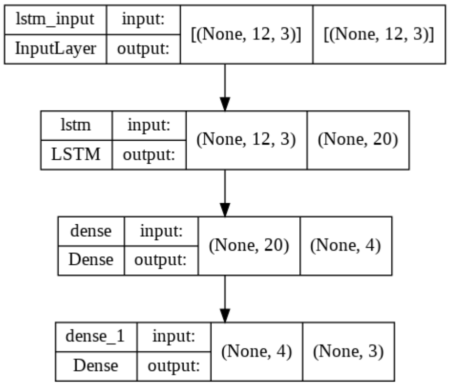
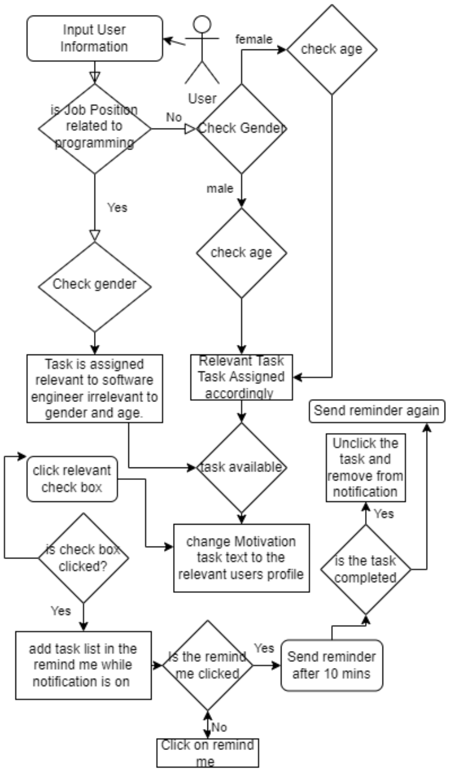
Documentation Report

1. Model Data Preparation

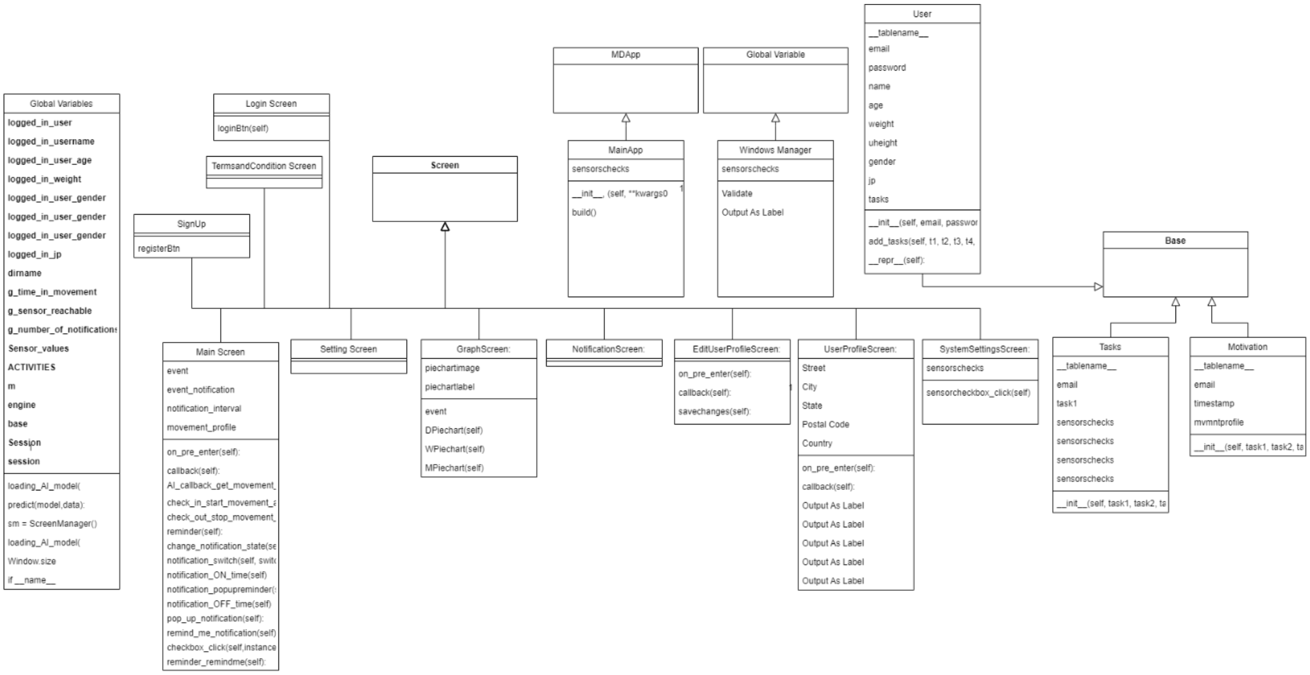
1. Import all the necessary libraries relevant to the documentation.
2. Import all the recordings by reading data using pandas while using the path of the recording where we remove the head and drop, time column. And append dataset in an array[n,3] matrix.
3. Used minmaxscalar to fit transform. Function to Loop the data to get 100 data in frequency 100m lookback 12 (since 12 sample rate) in 120 sec interval and convert to numPY matrix [n,12,3] and goies through the one hot encoder for transforming target values. We concatenate similar dataset x axis. where applicable get data [1,0,0], [0,1,0] and [0,0,1]. Use train, test, split from sklearn model selection to split dataset in training and testing set.

2. Model Preparation/3. Flowchart

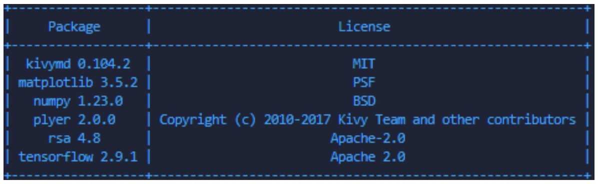



Input User Information
User
female
check age
is Job Position related to programming
No
Check Gender
male
Yes
check age
Check gender
Task is assigned relevant to software engineer irrelevant to gender and age.
Relevant Task Task Assigned accordingly
Send reminder again
Unclick the task and remove from notification
click relevant check box
task available
Yes
is check box clicked?
change Motivation task text to the relevant users profile
is the task completed
Yes
add task list in the remind me while notification is on
is the remind me clicked
Yes
Send reminder after 10 mins
No
Click on remind me

4 Class Diagram



5. We input the data from the user to the SQL database using an ORM, SQL achemy which has been integrated directly in the main file to edit, replace and input the user data. An Encryption of RSA was to be used. Numpy is used for matrix transformation. Plyer to call out notification. Tensorflow to read dataset and predict data from accelerometer. Kivymd is free to use, made by MIT.



```
+------------------+-----------------------------------------------------------+
|     Package      |                          License                          |
+------------------+-----------------------------------------------------------+
|  kivymd 0.104.2  |                            MIT                            |
| matplotlib 3.5.2 |                            PSF                            |
|   numpy 1.23.0   |                            BSD                            |
|   plyer 2.0.0    | Copyright (c) 2010-2017 Kivy Team and other contributors |
|     rsa 4.8      |                        Apache-2.0                         |
| tensorflow 2.9.1 |                        Apache 2.0                         |
+------------------+-----------------------------------------------------------+
```

6. All the software are licensed and they are free to use by any 3rd party to build an app. Our application is possible to be commercial distributed as it is relevant and authorized by the license owners.

7. The AI model is first used to train the dataset. It is then saved as a h5 file, pb file and tflite file. I opted to use the pb file saved in the directory. We call the load and use prediction soon the input values we get from the sensors or random values. As not connection to the accelerometer. When checked in, it will read the values provided by the sensors, after every 12 inputs, we change the values of the user, using rand.rand(). Every movement of the user is stored in the Motivation Task Table of the specific user(email is primary). Graph is plotted automatically from the data in the database.

8. The spec for using it in the mobile are mentioned below:

1. title = Build Yourself;
2. package.name = fitness_application ;
3. source.include_exts = db,tflite, pb, py,png,jpg,kv,csv ;
4. version = 0.1;
5. orientation = portrait/ vertical;
6. osx.python_version = 3.10 ;
7. osx.kivy_version = 2.1.0 (latest);
8. fullscreen = 0;
9. warn_on_root = 1
10. android.permissions = Access to Gyroscope, Bluetooth, Location, Accelerometer