

Smart Locker

Peripherals Lab (CS 341)

(Group 19)

Introduction

We have created a very basic prototype for a smart locker for our Libraries, where students can leave their baggage.

We have an inbuilt automated locking mechanism, where the Student just needs to scan his/her Student ID Card on the RFID Scanner, and the specific locker gets registered with the student and also sends a confirmation SMS to the student.

Now the Locker can only be opened by scanning that specific ID Card back again onto the scanner. The Intelligent anti-theft system ensures the safety of your belongings and starts the buzzer alarm in case of any tampering.

The Student is also subscribed to SMS messages in case any tampering is detected by the Locker.

When emptied, the locker automatically gets unassigned and is available for use by any other student.

We are also using MQTT to provide the locker allocation data.

Installation

Install paho-mqtt

- Install mosquitto to run python scripts for wireless communication
- Run the commands: “pip install paho-mqtt” and “sudo apt-get install mosquitto”

Install Twilio Python Helper Library

- The Twilio Python Helper Library makes it easy to interact with the Twilio API from our python application.

- Run the command: “pip install twilio”. (We have provided our API key and authentication key for the demo. For production purposes, you need to add your own)

Connect the sensors to the PI as follows:

1. Connect the PI to a power Source and and Ethernet Cable, or WIFI Dongle (and set it up) (GPIO Mode is set to GPIO.BOARD)
2. Connect the Following sensors:
 - a. Stepper Motor to pin 32, 36, 38 and 40
 - b. Buzzer to pin 15
 - c. Ultrasonic Sensor to pins (GPIO_TRIGGER: 35, GPIO_ECHO 33)
 - d. RFID to pin 8,9,10
 - e. Vibration Sensor to pin 16
 - f. Led to pin 18
3. Connect Vcc and Gnd for each of the previously given sensors appropriately using a breadboard

Follow the following steps to get the locker running

- Connect your computer to the PI via SSH
- Start executing the program by executing the files main.py and vibration.py on separate terminals connected to the pi via ssh

(Note: since we are using the messaging APIs along the code. We need to use proxy free internet (NOT IITG_WIFI))

Group - 19

Siddharth Sharma	160101071
Nikhil Kumar	160101047
Mohit Singh	160101043
Ritik Agrawal	160101055