**Instructions on the execution of Patients Monitoring System (PMS) mobile application in debug mode**

**Prerequisites**

1. A desktop computer

2. A mobile device

3. An USB or any suitable cable to connect the desktop and mobile devices

4. Node.js must be installed

5. Flutter must be installed for developing Android apps with the steps specified in https://docs.flutter.dev/get-started/install

6. A local Mongo Database must be installed, for example, using Homebrew with the following commands executed one after one:

* brew update
* brew tap mongodb/brew
* brew install mongodb-community

**Execution Procedures**

1. Turn on the mobile hotspot on your mobile device and connect your desktop computer to the mobile hotspot with the specific Wi-Fi.

A screenshot of a phone

AI-generated content may be incorrect. A screenshot of a computer

AI-generated content may be incorrect.

2. Then check your desktop computer's private IP address

A screenshot of a computer

AI-generated content may be incorrect.

3. Open the source codes of the app in the directory pms\_app/lib and change every occurrence of <ip\_address> into your computer's private IP address

4. Start the local MongoDB server with the command in Terminal:

brew services start mongodb/brew/mongodb-community

5. In Terminal, go to the directory PMS\_Mobile, start the backend server with the command: node backend3.js

6. Set the mobile device to allow USB debugging, then use an USB cable or other suitable cable to connect the mobile device to the desktop computer

A screenshot of a phone

AI-generated content may be incorrect.

7. Open a new Terminal, go to the directory pms\_app, type the command flutter devices to ensure the mobile device is available for running the mobile app in debug mode

A close-up of a computer screen

AI-generated content may be incorrect.

8. Enter the command flutter run, then it will let you choose the device to run the app

9. After choosing the connected mobile device, if the app starts successfully, the mobile device will automatically launch the app

A screenshot of a computer program

AI-generated content may be incorrect.

10. After the app is launched, you will see a login page.

A screenshot of a login screen

AI-generated content may be incorrect.

You can use one of the following pairs of username and password:

|  |  |
| --- | --- |
| **Username** | **Password** |
| Abel | abel |
| Disable | disable |
| Usable | usable |

Note: each pair of username and password represents a doctor

11. If login is successful, a page will be launched to show a list of patients that the doctor manages. Each card widget shows the information of a patient.

A screenshot of a phone

AI-generated content may be incorrect.

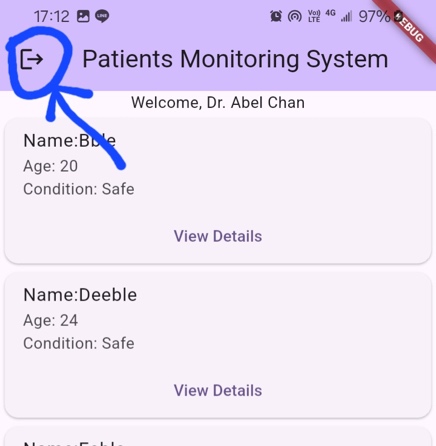
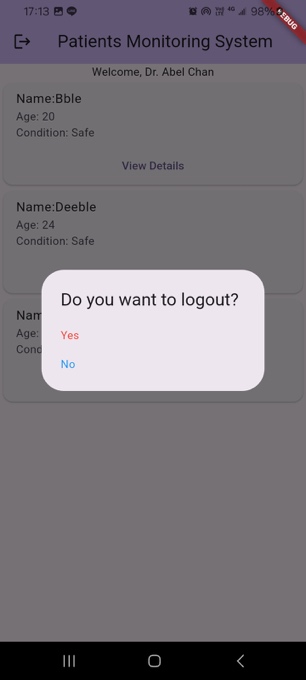
12. There is a button "View Details" on each card. After clicking that button, it will launch a page showing the detailed information of the specific patient

A screenshot of a phone

AI-generated content may be incorrect.

13. You can return to the previous page by clicking the left arrow icon on the app bar

14. Then you can logout by clicking the logout icon on the app bar on the patients page

**Stopping the execution**

To stop the app's execution, you can type q on the Terminal which is running the app. Then you can stop the backend server by simply pressing Control+C.

Then, stop the MongoDB server by the command brew services stop mongodb/brew/mongodb-community.

Then, unplug the cable that connects the mobile and device devices. And switch off the mobile hotspot on the mobile device.