# Smart Home Security System

**by**

**Archana Srivastava**
**Shahid Ali**
**Umar Ahmad**
**Monika Edla**

A project on Smart Home Security submitted to the Graduate College of Marshall University. In partial fulfillment of the requirements for the degree of Master's in Computer Science and Information Systems

Approved:

_____

Instructor of Capstone course

_____

Director of Major Department

_____

Chairman of Graduate Programs

Marshall University
Huntington, West Virginia
Dec 2017

# Acknowledgements

We would like to thank our Advisor Dr. Wook-Sung Yoo, for all his help and guidance he has given us through the completion of our masters.

We would also like to thank Dr. Haroon Malik who made considerable contribution in meeting all the requirements. He has helped us learn to be professionals as working in a software team, and has patiently guided us through every phase of our project. The Smart Home Security team would collectively like to thank Dr. Paulus Wahjudi, for his mentoring and guidance and would also like to thank and appreciate all the helpful people at Smart Home Security, for their support. Finally, we would like to thank all the Computer Science and Information System Faculty members for supporting us in completion of our course.

# Abstract

To implement a prototype that allows the user to remotely connect and control the security devices that are implemented in the home, apartment or a building. This system consists of multiple sensors and microcontrollers that communicates with the base station(Laptop). The reed switch sensor will generate alert when an event occurs open/close and send a notification to the base station(Laptop). Smart home security will be implemented in a way that allows the user to add or remove and enable or disable the devices, whereas also arm and disarm the system.

\

# TABLE OF CONTENTS


**1. Introduction**
- 1.1. Introduction 5
- 1.2. Background 5
- 1.3. Project Description 6
- 1.4. Scope 6
- 1.5. Summary 6

**2. Requirement Analysis 7**
- 2.1. Technology Selection 7
- 2.2. Electronic Equipment 8
- 2.3. Steps to Install Required Software 11
  - 2.3.1. Raspberry Pi Functionality 11
  - 2.3.2. Creating Hotspot 14
- 2.4 System Overview 14
  - 2.4.1. UI Functionalities 15
- 2.5. Functional Requirements 16
  - 2.5.1. System Functionalities 156
- 2.6. Context-Level Diagram 167

**3. Design Process** 168
- 3.1. User Process 178
- 3.2. Functional Design 19
  - 3.2.1. User Activity Diagram 19
  - 3.2.2. Sequence Diagram 20

**4. System Prototype** 23
- 4.1. Smart Home Security UI Homepage 23
- 4.2. Add Device 24
- 4.3. Remove Device 26
- 4.4. Arm System 27
- 4.5. Disarm System 28
- 4.6. Enable System 29
- 4.7. Disable System 30
- 4.8. Notification Panel 32

**5. System Integration and Validation** 324
- 5.1. General Assumption 324

**6. Conclusion** 35

# 1. Introduction to the System

## 1.1.Introduction

Smart Home Security system allow the user to remotely connect and control the security devices that are installed on doors and windows in home. This system consists of multiple sensors and microcontrollers that communicates with the central hub installed on users laptop. Smart home security system will be implemented in a way that allows the user to Add or Remove a device, and Enable or Disable a device, whereas also Arm or Disarm the entire system.

## 1.2. Background

A Smart Home Security System controls and monitors the security devices such as sensors and motion detectors that are installed in house. The available Smart Home Security Systems are expensive and have less possibility of modification. Our team will be implementing a smart home security prototype that will provide security to the doors and windows. The prototype will require less maintenance as the owner of the system can make modifications to the prototype, for example, replacing a microcontroller and sensors when required. Our team will provide a DIY (Do It Yourself) prototype that will be cost effective (affordable by any individual). The prototype will allow user to monitor the events generated by the sensors mounted on doors and windows as well as permits the user to arm or disarm the system with the audio play by one click feature. The user will also be able to add and remove single devices or multiple devices from the system. Our prototype will be centrally controlled by a central hub. The central hub is a software solution that will run on laptop. Moreover, we plan to have our product available at GitHub so to receive contributions from smart home security community.

### 1.3. Project description

The project proposal for smart home security will consists of set of microcontrollers or IoT devices with reed sensor mounted on it and the central hub to configure and monitor the microcontrollers or the IoT devices. The microcontrollers along with the sensors will be mounted on doors and windows. These sensors will be used to trigger an event notification to the hub if there is any security breach. There will be no programming required on the user side.

### 1.4.Scope

- Add or remove microcontroller from the deployed security system with minimal effort. This means user do not need to code or write a separate program for any additional microcontroller. The act can simply be achieved by selecting the appropriate option (feature) from the base station, e.g., ‘Add new security device’ and vice-versa
- Temporarily activate and deactivate existing microcontroller or any specific sensor mounted on them from the base station.
- Allow to reset or flush any false notification.

### 1.5. Summary

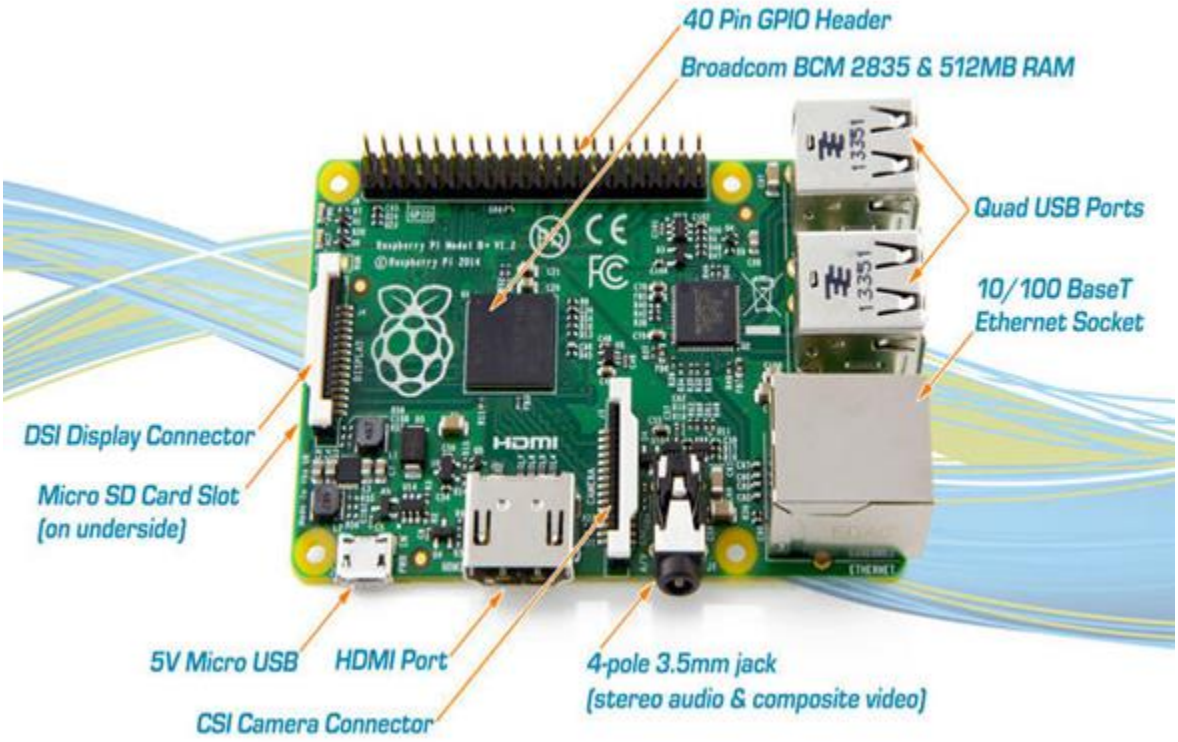
The Smart Home Security is a prototype that safeguards doors and windows of a house, apartment or a building. The reed switch (sensor) will generate alert when an event occurs (open/close) and sends a notification to the central hub (Laptop). A user can add or remove microcontroller from the base station, and the project constraints will allow him to copy an existing code from one microcontroller into another microcontroller.

# 2. Requirement Analysis

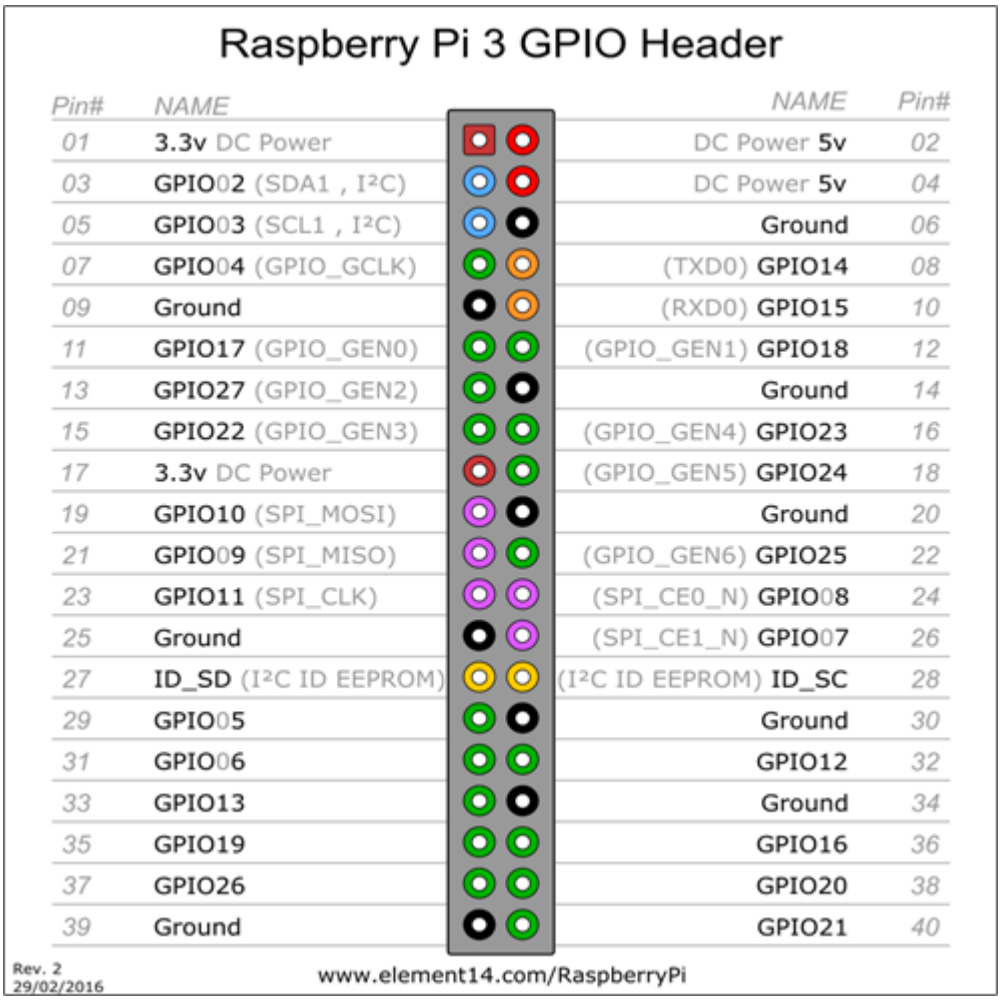
## 2.1. Technology Selection

- JAVA
- NOOBS
- PUTTY
- VNC
- NETBEANS

**Raspberry pi**



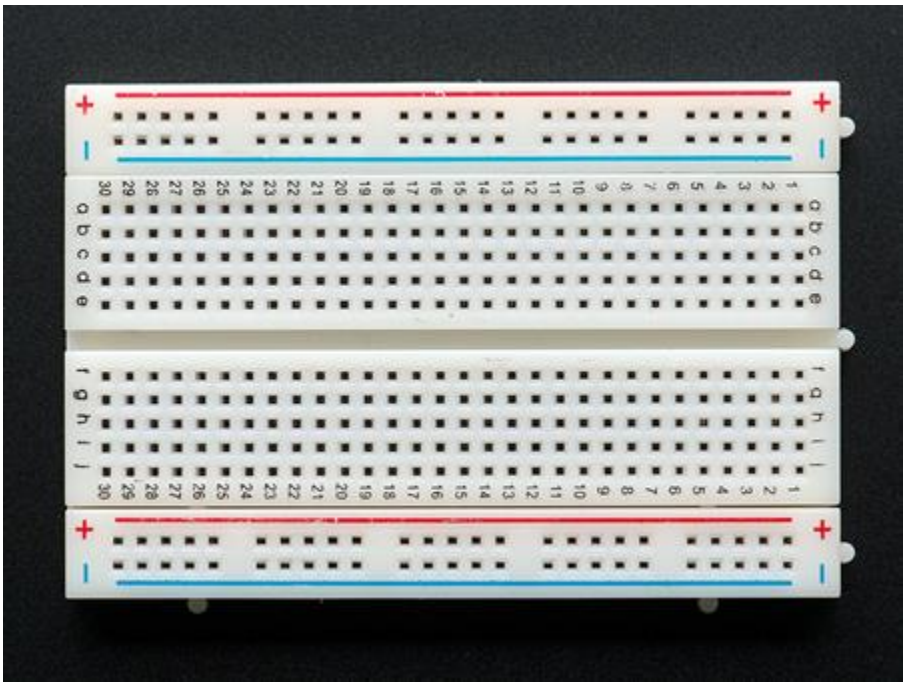
**GPIO:** General purpose I/O (GPIO) is a generic pin on a chip whose behavior can be controlled by the user at runtime.




## 2.2. Electronic Equipment:

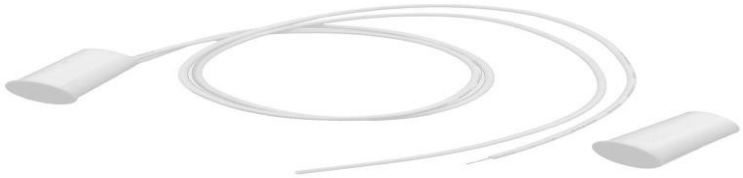
- Bread Board
- Jumper Wires
- Read Switch
- Resistors

**1. Breadboard**: A breadboard also known as protoboard is a type of solderless electronic circuit building.You can build a electronic circuit on a breadboard without any soldering.

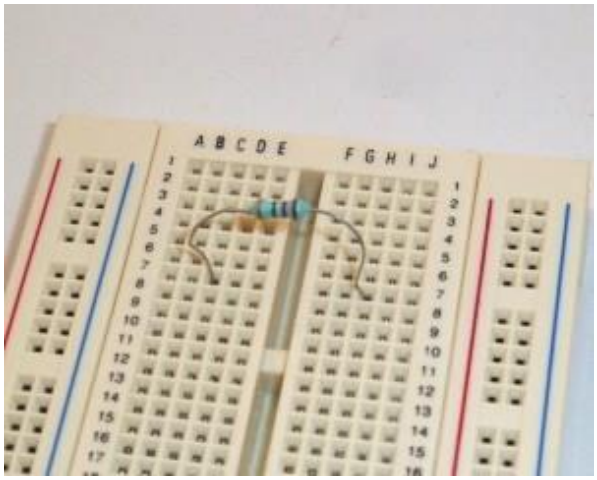


**2. Jumper Wires:** Try to use distinct colors to differentiate different purposes

**3. Reed Switch**: A reed switch is an electrical switch operated by an applied magnetic field.



**4. Resistors:** A resistor is a passive two-terminal electrical component that implements electrical resistance as a circuit element.




### 2.3. Steps to Install Required Software

It's Important for the user to know all the required software packages as well as the ways to install them as it is a DIY system. This category will allow the user to know about all the software programs along with the installation requirements.

#### 2.3.1 Raspberry Pi Functionality

The softwares programs that are required for Raspberry Pi are:

1. NOOBS

2. VNC Server
3. NETBEANS

## NOOBS Installation

**Requirements to install NOOBS on the Raspberry Pi**

- A computer with an SD card slot
- An SD or microSD card of at least 8GB

### Step 1: Download NOOBS and extract it

NOOBS should be downloaded as a .zip file, so before doing anything else, the file should be extracted.
It can be downloaded from the following link:
https://www.raspberrypi.org/downloads/noobs/

### Step 2: Format an SD card

To format the SD card, download any "SD Formatter" software. After installing the software the user needs to format the card inserted in the computer.

### Step 3: Put the NOOBS files on the SD card

Now, just drag and drop the NOOBS files into your formatted SD card. You want the files only, so if your .zip extracted to a folder, open that folder and select only the folder contents inside.

### Step 4: Put your SD card into your Raspberry Pi and boot it up

Once you have NOOBS on your SD card, using it is incredibly easy. Just put the SD card into your Raspberry Pi and start it up. This will initiate the installation.

## VNC Installation

A VNC viewer (or client) is installed on the local computer and connected to the server component such as Raspberry Pi, which must be installed on the remote computer. The server transmits a duplicate of the remote computer's display screen to the viewer.

It can be installed from the following link on the local computer:
https://www.realvnc.com/en/connect/download/viewer/

The user have to Add the IP address of the Raspberry Pi in the Installed VNC Viewer and the user will be asked to provide admin ID and Password. Upon providing the details, the user will be remotely connected to the Raspberry Pi and can see the screen on the computer.

For the Raspberry Pi the user can just open the terminal from the taskbar and write:

**Command**: sudo apt-get update

**Command**: sudo apt-get install realvnc-vnc-server realvnc-vnc-viewer

Next Step,

- On your Raspberry Pi, boot into the graphical desktop.
- Also ensure that VNC is Enabled.
- Select **Menu** > **Preferences** > **Raspberry Pi Configuration** > **Interfaces**

Then,

Type the Following Command in Terminal

**Command** : Sudo Raspi-config

VNC Server/Viewer is installed for Raspberry Pi. To test the connection, enter the IP address from the VNC server into VNC viewer and click connect.

**NETBEANS Installation**

NetBeans 8.0.2 is highly recommended as it has pre-installed libraries.

**NetBeans step by step procedure to download software:**

1. **NetBeans IDE 8.2 Installation Instruction:**

   https://netbeans.org/community/releases/82/install.html

2. **NetBeans IDE 8.2 Download:**

   https://netbeans.org/downloads/

* To use NetBeans for java you need to first install Java Development Kit.

* Download “NETBEANS IDE”

* Run the installer then run the downloaded installer.

**OTHER Software Packages (optional):**

**PUTTY**

This Software can create a remote access with the Raspberry Pi from the computer. It can be installed from the following link:

https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html

**Steps to run the server on startup in Raspberry Pi:**

- Download the code from Github for the server to run on the pi
- Save the file in **Documents** folder in the root directory of the raspberry pi
- In Documents folder unzip the downloaded server file
- Then in terminal type the following command line code:
  **Command:** sudo nano /etc/rc.local
- In the new popup file locate the keywords 'fi' and 'end' at the end of the file
- Type the following file path in between 'fi' and 'end' keywords:
  **Command:**
  **(sleep 10; python3 /pi/home/Documents/smart-home-security-server/shs-server.py)&**
- Then press ctrl+x to save the file and hit enter
- Then reboot the system:
  **Command:** sudo reboot
- We are rebooting the system to make sure the server runs.
- After the system reboots wait for 10 second after the pi startup

### 2.3.2. Creating Hotspot

Mobile hotspot from laptop will allow the user to share the network connection with other devices over Wifi, Ethernet, or any cellular data connection. Follow the steps below to create your personal connection:

- Select the **Start** button, then select **Settings** > **Network & Internet** > **Mobile hotspot**.
- For **Share my Internet connection from,** choose the Internet connection you want to share.

- Select **Edit** > enter a new network name and password > **Save**.
- Turn on **Share my Internet connection with other devices**.
- To connect on the other device, go to the Wifi settings on that device, find your network name, select it, enter the password, and then connect.

Once the devices are connected, to see and verify the MAC address of all the available devices in network by following the step below:

- Open **Terminal** by right clicking on **Start** button > click on **Command Prompt**
- In the **Terminal** type > **arp -a >** hit **Enter**

## 2.4. System Overview

Smart Home Security system allow the user to remotely connect and control the security devices that are installed on doors and windows in house. This system consists of multiple sensors and microcontrollers that communicates with the central hub (laptop). Smart home security system will work in a way that allows the user to add or remove a device, Arm or Disarm the System with the audio play and Enable or Disable the system. The System Consist of a GUI (Graphical User Interface) on the Laptop.

### 2.4.1. UI Functionality:

- ADD
- REMOVE
- ARM
- DISARM
- ENABLE
- DISABLE

**ADD**: To ADD a device, the user will have to select the MAC address of the Microcontroller appearing on the screen, after which the user can name the device and click on **Save.** The system will ask for confirmation and the device will be saved.

**REMOVE**: All the available devices are shown in the REMOVE panel. If the user selects the device and click on Remove option, this will remove the device from the system and a message will be generated "This device has been removed".

**ARM**: When the user clicks on ARM button, the complete system will be armed, and the status of the system will be 'ARMED' with an audio playing "SYSTEM IS ARMED".

**DISARM**: When the user clicks on DISARM button, the complete system will be disarmed, and the status of the system will be 'DISARMED' with an audio playing "SYSTEM IS DISARMED".

**ENABLE**: All the disabled devices will be appear in the ENABLE Panel. When the user selects the device from the disabled devices and clicks on Enable, that device will be automatically enabled and won't be seen on the disabled screen.

**DISABLE**: All the enabled devices will be appear in DISABLE panel. When clicked on anyone of the enabled devices that device will be disabled and won't be seen on the enabled screen.

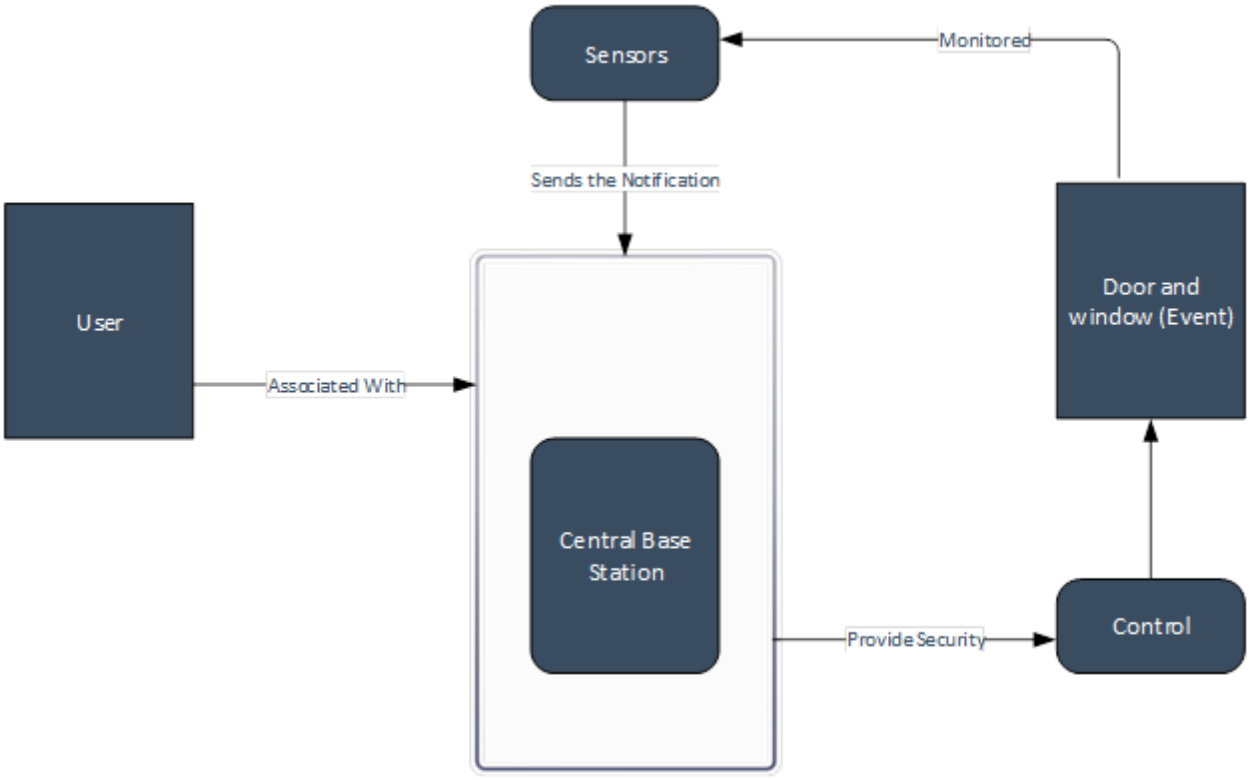
## 2.5. Functional Requirements

### 2.5.1. System Functionality

| Index | Item | Description |
|---|---|---|
| 1 | HomePage | HomePage shows the system status and all the six buttons through which user can function the Complete system |
| 2 | Status | System armed = activated<br>System Disarmed =Deactivated |
| 3 | (Arm) Button | Arms the entire system |
| 4 | (Disarm) Button | Disarms the entire system and silences all the alarms |
| 5 | (Add) Button | To Add a new device to the system, The user needs to select a MAC address and give it a Name |

| | | |
|---|---|---|
| 6 | (Remove) Button | To Remove an existing device from the system |
| 7 | (Enable/Disable) Button | For Enabling or Disabling a device in the system |

### 2.6. Context-Level Diagram




The Context-Level diagram shows the entities that interact with each other as this security system is for the user who will have a Central hub (laptop) that controls and provide security to doors and windows with the help of sensors and microcontrollers that are mounted on the doors and windows in the house.

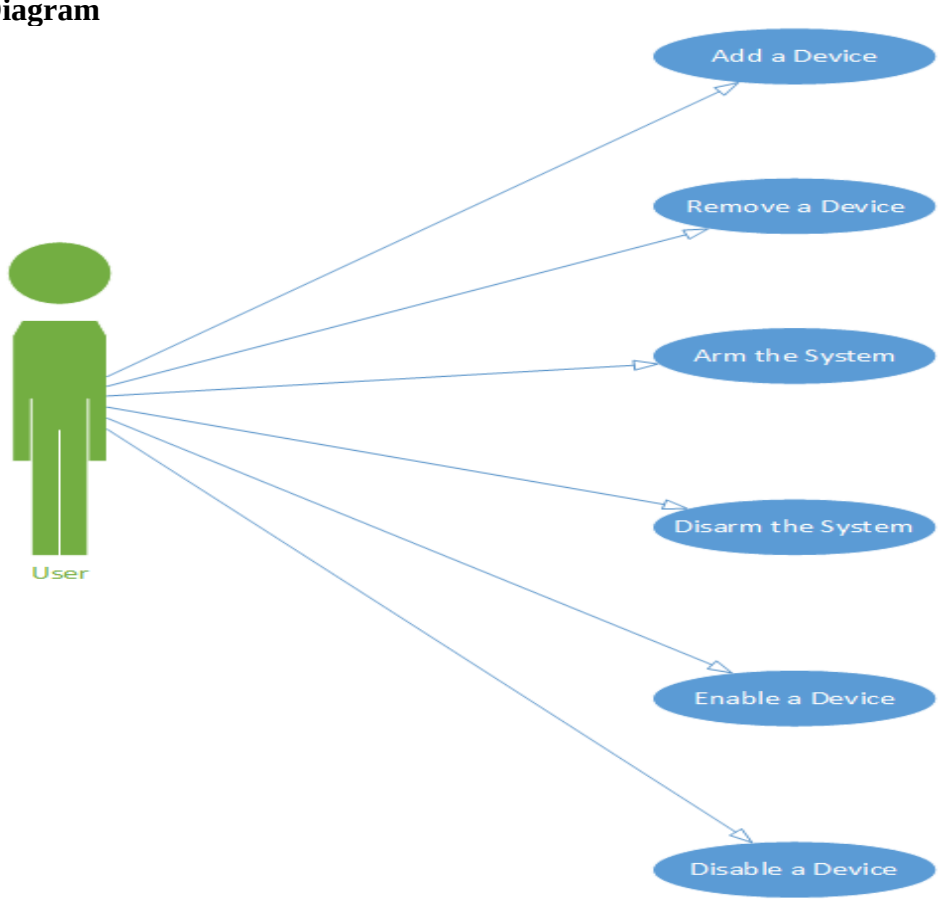
# 3. Design Process

This segment consults about the different outline parts of the project created. The whole design is characterized into different segments and then discussed elaborately. The different perspectives of the design that are considered are product flow and functional flow design.

## 3.1. User Process

In user process, the user can Add a device by selecting the MAC address and naming it. the user can also Remove a device by simply using a one click feature. The user can also Arm and Disarm the complete system. If the user does not wish to Arm or Disarm the complete system, the user can Enable or Disable a particular device. All functions can be performed with one click feature, At the left top, User can see the status of the system. If the system is Armed, the status will be "ARMED" and if the system is disarmed the status will be 'DISARMED".

**Use Case Diagram**



## 3.2. Functional Design

The functional design of the product is shown here using the activity diagram.

### 3.2.1. User Activity Diagram

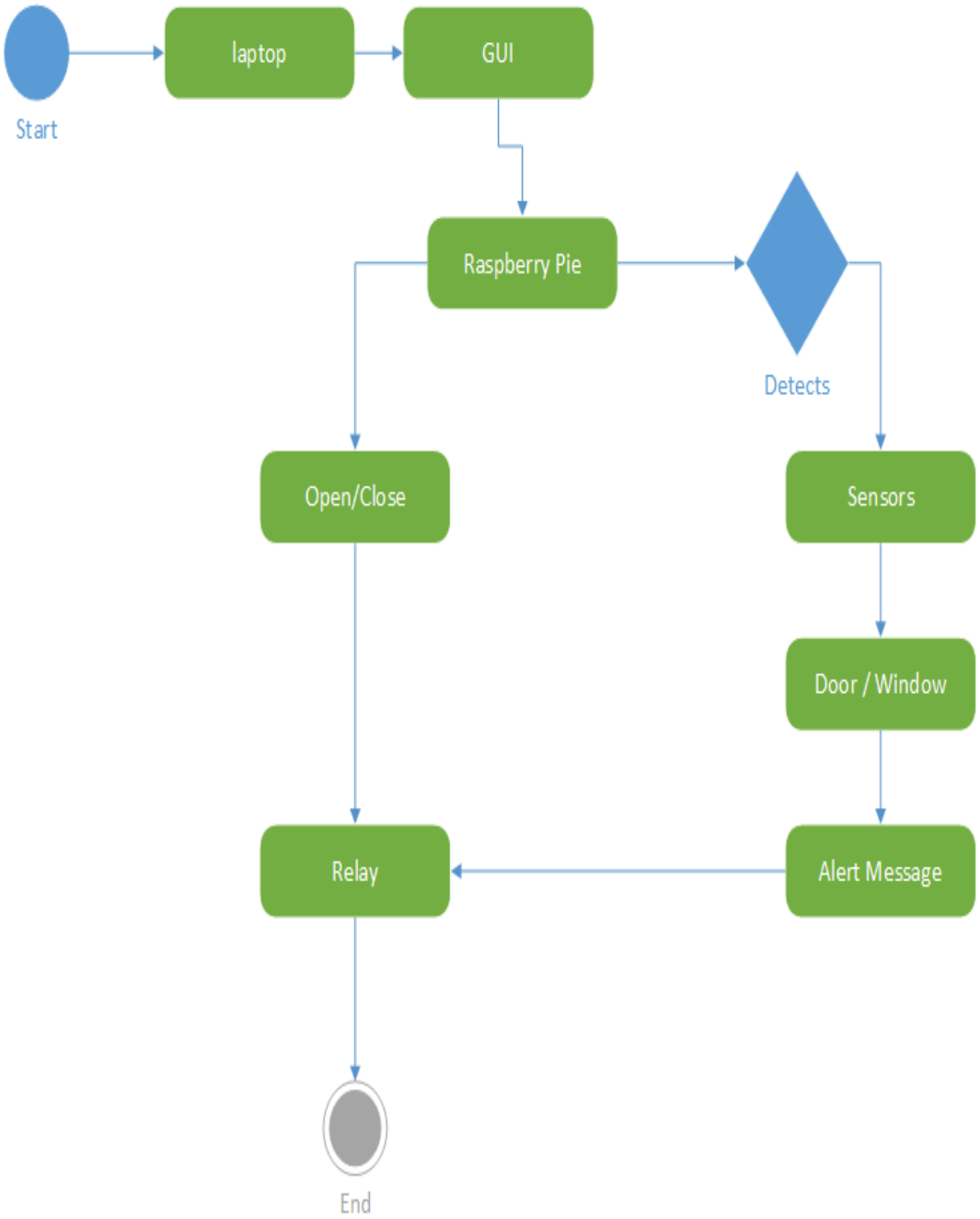



**Figure: User Activity Diagram**

In the above activity diagram, the process flow has been described as how the system will work and what are the ways to make the connection between the Door/Window and the User Interface (UI). There will be a Central Hub on user laptop and Reed Switch connected to microcontrollers (Raspberry Pi). When the magnetic connection of the reed switch breaks, it will generate a Notification that will be send to the laptop (UI). That means, if the door or window is opened and or closed, the user can see the alert on the central hub (UI) running on laptop.

### 3.2.2. Sequence Diagram

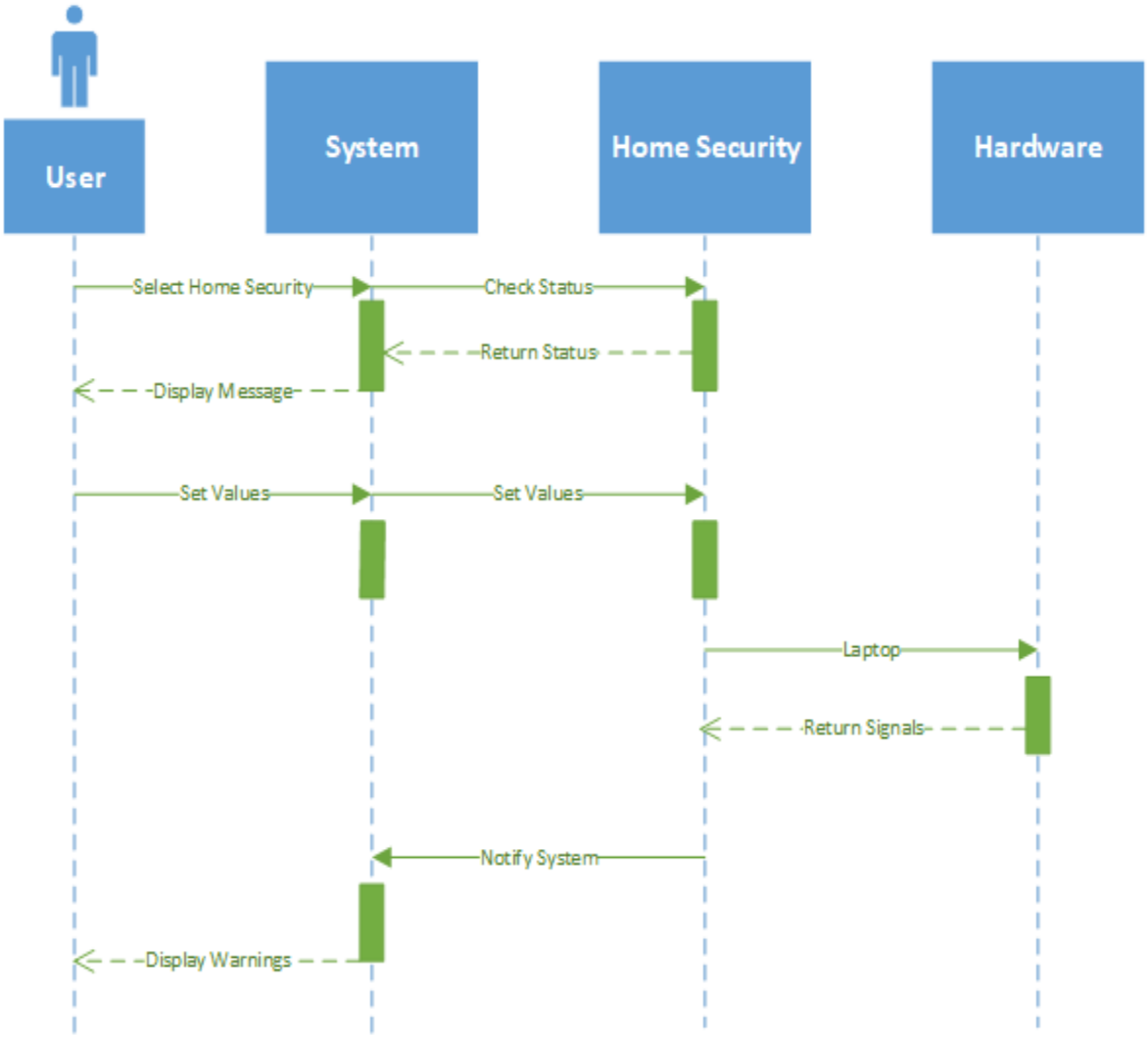


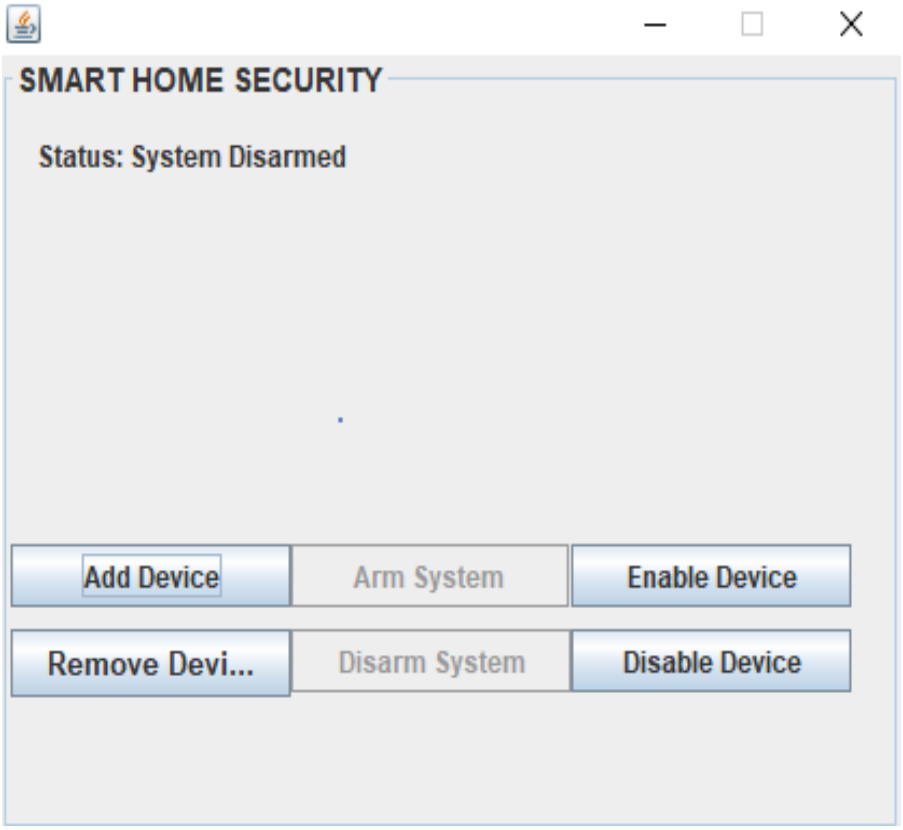

**Figure: Sequence Diagram**

The above Sequence diagram shows the flow of all the activities that are performed by the user on the system. It is the user who selects the system and acts as an administrator, who can check or update the system. The devices that are mounted on the door and window will send the signal to the central hub (laptop). The user can see the message on the Home Screen of the UI. The user can Enable and Disable the device if wish to.

# PROTOTYPE OF THE SYSTEM

# 4. System Prototype

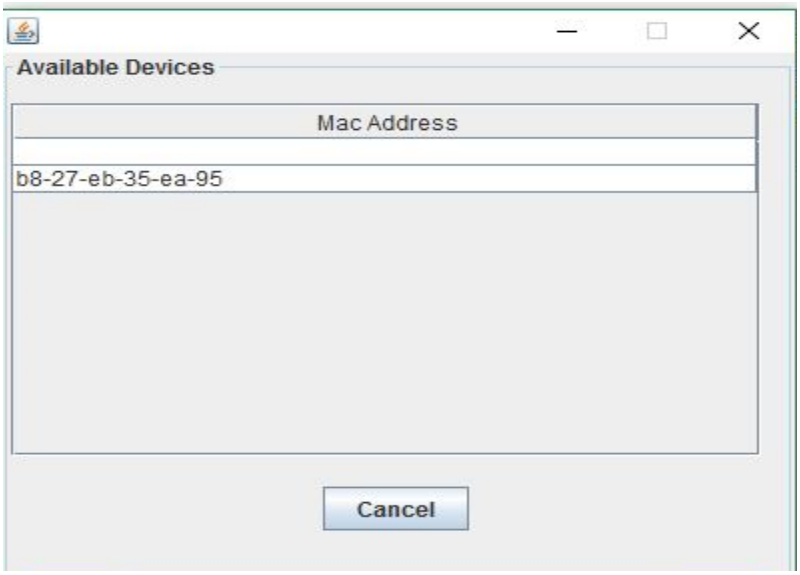
## 4.1. Smart home security UI Home page



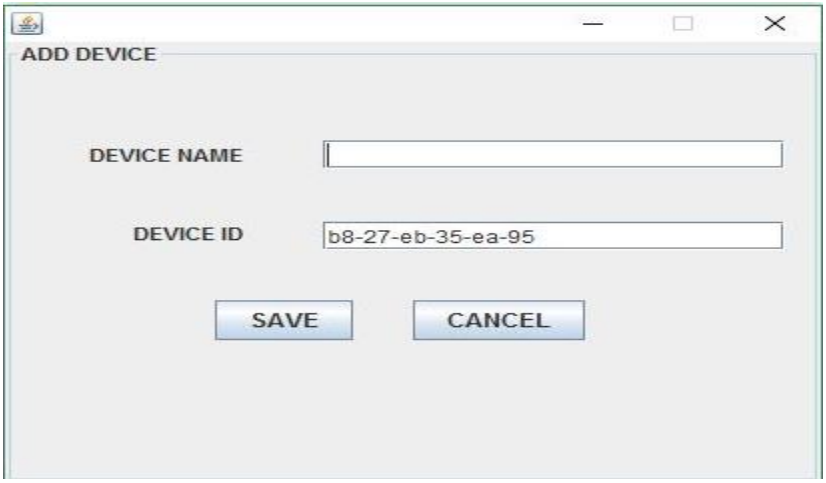

The Home screen buttons and Display Screen allow you to control all system functions. Additionally, the Screen display shows the description of all system occurrences. When the System is Armed, a full-function Message is displayed on the Home screen.

## 4.2. Add Device

To ADD a device, the user will have to select the MAC address of the microcontroller and give it a name, upon providing the information and clicking Save button, the system will ask for user confirmation and the device will be saved.

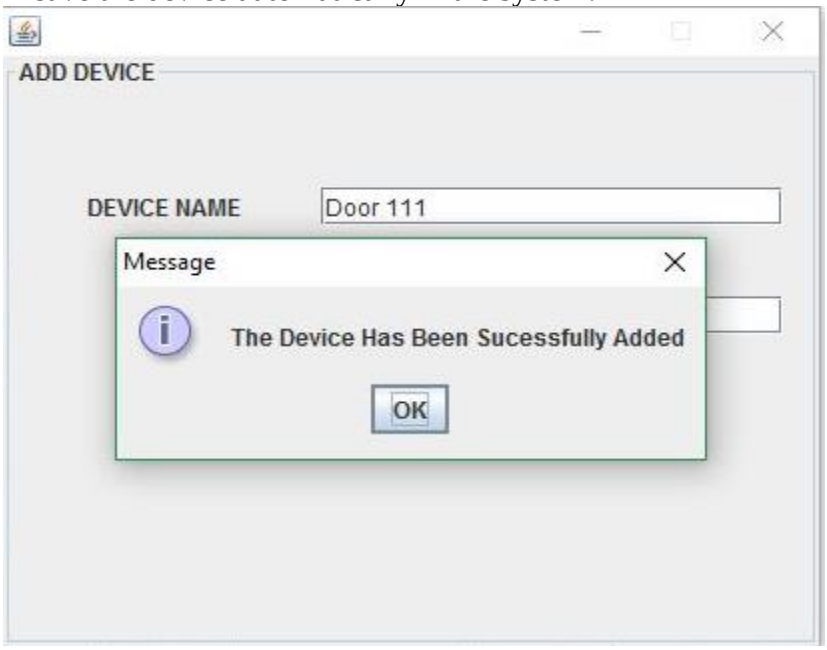



Once the MAC address is selected, it will ask to provide a name to the device.




Once the user confirm adding a device, the system will ask for a confirmation message.

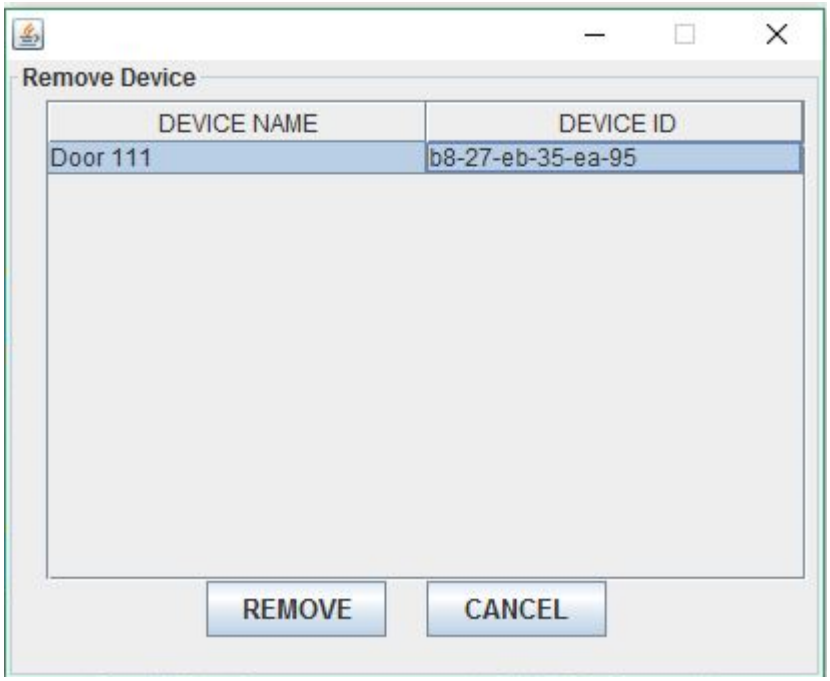

Clicking YES will save the device automatically in the system.



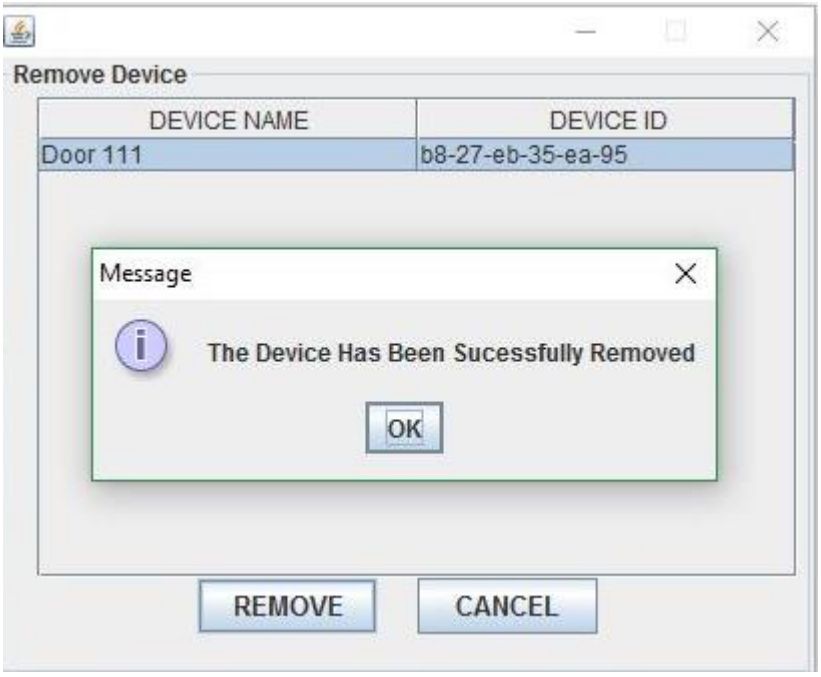

### 4.3. Remove Device

To remove a device, the user will have to select the device from the available devices. A single click on the Device that needs to be removed and click Remove button, the Device will be removed automatically.


Remove Device
DEVICE NAME
DEVICE ID
Door 111
b8-27-eb-35-ea-95
REMOVE
CANCEL

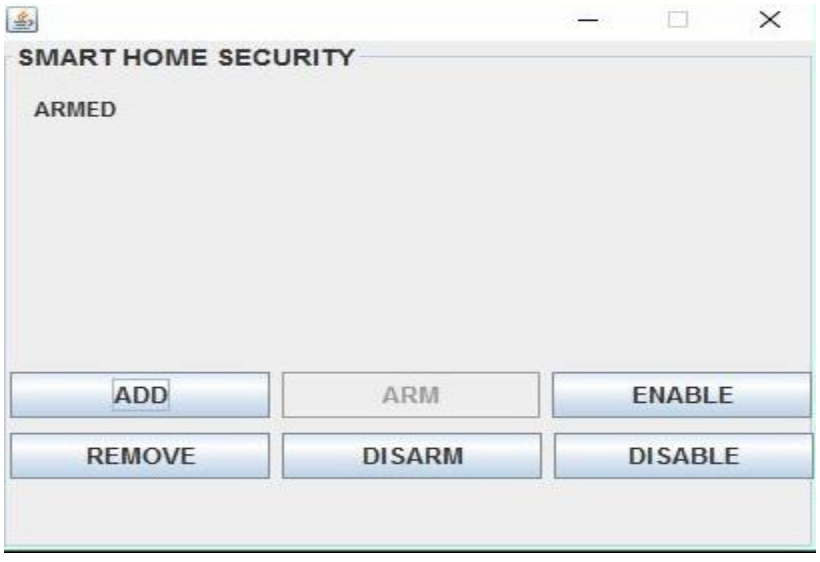

Remove Device
DEVICE NAME
DEVICE ID
Door 111
b8-27-eb-35-ea-95
REMOVE
CANCEL

The following will show that the device has been removed:




**4.4. Arm System**

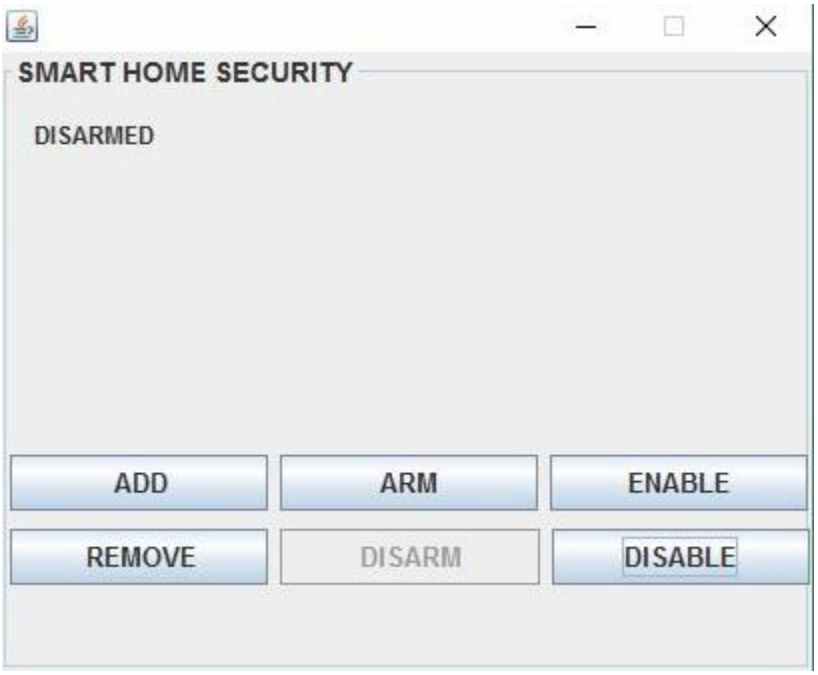
The Arm button is to Arm the entire system with a single click. There should be at least one device in the system if the user wish to Arm the system. When the system is Armed, the status will be shown on the top left of the HomePage.




.

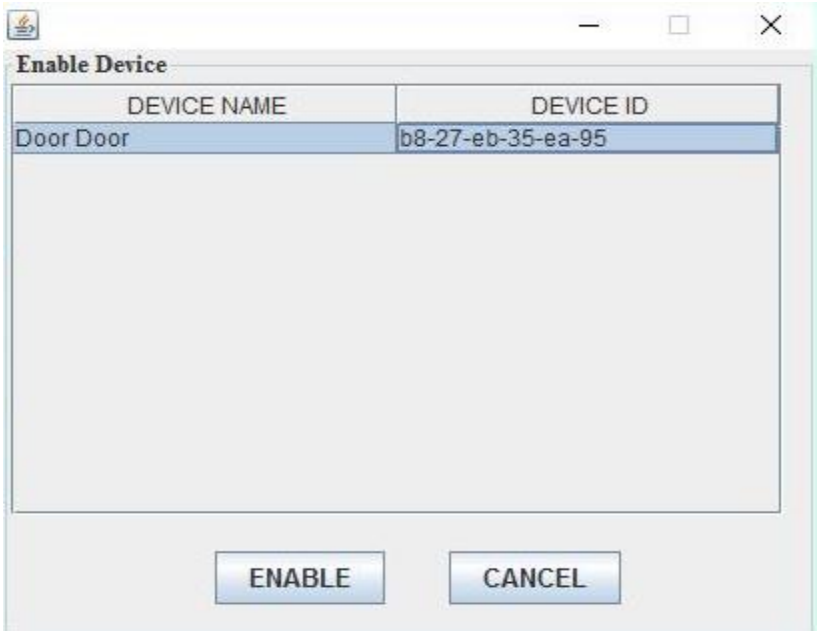
### 4.5. Disarm System

The Disarm button is to Disarm the entire system with a single click. There should be at least one device in the system if the user wish to Disarm the system. When the system is Drmed, the status will be shown on the top left of the HomePage.

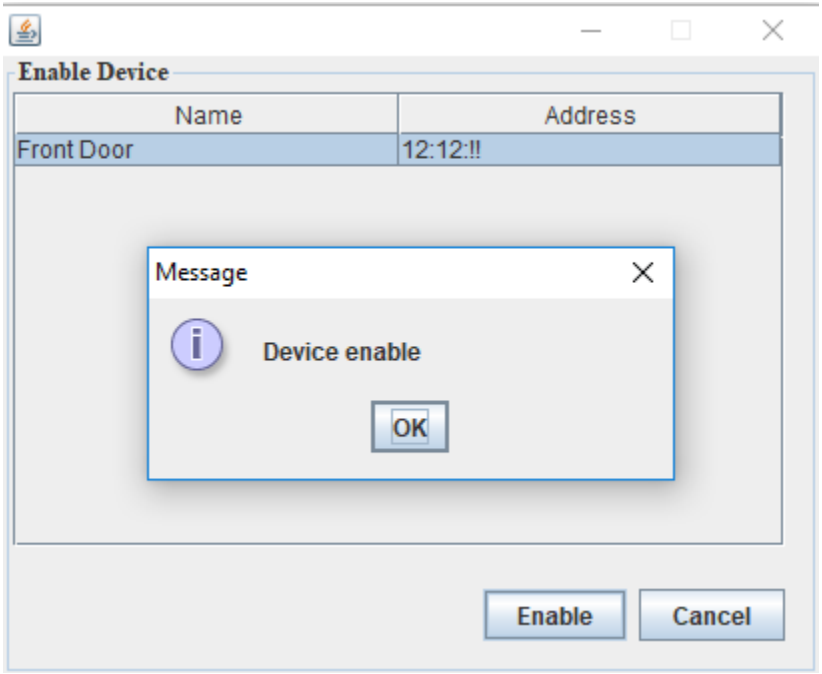



### 4.6. Enable System

The Enable panel will show all the Disabled devices. The user will have to select the device that needed to be enabled.

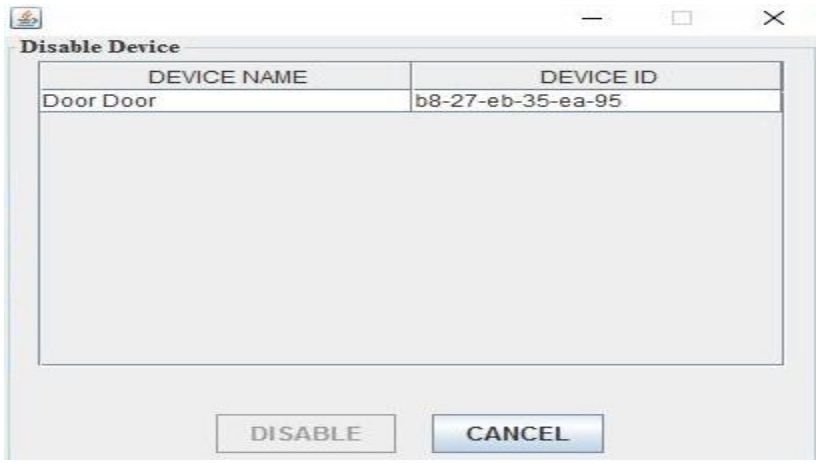



The Following screen will be shown after the device is set to enable:



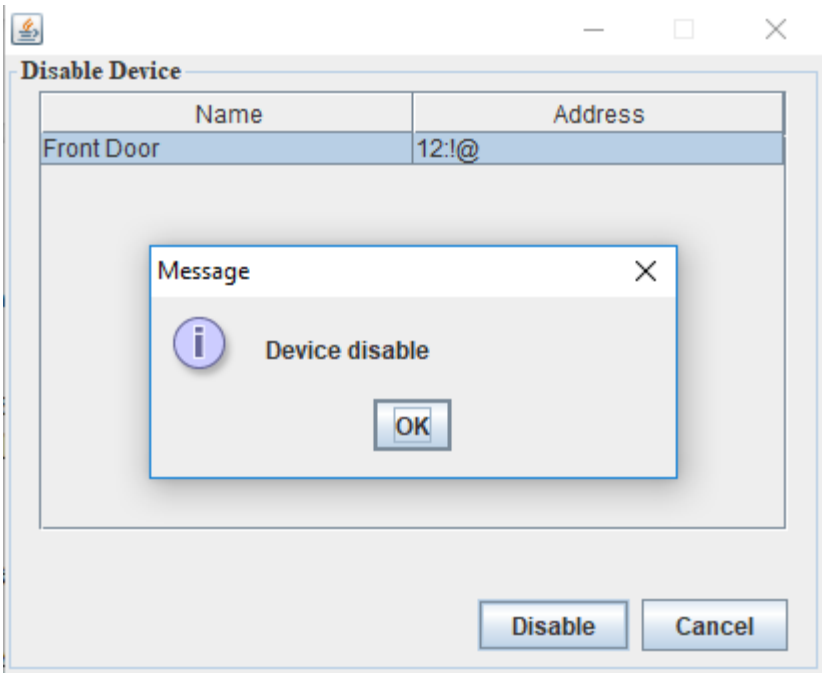

### 4.7. Disable System

The Disable panel will show all the Enabled devices. The user will have to select the device that needed to be disabled.



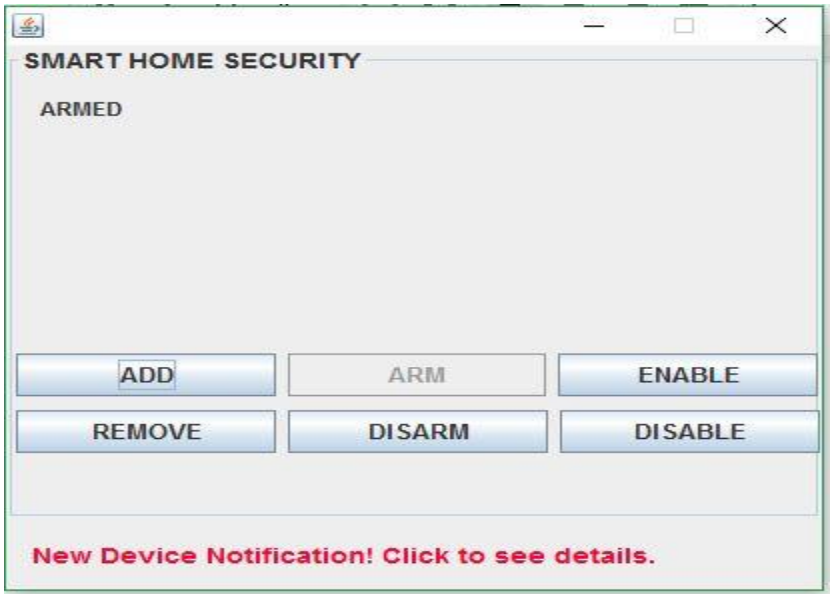


The following screen shows that the device has been Disabled:

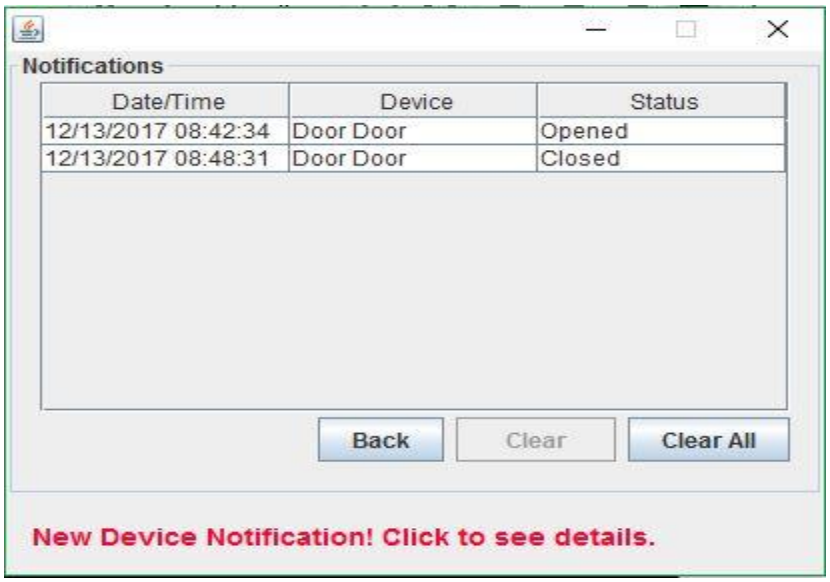



### 4.8. Notification Panel

If there is a notification on the system, an alert message on the Home Screen will appear stating that “New Device Notification! Click to see details.”.

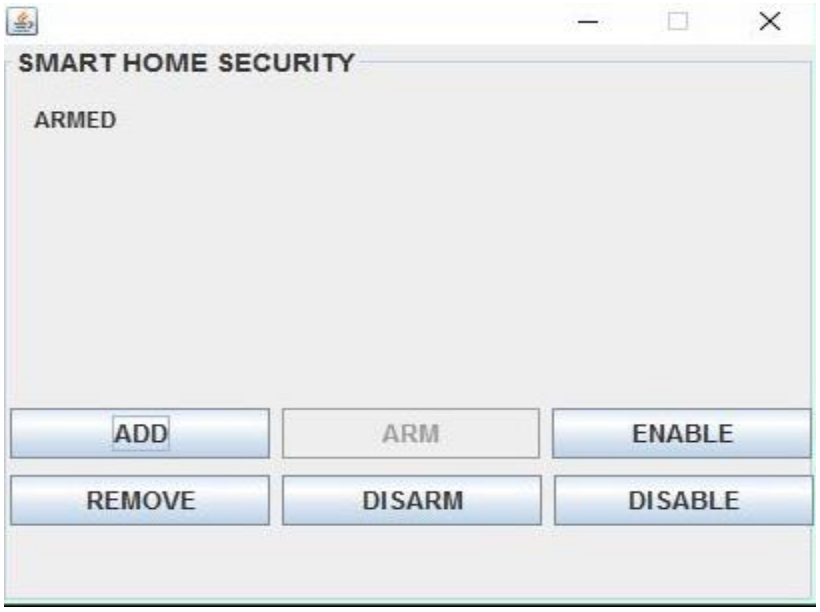



The user will have to click on the alert message to see which devices are creating the alert.




The User can select a single alert to Clear it or can clear all the alerts by clicking on Clear All button, If Clicked on Clear All button, All the alerts will be deleted and the system will take the user to the Homepage.

**HomePage**



# 5. System Integration and Validation

The system was tested 5 times at Weisberg Building Computer lab room 3142, and then it is handed to the mentor and faculty member for user testing. It is very important for every system to have testing and improve the system by the feedbacks.

## 5.1. General Assumption

- In this project we have assumed that the user has a laptop and all the required devices are installed in Home or Apartment.
- The user is assumed to have knowledge regarding the system with the help from the provided user manual.
- The user must have all the software programs installed on their laptop where the system is running.
- The user must be comfortable in making changes in the system from their laptop rather than a mobile phone.

# 6. Conclusion

This project is to provide a Smart Home Security system that safeguards doors and windows of a house, apartment or a building. This system is a DIY (Do It Yourself) with a one click feature. The reed switch (sensor) will generate alert when an event occurs (door or window open/close) and sends a notification to the central hub (UI) running on laptop. A user can add or remove microcontroller from the central hub. The project constraints will allow him to add several new devices with just connecting the microcontrollers to the hub and copying an existing code from the available source. The user can clear a single notification or can clear all the notifications. The notifications will be displayed with date and time. There is no code on the user side.

## 7. REFERENCES

https://thepi.io/how-to-install-noobs-on-the-raspberry-pi/
https://www.raspberrypi.org/downloads/noobs/
https://www.raspberrypi.org/documentation/
https://www.modmypi.com/blog/how-to-give-your-raspberry-pi-a-static-ip-address-update