“Home Automation through Internet of Things(IoT)”

Project report submitted in partial fulfilment of the requirement for the degree of Bachelor of Technology

In

Computer Science and Engineering/Information Technology

By

Kritika Verma (131291)

Shreya Sharma (133205)

Under the supervision of

Ms. Ruchi Verma

To

Department of Computer Science & Engineering and Information Technology

Jaypee University of Information Technology Waknaghat, Solan- 173234, Himachal Pradesh

Candidate's Declaration

I hereby declare that the work presented in this report entitled "Home Automation through Internet of Things(IoT)" in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology in Computer Science and Engineering/ Information Technology submitted in the department of Computer Science & Engineering and Information Technology, Jaypee University of Information Technology Waknaghat is an authentic record of my own work carried out over a period from August16 to December 2016 under the supervision of Ms. Ruchi Verma (Assistant Professor(Grade-1), Computer science and Engineering).The matter embodied in the report has not been submitted for the award of any other degree or diploma.

Kritika Verma (131291).....
Shreya Sharma (133205).....

This is to certify that the above statement made by the candidates is true to the best of my knowledge.

Ms. Ruchi Verma
Associate Professor
Computer Science & Engineering
Dated:

ACKNOWLEDGEMENT

We would like to express our special thanks of gratitude to our project guide Ms. Ruchi Verma who helped us in conceptualizing the project and actual building of procedures used to complete the project. We would also like to thank our Head of department for providing us this golden opportunity to work on a project like this, which helped us in doing a lot of research and we came to know about so many things.

Secondly we would like to thank our family and friends who guided us throughout the project so as to complete our project on time.

Thanking you,

Kritika Verma (131291)
Shreya Sharma (133205)

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

S. No.	Abbreviations	Description
1	IDE	Integrated Development Environment
2	ADT	Android Development Tools
3	D2D	Device to Device
4	D2S	Device to Server
5	MQTT	Message Queue Telemetry Transport
6	XMPP	Extensible Messaging & Presence Protocol
7	DDS	Data Distribution Service
8	AMQP	Advanced Message Queuing Protocol
9	IR	Infrared
10	LED	Light Emitting Diode
11	LDR	Light Dependent Resistor
12	CMOS	Complementary metal-oxide semiconductor
13	DTR	Data Transmission Rate
14	SPI	Serial Peripheral Interface
15	EEPROM	Electrically Erasable Prog. Read-only
16	RFID	Radio-frequency identification

ABSTRACT

In the present arena, the employment of home automation is increasing day by day exponentially. Home automation is an advancing field in information technology. Home automation is simply managingvarious activities of home with the new advanced technologies. In this project we are developing a prototype to demonstrate various aspects of Home Automation, like controlling electronic devices, using only when needed according to the external conditions, putting on/off the electrical appliances with cellular devices from a separate location. It also aims at saving the environment through efficient water use, fire detection, etc. It permits the owner to carry out different activities of home from a distant place. Home security is a field of home automation. It concentrates on the security aspects of homes and offices. This can be achieved by local networking or by remote control. These systems safeguard homes from intruders and burglars. Home security begins with home safety. Home safety starts with homeowners carrying out steps to safeguard their home and the people who live in it. The popularity of home security automation has been increased in greatly in recent years due to much higher affordability and simplicity. Thus,our project.

1. INTRODUCTION

From flipping a switch of light to opening your door of your store room with a remote control; our homes have witnessed automation now for tens of years. This concept goes as long back as in the 1934 World's Fair in Chicago that witnessed the "home of the future" and where it was presented. In the past eighty years, however, the automated homes have been made synonymous to smart home, courtesy to the Internet, sensors, connectivity and other emerging technologies. The modern automated home systems can do more than switching on our heating and our lights—it can actually think for us.

Security and technology are hastily offering solutions for home automation and building out the Smart Home of devices, that are connected and applications. As cameras, alarms, as well as, thermostats and other appliances have become controllable and programmable using Internet-enabled smart phones using android or ios, they are becoming part of a vast web called the Internet of Things (IoT). As devices have become linked to the inter network, they join others on a two-way street. Similar to the way the commands are sent to these devices, information about their activities can also be gathered and downloaded. This process allows for massive ability of data collection and analysis not just for home owners, but firms as well.

When we carry out the market research surveys we enquire from the target group using e-mails, phones, or personal surveys. In the era of the internet of things, why call on people who have constrained memory when we can enquire directly from the washing machine or the dish washer to ask them what happened now that we will have a variety of sensors to get the information.

A security installer can also check the status of systems they installed. Has anything gone wrong with the storage system or the cooler? As an alternative of contacting a technician, which is possible after making an appointment with the home owner, a technician in a remote station can perform the same kind of diagnostics and operations and often correct problems. Hence, IOT has become an important need.

1.1 Smart Home

The phrases “Home Automation,” “Connected Devices” and “Internet of Things” are not seldominterchangeable, but these are unique parts of the concept of Smart Home:

- **Home automation-** This refers to a home’s electrical devices connected to a central system automating those devices based on user input. Take an instance, you press the button and your shades go up and/or you give a voice command and hence your lights turn on.

- **Connected devices-** They are electrical/electronic devices that are intelligent, with courtesy to the connection to the Internet as well as the sensors. These devices know and/or are capable of guessing what a user needs. Firstly, the intelligence comes from the user programming, but with the course of time the device may learn and adapt to the patterns and engage in interaction with the users.

- **Internet of Things-** IoT may be called as the magical wand that turns the automated homes into the smart homes. With mixture of different sensors, smart systems, IoT connects everyday usable objects to a network, enabling these objects to complete tasks and communicate with each other, with no input contribution from the user.

When you blend together home automation, connected devices and IoT you get a Smart Home. A modern smart home can be easily managed through the use of a Smartphone, tablet and/or computer.

- **Why do we need Smart homes?**

  - **Savings-** Connected electrical devices such as learning coolers, sprinklers that are smart, lights that are Wi-Fi enabled, monitoring the electricity outlets as well as water heater modules reduce the energy and water use.

Figure 1: Smart home

- **Control-** Many things in a home, from ovens and fridges to deadbolts and air conditioners, can be managed remotely via applications in smart phones and tablets. In most cases, the control works when you are out of the home too, meaning you can close the door from the airport, check on the dog from Australia, or confirm that you switched off your stove from the market.

- **Convenience-** Having your living room light switched on as you reach your home, the home theatre system automatically playing your favourite song and front door opened once you approach it with hands full of shopping bags, is perhaps the ultimate luxury of the smart home. However, convenience isn't all about luxury. Smart locks can allow you to permit access to particular individuals at particular times, so that you don't have to stay home and/or give out a key. Similarly, a sensor tells you when your fridge is out of milk, and a Wi-Fi enabled doorbell lets you "answer" your door from anywhere in the universe.

- **Security-** There are many easy, connected solutions for security for the smart home that are cheap alternatives to 24X7 monitored security systems. Wi-Fi-enabled cctvcameras, connected motion sensors as well as smart smoke alarms can be monitored from inside or outside a home through a live video, email and text alerts.

- **Safety-** Smart sensors that can find out water leak, level of humidity, carbon di oxide, movement, heat and every environmental concern that could be imagined help prevent accident from turning into tragedies as they can communicate with owner directly, whenever you are, wherever you need.

- **Senior independence-** Automate audio reminder as well as voice activated alert systems are just a bunch of the features of home automation that help seniors' lead independent lives for a longer period of time. Moreover, cameras connected to the WiFi with two-way communication may help loved ones keep an eye on the senior citizens when they can't go and physically check on them.

1.2 Internet of things

The Internet of Things (IoT) is the network of objects or "things" embedded with electronics, software, sensors, and network connectivity, which allows these objects to collect and exchange a large amount of data. The Internet of Things permits objects to be sensed as well as controlled remotely across existing infrastructure of networks, developing opportunities for direct integration between the physical world and computer-based systems, leading to improved efficiency, accuracy as well as economic benefits. Each thing is identifiable uniquely through its embedded computing system however is able to operate within the existing Internet infrastructure. Various uses and applications of IoT are:

- Monitoring the environment
- Managing the infrastructure
- Manufacturing process
- Management of energy resources
- Medical as well as healthcare systems
- Automating homes and buildings
- For the purposes of transportation

Figure 2: Iot Applications

- **Benefits of IOT-**

- **Ubiquitous networks** -:personal Wi-Fi on your smart phones and on many of the other devices. Everyone (and everything) wants as well as needs to be connected.

- **Connected computing** -:we want all of the devices, smart phones, televisions(colored or black and white), dvd players, vehicles etc. to keep record of what we are doing, seeing, reading, and/or listening to as we sway through the day, from one place to another – the handoffs from device to device is happening already.

- **Intelligence at the periphery of the network** -:Jim Gray, the visionary database expert from Microsoft, envisaged smart sensors behaving as a small-database with embedded machine learning algorithms and pseudo codes. Here is how he mentioned it

(10 years ago): “Intelligence is swaying to the boundary of the networks. Each of the disk systems and each sensor component will be a competitive database machine.”

- **Analytics-as-a-Service-:** the API and App economies are already enormous and growing – which enables any “thing” to “do something interesting” as long as it can be connected to an API or can invoke an App that carries out a network-based service. The “thing” is a data generator as well as collector that learns from, makes predictions, and maybe even takes data-driven actions in response to the data that are collected (through the versatility and convenience of an App or API call) too.

- **Marketing automation-**: smart phone customer engagement, geological-location, Apple’s iBeacon etc. are all developing a network of knowledge and information regarding customers’ locations, intentions, preferences, as well as buying patterns. Obviously, the degree of geological location-based knowledge needs to maintain the right balance between user privacy as well as the timely delivery of important and significant products and services to the particular user.

- **Supply Chain Analytics-**: delivering the just-in-time products at the time of need (inclusive of the use of RFID-based tracking). Significantly, everything is a customer (inclusive of machines, automobiles, manufacturing plants, ATM machines, etc.), as well as the IoT is monitoring, watching, as well as waiting for the product needs to arise.

1.3 Objective

- To develop a prototype for an automated home controlled using Raspberry Pi, as well as relays.
- The aim is to both save electricity and provide its full use to the owner.
- To provide services such as water level sensing, fire detection, etc

- To develop a prototype for monitoring the user geographical area that will help the user in securing the domestic or industrial area.
- To provide real time monitoring of the area.
- To provide the real time information of the trespassing in the area/home and update the user by any source of notification.

1.4 Methodology

Smart home, which can be defined as a home that can identify as well as detect the owner, automatically adjusts the lighting of the house to your predefined and pre-mentioned taste, opens the doors automatically, plays your favourite music, waters your garden in the morning or as defined, turns on the security lights at night and turns them off in the morning, heats the water for bathe and prepares tea according to the defined recipe, streams to you anywhere in the world through the internet live video of what is happening in and around your house. It makes it possible to connect lighting, entertainment, security, telecommunications, heating, as well as air conditioning into one centrally-controlled system. This permits you to convert your house into an active partner that helps manage your busy life.

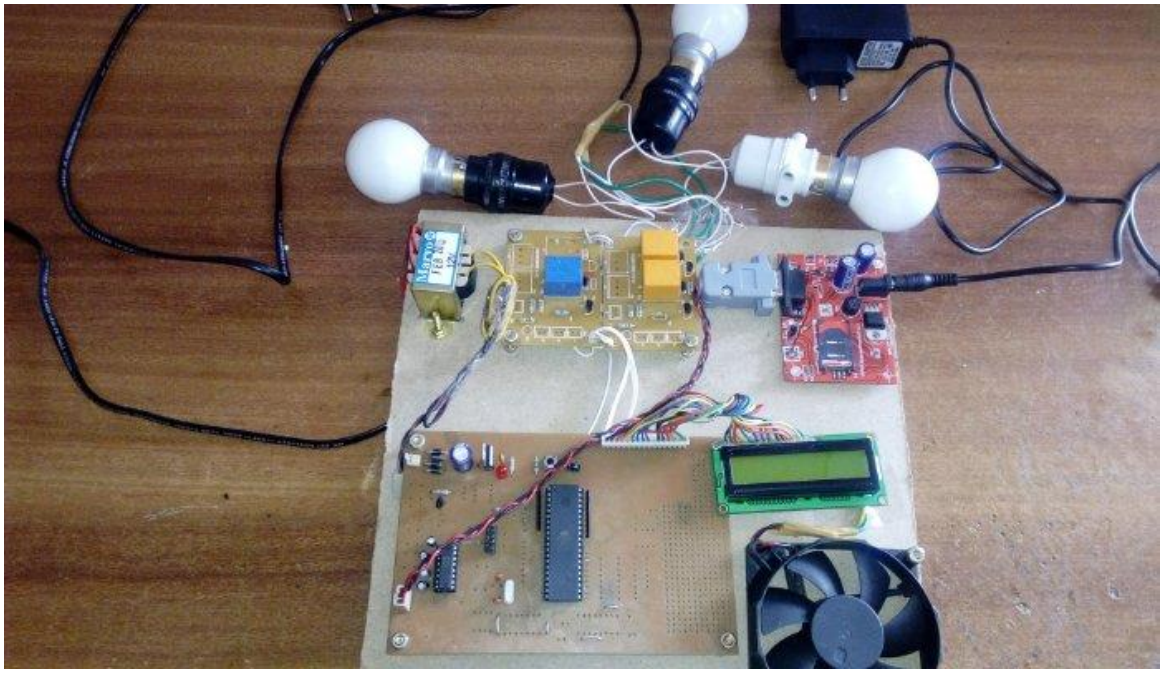


Figure 3: Methodology

Ours is a simplest of simplest, home automation system prototype whichranges from the burglar alarm, hi-tech security gates, water level sensing, light detection and an air conditioning system that is automated and that records the temperature at a predefined and pre-mentioned value.

A relay is a type of an electromagnetic switch. The one we have used in this project has basically two parts. The first part which is the coil, is the low-power part of the circuit, and shall be managed by the Raspberry Pi. The second part of the relay device is the switch, that can sustain higher powers. This part is actually the mechanical on the relay we have used, so one must hear a “click” sound when the relay is switching from one state to another state. Activating the low-power part involves applying 5V on the coil whichwill activate the switch and changes the state of the relay. To find in which state the relay is, the manufacturers have used one LED on one part of the switch. As expected, the LED mayalso be replaced by any of the appropriate device you want to switch on or off, for instance a bulb.

A major component in the world of makers, electrical and electronics, The Raspberry Pi ,is single-board, extremely cheap, high-performance computer first developed in the United Kingdom, by the renowned Raspberry Pi Foundation. It has not only helped in bringing the fun of electronics as well as computer programming to the people around the globe, but has also become a significant part of the maker's community. The wherever-present Raspberry Pi is now in the sixth iteration stage: Raspberry Pi 3—as the Pi Firm has continued to improve on an already existing, excellent product. There is also a five-dollar Pi Zero, which is a game changing Raspberry Pi in much smaller package.

Ever since the release of first Raspberry Pi, many a products have been developed to accompany, modify as well as enhance the Pi's capabilities and performance. From touch screens along with displays to HATs, cameras as well as plates, the possibilities are infinite when it comes down to creative project ideas.

2 LITERATURE REVIEW

2.1 RASPBERRY PI HOME AUTOMATION WITH WIRELESS SENSORS USING SMART PHONE: P BHASKAR RAO, S.K. UMA

The above stated paper presents a low cost as well as flexible home automation and monitoring system making use of the embedded microprocessor and/or microcontroller, along with connectivity using IP for getting access to and managing the devices as well as appliances ubiquitously using Smart phone application. System proposed here in this paper, does not requires a dedicated PC server with respect to similar systems along with offering a novel communication protocol that monitors and controls the house environment with more than just the turning on/off functionality. To establish the practicality as well as the potential of these systems, device, for instance, switches of light, power plug, sensors for temperature, sensor for currents etc may be used along with the home automation systems.

2.2 ANDROID BASED HOME AUTOMATION USING RASPBERRY Pi: T. ANITHA1, T. UPPALAIAH2

Over the years, the homely arena has seen a very fast establishment of network enabled digital technology that has improved our lives for good. These new technologies offer fresh and amazing options to expand the devices connectivity within the house for the motive and the purpose of home automation. Devices that are mobile are ideal in providing an interface for the users in such a home automation system, due to their abilities that include portability and including their wide range of capabilities. They may communicate with the home automation networks through the use of Internet services however they cannot directly contact the with the devices in the network, because these devices usually implement low-power protocols for communication, for example, Zigbee, WiFi etc. In this paper the authors aim at controlling and managing the appliances in the homes through Android device using Wifi as the protocol for communication along with Raspberry Pi as server system. They have created an interface that is user friendly for the android devices that allows the user to contact with the Raspberry Pi server of the system. The server has been interfaced with a relay circuit board which controls the appliances operating in the house. The communication with server permits the user to select and/or deselect the appropriate device. The server communicates with the corresponding relay device. Through this project the authors offer a scalable and a Home automation system that is cost effective.

2.3 Home Automation through IOT : Vinay Sagar, KN. Kusuma, SM. (2013)

In today's world, there are four main challenges faced by the home automation system today; these challenges include: high cost of ownership, inflexibility, poor manageability, as well as difficulty in achieving security. The main objectives of this project is to design and implement a home automation system using Internet of things technology, that is capable of controlling and automating most of the appliances in the house through an easy and manageable web interface. The system put forward in this paper, has a great adaptability of using Wi-Fi technology interconnecting the distributed sensors to home automation system server, which will ultimately reduce the cost of deployment along with increasing the upgrading ability and system reconfiguration.

2.4 Ramani, R. Olatunbosun , A. (2010) Internet of Things (IoT)

Information Technology is Internet of Things (IoT) has gained wide acceptance and popularity in recent years. The future is Internet of Things, that has the capability of metamorphosis of real world passive devices into virtual world nodes. The IoT strives to achieve unification of everything in our world under a common infrastructure, this will not help us to gain control but also enforce information symmetry. The prime aim of this paper is to provide an insight into Internet of Things, architectures, and essential technologies and their utility in our day to day life.

Arrival of IT and ITeS technologies has prompted a revolution in life at individual level as well as organizational working level. IoT has in store something for everyone ranging from many longitudinal and vertical markets encompassing a common man's everyday life in the society. Necessities of large corporations have driven the exponential growth in IoT infrastructure as these organisations tend to gain immensely by the improved predictability and control offered over its value chain system. This increased capability to track objects has manifested itself in companies becoming more efficient, speeding up of processes, marginalise error, prevent pilferage, through IoT. The IoT is a technological revolution that is going to spread out to each of the fields humans have ever created and revolutionise the future of computing and communications.

2.5 **P BHASKAR RAO, S.K. UMA A(2015), RASPBERRY PI HOME AUTOMATION WITH WIRELESS SENSORS USING SMART PHONE**

The aim of this project is to help users to operate devices with smart phones and to help people live a more sedentary life besides aiding energy savings. The android application will allow the user to control a device that is connected to any home appliance that is Pi enabled. The work area of this application is to make a security system with webcam surveillance, door sensor notification and a light control system. Sensors that have a connection to the home appliances can be monitored and controlled. In absence of authorised owner, a thief may sneaks up into the house using any means. In this case alarm will be triggered and client can monitor his home accordingly Client monitors his

home with webcam and could immediately pass on the information to local law and order authority. User can get the status of lights outside the house and accordingly change the status without any need of physical movement on his part. User interface must be organised, simple so as to facilitate ease of control, usage, monitoring on part of the user.

Python programming language is used to write server and client code and also interfacing raspberry pi with door sensors.

Following points clearly define the overall implementation.

Raspberry Pi: This minicomputer will provide an interface whereby app control systems can facilitate automation. Signals transmitted by the app on the phone will be received by the SoC processor and processes thereafter.

Android app development Mobile: With the help of an android application that will run on the user's Smartphone, communication is established between pi and home devices without any hassle. In order to establish communication between devices and other devices, an android app is developed on a well known platform of eclipse and android development tools.

Interfacing devices and Pi: The Raspberry Pi has to be interfaced or integrated to be more precise with the various sensors and webcam controller.

3.SYSTEM DEVELOPMENT

3.1Tools and Technologies used

3.1.1 Hardware Used

1. Raspberry Pi

The processor used by the Raspberry Pi system is a BroadcomBCM2835 system-on- chip (SoC) multimedia processor. It implies that the large number of the components, which includes its central processing unit and graphics processing units along with various audio components, video components and other communications infrastructure, are provided on a single set. It's the SoC design that is the main point of difference between Broadcom processors and those found in personal computers. In addition, pi employs a divergent instruction set architecture (ISA), which we call ARM.

Raspberry Pi has the advantage of running on just 5V 1A power supply that is supplied by the micro-USB port present on the chip itself. The Raspberry Pi is primarily designed to run OS named as GNU/Linux.

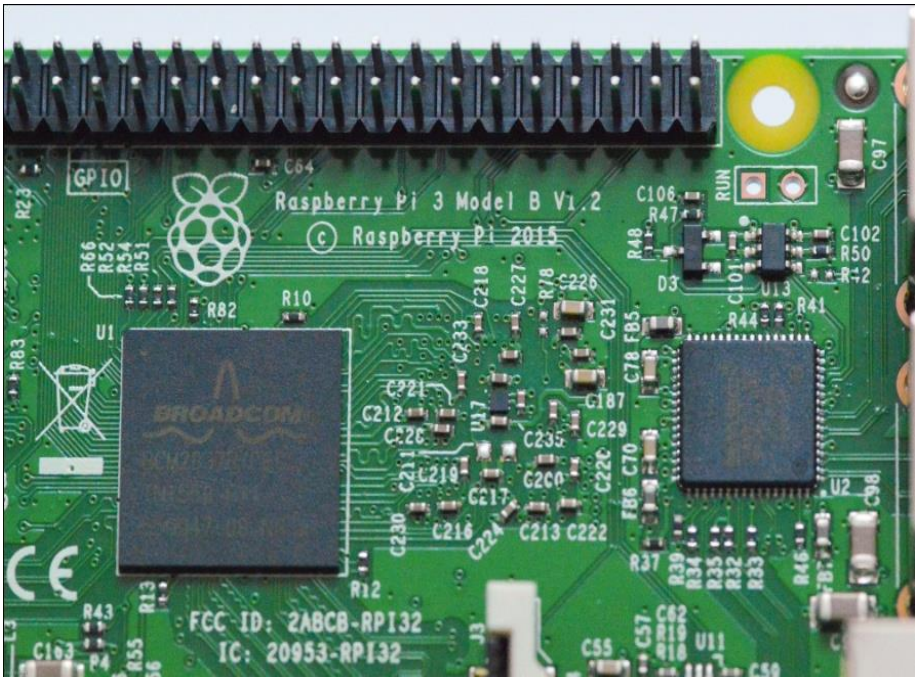


Figure 4: Raspberry Pi[8]

Table 1: Technical specifications

SoC Type	Broadcom BCM2387
Operating Voltage	5V1
Power	5V1,2.5A
Video Output	HDMI
GPIO Pins	14
No of cores	4
Operating System	Boots from SD card, Linux/ Windows 10 IoT
RAM	1 GB
Clock Speed	1.2GHz
Length	56.5 mm

Width	85.6 mm
Weight	45 g

Raspbian, a Debian-based Linux OS that is specially designed for Raspberry pi is available for download purposes, besides providing third party Ubuntu, Windows 10 IOT Core, RISC OS. It promotes Python and Scratch as the chief programming language, but the list is not limited to these two only.

Open source version is present unofficially.

Comparison chart of Arduino and raspberry pi alongside Intel Edison is given to ascertain the best among the three:

	Arduino Uno	Raspberry Pi Model B+	Intel Edison
Price	$30	$35	$50 (board not included)
Size	7.6 x 1.9 x 6.4 cm	8.5 x 5.6 x 1.7 cm	3.55 x 2.5 x .39 cm
Memory	0.002MB	512MB	1 GB
GPIO	14	40	40
Clock Speed	16 MHz	700 MHz	500 MHz, 100 MHz
On Board Network	None	10/100 BaseT Ethernet socket	Dual-band (2.4 and 5 GHz) Wifi, Bluetooth 4.0
Multitasking	No	Yes	Yes
Input voltage	7 to 12 V	5 V	3.3 to 4.5 V
Flash memory	32KB	Micro SD card	4 GB eMMC
USB	One, input only	Four, peripherals OK	One, peripherals OK
Operating System	None	Linux distributions	Yocto Linux v1.6
Integrated Development Environment	Arduino IDE	Scratch, IDLE, anything with Linux support	Arduino IDE, Eclipse, Intel XDK

GPIO Port:

The GPIO port that is present in Raspberry pi is located on the top-left of the circuit board.

Gpio pins can be configured to be made either on or off, work as either input or output besides input and output values are also readable.

Pins present in the GPIO port are assigned their own specific functionality. Coming together of various pins form distinct circuits.

Pin numbers for the GPIO port are divided into two rows, with the bottom row taking the odd numbers and the top row the even numbers. There are certain pins that are labelled as Do Not Connect. These pins are not supposed to be used as these are reserved for internal use of the processor. Using pins labelled such can cause damage to the hardware. The Pi's GPIO port provides a 5 V power supply on Pin 2 but the Pi's internal components operates on 3.3V. That implies that the components of Pi work with a 3.3 V power supply. Therefore, circuit has to be passed through a voltage regulator before it comes to the Pi in case interface with the Raspberry pi happens through the GPIO port.

Straightway connection to a 5 V supply to a pin on the Raspberry Pi's GPIO port, or direct connection established between pin1 or pin2 to various other pins present on the port can result in significant damage to the minicomputer.

There are 7 general purpose pins provided by the Pi by default, these are: Pin 11, Pin 12, Pin 13, Pin 15, Pin 16, Pin 18 and Pin 22. Apart from this, Pin 7 that provides clock support by default can also be employed as a general- purpose pin, taking the number of truly general purpose pins to 8. These pins works on binary logic i.e. possess two states: high, when they can supply a positive voltage of 3.3 V; and low, where they cannot supply any voltage as they are connected to 0V or ground. This binary logic is employed to change the state of other components to either on or off. Internally pi operates at a voltage logic level of 3.3V. This distinguishes it from other microcontrollers like Arduino operate at 5 V. Apart from these pins that are general purpose, certain pins are dedicated to the buses.

The GPIO pins can be used to interface physical devices like sensors with processor. For Python developers, Raspberry pi supports a library known as RPi.GPIO that provides functionality for interfacing the pins.

RPi.GPIO comes up with two numbering systems BOARD or the Broadcom GPIO numbers (BCM), but at a given point of time and in a given program only one of the above can be employed

The layout of the GPIO port is given in the figure below:

Various buses are summarised as follows:

UART Serial Bus

UART stands for universal asynchronous receiver/transmitter. It provides a simple two-wire serial interface. UART bus is employed as a port for transmitting or receiving messages when a serial port is set up. When a serial port is configured it's this serial bus that is used as the port for the messages. In order to expose messages from the kernel, pi's UART serial bus may be connected to a display node. Pins 8 and 10 provides the functionality for UART serial bus, whereby Pin 8 is set up as a node for the transmission

signal and Pin 10 is set up to act as a receiving terminal. Speed and data format are both configurable.

I2C Bus: In order to establish communication between two Integrated circuits I2C is used. I2C stands for Inter-Integrated Circuit. One of the two ICS to be connected is BCM processor that is SOC in case of Raspberry pi. Resistors that are not internal are not required to access the I2C functionality because pins already include pull up resistors on the pi.

Pin3 and pin5 provide the functionality of I2C bus whereby pin3 facilitates serial data signal while pin5 facilitates serial clock signal. BCM processor provides us with first I2C bus on the chip only and this bus is known as I2C0 in terminology. The second, I2C1, is terminated at resistors on the Raspberry Pi circuit board itself and this bus cannot be employed for tasks that are general and is dedicated.

SPI Bus: SPI is an acronym for Serial Peripheral Interface.SPI bus works in synchronous mode used mainly for in-system programming (ISP) of devices.

SPI bus follows master slave system architecture and has a unit master. The main point of difference between SPI and the buses previously discussed is that is four wired and have the facility of more than one select signal that enables it to communicate to more than one node at a point of time. Pin 24 and Pin 26 have the lines for chip select for two slaves are not dependent and pin19, pin21, pin23 provides the bus functionality for master o/p slave i/p, master i/p slave o/p and clock respectively.

Components:

RASPBERRY PI :-The Raspberry Pi ships simplistically is said to be a single-board minicomputer. Additionally, a few components are required to make the system ready:

1. Raspberry Pi — The Pi 2 Model B comes up with quad-core Processor and RAM has a capacity of 1GB which simply means the support for operating systems that are heavy in terms of memory and processing can be provided. Examples are Ubuntu and windows 10.

2. Power supply — 5V micro-USB power supply will be needed to make the system work that can be provided from a mobile device lying around.

3. USB keyboard

4. USB mouse

5. MicroSD card — the minimum required storage capacity for microSD is levelled at 8GB. Certain cards that already have NOOBS (New out of the box software) loaded into them are also present and available. In order to avoid purchasing a SD card already loaded with NOOBS, Noobs can directly be downloaded from the internet without incurring any cost.

6. MicroSD USB card reader -In order to download software onto PC, we need to connect SD card to it. So interfacing card with PC requires card reader in the form of an intermediary.

7. A monitor or TV that supports HDMI or composite video — Though Composite video display can be employed without any compromise ,HDMI caters much better to the needs and supports audio transfers.

8. An HDMI cable or composite video cable, that depends on the type of screen supported although HDMI has an advantage over the composite ones

9. An Ethernet cable (or Wi-Fi dongle)

.

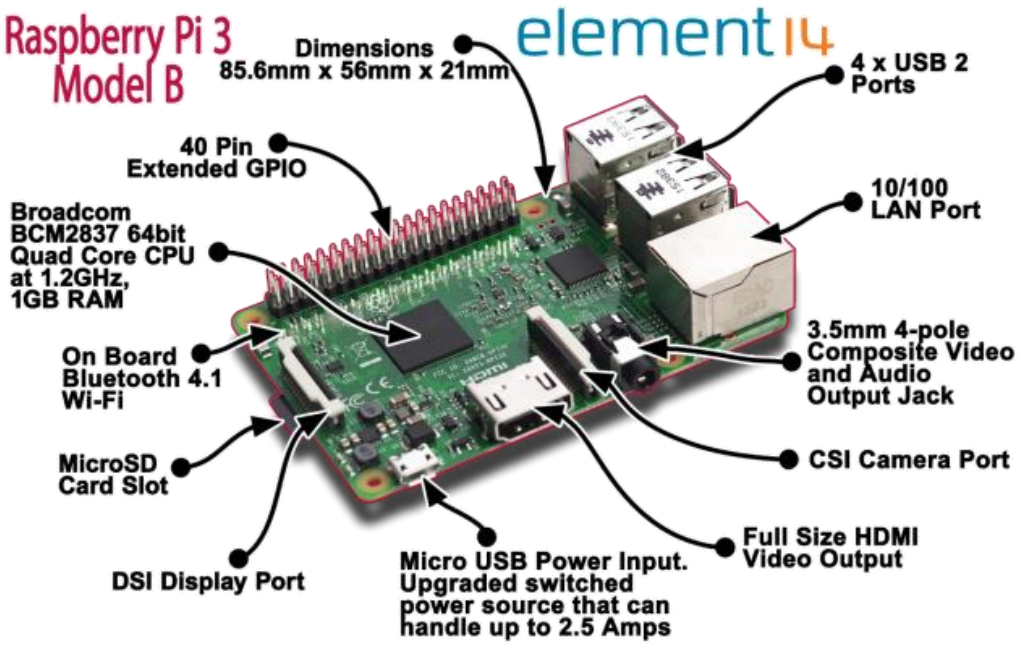


Figure 5: Raspberry Diagram [8]

2. Relay Module V2.0:

A relay driver circuit is essentially an electromagnetic switch that can turn on and off the devices connected to 220 v power supply

Input comprises of the following:

VCC as per the convention is connected to the positive supply voltage.

Ground which is marked as GND as the name suggests, is connected to negative terminal of the supply voltage

IN1 and IN2 are the communication pins

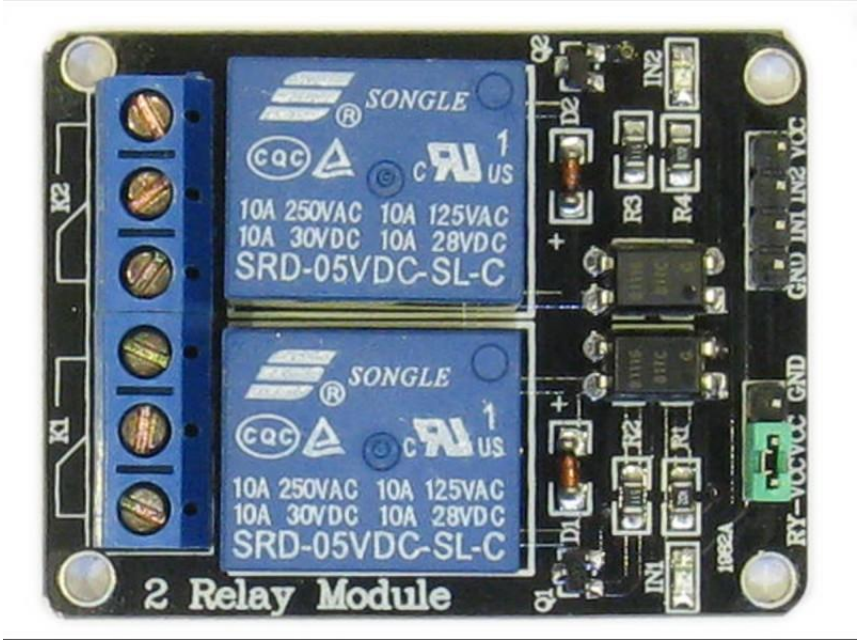
Each sub module in the relay has one NC (normal close), one NO (normal open) and one COM (Common).

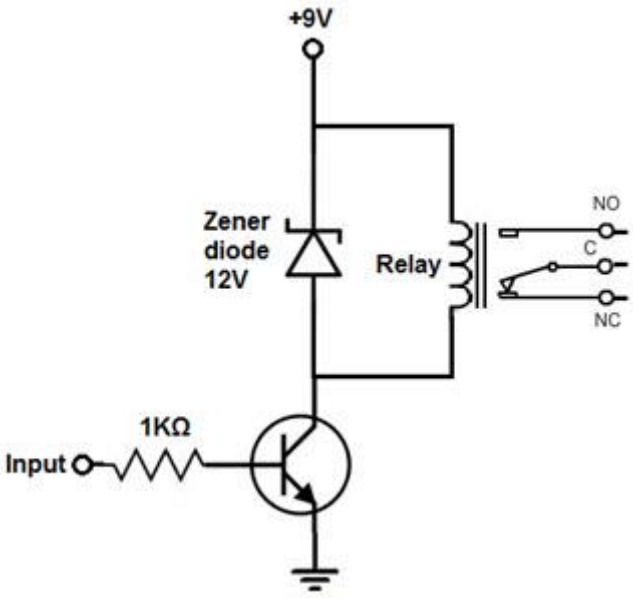
So relay comes with 2 NC, 2 NO and 2 COM of the channel relay in total.

NC is an acronym for the normal close port contact and the state without power.

NO is an acronym for the normal open port contact and the state with power.

COM stands for the common port.


SONGLE
10A 250VAC 10A 125VAC
10A 30VDC 10A 28VDC
SRD-05VDC-SL-C
SONGLE
10A 250VAC 10A 125VAC
10A 30VDC 10A 28VDC
SRD-05VDC-SL-C
2 Relay Module
K1
K2
IN1
IN2
GND
VCC


+9V
Zener
diode
12V
Relay
NO
C
NC
1KΩ
Input

3.1.2 Software Used

1. Android Studio or Eclipse IDE

Eclipse is an integrated development environment (IDE) widely employed in computer programming, and is the commonly used Java ID· The primary advantage of it is that it supports extensible plug-in system for customizing the developing environment. The main programming language is JAVA and it is most widely for writing JAVA application but it is also increasingly utilized to build applications in other programming languages which can be achieved by the use of plugins. The other languages supported include C, C++, COBOL and Python.

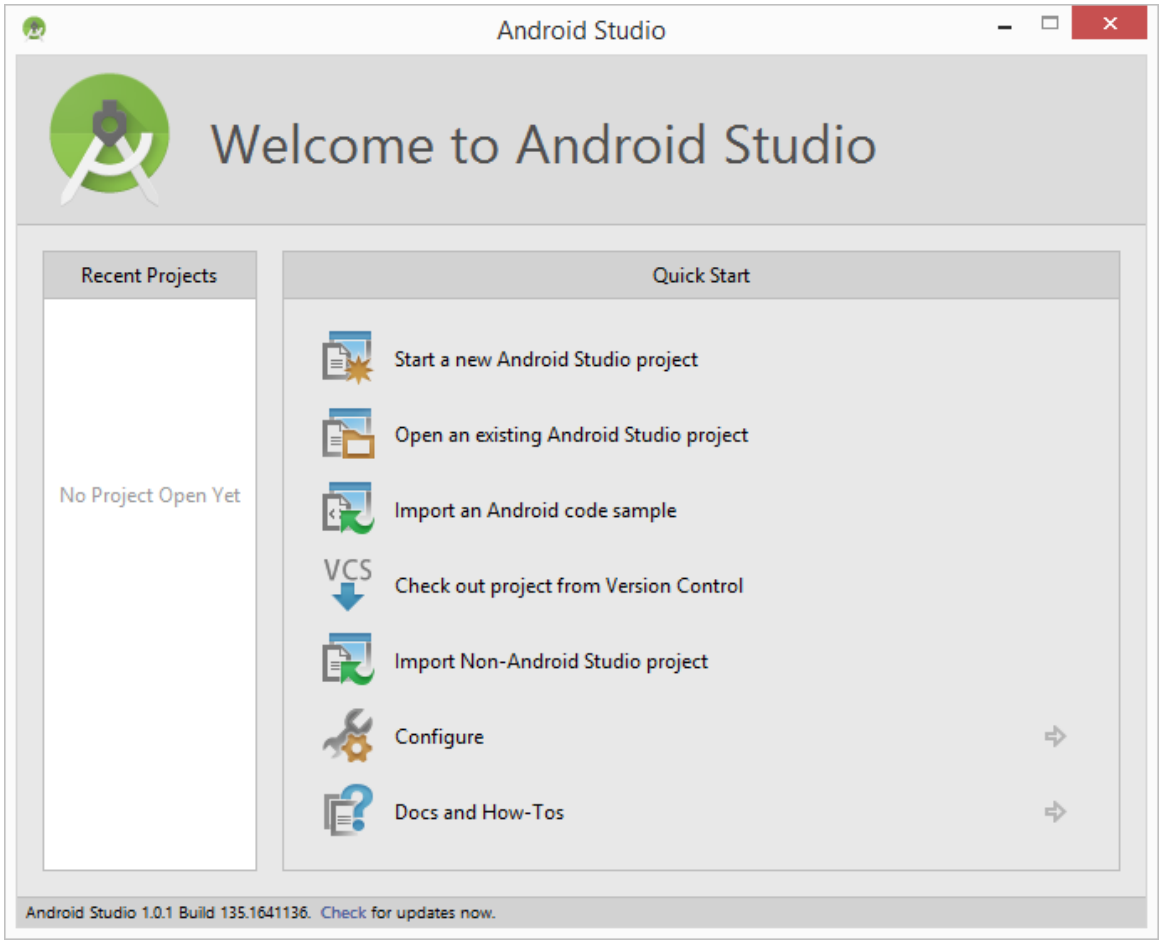


Figure 9: Android Studio IDE

2. Android Development Tools

Android is an intent based operating system. Android was primarily designed with the objective that developers must have necessary freedom and flexibility so as to design applications that are innovative and creative. Plus they could use effectively all the capabilities offered by a mobile phone. What adds further to the advantage is the fact that Android is open source and is simple and reliable to learn as well as create.

Android Development Tools (ADT) can be stated to be plug-in for the Eclipse IDE. These are designed and developed by search engine giant Google to give the developers a comprehensive and integrated environment for android app development. ADT can be seen as an extension to already much revered capabilities of Eclipse. It primarily allows developers to implement Android projects. Besides, it provides multiple benefits to create an application UI, debugging facilities provided by the Android SDK tools, and export .apk files for distribution. It has been replaced by Android Studio. Within the Android SDK Manager, following packages are to be installed:

SDK Tools

SDK Platform tools

SDK Build-tools

ARM EABI v7a System Image

Android Support Repository

Library Extras

Google Repository

Google USB Driver (for Windows systems)

Intel x 86 Emulator Accelerators

3.Python Shell

Python already comes with GPIO library but it is not yet loaded therefore necessitating user prompted explicit loading. There is a simple way to do it by starting the program with a statement:

Import RPi.GPIO at the top

Python is a high-level, interpreted, interactive language. It is developed to write clear programs making it highly readable. Syntax rules are relatively simpler than other languages and are general purpose.

- **Python is Interpreted:** Python is processed at runtime by the interpreter thereby eliminating need to compile it prior to execution. Therefore it can be placed in the same category as PHP and pearl.
- **Python is Interactive**
- **Object-Oriented:** Python supports encapsulation
- **As a Language for the beginners:** Python is diverse in the sense that it can be employed in wide ranging application from internet of things to gaming applications. Its lesser syntactical construction, readability makes it a great language to start with especially for the beginners

4. Firebase

It is a real-time database that is stored in the cloud and provides an API (application programming interface) to store and sync data in real time. It could act as a single point storer of data from multiple devices and also for multiple use-cases even sync to other clients. Firebase comes with many services apart from providing a cloud hosted database. Firebase Auth can authenticate users using only client-side code eliminating the need of any server side code. It supports login providers Facebook, GitHub, Twitter. Besides, it includes a user management system that allows developers to build systems that can authenticate users with email and password stored with Firebase.

Firebase cloud messaging (FCM): it is a cross platform messaging platform .Whenever user sign into an app firebase will receive credentials form the user which can be either related to Facebook, twitter, GitHub or prestored in firebase realtime database. These credentials are forwarded to authentication module of firebase that will verify these on backend and return the response to the client.

The Firebase Realtime Database is a cloud based database that inherently implies that it is not stored physically. Since data is stored on cloud real time synchronisation of

information can happen. It is particularly useful for cross platform applications, when all clients share one Database instance and whenever data is updated all the clients receive the updates in realtime. Data format is based on JSON.

The Realtime Database is a NoSQL database and therefore varies significantly in terms of its functionality to relational database. Database has immense utility for the applications for which response is in realtime. It can serve the needs of multiple users and without any compromise on reliability and responsiveness.

Firebase as an appropriate choice for a database:

• It provides support for ios, android, os x making it highly flexible.

• It eliminates the need for any server side bulky code or complex server infrastructure.

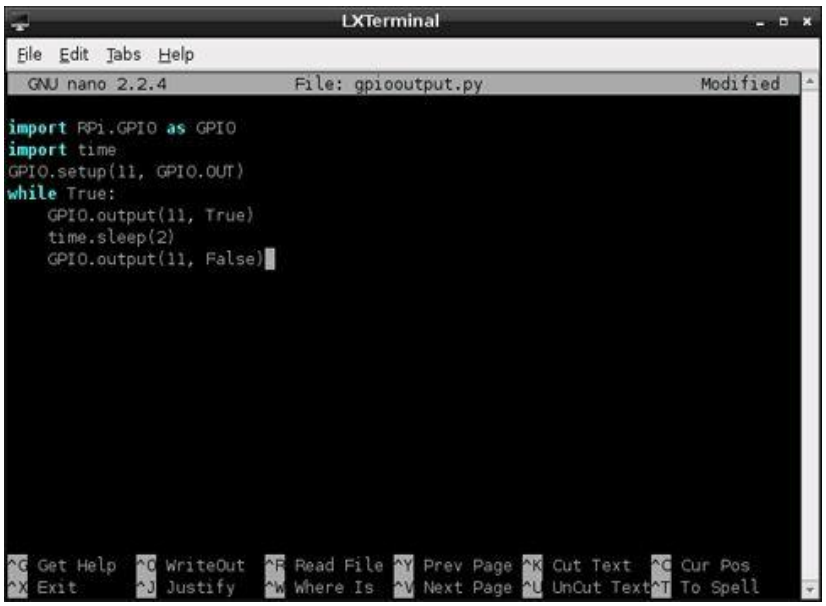
• Scalability without overwhelming the process.

• It gives cloud benefit, so there isn't any setup included.

• Data storage is based on native JSON that is simple to comprehend.

• Developer need not worry about nuances of storage, synchronization.

• It coordinates nicely with systems like Angular JS. So it's extremely valuable and permits you to make an application in a brief timeframe.



3.1.3 Protocols

For a IoT system to work, there must be a communication between the device nodes (D2D). Data from the devices needs to collated and sent to the IT/server infrastructure

(D2S). Possibly, servers may need to communicate with each other so connections between server infrastructures needs to be established so that sharing can happen (S2S), servers can possibly send it back to device nodes, or send them to other processing elements, or to the user itself (S2D).

Various protocols that are used are summarised as follows:

1. MQTT

MQTT, the Message Queue Telemetry Transport, targets collection of data from various device nodes. The prime motive is telemetry, or remote supervision and control. It provides the facility to collect data from various nodes and transmission of that data to IT infrastructure. It works effectively on huge networks that consist of constrained or small devices that are supervised or managed from the cloud.

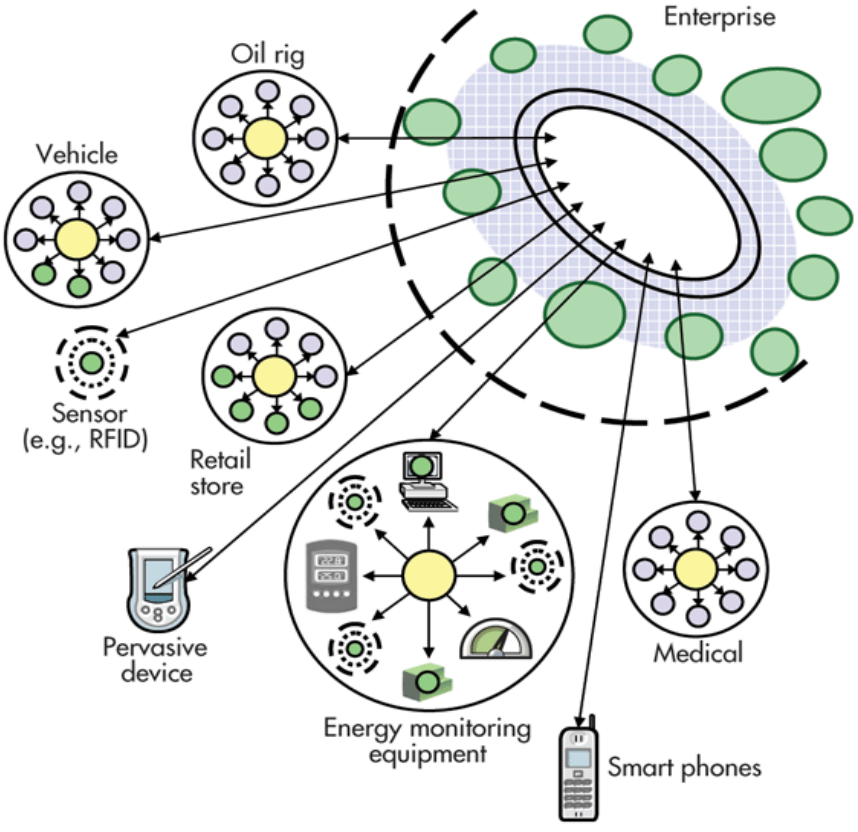


Figure 11: Message Queue Telemetry Transport[13]

Message Queue Telemetry Transport (MQTT) is primarily follows the architecture of hub-and-spoke. MQTT makes less effort to provide D2D transfers and fan out is not high in case of MQTT. It has a clear single application that does not offer many control options. Real time in case of MQTT is measured in units of seconds as it need not be fast. All the devices in the MQTT are connected to a data server (data concentrator to be specific). Since the protocol works on top of TCP, it facilitates a simple, reliable stream. This protocol is employed when the bandwidth is not too high.

2. XMPP

XMPP was earlier named as "Jabber." It originated for message oriented applications to connect communication nodes through text messages. Extensible Messaging and Presence Protocol is abbreviated as XMPP. XMPP is based on the XML text format as its indigenous format, providing simple and real P2P messaging. Both MQTT and XMPP runs over TCP, but XMPP can sometimes over HTTP (i.e. on top of TCP). The fundamental plus point of XMPP is its name@domain.com addressing scheme.

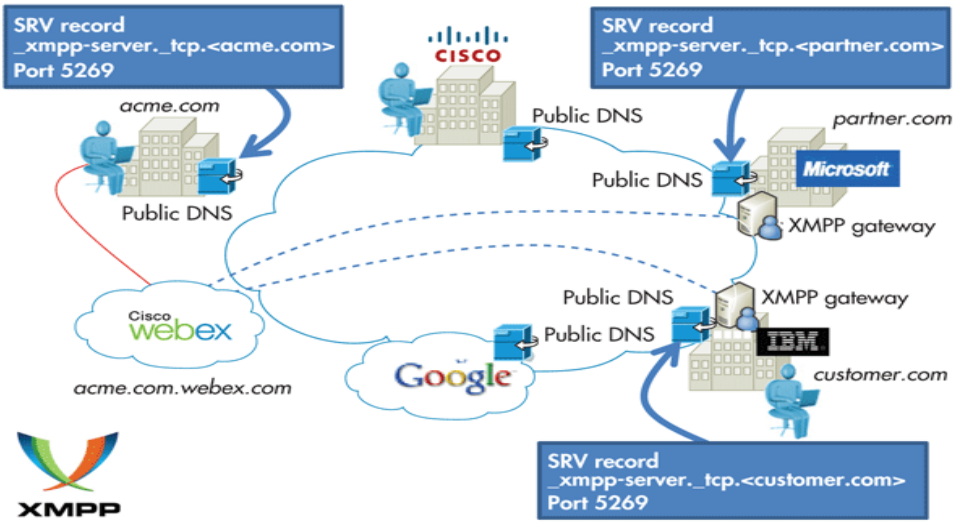


Figure 12: Extensible Messaging and Presence Protocol [13]

3. DDS

Data Distribution Service (DDS) is for those devices that directly use device data and this feature of DDS distinguishes it from MQTT and XMPP. As the name suggests, it distributes data to other nodes. The primary function of DDS is to establish communication/connection between various device nodes; however it provides a support system for interfacing with IT infrastructure. It is a middleware standard with utility in defence, industrial, and embedded applications that needs high performance, scalability, dependability, interoperability. DDS can effectively deliver tens of lakhs of messages in one second to many receiver devices in parallel. User can also specify service quality parameters. DDS provides services that far exceed those provided by publish-subscribe architecture.

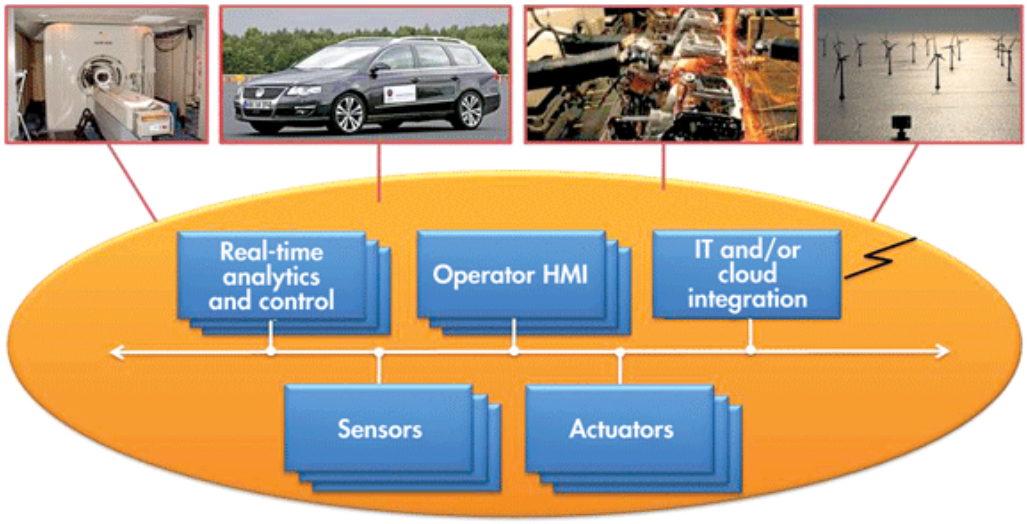


Figure 13: Data Distribution Service [13]

The main difference in terms of demand for data in IT infrastructure and devices comes from the fact that devices are comparatively much faster. “Real time” is defined in microseconds in the scenario of devices. There are multiple complex ways in which devices establish communication among themselves, so simplicity of TCP and its reliable point-to-point streams restricts D2D communications. On the other hand, quality-of-

service (QoS) control, multicasting capability, reliability, and redundancy provided by DDS are significant in complex network programming. Unlike XMPP, fan-out is a key advantage. DDS comes with powerful processes to filter and select exactly data destinations when the number of devices to which simultaneous data to be send is in thousands. For devices that work in constrained low capacity environments, lightweight versions of DDS are available.

4. Advanced Message Queuing Protocol

AQMP is an acronym for Advanced Message Queuing Protocol. As is evident from its name, AMQP is based on queues. Messages concerning the user-transactions are sent between servers using AQMP. It was first used in banking sector; and can process thousands of transactions based upon queues with increased reliability. AMQP is always striving on not losing track of messages. Communications between subscribers, exchanges and publishers happen over TCP that provides simple point-to-point connection that is reliable. Nodes must acknowledge the receipt of every communication stream/packet. It primarily originated from the banking industry thereby its middleware extensively targets upon real time tracking of messages and delivery assurance.

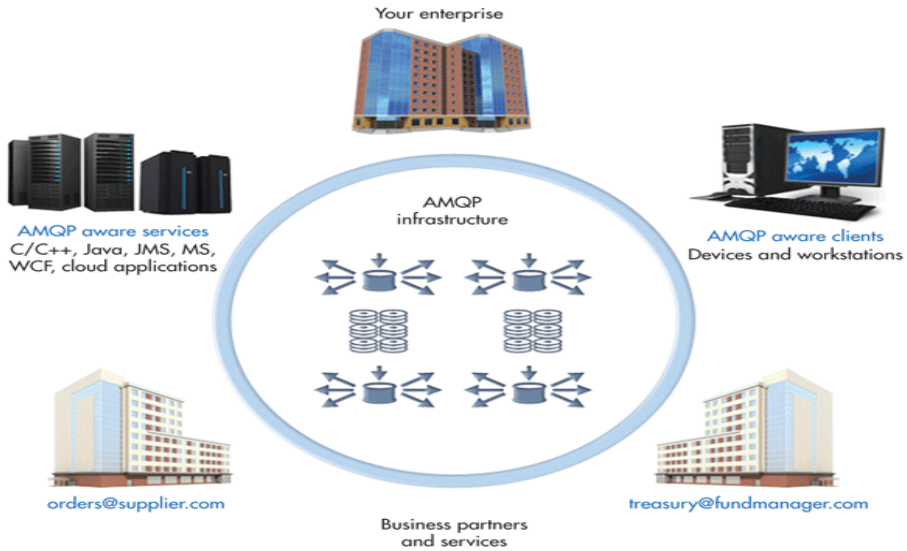


Figure14: AMQP

3.2 **Design**

3.2.1 Light control Module Design:

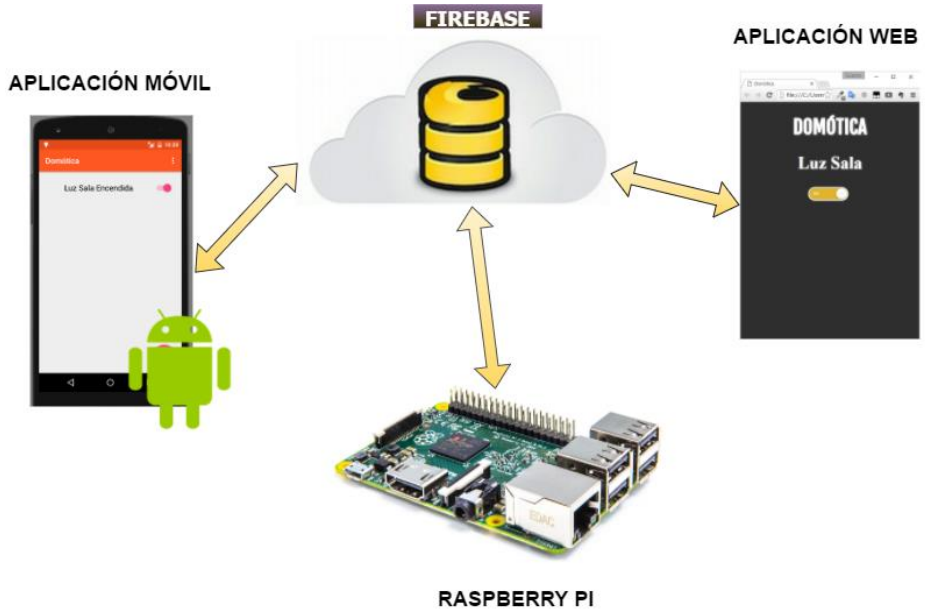


Figure 15: Light controller Module Design

3.3 Development:

i) **Light control circuit diagram:** Circuit diagram has Gpio pins connected to the relay switch. Neutral terminal of the power supply is connected to the device while live terminal of the power supply is connected to the relay switch. That is relay will decide whether device will be toggled on or off instead of the power supply.

Whenever user or gps changes the state in the firebase database, raspberry pi will read the changes, pass it on to the relay and relay will perform the action on the device.

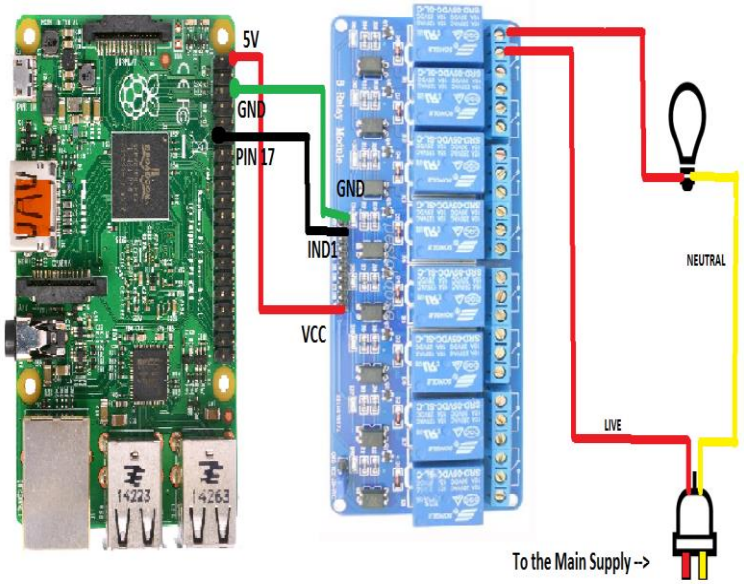


Figure 18: Light controller Circuit [11]

3.3.1 Algorithm:

1) User controlled lights Module Algorithm:

Label 1: while(1)

{

Read the GPIO pin

If(current status of GPIO pin changes)

Go out of the while loop

}

If (pin==true)

Turn the light on

Else

Turn the light off

Go to label 1

2) GPS controlled lights module Algorithm:

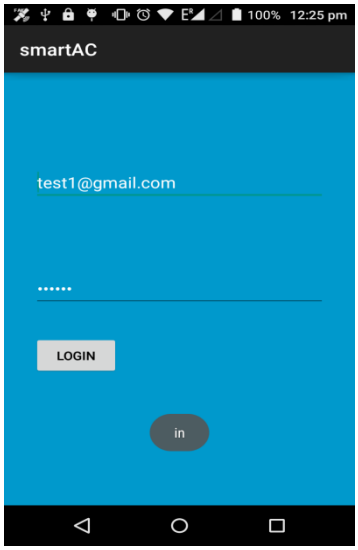
```
while(1)
{
Read the GPS coordinates and radius of the user
Compare the GPS coordinates with the coordinates saved in the database
If( both the coordinates system and radius are equal)
    {
       Turn the lights on
     }
Else
    {
      Turn the lights off
     }

}
```

4 PERFORMANCE ANALYSIS

4.1 Output Screenshots:

Step 1: User will enter his credentials and he will be directed to the next activity if firebase authentication successfully verifies his credentials.

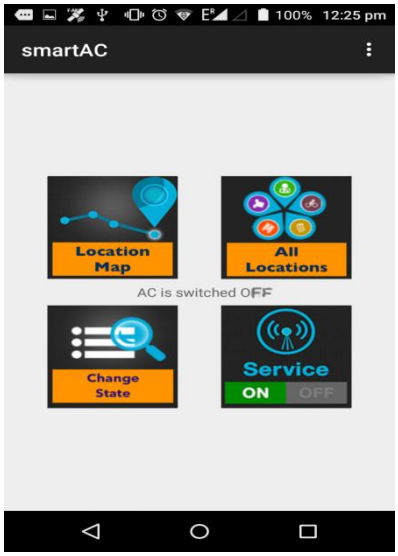


Step 2: User will be taken to the following page if his credentials are authenticated. Using "change state" he can manually turn on and off the devices connected to the pi.

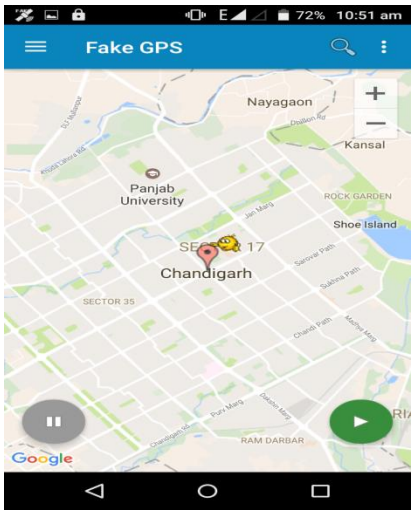
"All locations" contain the details of the gps locations that will be used to automatically turn on and off the devices if the user is within a stipulated diameter from the stored locations.

"Location Map" will contain the map.

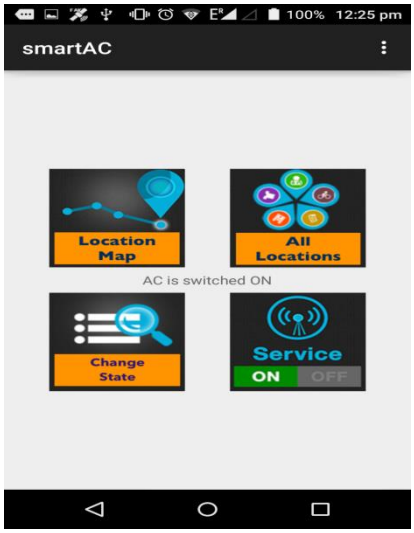
"Service" can be used by the user to turn the android service (that will keep on running in background if turned on) on and off.



Step 3: Using an app called “fake gps” set your location to the locations stored in the database



Step 4: Since the two locations matched, device will be triggered on and a message will be displayed on the user’s screen


100% 12:25 pm
smartAC
Location Map
All Locations
AC is switched ON
Change State
Service
ON
OFF

5. CONCLUSION

The home automation using Internet of Things has the potential to revolutionise home networking. It is experimentally proven to work satisfactorily when devices are controlled and connected successfully. The designed system monitors light and actuates a process according to the requirement, for example switching on the light. This will help the user to analyze the condition of various parameters in the home anytime.

Raspberry Pi can control devices from a longer range of distance as the connection which is to be used would be based on the internet. Due to which appliances, can be accessible easily. Home Automation system reduces the probability of human errors and can aid countries in reducing their carbon footprint.

The goal of saving energy and efficient use of energy can be achieved through IOT which is dependable, reliable and simplifies user side interaction. IoT can prove to be foundation for future artificial intelligence.

5.1 FUTURE WORK:

Using this system as framework, the system can be expanded to include various other options which could include home security feature like capturing the photo of a person entering and sending it to the owner through Whatsapp or implementing the door lock system. This will increase the security and enable to keep a check on his house from anyplace. There will be sensors that will work on real-time to unlock the door when an authorized person is within stipulated radius from the door. The system can be expanded for automation of various other devices at home.

5.2 SMART HOME EXPANSIONS

- **Wireless connectivity:** A hub which is wifi connected that is not having any physical connection to router can be a good enhancement, as this gives you more flexibility to choose an appropriate location for it in the house.

- **Scalability:** Hub must support a significantly large number of devices to achieve full automation.

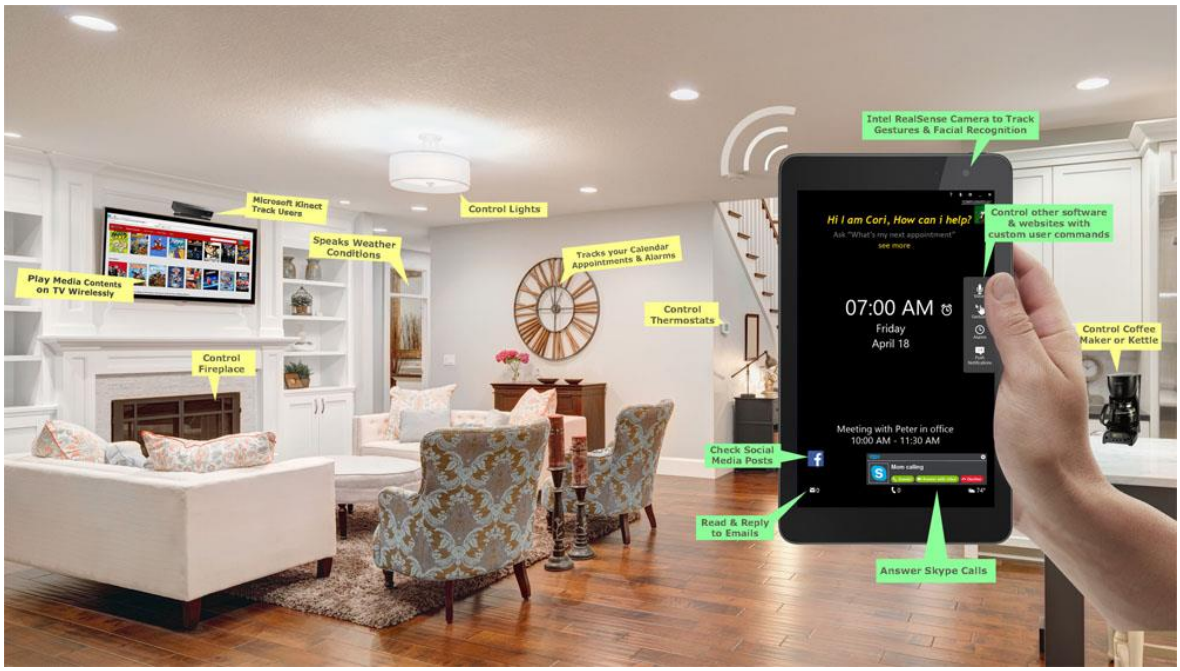


Figure 28: Smart Home

- **Compatibility test:** A hub that we choose to use must be able to support and offer compatibility to all the major protocols like Bluetooth, z wave, zigbee etc.
  Besides, compatibility with the devices around the house should be tested.

- **App control:** Web interfaces may look attractive but mobile applications are much easier to operate from the point of view of an user so making sure app compatible with all smart phones is developed.

- **Scheduling the devices and actions:** The system must possess the capability to draw schedules concerning various devices and create actions to connect these nodes.

- **Real time Alerts/Messaging:** The designed system should possess the ability to transmit the alerts when user actions are accomplished, for instance an alert message when someone opens main door of the house with her authenticated key.

REFERENCES

1. S.Praveen,"IOT and its Signifance ", 2015,Online.
Available: http://internetofthingswiki.com/internet-of-things-definition.

2. S.Prasad , P. Mahalakshmi "Smart Surveillance Monitoring System Using Arduino and PIR Sensor ",International Journal of Computer Science and Information Technologies, pp 45-65 ,Vol. 5 ,issue 1,2014.

3. Pyarie, R. Tyarize,"Bluetooth based home automation system using Iot", International Journal Of Computer Science and Information Technologies, pp 103-130,Vol 2 ,issue1,2013.

4. V Sagar, KN. Kusuma,"Home Automation through IOT ", International Research Journal of Engineering and Technology, pp 117-128, vol 2 ,issue 3 ,2015.

5. Chiemeziem ,E. Chukwubuikem, "Water Level Monitoring & Control Using Arduino Microcontroller Module", International Journal of Engineering Science and Technology ,pp119-122, vol 4,issue 3 ,2014.

6. Ramani, R. Olatunbosun , " Internet Of Things", International Journal Of Computer Science and Technology ,pp 120-145,vol 2 ,issue 3,2014.

7. Reza, K. Ahsanuzzaman,"Advance Research in Computer Science and Software Engineering ",pp 110-130,vol 3 ,issue 2 , 2013.

8. S.Mandeep ,"Arduino and Its Working ",2015,Online
Available : https://www.arduino.cc/en/main/arduinoBoardUno

9. M.Rawashed ,"Arduino and Bluetooth Connectivity ",2015,Online
Available: http://www.instructables.com/id/Arduino-AND-Bluetooth-HC-05

10. T.Mojidar ,"Arduino Temperature Sensor",2014,Online
Available:http://www.instructables.com/id/ARDUINO-TEMPERATURE-SENSOR-LM35/

11. Akellyirl ,"Light Sensor Arduino ",2015,Online
Available:http://www.instructables.com/id/How-to-Use-a-Light-Dependent-Resistor-LDR/

12. R.Sharma,"Water Level Sensor ",2014,Online
Available :http://www.instructables.com/id/How-to-Use-a-Light-Dependent-Resistor-LDR/

13. S.Schneider,"Various Protocols of Internet of Things ",2013,Online
Available:http://electronicdesign.com/iot/understanding-protocols-behind-internet-things

14. R.Dovina,"IR sensor with Arduino ",2013,Online
Available:http://www.instructables.com/id/Simple-IR-proximity-sensor-with-Arduino/

15. GreenPeak Technologies,2014,Online

Available : http://www.greenpeak.com/Application/SmartHome.html

6 APPENDICES

6.1 Code for Raspberry pi:

```
import RPi.GPIO as GPIO
import time
import pyrebase
import urllib2
import sys
GPIO.setmode(GPIO.BCM)
relay = 2
GPIO.setup(relay, GPIO.OUT)
config = {
  "apiKey": "apiKey""AIzaSyCMcEuZJEbI_hLTLYEMUHQSCmm_Rn16q_I",
  "authDomain": "testfire-7f970.firebaseapp.com",
  "databaseURL": "https://testfire-7f970.firebaseio.com",
  "storageBucket": "testfire-7f970.appspot.com"
}
global db def stream_handler(post):
    print(post["data"])
    if(post["data"] == "on"):
        GPIO.output(relay, GPIO.LOW)
        print "Ron"
    if(post["data"] == "off"):
        GPIO.output(relay, GPIO.HIGH)
        print "Roff"
```

```
def initdb():
 global db
 db = firebase.database()
 test = db.child("smarthome").child("state").stream(stream_handler)
def internetcheck():
   while True:
        try:
            res = urllib2.urlopen('http://google.com',timeout=1)
            return
        except urllib2.URLError:
            pass
#internetcheck()
print "Internet working"
firebase = pyrebase.initialize_app(config)
initdb()
while True:
       try:
       time.sleep(2)
     except KeyboardInterrupt:
        print "user terminated program"
        GPIO.cleanup()
        sys.exit()
```

6.2 Code for android app:

User registration Page

```
public class LoginReg extends AppCompatActivity implements View.OnClickListener {
```

```
 EditText e1, e2;
 Button b1,b2,b3;
 private static final String TAG = "MyAPP";



    private FirebaseAuth mAuth;
    private FirebaseDatabase firebaseDatabase = FirebaseDatabase.getInstance();
    private DatabaseReference mref;
    String lusername, lpassword;
@Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.loginreg);
     b1 = (Button)findViewById(R.id.button12);
        b1.setOnClickListener(this);
   b2 = (Button)findViewById(R.id.button13);
        b2.setOnClickListener(this);
e1 = (EditText)findViewById(R.id.editText13);
e2 = (EditText)findViewById(R.id.editText8);
mAuth = FirebaseAuth.getInstance();
mref = firebaseDatabase.getReference();
 e1.setText("test1@gmail.com");
 e2.setText("123456");
}
@Override
```

```
   //when the button is clicked
  public void onClick(View v) {
lusername = e1.getText().toString();
lpassword = e2.getText().toString();
 switch (v.getId())
  {
   case R.id.button13:
   createAccount(lusername,lpassword );
    break;
    case R.id.button12:
     boolean b= haveNetworkConnection();
              if(b==true) {
                  signIn(lusername, lpassword);
              }
              else
              {
                  buildAlertMessageNointernet();
                  signIn(lusername,lpassword)
 }
Break;
 }
 }
   private boolean haveNetworkConnection() {
       boolean haveConnectedWifi = false;
       boolean haveConnectedMobile = false;
```

```
   //android.net : classes that help with network access
//connectivity manager: Class that answers queries about the state of network connectivity
ConnectivityManager cm = (ConnectivityManager)
getSystemService(Context.CONNECTIVITY_SERVICE);


//Describes the status of a network interface.
   NetworkInfo[] netInfo = cm.getAllNetworkInfo();
   for (NetworkInfo ni : netInfo) {
      if (ni.getTypeName().equalsIgnoreCase("WIFI"))
         if (ni.isConnected())
            haveConnectedWifi = true;
      if (ni.getTypeName().equalsIgnoreCase("MOBILE"))
         if (ni.isConnected())
            haveConnectedMobile = true;
   }
   return haveConnectedWifi || haveConnectedMobile;
}
private void buildAlertMessageNointernet() {
   final AlertDialog.Builder builder = new AlertDialog.Builder(this);
   builder.setMessage("Your Internet Connection seems to be disabled, do you want to
enable it?")
         .setCancelable(false)
         .setPositiveButton("Yes", new DialogInterface.OnClickListener() {
            public void onClick(@SuppressWarnings("unused") final DialogInterface dialog,
@SuppressWarnings("unused") final int id) {
```

```
                startActivity(new Intent(Settings.ACTION_WIFI_SETTINGS));
            }
        })
        .setNegativeButton("No", new DialogInterface.OnClickListener() {
            public void onClick(final DialogInterface dialog, @SuppressWarnings("unused")
    final int id) {
                dialog.cancel();
            }
        });
    final AlertDialog alert = builder.create();
    alert.show();
  }
private void createAccount(String email, String password) {
mAuth.createUserWithEmailAndPassword(email, password)
        .addOnCompleteListener(this, new OnCompleteListener<AuthResult>() {
            @Override
            public void onComplete(@NonNull Task<AuthResult> task) {
                Log.d(TAG, "createUserWithEmail:onComplete:" + task.isSuccessful())
if (!task.isSuccessful()) {
                    Toast.makeText(LoginReg.this, "Authentication failed.",
                            Toast.LENGTH_SHORT).show();
                }
                else
                {
                    FirebaseUser user = task.getResult().getUser();
```

```
                    createUserDatabase(user);
                }
}
            });

    }

    private void signIn(String email, String password) {
mAuth.signInWithEmailAndPassword(email, password)
            .addOnCompleteListener(this, new OnCompleteListener<AuthResult>() {
                @Override
                public void onComplete(@NonNull Task<AuthResult> task) {
                    Log.d(TAG, "signInWithEmail:onComplete:" + task.isSuccessful());

                    if (!task.isSuccessful()) {
                        Log.w(TAG, "signInWithEmail", task.getException());
                        Toast.makeText(LoginReg.this, "Authentication failed.",
                                Toast.LENGTH_SHORT).show()
}
                    else
                    { Toast.makeText(LoginReg.this, "Authentication ok.",
                                Toast.LENGTH_SHORT).show();

                        Intent in = new Intent(LoginReg.this, MainActivity.class);
                        startActivity(in);
```

```
  }
}
                });
}
private void signOut() {
       mAuth.signOut();
        finish();
    }
```