Mobile Application Final Report

Group 15 - ICT

University of Science and Technology of Hanoi

November, 2021

Contents

1 Introduction

1.1 Overview

Coronavirus first outbroke in Wuhan in November 2019, and it has spread to almost all countries in the world. The virus is deadly; as a result, people want to keep track of the virus information daily. However, it is not easy since there are lots of false information on social media. As a result, we develop a Corona tracker application that aims to solve all of these problems. The system we introduce will give the user information about Covid-19 cases, news, and related details.

1.2 Tools and techniques

In this project, the tools that we used to finish the application are:

- **Java**: our main programming language.
- **Firebase**: the tool for the database (backend) because of its simplicity, portability.
- **Third-party libraries**: multiple libraries for charts display, login and register verification, maps.

1.3 Authors

Nguyễn Xuân Tùng	BI10 - 188
Nguyễn Quang Anh	BI10 - 012
Trần Hồng Quân	BI10 - 149
Lữ Khánh Huyền	BI10 - 083
Vũ Đức Chính	BI10 - 024

2 Architecture and methods

2.1 Architecture

In this project, we choose MVC since the project is not that big. Here is the detailed information of this architecture.
Model: This component stores the application data. The model handles the domain logic(real-world business rules) and communication with the database and network layers. Here in our project, we have six models for fetching the data from the API and used for Firebase Database.
View: The UI(User Interface) layer holds components that are visible on the screen. Moreover, it provides the visualization of the data stored in the model and offers interaction to the user. In our app, we borrow the UI from Instagram. We display multiple pages using bottom navigation. We only have three activities.

Controller: This component establishes the relationship between the View and the model. It contains the core application logic, gets informed of the user's behavior, and updates the model as needed. We use controllers to fetch data from API, display charts, authorization, insert to and query from the database.

2.2 Use Case Diagram

Below is the general use case diagram of our mobile application.

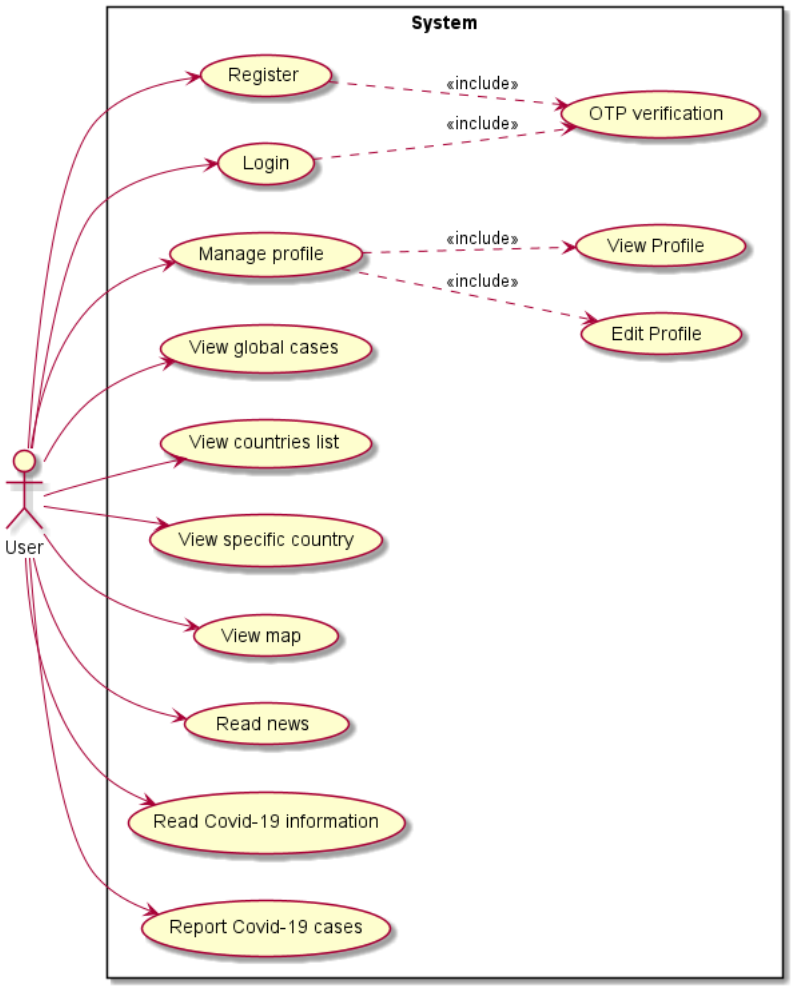


Figure 1: Use case diagram

3 Dataset and library description

3.1 Dataset

We take the dataset from Open Disease Data Api (https://corona.lmao.ninja/). The data is updated daily and this site does not need API key in order to access.

3.2 Library

Here are the list of all third-party libraries that we use in our app

- Retrofit : For fetching Api such as covid cases.
- Volley: Another library for fetching Api. In our project, we use Volley to fetch the vaccination numbers and its corresponding date.
- MP Android Chart : A powerful Android chart view / graph view library.
- Google Maps Clustering for Android : Display clustering on maps.
- Glide : For image uploading.
- Android Navigation Component: Transition between fragment, created by Google.
- Country Code Picker : Pick country code, used in register or login.
- **Firebase and google services for map, OTP, captcha**

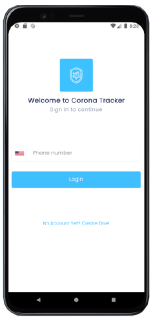
4 Remarkable Features

Here are the list of all the features that our app provides:

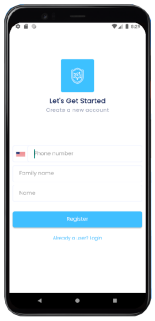
- **Register screen** : The user will fill in his first name, last name, mobile phone .
- **Login screen**: The user will fill in his mobile phone in order to login.
- **Verification screen** : In this section, the user receives the OTP that is provided from Firebase. This 6-number pin code will act as the verification for register or password for login.
- **Dashboard screen** : The user has 2 choices on this screen. He can go to the global cases of Covid-19 screen or go to Covid-19's map screen.
- **Global cases screen** : The user can view the data we fetch from API includes total cases, active, recovered, and deaths. The data is represented in pie chart as well as text.
- **List of countries screen**: List of all countries with their corresponding cases. The user can search for a specific country as well as sort bases on total,recovered, and death cases.

- **Specific country screen** : The detailed information of Covid-19 cases of a specific country will be shown. We also include the line chart that display the total number of vaccination in that country.
- **News screen** : The user can read news related to Covid-19 in the world.
- Corona info screen: Contains detailed information such as how to prevent Covid-19, the symptoms as well as detailed the virus.
- Profile screen: Show the user's information such as full name and mobile number.
- Edit profile screen: The user can edit his profile. He/she can also upload his/her avatar as well as fill in some information like ID number, gender.
- **Report Covid-19 cases screen**: If the user think that there are someone might have Covid-19 symptoms, he will go to this section. It provides real-time map so that the user can mark his position

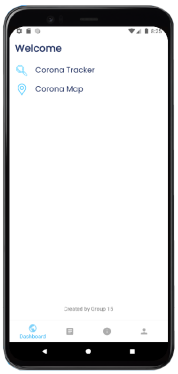
5 Outputs and Results



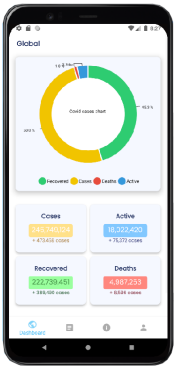
(a) Login screen



(b) Register screen

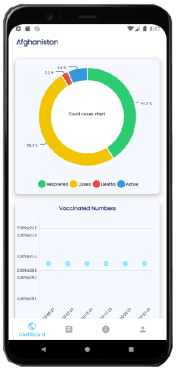


(a) Dashboard screen



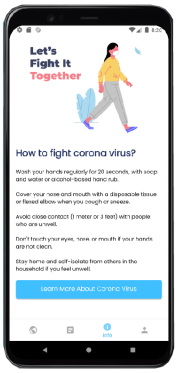
(b) Global screen

(a) List of countries screen

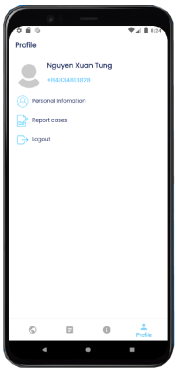


(b) Country's detail screen

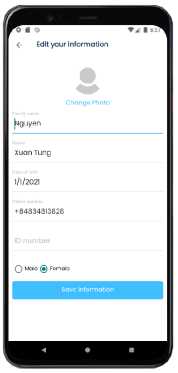
(a) News related to covid screen



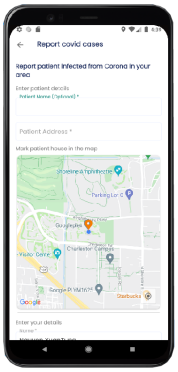
(b) Corona's info screen



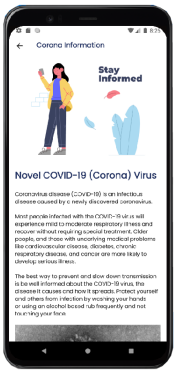
(a) Profile screen



(b) Edit profile screen



(a) List of countries screen



(b) Corona's detail screen

6 Conclusion and Future work

6.1 Conclusion

In conclusion, this report has shown specific details and information regarding the application. The app shows Covid-19 cases of a specific country and globally. Currently, the user can log in, register, log out from the app, read news related to Covid-19, and detailed information on this virus. He can also report covid cases if he doubts that a person might have Covid-19 symptoms. The application runs smoothly with no bugs found yet; however, several things need to be done or improved in the future, which will be noted in the next section.

6.2 Future Work

However, there are several things that can be improved in the future. We will rebuild our code based on Model-view-view model architecture to reduce the amount of code written for future work. We may include QR scan and health care declaration, just like the health care note made by the Vietnamese's government.

References

[1] OTP verification using firebase
https://www.youtube.com/watch?v=ZLOUggIEqzo

[2] Multiple back stack navigation
https://github.com/mabahmani/MyMultipleBackStackNavigation