**CA1 project specification**

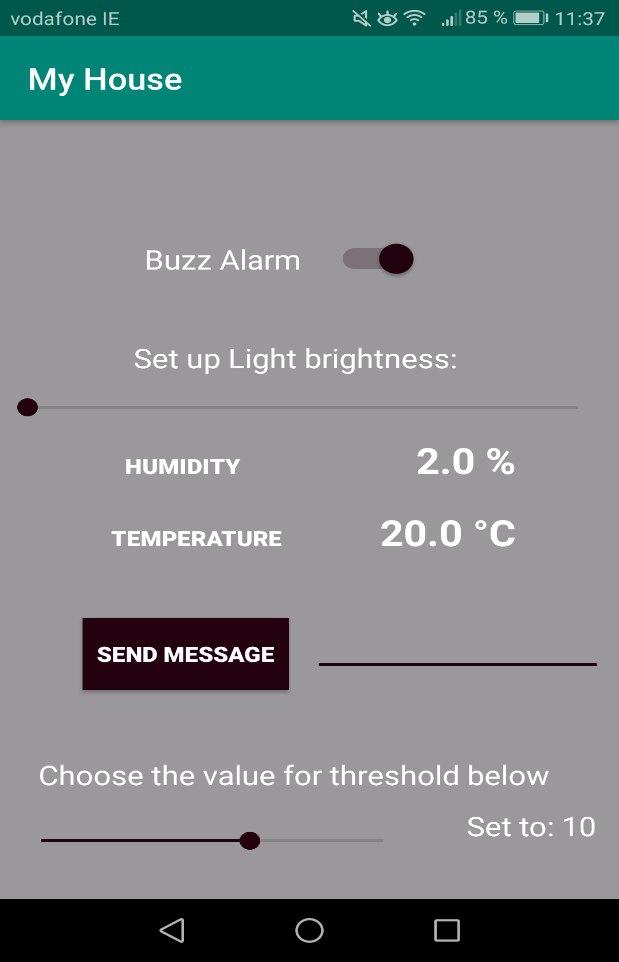
The idea behind this project is to control sensors inside the house via a mobile application and get real time data and notification. The application controls five different sensors; Buzzer, Temperature and Humidity sensor, LCD Display Screen, Light sensor and a LED sensor.

The application is developed using Java language and android studio for the mobile part, Python mainly in the raspberry part and server-side Java program using NetBeans to act as a receiver to display the data for the user

A multithreading is used to control how the python script sends and receive data, one thread is acting as a publisher sending data to the firebase database, another thread is acting as a receiver that gets readings from the firebase database. This gives the processor to run faster and avoid crashing

Firebase is used as a message broker between the android part and raspberry, it is used mainly for storing and retrieving it in real time as it offers real time database feature.

In the application screen we have a toggle button for the buzzer that sets the value in the firebase to on/off, a stream listener in the python side is listening to all changes and gets the values/notifications, once the values changes, another script is run to check if it's on or off, depending on this, the buzzer port is set to 1 or 0



The other feature is the side bar that controls the light, the value passes from 0 to 100, when the side bar is slid, the values gets changed in real time in the firebase database and another script is reading from the python side to control the intensity of the Led and sets the light intensity accordingly. The light gets dimmed to lightened instantly when sliding

The Humidity and Temperature are displayed in real time, if the value changes, it gets sent to firebase, a script listener from the android part is getting the values and sets the value in the application main screen. The next sensor is the LCD Display Screen, we have given the ability to send a message from the mobile application to display it on the LCD screen. In the android part, a text field to enter the message and button click to send it. A toast message that gets displayed to confirm that the message was sent successfully

The last functionality is the ability to set a threshold to switch the light on/off depending on the value that was chosen from the second sidebar. The light sensor is getting the actual light value, then we have limited the values only to 0,10,20,30,40,60,80 and 100 to be selected. We did this by checking if the modulus of the values selected from the sidebar is equal to 0 when divided by 10, then display them into the screen and sent them to the database. The user can select one value and sets the threshold in the python screen to this value. The benefit of this operation is to change the threshold value according to the outside light

In able to allow safe error handling and avoid system to crash, we have implemented error handling both from the python side and the android codes. When we are working JSON, we have made sure that if the JSON object is correct/malformed, we handle the right error and catch it at the end. Other methods used to handle errors when working with sensors is to set the sensors to off if there is an error. As an example, if the Ctrl +C is clicked to exit the program, we make sure that the LED sensor is switched off and the buzzer set to “0”. Other error handling will be explained in the video

***Link to the video:*** <https://youtu.be/tcGUz_AaGtM>