KANTIPUR ENGINEERING COLLEGE

(Affiliated to Tribhuvan University)

Dhapakhel, Lalitpur

[Subject Code: CT654]

A MIDTERM DEFENSE PROJECT REPORT ON

"HOME AUTOMATION SYSTEM WITH RPi"

Submitted by

Abhishek Bhattarai [2072/BCT/90]

Ankit Kharel [2072/BCT/95]

Kritish Dhaubanjar [2072/BCT/102]

Pradyumna Subedi [2072/BCT/110]

Submitted to

Department of Computer and Electronics Engineering

July, 2018

HOME AUTOMATION SYSTEM WITH RPi

Submitted by

Abhishek Bhattarai [2072/BCT/90]

Ankit Kharel [2072/BCT/95]

Kritish Dhaubanjar [2072/BCT/102]

Pradyumna Subedi [2072/BCT/110]

Submitted to

Department of Computer and Electronics Engineering

Kantipur Engineering College

Dhapakhel, Lalitpur

July, 2018

ABSTRACT

Home Automation System (HAS) is a system that uses computers to control basic home functions and features automatically through local network, an automated home is sometimes called a smart home. It is meant to save the electric power and human energy. In this paper we present the working progress of our Home Automation System (HAS) using Raspberry Pi 3 that will employ the integration of networking, wireless communication, to provide the user with remote control of various lights, fans, and appliances within their home and storing the data in the database. The system will automatically change on the basis of user's control and synchronize in the system. This system will be designed, to be of low cost and expandable allowing a variety of devices to be controlled. The switches status is synchronized in all the control system whereby every user interface will indicate the real time existing switches status. The system will be intended to control electrical appliances and devices in house with relatively low cost design, user-friendly interface and ease of installation.

Key Words:

Home Automation, **Raspberry Pi 3**, **Server Client**.

TABLE OF CONTENTS

LIST OF FIGURES

FIGURE **PAGE**

LIST OF TABLES

TABLE **PAGE**

CHAPTER 1: INTRODUCTION

1.1 Background

With advancement of Automation technology, life is getting simpler and easier in all aspects. In today's world Automatic systems are being preferred over manual system. With the rapid increase in the number of users over a network over the past decade has made technology a part and parcel of life. Automation of things is a growing network of everyday object - from industrial machine to consumer goods that can share information and complete tasks while you are busy with other activities.

Home automation gives you access to control devices in your home from a mobile device anywhere in the house. The term may be used for isolated programmable devices, like thermostats and sprinkler systems, but home automation more accurately describes homes in which nearly everything -- lights, appliances, electrical outlets, heating and cooling systems -- are hooked up to a remotely controllable network. From a home security perspective, this also includes all of the doors, windows, locks and any other sensors that are linked to it.

The first and most obvious beneficiaries of this approach are devices and appliances that can be connected to a local area network, via Raspberry Pi. However, electrical systems and even individual points, like light switches and electrical outlets, can also be integrated into home automation networks.

Automation is, unsurprisingly, one of the two main characteristics of home automation. Automation refers to the ability to program and schedule events for the devices on the network. The programming may include time-related commands, such as having your lights turn on or off at specific times each day. It can also include non-scheduled events, such as turning on all the lights in your home when your security system alarm is triggered.

1.2 Problem Statement

Until fairly recently, automated central control of building-wide systems was found only in larger commercial buildings and expensive homes. Typically involving only lighting, building automation rarely provided more than basic control, monitoring and scheduling functions and was accessible only from specific control points within the building itself.

Home automation is a step forward, in which everyone has an assigned IP address, and can monitor and access the system locally. As people constantly run over places, working to accomplish everything on our never-ending "to-do" list. With Home Automation System, we will never have to worry about closing the door or switching off the appliances and so on. In short, we can save precious time and experience for more daily productivity.

1.3 Objectives

Application controlled Home Automation System should be able to control the home appliances wirelessly effectively and efficiently.

I. To develop a system, that can integrate most home appliances into network. And make it flexible for users to control appliances from Wi-Fi network.
II. To enable real time capability in the system, to ensure validity and consistency.

1.4 Applications

The project aims at designing a prototype for controlling the home appliances that can be controlled wirelessly via an application. The system can be used in wide range of areas. The system integrated with different features can be applied in the following fields.

I. The system can be used in home, small offices to the big malls.
II. Reduction in the energy usage by configuring the system to adjust and turn off some or all the appliances with automation.
III. Scripts and processes that require low cognition and with repetitive processes are ripe for the choosing for automation.

1.5 Project Features

The ability to manage your home's electronic systems from one main control system can make your household run smoother, feel better, and save energy.

I. Ability to tie diverse electronic devices together so they can perform as one unified system.
II. Save energy by turning off electronic devices automatically.
III. Easily expanded vertically, to incorporate additional products, and horizontally to support additional rooms.
IV. Able to change the settings quickly and easily if your plans change.
V. Local access capabilities allow you to monitor your home's environment and alter the settings of the lights, and other gear, all from your computer.

1.6 Feasibility Analysis

The project can be implemented using affordable electronic and software technology making it economically, technically and operationally feasible.

1.6.1 Economic Feasibility

This project will be based on a Raspberry Pi 3 microcomputer and few electronic components like relay switches, door sensors etc., which are affordable, making it economically feasible to implement.

1.6.2 Technical Feasibility

This project is based on embedded system and wireless and wired technology which are reasonably in phase with currently used technology. Therefore, it is very much favored by the technology.

1.6.3 Operational Feasibility

The application will have a very easy to use, with a user friendly interface so it will be pretty much operable by anyone having little experience of using other devices. It could even be helpful for physically disabled person as well, for controlling home appliances with the click of a button. So it is operationally feasible.

1.7 System Requirement

1.7.1 Hardware Requirement

I. Raspberry Pi 3 as the controller for its processing power and large developer community.

II. Relays to connect electrical appliances to low voltage control of Raspberry Pi.

III. GPIO pins are connected to transistor, which are used as a switch.

1.7.2 Software Requirement

I. Raspbian OS

- Linux based OS for Raspberry Pi.

II. RPI-GPIO library

- GPIO interface library for the Raspberry Pi.

III. pi4j library

- object-oriented I/O API and implementation libraries for Java to access the full I/O capabilities of the Raspberry Pi platform.

IV. MySQL Server

- database management tool.

V. Java Virtual Machine and Java Runtime Environment

VI. Windows Vista or higher, OSX or Linux

CHAPTER 2: LITERATURE REVIEW

Ahmed Elshafee and Karim AlaaHamed, "Design and Implementation of a Wi-Fi based Home Automation System" - International Journal of Computer, Electrical Automation, Control and Information Engineering, is Arduino based distributed HAS system which consists of a server, hardware and interface modules. Server controls hardware, one interface module and can be easily configured to handle more hardware interface module. The hardware interface module controls its alarms and actuators. The web server software is developed using ASP.net technology. They have used Wi-Fi as a medium for internet connectivity. [1]

In this setup mode user, can create a basic macro involving simple triggers and to customize the macros to perform complex series of events. Macros can be activated manually or as a reaction for certain trigger light motion sensors and surveillance cameras.

Since they have used Arduino as the base of the system while adding more hardware interface modules we require number of Arduino chipset to be installed in the system which results in the increases of complexity and the overall cost of the system and also eventually increase the power consumption of the entire system.

Shaiju paul, Ashlin Anthony and Aswathy B, "Android based Home Automation Using Raspberry Pi", IJCAT - International Journal of Computing and Technology, is very close to the paper we are introducing. They have created Home the HAS wherein they have simply connected the home appliances to the Raspberry Pi using relay circuit and they have used an android application for a user interface. [2]. Their system consists of mainly three components a Wi-Fi module, Raspberry Pi board, Relay circuit. Wi-Fi is used as a communication channel between android phone and the Raspberry Pi board. They have hidden the complexity of the notions involved in the HAS by including them in a simple, but comprehensive set of related concepts.

In this paper, they have not used sensors due to which the project doesn't provide the base for an automated system. Only switching on and off the appliances is just not enough for reducing the power consumption of the system.

Paper Title	Authors	Year of publishing	Technology Used	Remarks
Design and Implementation of a Wi-Fi based Home Automation System	Ahmed Elshafee, Karim AlaaHamed	2012	Arduino board, ASP.net, Wi-Fi.	Complexity increases with the expansion of system.
Android based Home Automation Using Raspberry Pi	Shaijupaul, Ashlin Anthony, Aswathy B	2014	Android, Raspberry Pi, Wi-Fi.	Doesn’t provide sufficient base for an automated system.

Fig 2.1 : Research Paper Analysis Table

CHAPTER 3: METHODOLOGY

3.1 System Development

The section of the report explains about what are we going to do and how are we going to do it. We will be following incremental development mode for the completion of this project which is described below briefly:

3.1.1 Incremental development model

Incremental development is based on the idea of developing an initial implementation, exposing this to user comment and evolving through several verisons until a complete system has been developed. It involves both development and maintenance. It interleaves the activity of specification, development and validation. Development as a series of version i.e. increments with each version adding functionalities to the previous one. This model combines the elements of the waterfall model with the iterative philosophy of prototyping.

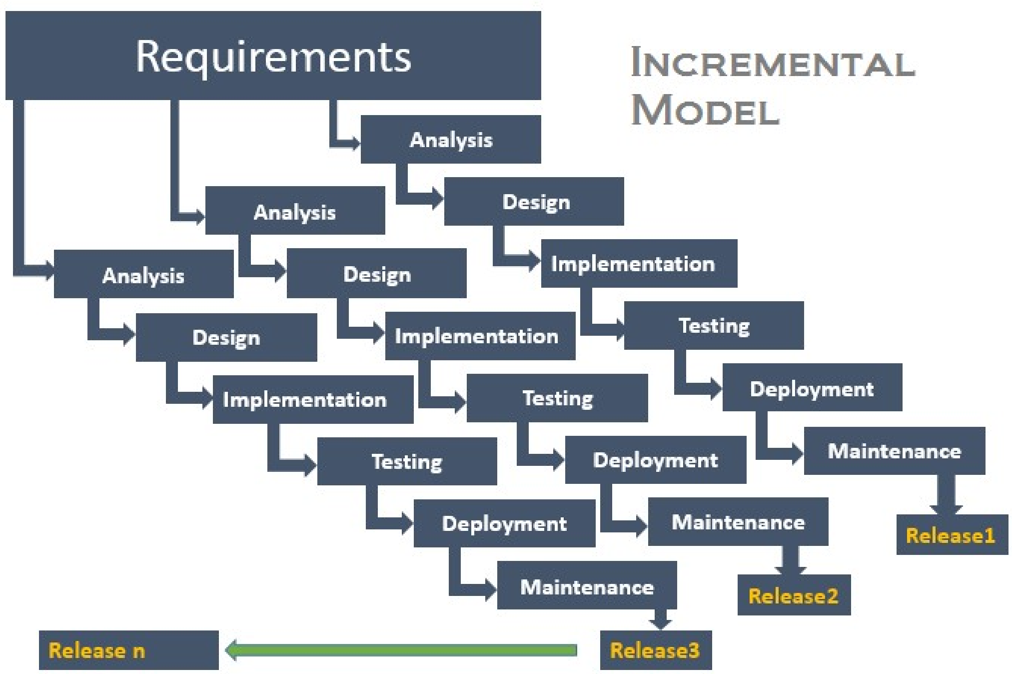


Fig 3.1: Incremental Development Model *Image downloaded from*: http://testingfreak.com/incremental-model-software-testing-advantages-disadvantages-incremental-model/

We will closely follow this model of software development in our proposed project. If the requirements change, new feature will be added to enhance the functionality as a whole, we will integrate them as they come up.

3.2 Research Phase

3.2.1 Hardware Recognition

The Raspberry Pi is a low cost, **credit-card sized computer** that plugs into a computer monitor or TV, and uses a standard keyboard and mouse. It is a capable little device that enables people of all ages to explore computing, and to learn how to program in languages like Scratch and Python. It's capable of doing everything you'd expect a desktop computer to do, from browsing the internet and playing high-definition video, to making spreadsheets, word-processing, and playing games.

One powerful feature of the Raspberry Pi is the row of GPIO (general purpose input/output) pins along the edge of the board.

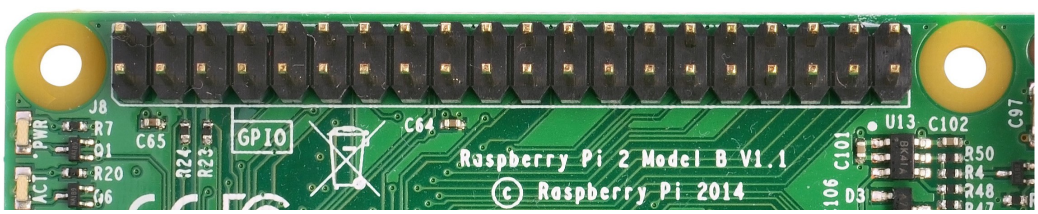


Fig: Raspberry Pi GPIO Pins. *Image downloaded from*
https://www.raspberrypi.org/documentation/usage/gpio-plus-and-raspi2/images/

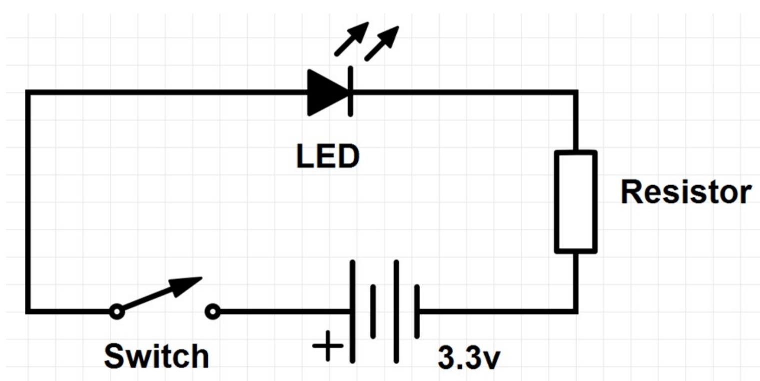


Fig: For Working of GPIO . *Image downloaded from*
https://www.raspberrypi.org/documentation/usage/gpio/

When we use a GPIO pin as an output, the Raspberry Pi replaces **both the switch and the battery** in the above diagram. Each pin can turn on or off, or go HIGH or LOW in computing terms. When the pin is HIGH it outputs 3.3 volts (3v3); when the pin is LOW it is off.

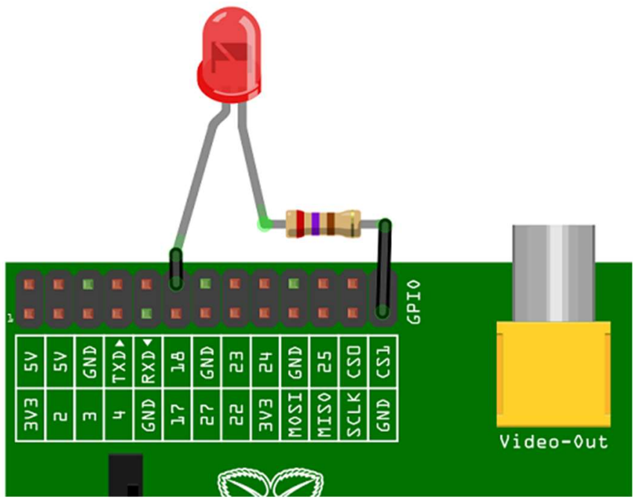
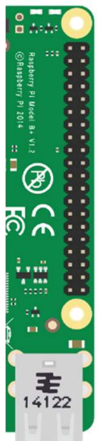


Fig: LED connection over GPIO pins. *Image downloaded from* *https://www.raspberrypi.org/documentation/usage/gpio/*

Here's the same circuit using the Raspberry Pi. The LED is connected to a GPIO pin (which can output +3v3) and a ground pin (which is 0v and acts like the negative terminal of the battery):

General purpose input-output (GPIO) connector

(Source:https://en.wikipedia.org/wiki/Raspberry_Pi)



Peripherals	GPIO	Particle	Pin #			Pin #	Particle	GPIO	Peripherals
	3.3V		1	X	X	2		5V	
I2C	GPIO2	SDA	3	X	X	4		5V	
	GPIO3	SCL	5	X	X	6		GND	
Digital I/O	GPIO4	D0	7	X	X	8	TX	GPIO14	UART
	GND		9	X	X	10	RX	GPIO15	Serial 1
Digital I/O	GPIO17	D1	11	X	X	12	D9/A0	GPIO18	PWM 1
Digital I/O	GPIO27	D2	13	X	X	14		GND	
Digital I/O	GPIO22	D3	15	X	X	16	D10/A1	GPIO23	Digital I/O
	3.3V		17	X	X	18	D11/A2	GPIO24	Digital I/O
SPI	GPIO10	MOSI	19	X	X	20		GND	
	GPIO9	MISO	21	X	X	22	D12/A3	GPIO25	Digital I/O
	GPIO11	SCK	23	X	X	24	CE0	GPIO8	SPI (chip enable)
	GND		25	X	X	26	CE1	GPIO7	
DO NOT USE	ID_SD	DO NOT USE	27	X	X	28	DO NOT USE	ID_SC	DO NOT USE
Digital I/O	GPIO5	D4	29	X	X	30		GND	
Digital I/O	GPIO6	D5	31	X	X	32	D13/A4	GPIO12	Digital I/O
PWM 2	GPIO13	D6	33	X	X	34		GND	
PWM 2	GPIO19	D7	35	X	X	36	D14/A5	GPIO16	PWM 1
Digital I/O	GPIO26	D8	37	X	X	38	D15/A6	GPIO20	Digital I/O
	GND		39	X	X	40	D16/A7	GPIO21	Digital I/O

Fig : Raspberry Pi 3 GPIO Pinout Diagram

3.2.2 Software Recognition

There are two major pieces of software in this project. First one is the application, which will be the front end and next is the server side software running on the Raspberry Pi at the backend.

3.2.2.1 Raspbian OS

Raspbian is a Debian-based computer operating system for Raspberry Pi. There are several versions of Raspbian including Raspbian Stretch and Raspbian Jessie, as the primary operating system for the family of Raspberry Pi single-board computers.

3.2.2.2 Java

Java is a general-purpose computer-programming language that is concurrent, class-based, object-oriented, and specifically designed to have as few implementation dependencies as possible. It is intended to let application developers "write once, run anywhere" (WORA), meaning that compiled Java code can run on all platforms that support Java without the need for recompilation. Java applications are typically compiled to bytecode that can run on any Java virtual machine (JVM) regardless of computer architecture.

3.2.2.3 JavaFX

JavaFX is the next step in the evolution of Java as a rich client platform. It is designed to provide a lightweight, hardware-accelerated Java UI platform for enterprise business applications. With JavaFX, developers can preserve existing investments by reusing Java libraries in their applications. They can even access native system capabilities, or seamlessly connect to server-based middleware applications.

3.2.2.4 pi4j

pi4j intended to provide a friendly object-oriented I/O API and implementation libraries for Java Programmers to access the full I/O capabilities of the Raspberry Pi platform. This project abstracts the low-level native integration and interrupt monitoring to enable Java programmers to focus on implementing their application business logic.

3.3 Design Phase

The user's basic requirements analyzed, referenced to the previous report, for better understanding of what was required of the system was thought well of and implemented with different prototypes, and ways of implementing these requirements were further discussed. Physical modules of the system were designed and identifying of the operating environment in which they were to work on. The server and client systems were written in Java and allowed the basic objectives of the system to be fulfilled partially as proposed.

System design involved transforming the software requirements into an architecture that described its top-level structure and identified the software components and developed detailed design for each software components. For each requirement, a set of one or more design elements was produced and implemented.

3.3.1 Server Design

The server side terminal application was designed to accept a client connection over a local network and listen for a message from the client to control the GPIO as to toggle the status Logic level of GPIO Pins. The Server application, would run from a

Raspberry Pi, with hostname of "**raspberrypi.local**" which eliminates the necessity of IP configurations.

3.3.2 Client Design

The client side application is a JavaFX application, which connects to the Raspberry Pi Server, and issues the request message to control the status through Remote Method Invocation and UDP.

3.4 Use Case Diagram

Fig 3.4: Use Case Diagram

In our system, user acts as a primary actor who can read status of appliances either in ''ON'' state or ''OFF'' state through the database. Further, the user may change the state of appliances and control them according to the need by sending the status signals to the database and raspberry pi gets access to the database and send control signals to the appliances.

3.5 Class Diagram

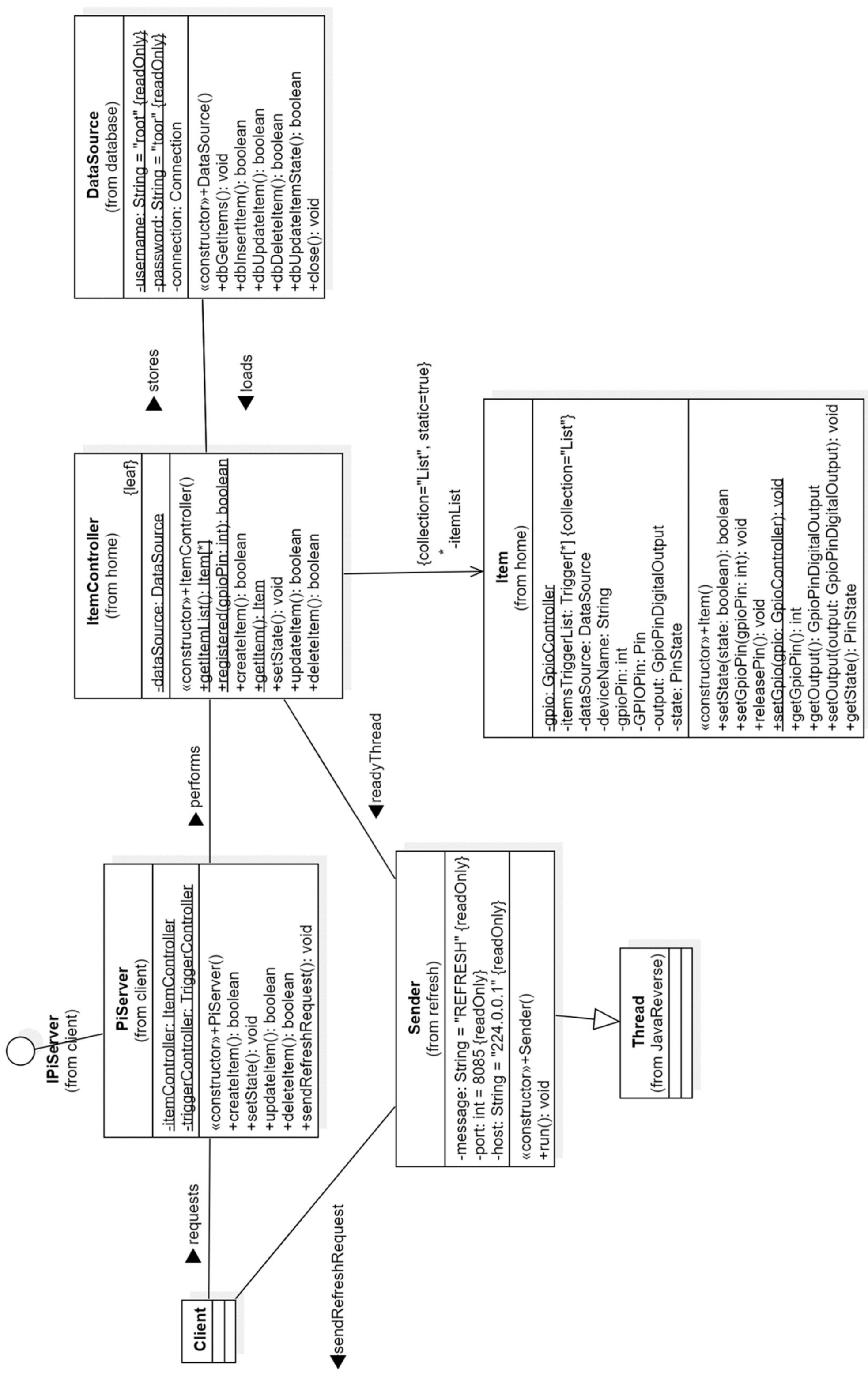


Fig 3.5: Class Diagram

3.6 Sequence Diagram

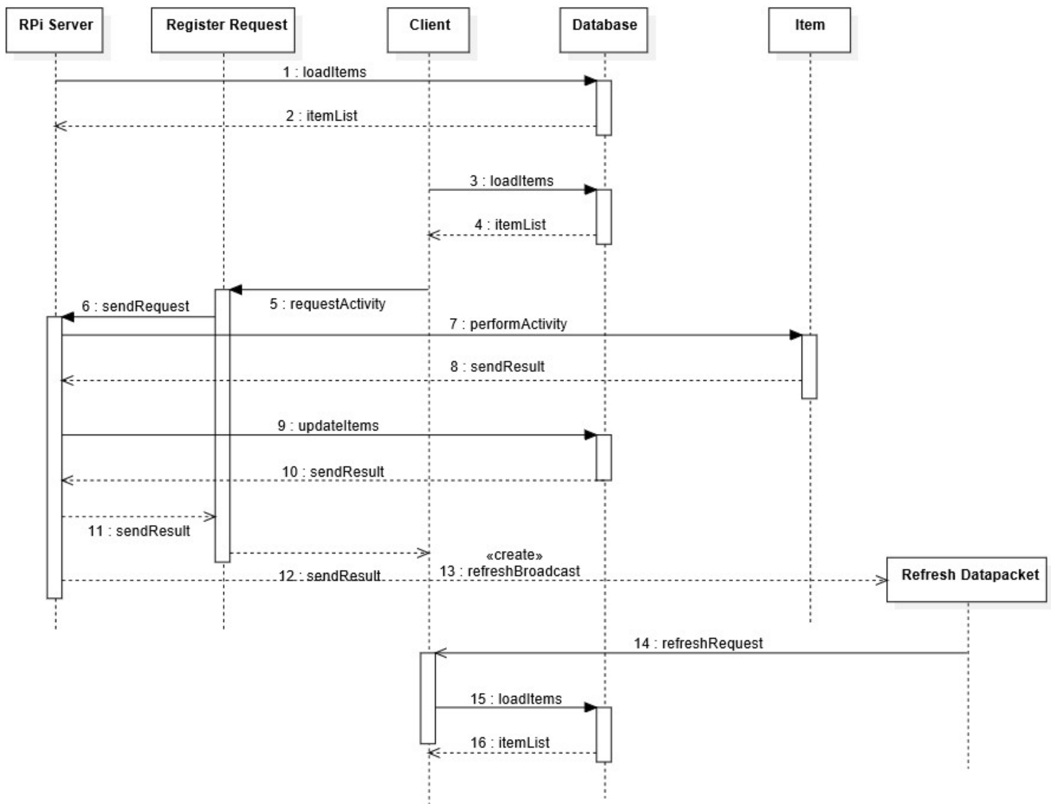


Fig 3.6: Sequence Diagram

CHAPTER 4: EPILOGUE

4.1 Work Completed

A basic client and server network was designed to issue and handle the requests from the multiple client to the server, for controlling the status of logic of the GPIO pins and further the relay. The works accomplished are listed below:

- The server side application can accept a client and handle requests from **multiple** clients.
- Database for synchronization of client actions, GPIO status and the Server and all the clients was designed.
- The client side application can connect and issue the request action to server.
- A dynamic host was setup to eliminate the necessity for manual IP configuration.
- UDP was used as a layer to ensure real time capability.
- Threads were optimized to ensure performance enhancements.
- The configuration and existing system is entirely dynamic and the option for dynamic controls is available.
- Automation through trigger was implemented, in addition to Client Control, that could chain the action over a single item, over multiple responses.

4.2 Work Remaining

Prototypes for upcoming tasks such as scheduling, user access control, are still under design and consideration phase. However, the research and implementation for the features as defined in the objectives is ongoing and will be accomplished in the defined time.

4.3 Work Schedule

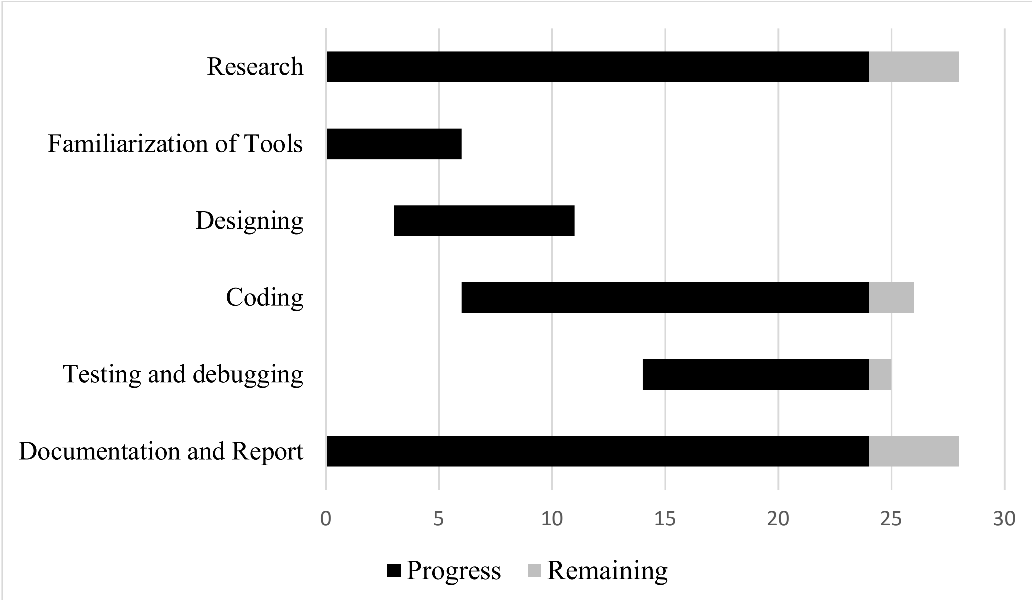
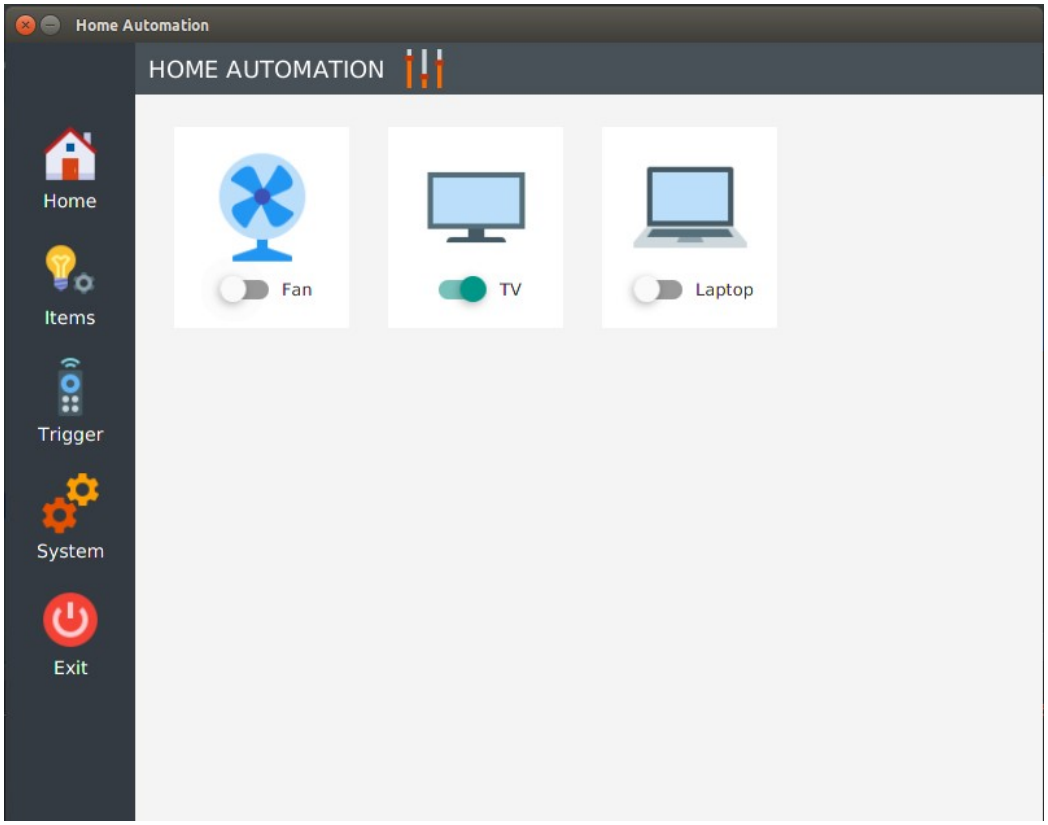


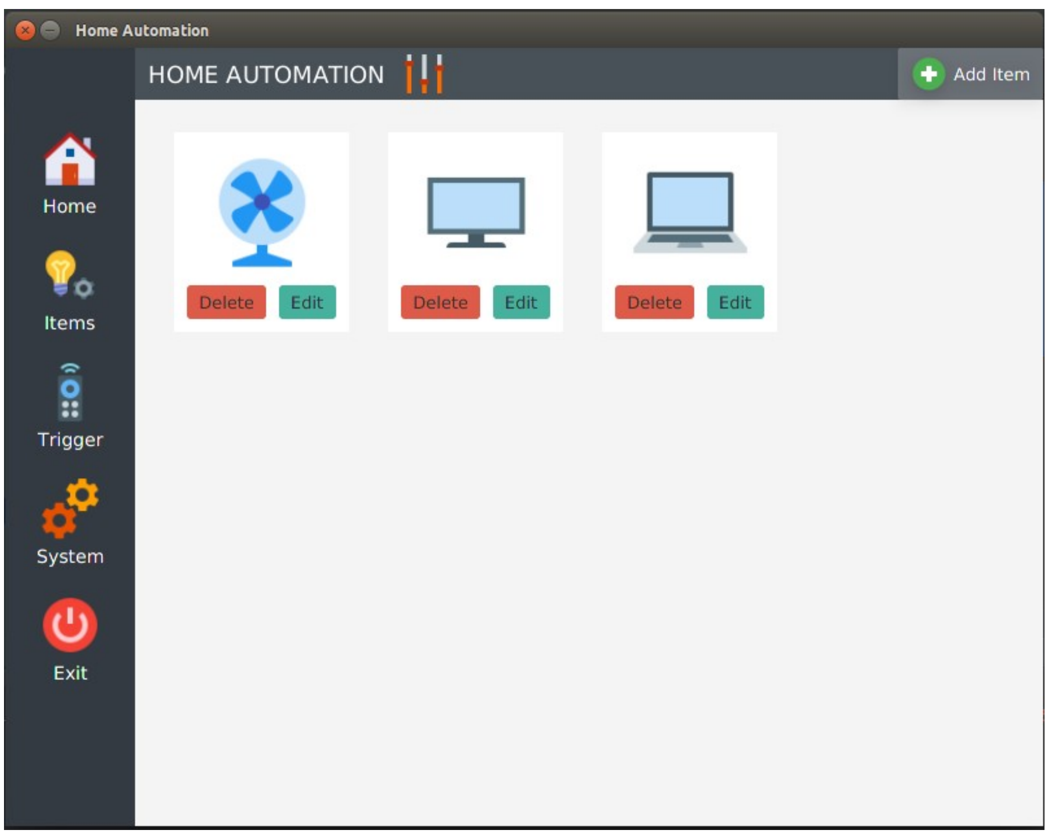
Fig 4.3 : Gantt Chart

The estimated time period of this project is 28 weeks. The work is divided into several phases as shown in Gantt chart. We will be continuously involved in the proper documentation and report preparation throughout the whole project period. Firstly, we researched and got familiarized with the tools which we used during later phases. Then the design and planning of software and hardware part followed. After that we started coding for an application and side by side fabrication of relay board and related hardware. Testing and debugging is performed along the way.

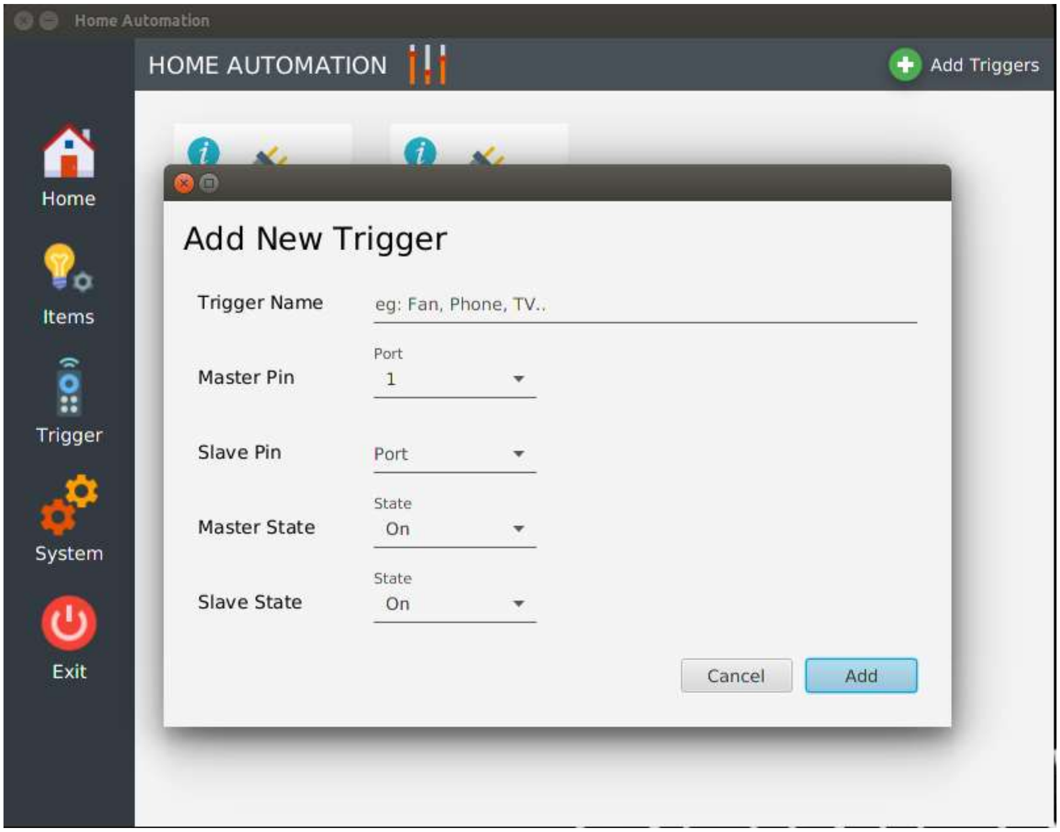
4.4 Output

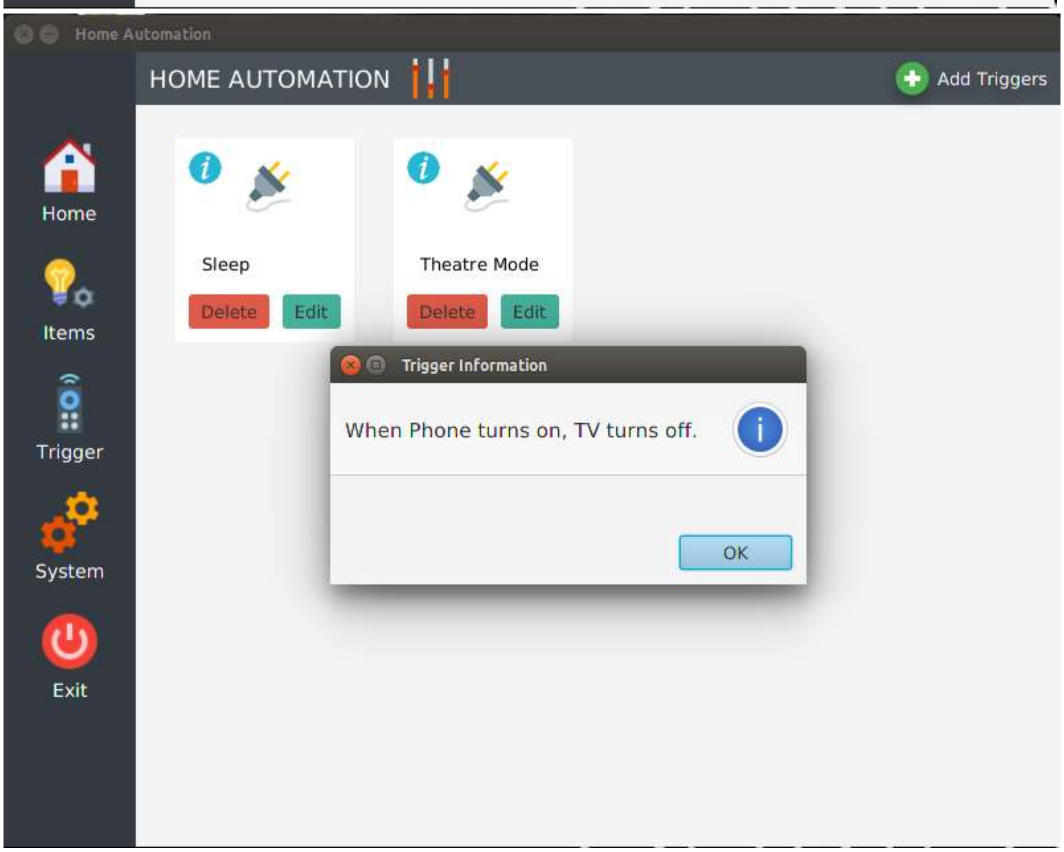


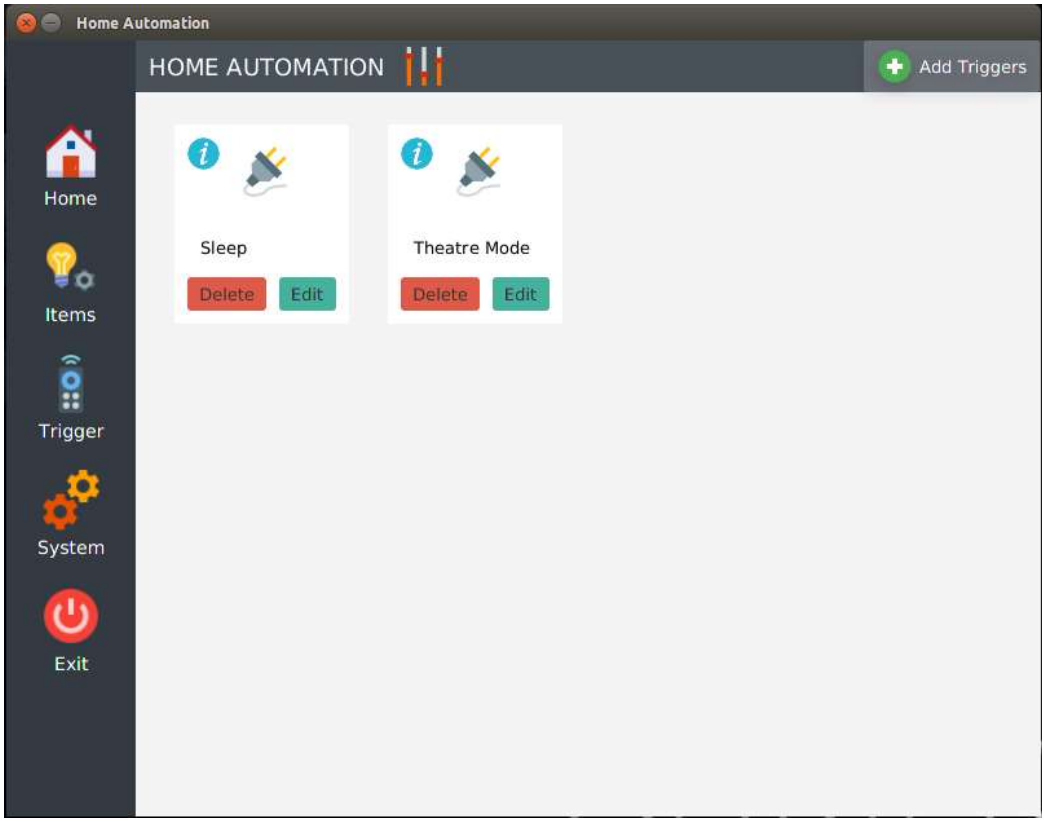
Home Automation
HOME AUTOMATION
Home
Items
Trigger
System
Exit
Fan
TV
Laptop



Home Automation
HOME AUTOMATION
Add Item
Home
Items
Trigger
System
Exit
Delete
Edit
Delete
Edit
Delete
Edit


Home Automation
HOME AUTOMATION
Add Triggers
Home
Items
Trigger
System
Exit
Add New Trigger
Trigger Name
eg: Fan, Phone, TV..
Master Pin
Port
1
Slave Pin
Port
Master State
State
On
Slave State
State
On
Cancel
Add


Home Automation
HOME AUTOMATION
Add Triggers
Home
Items
Trigger
System
Exit
Sleep
Delete
Edit
Theatre Mode
Delete
Edit
Trigger Information
When Phone turns on, TV turns off.
OK


Home Automation
HOME AUTOMATION
Add Triggers
Home
Items
Trigger
System
Exit
Sleep
Delete
Edit
Theatre Mode
Delete
Edit

REFERENCES

[1] Ahmed Elshafee, Karim AlaaHamed, "Design and Implementation of a Wi-Fi based Home Automation System”, *International Journal of Computer, Electrical Automation,Control and Information Engineering*, Vol:6,No:8,2012,pp 1074 - 1080.

[2] Shaijupaul, Ashlin Anthony, Aswathy B, "Android based Home Automation Using Raspberry Pi”, *IJCAT - International Journal of Computing and Technology,* Volume 1,Issue 1, February 2014, pp 143 - 147.

[3] "Raspberry Pi - Teach, Learn, and Make with Raspberry Pi", *Raspberry Pi*, 2017. Available: http://www.raspberrypi.org/.

[4] "The Pi4J Project - Home", *Pi4j.com*, 2017. Available: http://pi4j.com/.

[5] "RPi Hub - eLinux.org", *Elinux.org*, 2017. Available: https://elinux.org/RPi_Hub .