

QBKL - Group 16:

The project uses both Arduino and Raspberry Pi, and depends on the following sensors/motors:

1. Ping sensor
2. PIR sensor
3. Vibration Motor
4. MPU 9150 (Accelerometer, Compass, and Gyroscope)
5. RFID Scanner
6. Webcam (both laptop and Logitech C170)
7. GY-30 BH1750FVI Digital Light Intensity Sensor
8. 3v High Output LEDs

A separate document named "Features" describes the various features that are implemented in this project.

1.Arduino

Arduino deals with the Bad Posture Detection feature of this project. 90 degrees (the pitch, the position of the back/collarbone with respect to the ground) is defined as the standard posture, the correct range was defined as 60-120 degrees. The range was defined to give the required flexibility to the individual. Further we also plan to allow user to set his/her ideal posture by himself.

Installation Steps:

1. Copy all the files provided in the zip folder to your Arduino directory.
2. For calibration run code in the Arduino MagiCal folder and move the accelerometer 360 degrees in three planes (for calibration of the MPU9150 sensor).
3. Run the Kingtide9150_vib_motor.ino code in Kingtide9150_vib_motor folder.
4. The serial monitor will display the current pitch readings and the count(vibration count)

2.Raspberry Pi

Raspberry Pi deals with the Security of the cubicle, Break reminder, Concentration reminder and lightening of the cubicle.

Installation Steps:

1. `sudo apt-get install espeak python-espeak`
2. `sudo apt-get install fswebcam`

Guidelines for using the system:

1. Run the code `final.py`.
2. In order to unlock system user needs to scan its RFID.
3. User is asked to set time after which he needs to be reminded to take a break.

We have used counter variable (`counter=1000ms`) which indicates the amount of time for which the system should wait before locking itself if the user is not present at cubicle.

3. Concentration Reminder

Since the Concentration reminder (Real-time) feature is Computation Intensive , it was decided to do it using Computer's processor and GPU. To run the code , user needs to have a webcam and image processing library - `opencv2` , `dlib` and other GPU specific software (`CUDA`, `Nvidia Drivers`).

Installation Steps and Guidelines :

1. Install facial recognition library (`pip install face_recognition`)
2. Copy your picture where the code is , so that we can recognise your face. Change the code on line 20 to enter the path of your picture.
3. Run `mqtt.py`
4. Anyone using the Data from the concentration reminder (Whether the person is concentrated or not) , can subscribe to the channel .Eg. Smart Classroom (Group 15) use it to detect whether every student is concentrated or not .