
ANTI-THEFT PASSWORD-BASED IGNITION SYSTEM

Suman Naidu R - 1AP21IS036

Bidhan Gope - 1AP21CS011

Sandeep J - 1AP22CS405

1. Software Installations

1.1. Raspberry Pi:

- Update your Raspberry Pi OS:

`'sudo apt update && sudo apt upgrade'`

- Install Python 3 and pip (if not already installed):

`'sudo apt install python3 python3-pip'`

- Install required Python libraries:

`'pip3 install RPi.GPIO requests'`

1.2. Laptop:

- Install Python 3 (ensure it's added to the PATH during installation).

- Install pip (if not already installed):

`'python -m ensurepip --upgrade'`

- Install OpenCV dependencies:

`'pip install opencv-python'`

2. Project Setup

2.1. Raspberry Pi:

- Create a directory for the project:

‘AntiTheftIgnitionSystem’

- Place the ‘pi_main.py’ file in this directory.

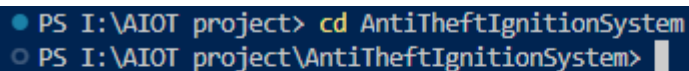
2.2. Laptop:

- Create a directory for the project, ‘AntiTheftIgnitionSystem’.
- Place ‘app.py’ and ‘authorized_face.jpg’ in this directory.
- Update the ‘laptop.py’ file to load ‘authorized_face.jpg’ for face recognition.

3. Running the Project

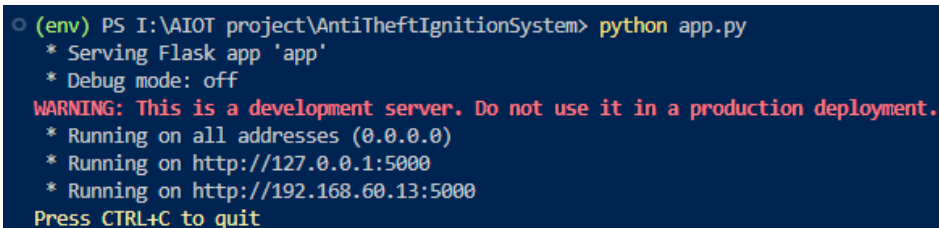
Step 1: Start the Flask Server on the Laptop.

- Open a terminal or command prompt from the project directory:



```
PS I:\AIOT project> cd AntiTheftIgnitionSystem
PS I:\AIOT project\AntiTheftIgnitionSystem>
```

- Run the ‘app.py’ script:



```
(env) PS I:\AIOT project\AntiTheftIgnitionSystem> python app.py
* Serving Flask app 'app'
* Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment.
* Running on all addresses (0.0.0.0)
* Running on http://127.0.0.1:5000
* Running on http://192.168.60.13:5000
Press CTRL+C to quit
```

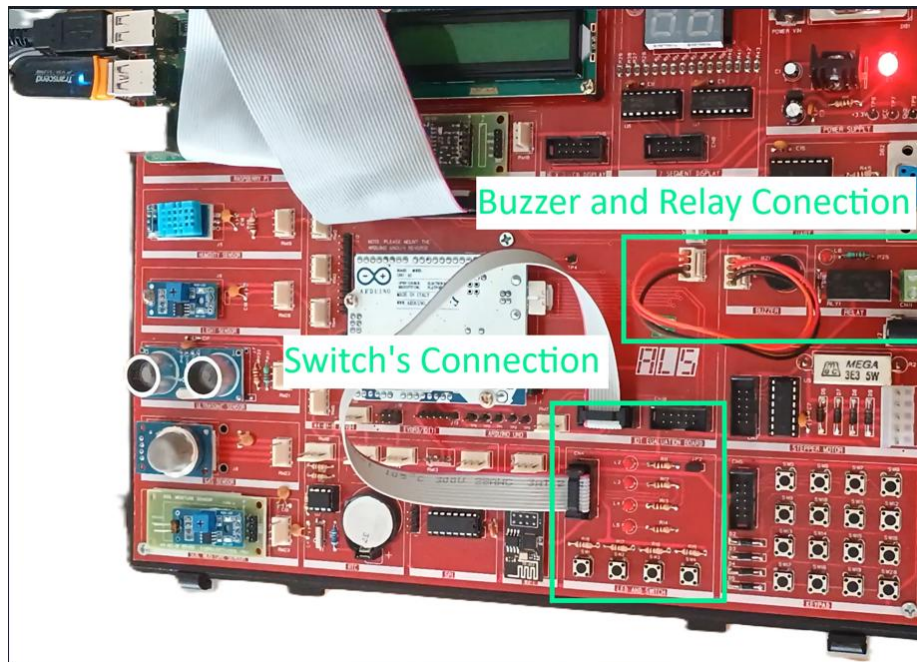
- Note the IP address and port (default is ‘http://0.0.0.0:5000’).
- Ensure the laptop and Raspberry Pi are on the same network.

Step 2: Run the Raspberry Pi Script

- Open a terminal from the project directory
-

- Run the `pi_main.py` script with Python 3
`python3 pi_main.py`
- The system will initialize and wait for input from the switches.

```
Security System Initialized...
```



Step 3: Test the System

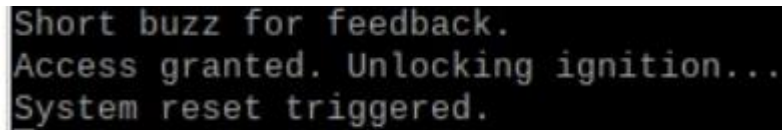
1. Press **Switch 2** on the Raspberry Pi to trigger face recognition.
 - a. If the face matches `authorized_face.jpg`, the relay will activate.
 - b. If the face does not match, the system will prompt for a password.

```
Security System Initialized...
Face recognition request received...
Face not recognized.
Retry face recognition? (y/n): yes
Face not recognized. Requesting password authentication.
Enter password: 1234
Password accepted.
Short buzz for feedback.
Access granted. Unlocking ignition...
```



```
Face recognition request received...  
Face recognized successfully!  
Access granted. Unlocking ignition...
```

2. Press **Switch 3** to reset the system state.



```
Short buzz for feedback.  
Access granted. Unlocking ignition...  
System reset triggered.
```

3. Press **Switch 4** for an emergency shutdown (with confirmation).
 4. Notes.
 - a. Ensure that the `SERVER_URL` in `pi_main.py` is updated with the laptop's IP address and port.
 - b. Verify that the `authorized_face.jpg` file is clear and the face is properly visible for recognition.
 - c. If running into issues with dependencies, ensure Python and pip are correctly installed on both devices.
 - d. Keep the Raspberry Pi powered with a reliable source to avoid interruptions.
-