Instructions for Deploying

Software stack required

- PHP 7
- MySQL 8.0.20
- PHPMyAdmin 4.8.5
- Python 3 (and required libraries)
- Android Studio version 3.4.1
- Java JDK 8
- IDE / Code Editor of choice for working on PHP and Python Code

Build Instructions

Before you begin, please ensure that both PHP and MySQL have been installed and are running.

Note:

- The specific commands for starting/stopping the MySQL services depends upon the host OS and method of installation. For further info, please refer to the official documentation.
- Before starting the server, please make sure that you have a mysql user with an username "root" (and no password). Please refer to mysql documentation for information on how to create a new user.

1. Copy the server directory and place it in the PHP site folder in your computer.
   Note: the site folder path depends upon your OS. Here is the list of default paths according to your OS:
   Linux: `/var/www/`
   MacOS: `/Library/WebServer/Documents/`

2. Now, open http://localhost/test.php on any web browser. If it displays “Success!”, then the files have been copied correctly. Else, go back to step 1.

3. Open PHPMyAdmin (http://localhost/phpmyadmin), and import `(Import -> Choose File)` all the files, in the `database` directory while leaving all options set as default.

4. Next, Create a virtual environment for Python3, using the `virtualenv` or `venv` packages, for creating an isolated environment for running the models
   Note: Here are the list of commands for creating the virtual environment:
   - Using virtualenv:
     ```
     Virtualenv -p python3 venv
     source venv/bin/activate
     ```
   - Using venv:
     ```
     python3 -m venv myenv
     source myenv/bin/activate
     ```

5. Install all the python dependencies mentioned in the `requirements.txt` file using the command:
   ```
   pip3 install -r models/requirements.txt
   ```

6. Run the models using the command:
   ```
   cd models
   python3 main.py
   ```

7. Now, open the project in Android Studio, and allow Android Studio to Index all the files (this may take a while).

8. Try to build the APK. This installs all the required dependencies and Gradle Libraries. You will be prompted to install the missing SDKs and Build Tools along the way. Please follow the prompts.

9. Check IP of your server (the computer having the php files, the database as well as the python models)
   Note: you can use the `ifconfig` command or GUI (`Settings -> Networks`) to get the IP.

10. Now replace the IP in the `server` field with this IP, in the `Constants.java` file (java/iitg/vedinkaksa/Constants.java) in the Android Project.

11. Next, build and install (`run -> run 'app'`) the APK on all the test devices.
    Note: Do not disconnect the device from the computer in case you want to check the logs (containing all the sensor data and other events)

12. Now, run the app on all the devices, and log in using the credentials provided in the `Credentials` folder. In case QR Code is required, use the ones provided in the `qr_codes` folder

13. Now, open the `Logcat` pane on Android Studio to view all the logs while the app is running

Iterating upon the Models

The python State detection models are independent of the Android App. Thus, iterating upon the model would not require you to recompile the Android APK (unless the server IP is changed). However, you are required to restart the Python server.

There are different functions corresponding to each model in the Code (in `models/classifier/classifier/main.py`). In order to update any model, please change code in the function corresponding to that model.

Before restarting the server, please verify that the path to the training data for each model has been specified correctly in the corresponding functions.

Finally, interrupt the server (`ctrl + C`), in case it's already running, and then restart the server using the commands given in Step 6.