DESIGN AND IMPLEMENTATION OF HOME AUTOMATION SYSTEM
USING RASPBERRY PI

A Project
Presented to the
Faculty of
California State Polytechnic University, Pomona

In Partial Fulfilment
of the Requirements for the Degree
Master of Science
in
Computer Science

By
Irwin Soni
2018
SIGNATURE PAGE

PROJECT: DESIGN AND IMPLEMENTATION OF HOME AUTOMATION SYSTEM USING RASPBERRY PI

AUTHOR: Irwin Soni

DATE SUBMITTED: Spring 2018

Computer Science Department

Dr. Yu Sun
Project Committee Chair
Computer Science

Dr. Sampath Jayarathna
Computer Science
ACKNOWLEDGEMENT

First and foremost, I would like to thank God for blessing me with such amazing people who have been there to support me in all that I have achieved. To my parents, who have filled my life with immense love, happiness and shaping me into the person I have become today. I would like to thank Dr. Yu Sun, my project advisor, for his selfless support and enlightening guidance. Working with Dr. Sun was such a valuable experience of learning computer science and life at the same time. I would also like to thank Dr. Sampath Jayarathna for his review, suggestions and encouragement. I would also like to thank my classmates for their companionship and friendship.
ABSTRACT

Home automation system achieved great popularity in the last decades as it increases the comfort and quality of life. Smartphone applications are used to control and monitor the home appliances using different types of communication techniques. As mobile devices continue to grow in popularity and functionality, the demand for advanced ubiquitous mobile applications in our daily lives also increases. The paper deals with the design and implementation of a flexible and low-cost Home Automation System for various mobile devices that leverages mobile technology to provide essential functionalities to our homes and associated control operations. In particular, with the help of mobile devices, the device manages operations on home appliances, such as turning ON/OFF a television or microwave or altering the intensity of lighting around the house.

The device has three ways of communicating with home appliances. First is through Apple Home app, which runs on iOS 10 or later, and lets you securely control smart home devices or any home appliance as long as the device is added in the Apple HomeKit as an accessory. The second way is through the means of an Android app. And lastly, the application is not only limited to smartphones but also can be used by feature phones through web browser.
# TABLE OF CONTENTS

SIGNATURE PAGE............................................................................................................. ii

ACKNOWLEDGEMENT........................................................................................................ iii

ABSTRACT............................................................................................................................ iv

LIST OF FIGURES................................................................................................................ vii

CHAPTER 1: INTRODUCTION............................................................................................ 1

1.1 INTERNET OF THINGS................................................................................................. 1
1.2 HOME AUTOMATION SYSTEMS................................................................................... 2

CHAPTER 2: RELATED WORK.......................................................................................... 7

CHAPTER 3: MOTIVATING EXAMPLES............................................................................. 10

CHAPTER 4: CHALLENGES............................................................................................... 13

CHAPTER 5: SOLUTION IMPLEMENTATION..................................................................... 16

5.1 ACCESSING THE DEVICE THROUGH BROWSER:..................................................... 16

5.1.1 Hardware Implementation:...................................................................................... 16

5.1.2 Software Implementation:....................................................................................... 17

5.2 ACCESSING THE DEVICE THROUGH iOS HOME APP:........................................ 20

5.2.1 Hardware Implementation:...................................................................................... 21

5.2.2 Software Implementation:....................................................................................... 21

5.3 ACCESSING THE DEVICE THROUGH ANDROID APP NAMED, HOMEPI:.......... 24

5.3.1 Hardware Implementation:...................................................................................... 24

5.3.2 Software Implementation:....................................................................................... 24
CHAPTER 6 : EMPIRICAL RESULTS ................................................................. 26

CHAPTER 7 : CONCLUSION AND FUTURE WORK ........................................ 28

REFERENCES ........................................................................................................... 29
LIST OF FIGURES

Figure 1.1 - M2M device connections and future predictions .............................................. 1
Figure 1.2 - The System Setup ............................................................................................ 5
Figure 5.1 - Hardware setup ............................................................................................. 16
Figure 5.2 - Raspberry Pi Pinout Mapping Guide ............................................................ 17
Figure 5.3 - The front view of PHP file showing all buttons .............................................. 19
Figure 5.4 - Flow chart showing hardware setup ............................................................. 22
Figure 5.5 - Screenshot of Home app with String Lights as an accessory ....................... 23
Figure 5.6 – (a),(b),(c) Screenshots of HomePi app ..................................................... 25
1.1 Internet of Things

The Internet of Things (IoTs) can be described as connecting everyday objects like smartphones, Internet TVs, sensors and actuators to the Internet where the devices are intelligently linked together enabling new forms of communication between things and people, and between things themselves. Building IoTs has advanced significantly in the last couple of years since it has added a new dimension to the world of information and communication technologies. According to [1], it is expected that the number of devices connected to the Internet will accumulate from 100.4 million in 2011 to 2.1 billion by the year 2021, growing at a rate of 36% per year. In the year 2011, 80% machine to machine (M2M) connections were made over mobile networks such as 2G and 3G and it is predicted that by 2021, this ratio will increase to 93% since the cost related with M2M over mobile networks are generally cheaper than fixed networks as illustrated in Figure 1.1 [1].

![Figure 1.1 - M2M device connections and future predictions](image-url)
Now anyone, from anytime and anywhere can have connectivity for anything and it is expected that these connections will extend and create an entirely advanced dynamic network of IoTs. The development of the Internet of Things will revolutionize a number of sectors, from automation, transportation, energy, healthcare, financial services to nanotechnology.

1.2 Home Automation Systems

IoT technology can also be applied to create a new concept and wide development space for smart homes to provide intelligence, comfort and to improve the quality of life. Thus, home automation becomes much more popular nowadays. In 1984, the National Association of Home Builders (NAHB) introduced the “smart house” concept. The researches and developments in this field continued over the years, but to actually automate a building is still a quite expensive job. This leads to the question: “If it is expensive, why home automation?”. First, home automation brings interoperability: the temperature can be set to a certain value according to certain conditions, lighting can be turned on, off or may be dimmed based on daylight. Second, home automation implies remote access, such as monitoring the house using a laptop or even the own cell phone. Third, a smart home automation system should have the possibility to be extended or reduced when needed. Therefore, it can bring expandability and also energy savings. Nowadays, one of the hottest topics in media is related to energy conservation. Automation systems can help the energy savings by, for example, turning off the electronic devices automatically when they are not in use. A house which is equipped with such a system offers much more comfort, flexibility, elegance, security, but most important, reduced maintenance costs through the
optimization of the consumption of electricity and heat. For example, some of these smart houses can include simple things like turning the sprinkler at some time during the day or detecting thieves in the middle of the night; others are more advanced and employ sensors for detecting the presence of a person in a room, used to adjust the ambient light, to control the temperature or the music volume depending on various factors. A field which encompasses all facilities of a smart home is “domotic”. The domotic term was coined in 1984 by the journalist Bruno de Latour. It is a combination of technologies and services that improve the life in the areas of safety, comfort and technical management, resulting in a complete system. Things can happen automatically, as they are necessary or as they are scheduled by the users. However, the user does not lose the control over the house, manual operation being available also. Typically, domotic systems need to collect data from various sensors and make things such as light and temperature to automatically adjust. Moreover, using these sensors, many tasks can be accomplished, such as, controlling curtains and windows without human intervention, opening, locking or unlocking the garage gate, controlling the climate inside the house, providing the corresponding light in each room, starting the sprinkler when the soil is too dry and so on. However, the concept of home automation gets increasingly louder on the market, being a relatively new concept which draws attention to researchers. This leads to new technologies that can perform home automation functions.

Different devices and the appliances in the home such as lightings, air condition, home security and entertainment systems are now being connected to the Internet so that it can be controlled remotely using the Smartphones or Tablets. Not only devices can be
controlled, but home environment can also be continuously monitored for maintaining certain desired temperature or monitoring amount of energy consumption. Hence, this will contribute to overall cost reduction and energy saving which is one of the main concerns of today.

This paper presents a low cost and flexible home control and monitoring system using a Raspberry Pi, with IP connectivity for accessing and controlling devices and appliances using Apple Home app and an Android based smartphone app. The proposed system does not require a dedicated server PC with respect to similar systems and offers a flexible and low cost medium to monitor and control the home environment. The device utilizes control functionality of home appliances also through web browsers as an additional medium of access apart from iOS and Android based smartphones.

The IoT includes everything from wearable fitness bands and smart home appliances to factory control devices, medical devices and even automobiles. Security has not been a high priority for these devices until now. There has been a lot of discussion regarding the hacking of devices and systems to obtain information and data. However, just as critical are cyber-attacks against the devices themselves - attacks which take over control of the device and cause them to operate in dangerous and insecure ways.

Unfortunately, many of these systems – thought to be safe – are still vulnerable. For instance, even though Industrial Automation and Critical Infrastructure devices are usually installed inside the secure perimeter of an enterprise network, that perimeter is porous and can be easily penetrated or disabled. On top of that, insider threats, whether malicious or accidental, make up 70% of cyber-attacks, and they usually originate inside that perimeter. It is necessary to secure the Things themselves. Security challenges in IoT include privacy,
authentication and secure end-to-end connection. In addition, with the presence of multiple smart home standards currently used in the market, any security scheme needs to consider inter-compatibility among the multiple standards.

Figure 1.2 - The System Setup

Figure 1.2 shows the setup of the proposed system, consisting of several IoT devices connected via the Home Wi-Fi network. The users can access and control the system using a mobile device by accessing the home Wi-Fi. The network is responsible for authenticating and monitoring the communication between devices in the system.

Each IoT device can only communicate with the home network. Based on user preferences, the information from one device can also trigger the network to send a message to another device in order to response with appropriate action.

The subsequent sections are divided in the following ways. ‘Introduction’ is followed by ‘Related Work’ section which gives an account of some similar projects achieving similar results but the design and implementation are carried out in different ways. Next section is ‘Motivating Examples’ which gives a brief account of different works that motivated me
to pursue this project. Next section describes some ‘Challenges’ that I faced during the design and implementation of this project. Then the section, ‘Solution and Implementation’ gives a detailed account of both the hardware and software implementation of this project. Next sections gives a brief review of the ‘Results’ achieved after the project implementation. And lastly, it is followed by ‘Conclusion and Future Work’. 
CHAPTER 2 : RELATED WORK

Home automation or Smart Homes (also known as domotic) can be described as introduction of technology within the home environment to provide convenience, comfort, security and energy efficiency to its occupant. Adding intelligence to home environment can provide increased quality of life for the elderly and disabled people who might otherwise require caregivers or institutional care. There has been a significant increase in home automation in recent years due to higher affordability and advancement in Smart phones and tablets which allows vast connectivity. With the introduction of the Internet of Things, the research and implementation of home automation are getting more popular. Much of the research attention has been given in academia. Various wireless technologies that can support some form of remote data transfer, sensing and control such as Bluetooth, Wi-Fi, RFID, and cellular networks have been utilized to embed various levels of intelligence in the home. The studies have presented Bluetooth based home automation systems using Android Smart phones without the Internet controllability. The devices are physically connected to a Bluetooth sub-controller which is then accessed and controlled by the Smart phone using built-in Bluetooth connectivity. However, due to limited range of operation (maximum up to 100 m) the system is unable to cope with mobility and can only be controlled within the vicinity. Researchers have also attempted to provide network interoperability and remote access to control devices and appliances at home using home gateways. Researches introduced a Wi-Fi based home control system using PC based web server which manages the connected home devices. Similar designs have also been presented in where a dedicated web server, database and a web page have been developed
to interconnect and manage the devices with the Internet. The disadvantages of these systems are twofold. Firstly, a high end personal computer has been utilized which not only increases the cost of installation but also increases the energy consumption. Secondly, development and hosting of web pages which also add to the cost. A GSM based communication and control for home appliances has also been presented where different AT commands are sent to the Home Mobile for controlling different appliances. The drawback of this system is that users are not provided with a graphical user interface and users have to remember different AT commands to control the connected devices.

Researches proposed mobile IP based architecture and its potential applications in Smart homes security and automation without any actual deployment and testing. Lately few researchers have also presented use of Web services, Simple Object Access Protocol (SOAP) and Representational State Transfer (REST) as an interoperable application layer to remotely access home automation systems. Researches introduced a smart home management scheme over the Ethernet network based on XML SOAP standards. The drawback of using SOAP based Web a service is that it is complex and adds overhead to the client and server when parsing the message, resulting in slower operation and higher Bandwidth. REST has been presented as a Web-based interaction for controlling household appliances using Web techniques such as HTTP caching and push messaging. Also, a Web-based graphical user interface has been developed to manage the home devices. Home automation using Cloud computing has also been proposed where users were able to control various lights and appliances within their home. The above-mentioned systems have made significant contributions to the design and development of home automation systems. However, the existing works were mainly focused on developing a single type of
accessing functionality for their home automation devices rather than combining different kinds of ways through which the device can be accessed and controlled depending upon the different kinds of mobile devices users in the same household may own.
CHAPTER 3 : MOTIVATING EXAMPLES

I was motivated to pursue this project due to numerous reasons. Although automation is not a new idea in our modern life. Large businesses and wealthy homeowners have already implemented this technology for years now. With this project, I wanted to see this concept getting more accessible to very home owner, due to cheaper cost, easier to setup and used modular concept and also higher internet penetration rate. The system needs to be simple enough to be used by one and all. With this project, I hoped that it will provide a better quality of life, while reducing the electricity wastage by giving user the power to control, conserve and react according to user needs.

On researching, I found several similar projects that motivated me further. [2] describes a system that provides a wireless remote-control solution for controlling the lights and fan via Wi-Fi capable handheld devices such as Smartphone, adding convenience and also reducing electricity wastage. While this concept is not new, all of this only appeals to tech savvy user, due to the complexity, feature and price, which are not important for this project target user. In this project, appliances such as light and fan that connected to the Main Control Unit (MCU) still can be controlled remotely from a computer screen or a smart phone. This is performed by using a very simplistic Graphical User Interface (Graphical User Interface, GUI), which is easily used and understandable for the target user. This system can also be equipped with the monitoring function by including a web camera to the MCU for a live video feed, or from wearable electronics wore by the user which for example include heartbeat sensor.
[5] presents a low cost secure cell phone based, flexible home automation system. Appliances at home are connected to the Arduino BT board. The communication between the cell phone and the Arduino BT board is wireless. Additional devices can be connected into the system with little modifications. Since the cell phone script is written in Python, it is portable and can run on any Symbian Operating System platform. The system is secured for access from any user or intruder. The users are expected to acquire pairing password for the Arduino BT and the cell phone to access the home appliances. This adds a protection from unauthorized users. This system can be used as a test bed for any appliances that requires on-off switching applications without any internet connection.

[24] project integrates locally and remotely controlled systems with the use of Cloud data network. This allows the system to operate without the dependence of a mobile provider, allows the system to be used with various mobile phone platforms, and allows the system to operate locally when phone or computer access is not available. Cloud networking and data infrastructure allow individuals to monitor, manage, and control their personal data points through the Internet. Each data stream is given a unique feed identification number to differentiate itself from all other data Streams on the Network. The home control is achieved via a hand held remote which uses Zigbee wireless communication to the home controller which is integrated with the cloud network via an internet connection.

[13] put forwards the design of home automation and security system using Android ADK. The design is based on a standalone embedded system board Android ADK (Accessory Development Kit) at home. Home appliances are connected to the ADK and communication is established between the ADK and Android mobile device or tablet. The
home appliances are connected to the input/output ports of the embedded system board and their status is passed to the ADK. It presents an authentication to the system for authorized person to access home appliances. The device with low cost and scalable to less modification to the core is much important. It presents the design and implementation of automation system that can monitor and control home appliances via android phone or tablet.

Although all these projects motivated me to pursue my idea, I still felt need for a better design and implementation. I wanted to design a home automation system which had all the specifications of above mentioned projects like low cost, ease of use, low energy consumption but I wanted the user experience to be similar for all kinds of device users. Hence, I pursued the idea of a device that has three ways of communicating with home appliances. First is through Apple Home app, which runs on iOS 10 or later, and lets you securely control smart home devices or any home appliance as long as the device is added in the Apple HomeKit as an accessory. The second way is through the means of an Android app. And lastly, the application is not only limited to smartphones but also can be used by feature phones through web browser.
CHAPTER 4 : CHALLENGES

Designing of home automation system comes with multiple challenges as follows:

1) Reliability: For home automation to succeed, developers should address considerations regarding the dependability of sensible devices compared with ancient home appliances and equipment. If connected devices don't possess similar practicality to precursor appliance, they might produce a replacement category of issues, like how to guarantee service continuity within the event of a sudden breakdown or service failure.

A large-scale service outage is one factor; however, a connected device or home automation merchandiser is additionally at the mercy of the consumer's broadband affiliation. If the product cannot fall back to some lower normal of helpful practicality once a web affiliation is unavailable, the consumer's valuation of your product are injured when their net connection has issues. This creates an oversized third-party dependency for sensible device corporations.

2) Date Collection and Use: Many connected home and smart products rely on value propositions that are in part about new functionality, and in part about the 'smarter' use of resources. In order to achieve this, data flows between the devices and servers operated by the device providers, between devices, and to and from the consumer's smart phone or computer.

This creates opportunities to gather information that may be accustomed to improve the service, or be analyzed by marketers to learn regarding consumers' habits to create and
grow existing relationships. Even if the systems aren't hacked by malicious third parties, 
users and customers ought to be confident that the vendors provision these product and 
services area unit by themselves trustworthy.

Vendors ought to see compliance with information protection laws as a value differentiator 
when developing their product offerings and selling methods. Vendors that fail to do this 
may step by step lose and more and more information and privacy aware market.

3) Data transformation and integration: The evolving 'connected home' means that many 
related professions, such as locksmith, heating engineer and electrician, need to consider 
putting software at the heart of their businesses and transforming themselves into digital 
providers to keep up with the market. These professionals still represent key intermediaries 
for consumer choices about major installation projects. Vendors that understand this, and 
provide software tools which can be deployed to interact with particular products, are more 
likely to benefit from the goodwill generated in the professional community.

Another factor to consider is standardization and the ability to connect to systems/devices 
from other manufacturers. Having APIs or other standards-based connectivity solutions 
that allow devices to control/be controlled by other devices can add significantly to the 
overall value proposition to the consumer. This raises the question of which company owns 
particular standards for device interconnectivity. Where any partnerships with other device 
manufacturers, app developers or platform providers are to be considered, both parties 
should address and carefully document how any newly created intellectual property will 
be owned at the outset to avoid difficulties down the line.
4) Liability: Solutions to smart device problems often come in the form of updates and patches, which aren't always completely reliable. Developers also need to bear in mind that not all users will download updates as they become available, leading to 'version lag' as devices continue to run older software. In addition to creating support challenges for vendors, this could leave devices vulnerable to attack. All of this creates a complex situation from a product liability perspective, as the device being used at any given point may function very differently to the device the consumer first bought.

Since many connected devices require an ongoing service component from the vendor to function, the consumer-facing T&Cs associated with a service are one way for manufacturers to try to limit and exclude liability. The effectiveness of this strategy will vary by jurisdiction, and the law is likely to step in to render exclusions or limitations invalid in jurisdictions with a more protective attitude to consumer rights.

Where the relevant manufacturer has partnered with another device manufacturer or platform provider, these kinds of liability issues can be addressed in the agreements that govern the commercial relationship. In many cases, where manufacturers simply follow a published standard for device interaction, or use a documented public API, liabilities will be less clearly delineated, and vendors will have to proceed on the assumption that they may bear a substantial part of the risk even if there are extrinsic factors involved.
CHAPTER 5 : SOLUTION IMPLEMENTATION

5.1 Accessing the device through browser:

5.1.1 Hardware Implementation:

The hardware requirements are- one Raspberry Pi computer, one 5V relay with multiple channels, multiple female to female jumper cables, one external power adapter and any home device or appliance (Here, a string light is used). Figure 5.1 shows the hardware setup.

![Hardware setup](image)

*Figure 5.1 - Hardware setup*

The output of the relay will be connected to the string lights. The GPIO pins on relay will be connected to Raspberry Pi pins. This is how the Raspberry Pi will be able to send commands to relay to close or open the connection to the string lights which will turn it
ON/OFF. The GPIO pins are connected according to the Raspberry Pi Pin Mapping guide as shown in figure 5.2 [3].

Figure 5.2 - Raspberry Pi Pinout Mapping Guide

5.1.2 Software Implementation:

The first step to get started is to install the operating system Raspbian in a Raspberry Pi computer. Raspbian is a Debian-based computer operating system for Raspberry Pi. There are several versions of Raspbian including Raspbian Stretch and Raspbian Jessie. Since 2015 it has been officially provided by the Raspberry Pi Foundation as the primary operating system for the family of Raspberry Pi single-board computers. Raspbian uses PIXEL, Pi Improved Xwindows Environment, Lightweight as its main desktop environment as of the latest update. It is composed of a modified LXDE desktop
environment and the Openbox stacking window manager with a new theme and few other changes. Raspbian is a free operating system based on Debian, optimised for the Raspberry Pi hardware. Raspbian comes with over 35,000 packages: precompiled software bundled in a nice format for easy installation on Raspberry Pi.

Next step is to install the Apache Web Server on Raspberry Pi which will allow it to serve web pages. On its own, Apache can serve HTML files over HTTP, and with additional modules can serve dynamic web pages using scripting languages such as PHP. So, the PHP library is installed through the Apache.

Now it’s time to create the PHP file which will be the front end and it will show all the buttons which will turn home appliances and devices ON/OFF on clicking as shown in figure 5.3. Each button on the file will send commands to the relay and then from the relay these commands will be sent to the GPIO pins on Raspberry Pi. To control the GPIO pins through PHP file, WiringPi is installed.

WiringPi is a PIN based GPIO access library written in C for the BCM2835, BCM2836 and BCM2837 SoC devices used in all Raspberry Pi versions. It’s released under the GNU GPLv3 license and is usable from C, C++ and RTB (BASIC) as well as many other languages with suitable wrappers (See below) It’s designed to be familiar to people who have used the Arduino “wiring” system.

WiringPi includes a command-line utility gpio which can be used to program and setup the GPIO pins. You can use this to read and write the pins and even use it to control them from shell scripts.
WiringPi is extendable and modules are provided to extend wiringPi to use analog interface devices on the Gertboard, and to use the popular MCP23x17/MCP23x08 (I2C 7 SPI) GPIO expansion chips, as well as module that will allow blocks of up to 4 74×595 shift registers to be daisy-chained together for an additional 32-bits worth of output as a single unit. (You can have several blocks of 4 74x595s if needed) One of the extension modules allows you to use an ATmega (e.g. Arduino, or the Gertboard) as more GPIO expansion too – via the Pi’s serial port.

WiringPi supports analog reading and writing, and while there is no native analog hardware on a Pi by default, modules are provided to support the Gertboards analog chips and other A/D and D/A devices can be implemented relatively easily.

The next step is to create the Python script on Raspberry Pi in order to work the home appliances. This is where we define the channel which will control the output of the relay.
to the GPIO Pins. The script mainly has two functions each for sending the command to turn the device ON and OFF.

When all of these steps are done, the device operates completely fine and can be used to operate any home appliance (once connected to the relay) through the web browser of any personal or mobile computing device.

5.2 Accessing the device through iOS Home app:

Apple launched its Home app with iOS 10 in 2016. It serves as a centralized hub for managing Apple HomeKit-enabled accessories. HomeKit is a software framework by Apple that lets users set up their iPhone or other Apple device to configure, communicate with, and control smart-home appliances. By designing rooms, items, and actions in the HomeKit service, users can enable automatic actions in the house through a simple voice dictation to Siri or through apps. HomeKit was first released with iOS 8 in September 2014. HomeKit support is not available in macOS, but it is available on all their other devices, including through Siri. Manufacturers of HomeKit-enabled devices were required to have a MFi Program, and all HomeKit products were required to have an encryption co-processor. Equipment manufactured without HomeKit support can be enabled for use through a "gateway" product, such as a hub that connects between those devices and the HomeKit service.

Many HomeKit-enabled smart accessories have their own separate apps, but the advantage of using the Home app is that you can access and control all of them from one centralised location.
With the new Home app in iOS 10, it’s easy to set up and manage all your HomeKit-enabled smart accessories. The app features integration with Control Center, 3D Touch quick actions, and of course, support for Siri.

The app's settings are also synced through Apple's iCloud storage service, so any iCloud-enabled Apple device - whether that be an iPhone or iPad - can be used to control your Home.

With this project, I am going to connect normal products like, a basic string light, which is not at all a smart device, to the apple HomeKit and will control it through the Home app.

5.2.1 Hardware Implementation:

The hardware requirements and implementation for this part are almost similar to the previous part. We require one Raspberry Pi computer, one 5V relay with multiple channels, multiple female to female jumper cables, one external power adapter and any home device or appliance (Here, a string light is used). Apart from that we need an iOS device having iOS 10 or later. The output of the relay will be connected to the string lights. The GPIO pins on relay will be connected to Raspberry Pi pins. This is how the Raspberry Pi will be able to send commands to relay to close or open the connection to the string lights which will turn it ON/OFF. Figure 5.4 shows the flow chart of the hardware setup.

5.2.1 Software Implementation:

First step is to install and run Homebridge on Raspberry Pi. Homebridge is a NodeJS server that acts as a HomeKit-enabled bridge, linking up non-HomeKit-enabled products to your HomeKit setup. Once you've set up a Homebridge server, you can download
plugins for non-HomeKit-enabled devices in order to access them with the iOS Home app and control them using your voice with Siri. Next step is to install and run the HAP-NodeJS(HomeKit) on Raspberry Pi. HAP-NodeJS is a NodeJS application which is installed to form a bridge between HomeKit requests and the Wi-Fi devices.

Next step is to make the software setup for Siri/HomeKit lights. We already have a Raspberry Pi with WiFi and HAP-NodeJS fully setup from the previous section. The channel relay with jumper cables is also already working and setup.

Now we need to download the Setup Light Script in the HomeKit directory. Then we need to make a python directory and download and install all the python accessories and files for the HomeKit. By doing this, we will get a Light Accessory file and a Python Directory at the root of HAP-NodeJS. We will then create two python files named light0.py and light1.py.

Now we need to go into the Accessories folder in HAP-NodeJS directory which contains a light accessory file. Then we need to make some changes in this file. We need to change
the display name of the device which is working as an accessory (String Lights). Then we need to change the user name of that accessory. We can also change several values related to our particular accessory according to our wish.

The coding related to the power state of accessories is already done in the file. It consists of two functions one for switching ON the lights and another that switches OFF the light. Next step is to install Forever on Raspberry Pi which is a node module and start it. Then we need to start the Apple Home BridgeCore file at the HAP-NodeJS root directory. Once started, we just need to go to the apple Home app and the device (String Lights) will have been added there as an accessory and now we can start controlling this accessory. A screenshot of the Home app with String Lights as accessory is shown in the figure 5.5.

![Figure 5.5 - Screenshot of Home app with String Lights as an accessory](image-url)
5.3 Accessing the device through Android app named, HomePi:

5.3.1 Hardware Implementation:
The hardware requirements and implementation for this part are almost similar to the previous part. We require one Raspberry Pi computer, one 5V relay with multiple channels, multiple female to female jumper cables, one external power adapter and any home device or appliance (Here, a string light is used). Apart from that, we need an Android device. The output of the relay will be connected to the string lights. The GPIO pins on relay will be connected to Raspberry Pi pins. This is how the Raspberry Pi will be able to send commands to relay to close or open the connection to the string lights which will turn it ON/OFF.

5.3.2 Software Implementation:
The system consists of a micro web - server based on Raspberry Pi, hardware interface modules and the Android compatible Smart phone. This system allows authorized home owners to remotely control and monitor connected devices at home using any Wi-Fi or 3G/4G enabled Smart phone which supports Java. The smart phone app provides a graphical user interface (GUI) for accessing and controlling the devices at home through server real IP. Upon starting the HomePi app, the user needs to select a particular room and on selecting room the user finds a screen showing several buttons. Each button is connected to a device which can be turned ON/OFF by clicking on that button. Figure 5.6 illustrates the interface of Android GUI. The interface is simple to use, user can simply touch on the icon to turn on/off the appliances.
Figure 5.6 – (a),(b),(c) Screenshots of HomePi app
CHAPTER 6 : EMPIRICAL RESULTS

The project was completed and is working as described in this paper. The system allows the users to control appliances and lights in their home from an iOS device or an Android device or from the web browser of any personal computing or mobile device through the home Wi-Fi network. It also allows the user to control their units within their home acting as a wireless remote. The project was tested to turn appliances on and off such as: radio, fan, coffee maker, and television. The project turned out to be a very flexible and low cost solution for the basic switching control needs of a home automation system. The interface design is completely user friendly. It’s basically like a virtual button pressing interface which controls devices on clicking. The device works fairly good in terms of security as well. Since, the device works under the home Wi-Fi network, it automatically adheres to all the security configurations implemented on the home Wi-Fi network. The iOS Home app's settings are also synced through Apple's iCloud storage service, so only the authorized iCloud-enabled Apple device - whether that be an iPhone or iPad - can be used to control the home automation device with it.

In a questionnaire I conducted with 8 other users of this device, I received multiple positive results. Almost all of them agreed to the statement that this device is indeed very beneficial in improving the convenience and comfort of its users. Some of them pointed to the fact that with this device, there is no need to carry separate remotes for various kinds of appliances. The device has its own kind of centralized hub which lets you control various home appliances all in one place. All of the users agreed that it is a very simple system to set up and everyone also acknowledged that the system is completely affordable for all
with a price range of $30-40. One particular user pointed to the fact that the system can be very useful for physically challenged and disabled people as they can control home appliances from a distance through their mobile devices with ease. Some users also had the observation that this device is very different from other home automation systems as it has multiple ways to control it using different types of mobile or computing devices since it is very normal for people in the same household to prefer different kinds of mobile devices like Android or iPhone.
CHAPTER 7 : CONCLUSION AND FUTURE WORK

In this paper, an architecture for low cost and flexible home control and monitoring system using Raspberry Pi is proposed and implemented. The home appliances are controlled by various kinds of mobile and smart devices. Such a design transforms a mobile phone into a portable remote controller for home automation. It is noted that the proposed system is not restricted to home automation, it can be applied directly to remote control of many industrial devices. Using this system as framework, the system can be expanded to include various other options which could include home security feature such as open-door and motion detection, energy monitoring, or weather stations.

Currently, the device is only offering internet security through the means of the Wi-Fi network it is connected but more dedicated security implementation needs to be done.

Some possible aspects of home automation which make us not forget that the need for new automated systems is very high and the degree of integration increases from year to year are presented. Therefore, more and more functions will be included into fewer and smaller systems that can perform several functions in parallel. Automation is inevitable, especially because it can be a determining factor in energy, time or in housing space efficiency.
References


