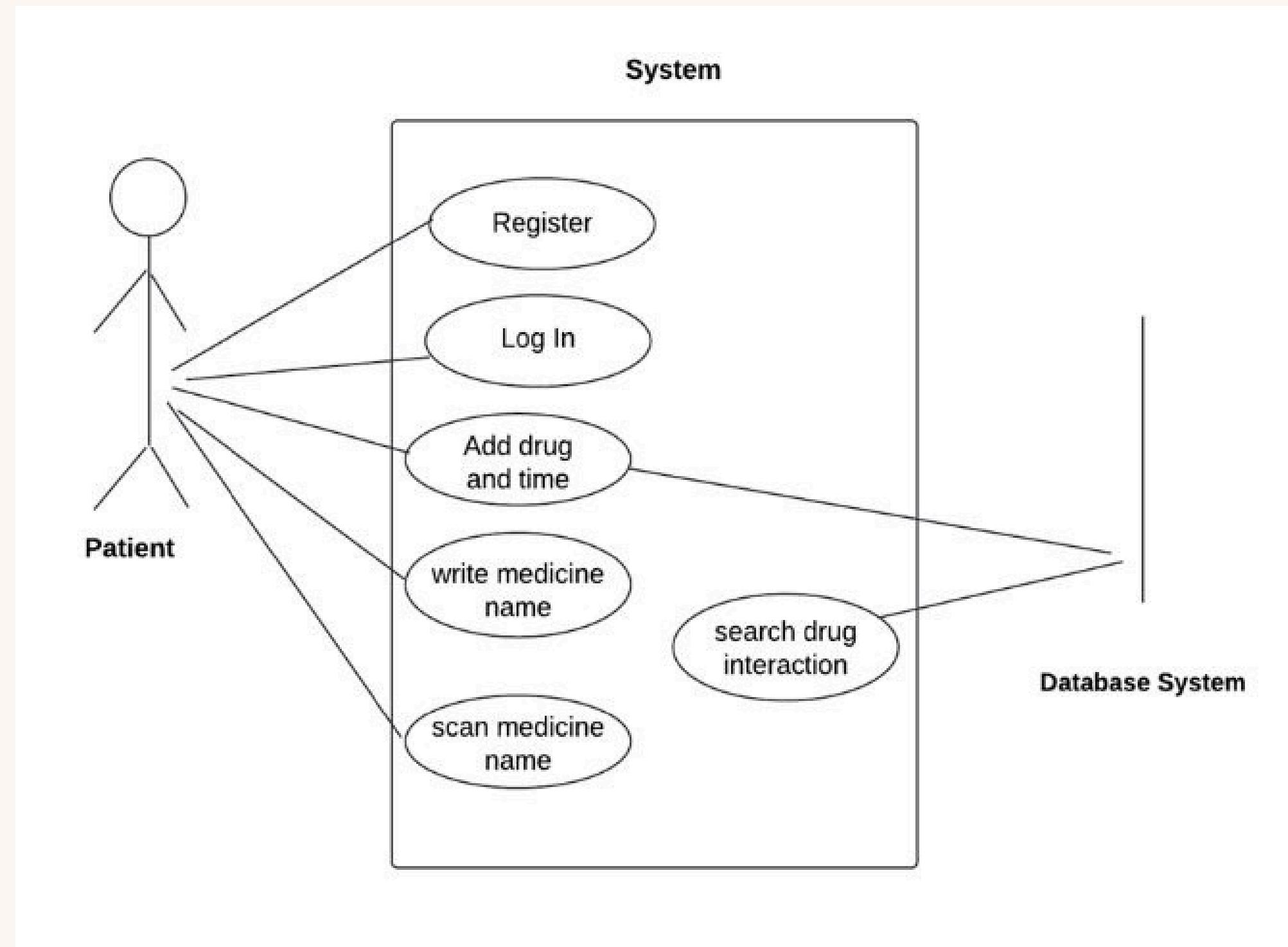
# ERD DIAGRAM



# ACTIVITY DIAGRAM



# USE CASE DIAGRAM



# UML SEQUENCE DIAGRAM

Account Creation and Login:

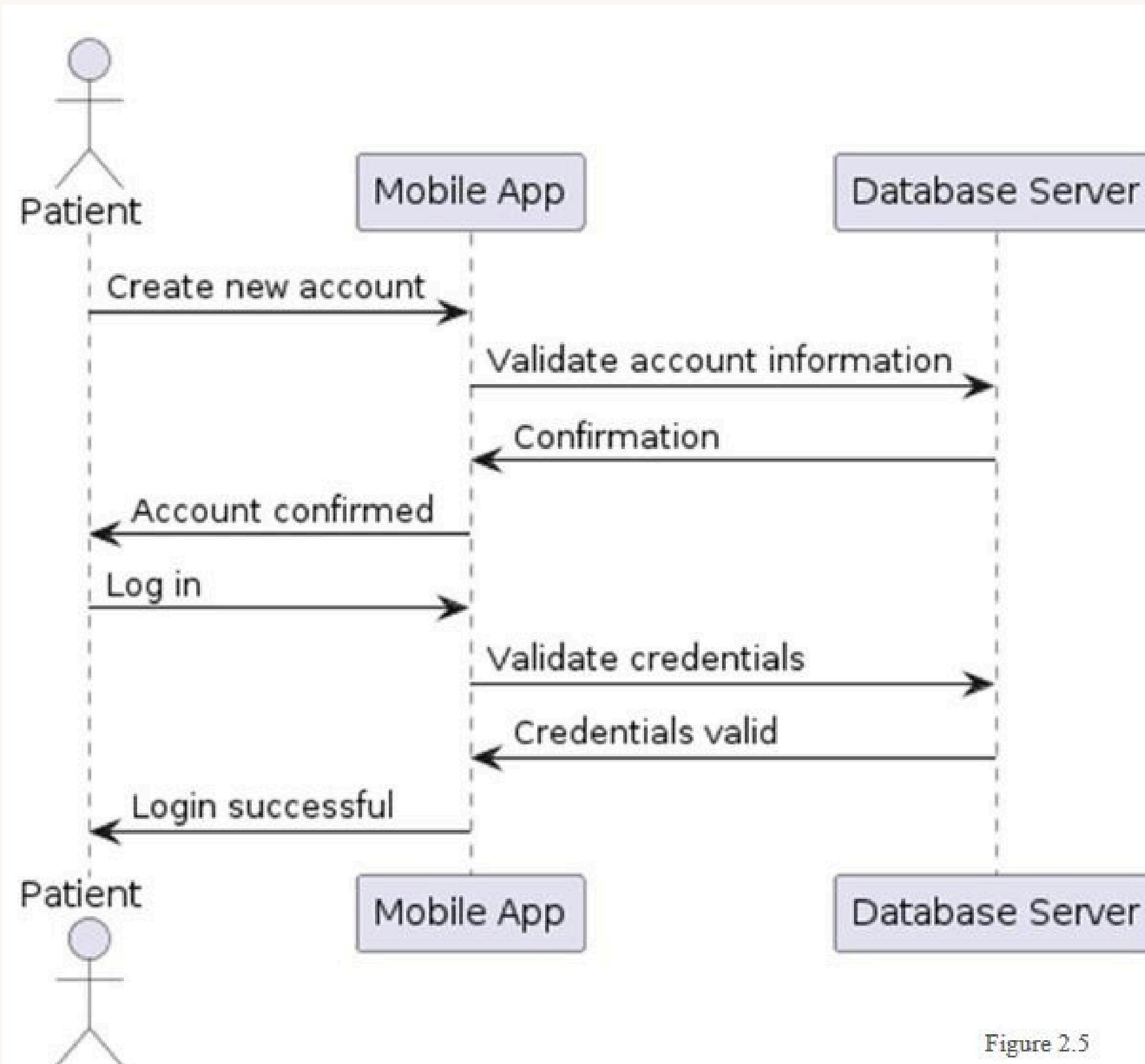
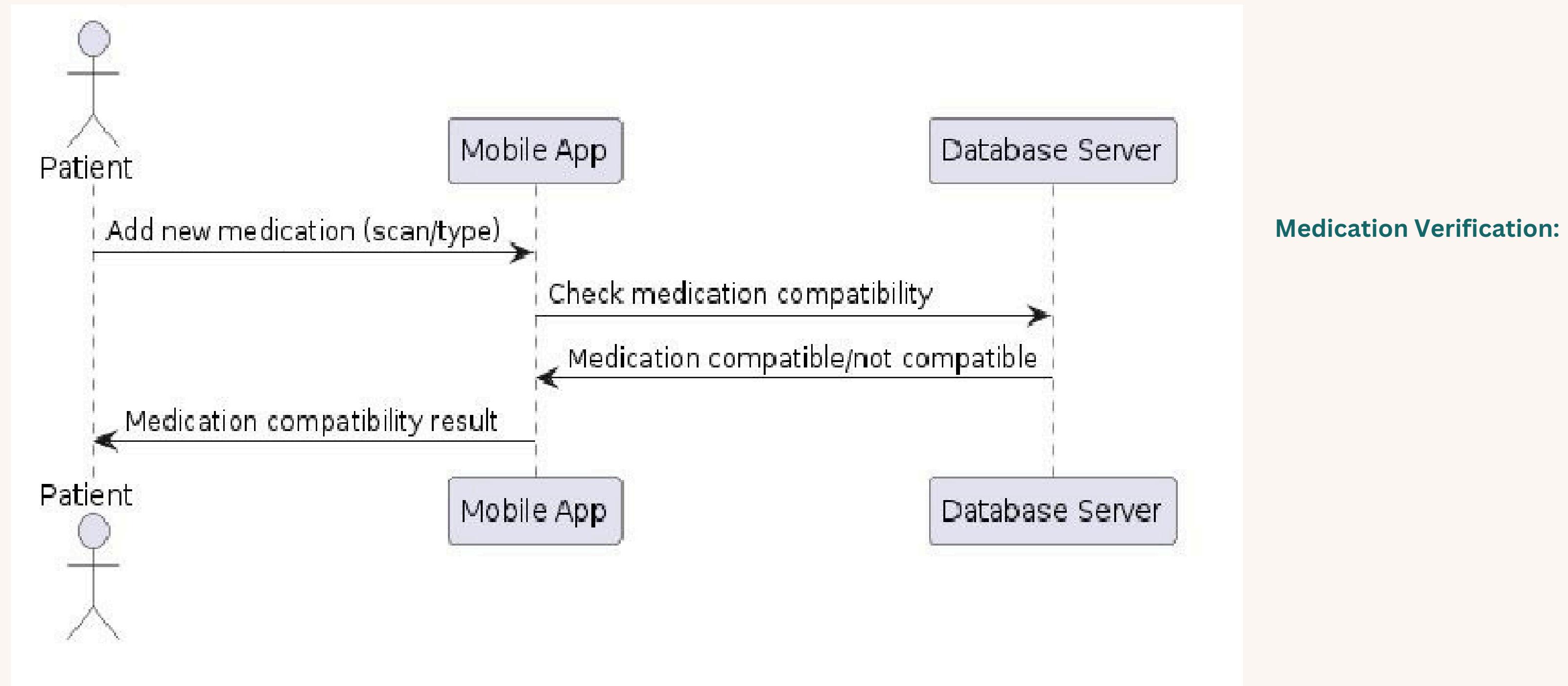
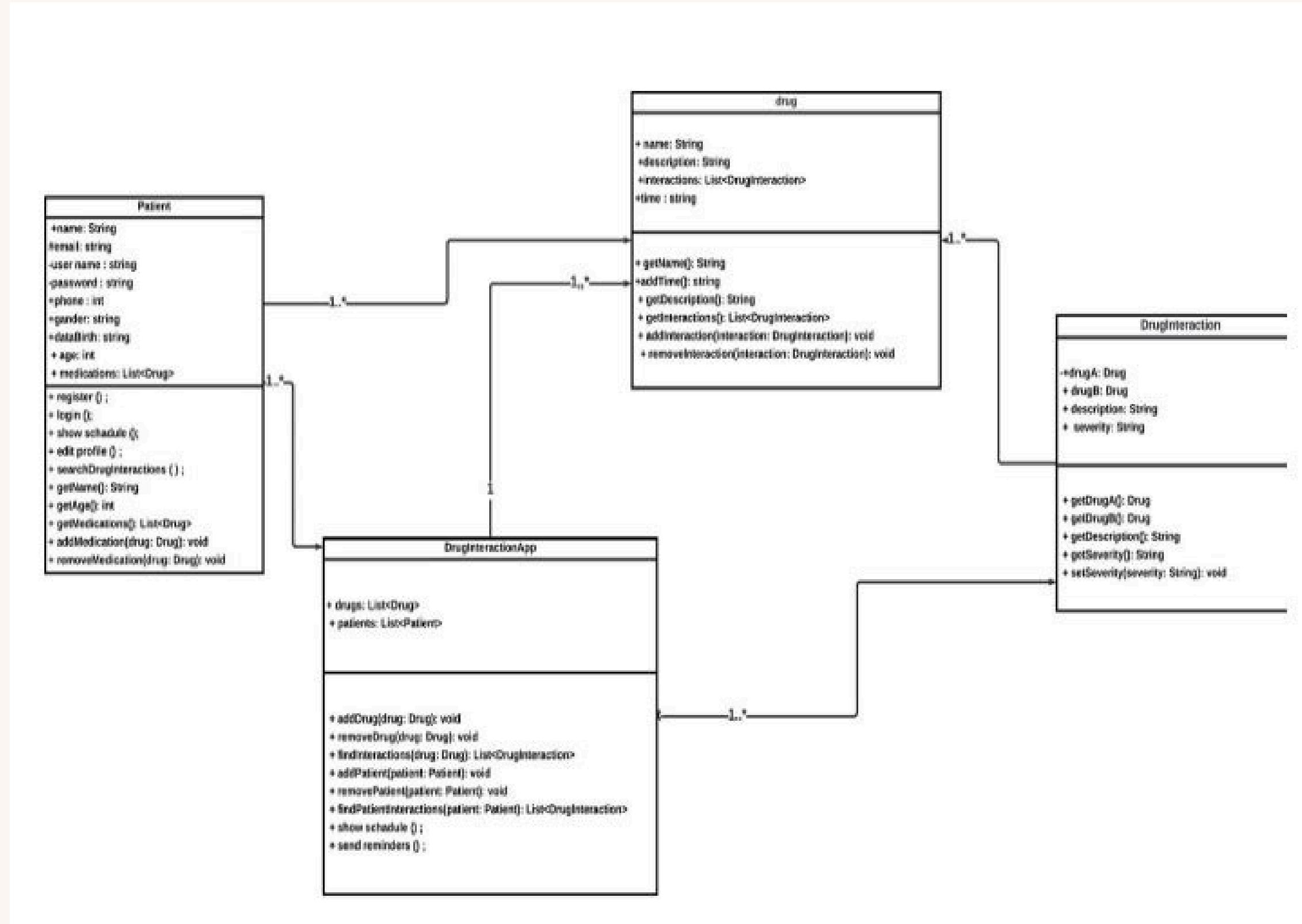


Figure 2.5

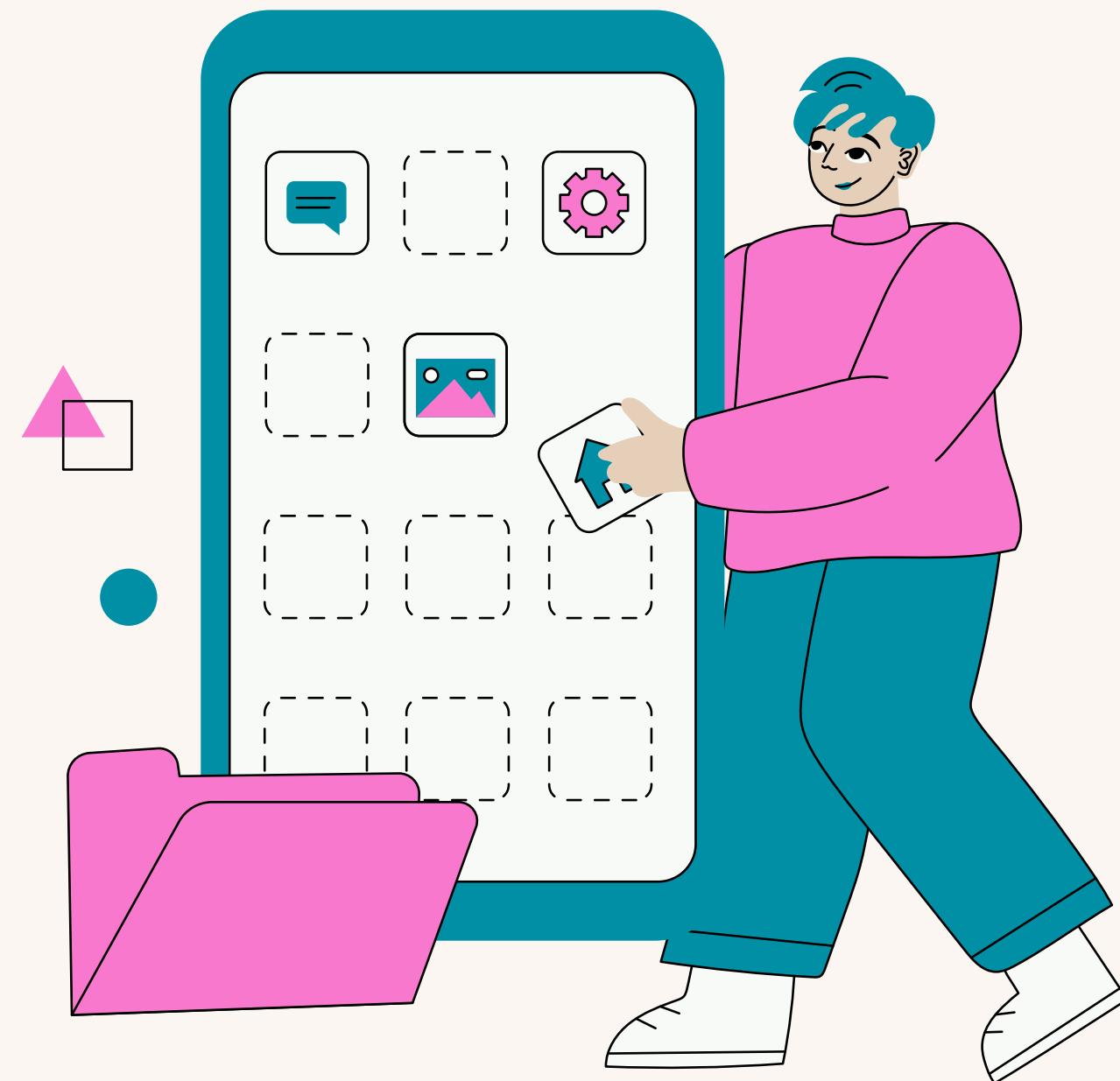
# UML SEQUENCE DIAGRAM



# UML CASE DIAGRAM



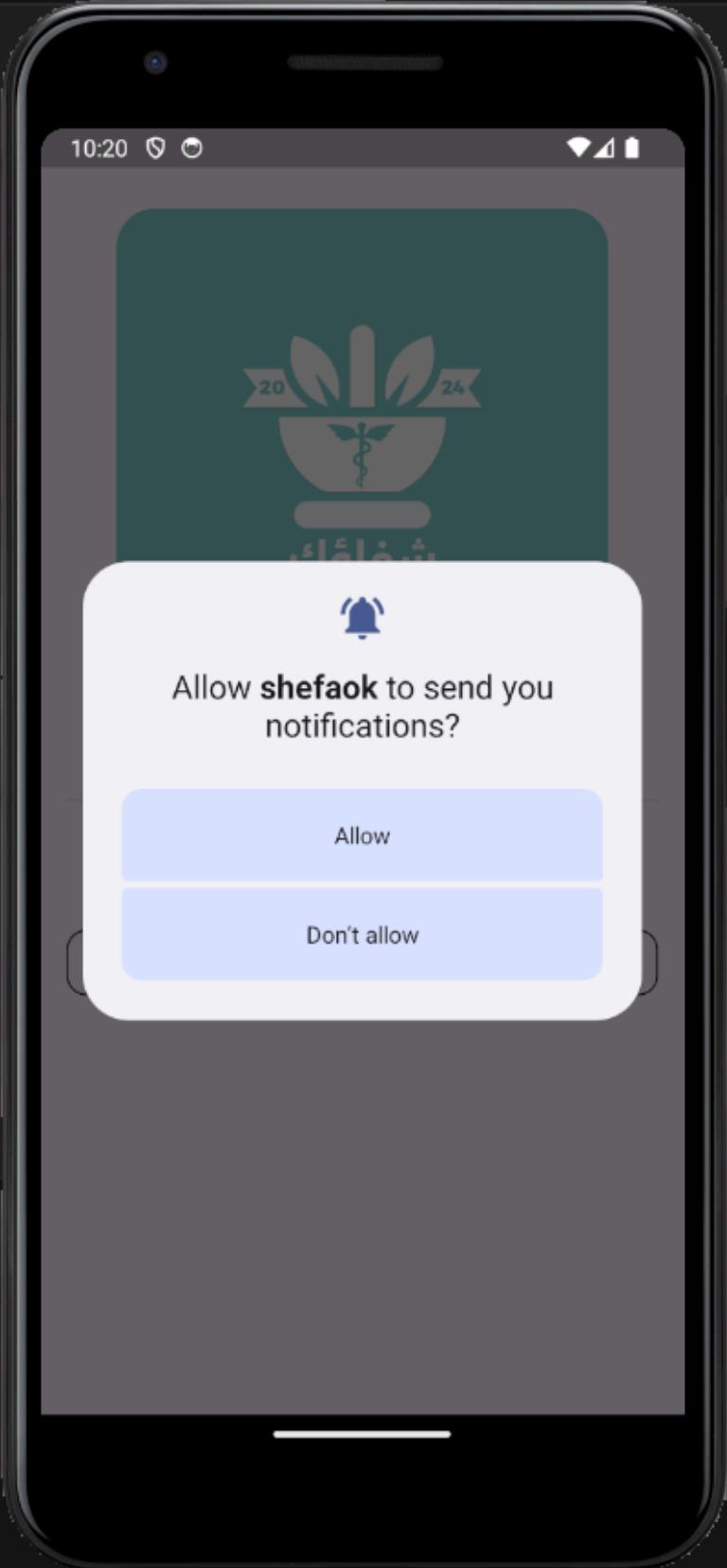
# App Devolpmnt



# Flutter implemantion



# Flutter implemantion



The screenshot shows a Flutter application running on an Android emulator. The app's splash screen features a green background with a white caduceus logo in the center. A notification permission dialog is overlaid on the screen, asking "Allow shefaok to send you notifications?" with "Allow" and "Don't allow" buttons. The top of the screen displays the title bar "shefaok". The left side of the interface shows the VS Code Explorer sidebar with project files like ".dart\_tool", ".vscode", "android", "assets\images", "ios", "lib", and "main.dart". The main code editor window shows the "main.dart" file with Dart code for initializing Firebase and starting the app. The bottom of the screen shows the VS Code status bar with tabs for PROBLEMS, OUTPUT, TERMINAL, PORTS, and DEBUG CONSOLE, along with dependency information.

```
lib > main.dart
1 import 'package:firebase_core/firebase_core.dart';
2 import 'package:firebase_messaging/firebase_messaging.dart';
3 import 'package:flutter/material.dart';
4 import 'package:flutter_bloc/flutter_bloc.dart';
5 import 'package:flutter_screenutil/flutter_screenutil.dart';
6 import 'package:shefa2ok/My_App/my_app.dart';
7 import 'package:shefa2ok/Screens/auth/bloc/auth_bloc.dart';
8 import 'package:shefa2ok/core/services/cache_service.dart';
9 import 'package:shefa2ok/core/services/firebase_api.dart';
10 import 'package:shefa2ok/core/services/simple_bloc_observer.dart';
11 import 'package:shefa2ok/firebase_options.dart';
12 FirebaseMessaging _firebaseMessaging = FirebaseMessaging.instance;
13
14 void main() async {
15   WidgetsFlutterBinding.ensureInitialized();
16   await Firebase.initializeApp(
17     options: DefaultFirebaseOptions.currentPlatform,
18   );
19   NotificationHelper().init();
20   configureFirebaseMessaging();
21   CacheService.init();
22   await ScreenUtil.ensureScreenSize();
23   Bloc.observer = SimpleBlocObserver();
24   runApp(BlocProvider<AuthBloc>(
25     create: (context) => AuthBloc(),

```

PROBLEMS 2    OUTPUT    TERMINAL    PORTS    DEBUG CONSOLE

```
> path 1.9.0 (was 1.8.3)
+ vm_service 13.0.0 (14.2.3 available)
  web 0.3.0 (0.5.1 available)
  win32 5.2.0 (5.5.1 available)
Changed 8 dependencies!
51 packages have newer versions incompatible with dependency constraints.
Try `flutter pub outdated` for more information.
exit code 0
```

# Flutter implemantion

The screenshot shows a Flutter project structure in the Explorer sidebar. The current file is `add_time.dart` under the `Screens/add_time` directory. The code defines a `AddTimeTap` widget and its state, which includes a time picker implementation.

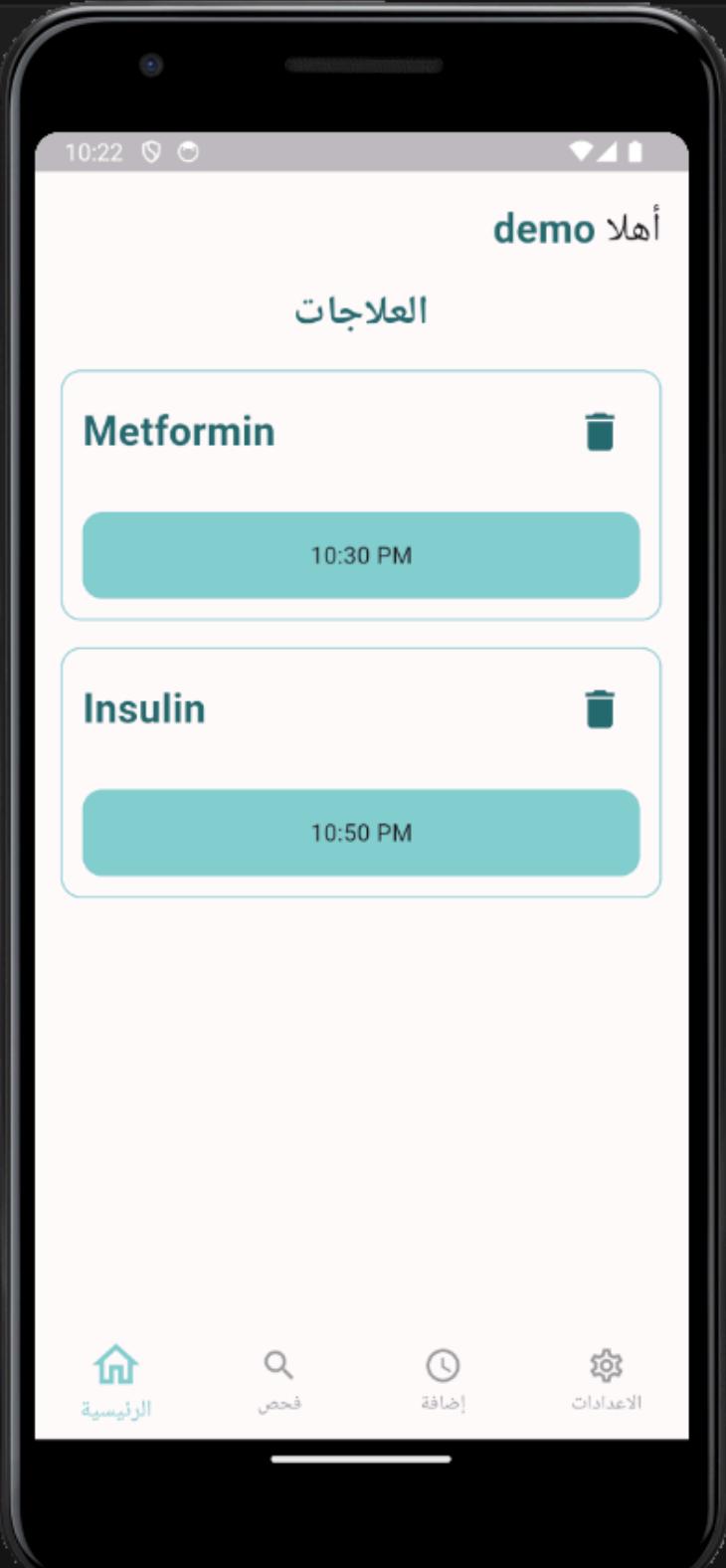
```
lib > Screens > add_time > add_time.dart > ...
10   class AddTimeTap extends StatefulWidget {
11     ...
12   }
13
14   class AddTimeTapState extends State<AddTimeTap> {
15     ...
16     TextEditingController dateController = TextEditingController();
17     String? _timeOfDay;
18     String? selectedMedicine;
19     bool isLoading = false;
20     List medicines = [];
21
22     void _showTimePicker() {
23       showTimePicker(
24         context: context,
25         initialTime: TimeOfDay.now(),
26         builder: (BuildContext context, Widget? child) {
27           return Theme(
28             data: ThemeData.light().copyWith(
29               colorScheme: const ColorScheme.light(
30                 primary: Color.fromARGB(255, 135, 205, 206),
31               ), // ColorScheme.light
32             ),
33             child: child!,
34           ); // Theme
35         },
36       );
37     },
38     ).then((value) {
39       setState(() {
```

The terminal output shows dependency updates:

```
PROBLEMS 5 OUTPUT TERMINAL PORTS DEBUG CONSOLE
> path 1.9.0 (was 1.8.3)
+ vm_service 13.0.0 (14.2.3 available)
| web 0.3.0 (0.5.1 available)
| win32 5.2.0 (5.5.1 available)
Changed 8 dependencies!
51 packages have newer versions incompatible with dependency constraints.
Try `flutter pub outdated` for more information.
exit code 0
```

The right side of the interface displays a mobile application running on an Android emulator. The screen shows a modal dialog titled "اختر الوقت" (Select Time) with a time picker and a "اضافه" (Add) button. The bottom navigation bar includes icons for الرئيسية (Home), فحص (Check), and إضافة (Add). The status bar shows the time as 10:23 and battery level.

# Flutter implemantion



The screenshot shows a Flutter application running on an iPhone X simulator. The app displays a home screen with a title 'أهلا' (Hello) and 'العلاجات' (Medicines). Below this, there is a list of two items: 'Metformin' and 'Insulin'. Each item has a delete icon and a timestamp: '10:30 PM' for Metformin and '10:50 PM' for Insulin. At the bottom of the screen, there are navigation icons for 'الرئيسية' (Home), 'فحص' (Check), 'إضافة' (Add), and 'الإعدادات' (Settings).

The code editor on the left shows the Dart file `home_view.dart` with the following content:

```
lib > Screens > home > view > home_view.dart > ...
19   class _HomeViewState extends State<HomeView> {
20       Widget build(BuildContext context) {
21           ...
22           ],
23           ),
24           const SizedBox(
25               height: 16,
26           ),
27           const Text(
28               'العلاجات',
29               style: TextStyle(
30                   fontSize: 24,
31                   fontWeight: FontWeight.bold,
32                   color: Color.fromARGB(255, 37, 108, 111), // TextStyle
33               ),
34           ),
35           const SizedBox(
36               height: 16,
37           ),
38           // ElevatedButton(onPressed: () {NotificationHelper.schedueledNotificati
39           //     'title', 'body');}, child: Text('schedule')),
40           medicineList()
41       ],
42       ),
43       ],
44       ),
45       ],
46       ),
47       ],
48       ),
49       ],
50       ),
51       ],
52       ],
53       ],
54       ],
55       ],
56       ],
57       ],
58       ],
59       ],
60       ],
61       ],
62       ],
63       ],
64       ],
65       ],
66       ],
67       ],
68       ],
69       ],
70       ],
71       ],
72       );
73   }
```

The output terminal at the bottom shows the following log:

```
PROBLEMS 5 OUTPUT TERMINAL PORTS DEBUG CONSOLE
> path 1.9.0 (was 1.8.3)
+ vm_service 13.0.0 (14.2.3 available)
web 0.3.0 (0.5.1 available)
win32 5.2.0 (5.5.1 available)
Changed 8 dependencies!
51 packages have newer versions incompatible with dependency constraints.
Try `flutter pub outdated` for more information.
exit code 0
```

# Flutter implemantion

The screenshot displays a Flutter application running on an Android emulator. The app's interface includes a title bar with Arabic text 'اختر طريقة البحث الخاصة بك' and two teal-colored buttons labeled 'كتاب' and 'مسح'. Below the interface, the code editor shows the Dart file 'check\_tab.dart' containing the following code:

```
import 'package:shefa2ok/Screens/search_view/search_screen.dart';
import 'package:shefa2ok/core/shared_widgets/button_builder.dart';

class CheckTab extends StatefulWidget {
    const CheckTab({super.key});

    @override
    State<CheckTab> createState() => _CheckTabState();
}

class _CheckTabState extends State<CheckTab> {
    File? image;
    bool isDone = false;
    // List<String> suggestions = [];
    // List<String> suggestionsLoc = [];
    List<dynamic> interactions = [];
    List<dynamic> nonInteractions = [];
    Future<void> pickImage(ImageSource source) async {
        var pickedimage = await ImagePicker().pickImage(source: source);
        if (pickedimage != null) {
            image = File(pickedimage.path);
            await _processImage();
        }
    }
}
```

The code editor also shows the output of the build process, indicating dependency changes and exits with code 0.

# Database Implementation



Firebase



# Database Implementation



# Firebase

Using Firebase in a mobile app focused on drug-drug interaction (DDI) checks and drug reminders is important for several reasons. It enhances the app's functionality, reliability, and user experience, while also simplifying the development process. Here are some key points outlining the importance of integrating Firebase into such an app:

- Real-Time Database and Data Synchronization
- User Authentication and Security
- Push Notifications and Engagement
- Analytics and User Insights
- Scalability and Reliability
- Ease of Integration and Development Speed
- Enhanced User Experience
- Data Analytics and Monitoring

# Database Implementation

The image displays two screens illustrating the implementation of a database. On the left, the Firebase Cloud Firestore console shows a user's data structure under the collection 'users'. The document 'cTzeMa9dlhSaE.' contains fields such as 'Password', 'Phone', 'birthday', 'gender', 'image', 'uid', and 'userMedicine'. The 'userMedicine' field is an array containing one item with 'medicine: "Metformin"' and 'time: "10:30 PM"'. On the right, a mobile application interface for a medication reminder app is shown. The screen is titled 'أهلاً' (Hello) and 'العلاجات' (Medications). It lists two medications: 'Metformin' taken at '10:30 PM' and 'Insulin' taken at '10:50 PM'. The bottom navigation bar includes icons for 'الرئيسية' (Home), 'فحص' (Check), 'إضافة' (Add), and 'الإعدادات' (Settings).

Cloud Firestore

shefaok

users > cTzeMa9dlhSaE.

(default)

+ Start collection

medicine

users >

cTzeMa9dlhSaENJTIb2R

+ Add document

Rankvh9aKJC...

S5jW0hxT24D...

UeuxY1M7vPf...

WWFT8xk40bh...

YK86hGtARf7...

Z2zED9ZoIhm...

aqZxxlac31m...

cTzeMa9dlhS... >

jBScYMm4ST0...

mywicH0ELE1...

nwjgVMvgZAZ...

r4h9v0LjFbS...

tF01wHojkM6...

y9yymh0W6Ir...

More in Google Cloud

Panel view

Query builder

10:26

demo

أهلاً

العلاجات

Metformin

10:30 PM

Insulin

10:50 PM

الرئيسية

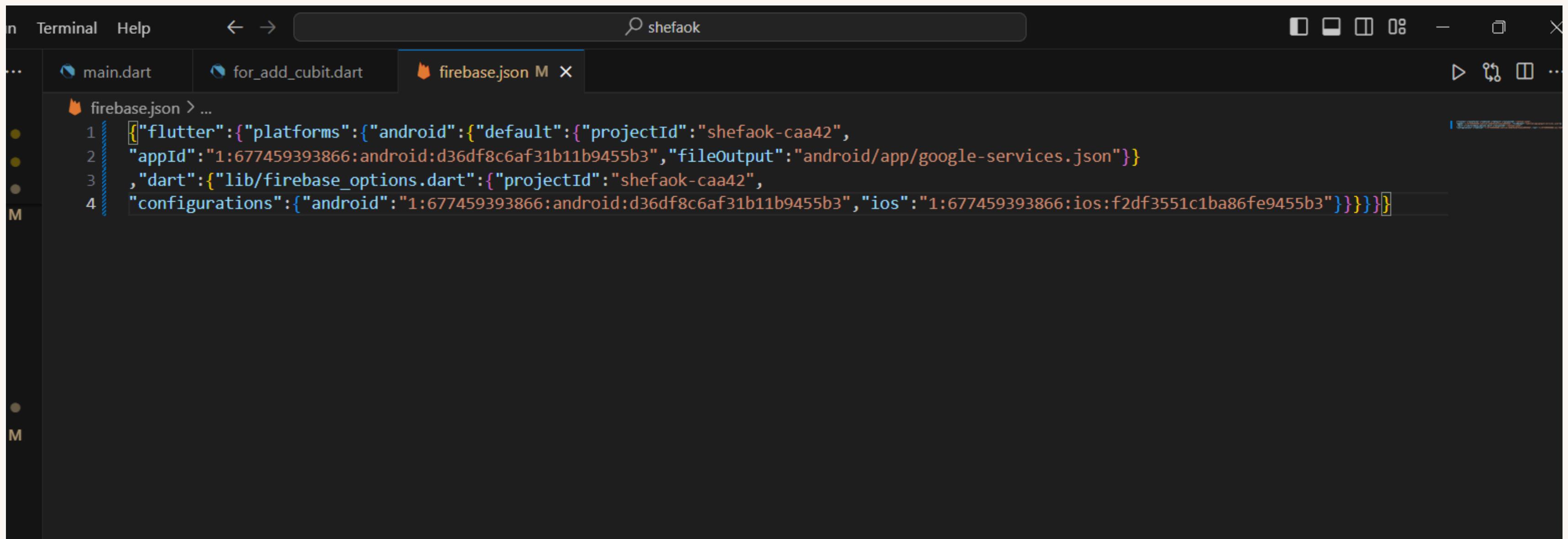
فحص

إضافة

الإعدادات

Database location: eur3

# Database Implementation



A screenshot of a dark-themed code editor, likely VS Code, showing the contents of the `firebase.json` file. The editor has a top bar with tabs for `Terminal`, `Help`, and file navigation. A search bar contains the text `shefaok`. The main area displays the JSON configuration for a Flutter project.

```
firebase.json > ...
1 {"flutter": {"platforms": {"android": {"default": {"projectId": "shefaok-caa42",
2 "appId": "1:677459393866:android:d36df8c6af31b11b9455b3", "fileOutput": "android/app/google-services.json"}},
3 "dart": {"lib/firebase_options.dart": {"projectId": "shefaok-caa42",
4 "configurations": {"android": {"1:677459393866:android:d36df8c6af31b11b9455b3", "ios": "1:677459393866:ios:f2df3551c1ba86fe9455b3"}}}}}}
```

# Database Implementation

Panel view    Query builder    :

More in Google Cloud ▾

(default)    medicine    addMedicine

+ Start collection    + Add document    + Start collection

medicine    nonInteractionSearch    + Add field

users    addMedicine    forAddMedicine

interactionSearch

0 "Metformin"  
1 "Insulin"  
2 "Glipizide"  
3 "Glimepiride"  
4 "Glyburide"  
5 "Pioglitazone"  
6 "Rosiglitazone"  
7 "Sitagliptin"  
8 "Saxagliptin"  
9 "Linagliptin"  
10 "Alogliptin"  
11 "Exenatide"

(string)   

This screenshot shows the Google Firestore database interface. The left sidebar lists collections: '(default)', 'medicine', and 'users'. The 'medicine' collection is selected, showing sub-collections 'nonInteractionSearch' and 'addMedicine'. The 'addMedicine' collection contains a single document 'forAddMedicine' which holds an array of 12 strings representing medicine names. The interface includes navigation icons, a search bar, and a bottom toolbar with 'Panel view', 'Query builder', and other options.

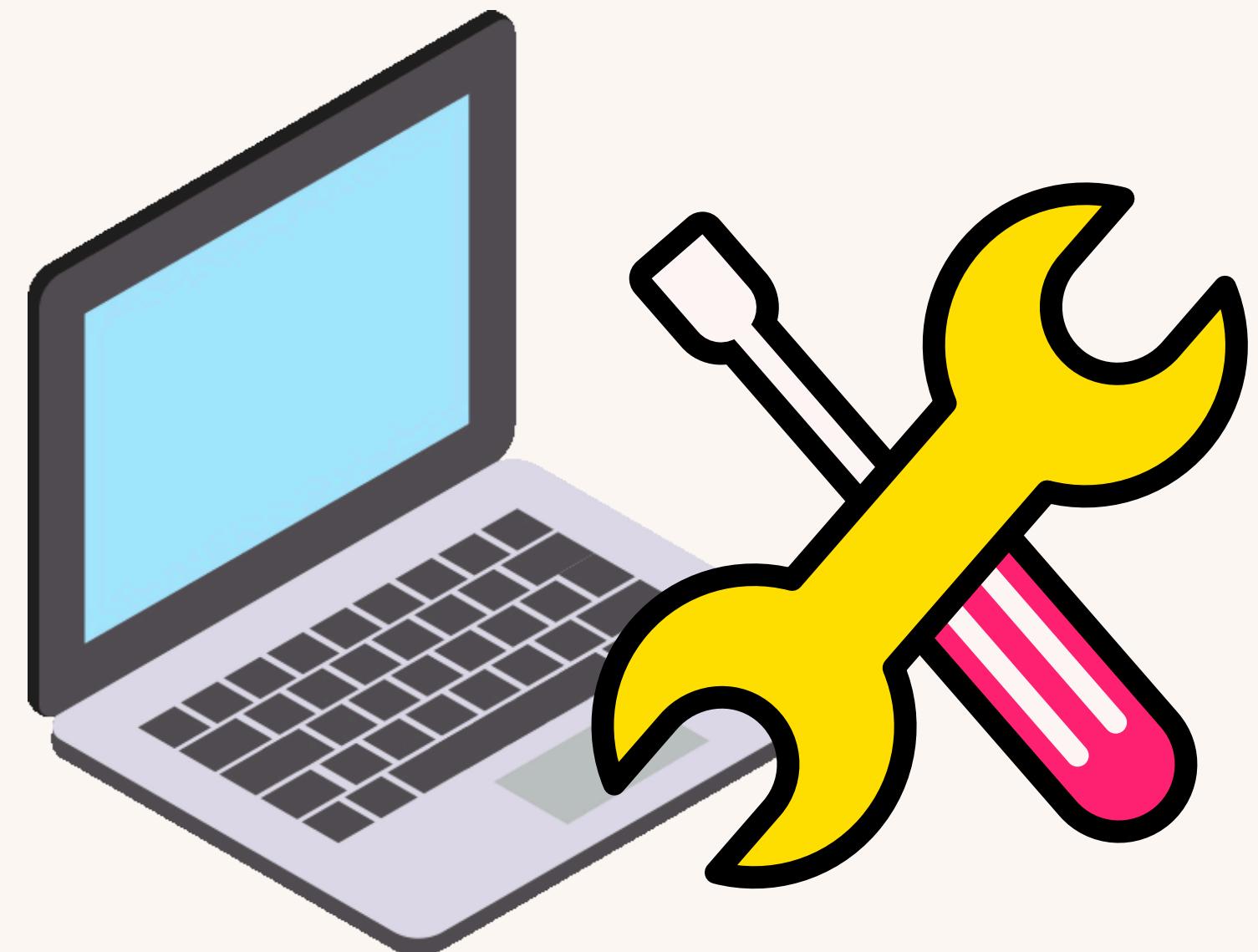
# Database Implementation

The screenshot shows a MongoDB database interface with the following structure:

- Collection:** users
- Document ID:** 90ZQd2zFNY7D.
- Fields:**
  - Email: "abacaxi@outlook.com"
  - Name: "nuser"
  - Password: "123456"
  - Phone: "01023456789"
  - birthday: "01-01-1990"
  - gender: "ذكر"
  - image: ""
  - uid: "vIDQyIS"
  - sub-collections:**
    - userHistory (empty)
    - userMedicine (empty)
    - medicine (empty)

# PROJECT TOOLS

- Visual Studio Code
- Android Studio
- Click-Up
- Figma
- Flutter
- Lucid Chart
- Firebase



# REFERENCES

1-Drug-Drug Interactions: Mechanisms and Implications for Drug Therapy" by T. E. Sheiner, B. Rosenberg

2-Drug Interactions: Analysis and Management" by Philip D. Hansten and John R. Horn

3-DrugBank

- Website: [DrugBank](#)
- A comprehensive resource that combines detailed drug data with drug-target and drug interaction information

4-FDA Guidance on Drug Interaction Studies

- Website: [FDA Guidance](#)
- Provides guidance on the study design, data analysis, and clinical implications of drug interaction studies.

5-Clinical Pharmacology & Therapeutics

- Publishes articles on the latest research in drug interactions, pharmacokinetics, and pharmacodynamics.

# CONCLUSION

**In the end, this application seeks to help patients with pressure and diabetes to improve their treatment period and lifestyle.**

**Enables simplifying medical information to them with ease to ease by touching everything related to their treatment system and what affects it from the rest of the diseases, medicines, food and even daily work.**

**The application is rich with a lot of information. which leads to Increasing medical and health awareness in the community.**

**And as said "Prevention is always better than a cure".**

THANK  
YOU