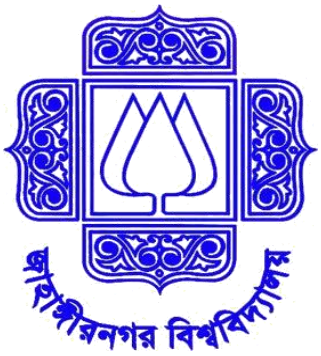

IoT based Smart Home System using Mobile Application

A project Submitted to the Computer Science and Engineering Department, Jahangirnagar University In partial fulfillment of the requirements for the award of the degree of Master of Science (MSc.) in Computer Science.

By

Ismith Pasha

ID. CSE 201702011

Supervised By

Dr. Md. Ezharul Islam

Assistant Professor

Department of Computer Science and Engineering,

Jahangirnagar University

July, 2018

ABSTRACT

"IoT based Smart Home System using Mobile Application" is based on IoT and Android platform. With this application anyone can control the electronics devices of his/her home from anywhere. A person can easily monitor the current status of electronic devices with his mobile phone using this application. He/she can turn on or turn off devices easily when it needs. The hardware part of the application is implemented with Arduino and different sensors like Bluetooth sensor, Relay module, GAS sensor, Buzzer, Ultrasonic sensor etc. The mobile application part is developed using Android Studio, Firebase Cloud System, API hosting with PHP server. The application can also track the locations of every person using GPS and update it in real time using Firebase Cloud. Every family members can know each other's location using this application. It will play a very important role for security when a user is outside from home. Emergency contact number, email such as hospitals, police stations etc. will be included with this application which may become very useful in emergency situations. This report provides the detailed description about the features and development process of the application.

Keywords

IoT, Internet of Things, Mobile Application, Smart Home.

Declaration

I hereby declare that this is submission is my own work and that to the best of my knowledge and belief it contains neither materials nor fact previously published or written by another person. Further, it does not contain material or fact which to a substantial extent has been accepted for the award of any degree of a university or any other institution of tertiary education except where an acknowledgment.

Signature of Candidate

……

Ismith Pasha

CSE201702011

Letter of Acceptance

The project entitled "IoT based Smart Home System using Mobile Application**"** submitted by Ismith Pasha, ID No. CSE201702011, to the PMSCS Program, Department of Computer Science and Engineering, Jahangirnagar University, Savar, Dhaka-1342, Bangladesh is accepted by the Department for the partial fulfillment of requirements for the degree of Master of Science (MSc.) in Computer Science on December, 2017.

Board of Examiners

_____

Dr. Md. Ezharul Islam
Associate Professor
Supervisor & 1st Examiner
Department of Computer Science and Engineering,
Jahangirnagar University, Savar, Dhaka-1342, Bangladesh

_____

Sarnali Basak
Assistant Professor
Supervisor & 2nd Examiner
Department of Computer Science and Engineering,
Jahangirnagar University, Savar, Dhaka-1342, Bangladesh

Accepted by:

_____

Prof. Md, Abul Kalam Azad
Professor,
Coordinator, PMSCS Program
Jahangirnagar University
Savar, Dhaka, Bangladesh

Acknowledgements

The application “IoT based Smart Home System using Mobile Application” would never run successfully without the valuable encouragement and guidance from my supervisor Dr. Md. Ezharul Islam, Associate Professor, Department of Computer Science and Engineering, Jahangirnagar University. He enlightened, encouraged and provided me with the ingenuity to transform my vision into reality. I am particularly grateful for his encouragement. I am also grateful to all my teachers and many of my friends who provided me with necessary hardware and suggestion to test the software as it developed. I would like to thank Google for creating android platform and keeping it free.

Table of Contents

Figures

Chapter 1

Introduction

1.1 Introduction

A smart home is a home where every part of the home is a part of modern world and latest technology. A smart home system is actually a home automation [1] system where different devices in the home is automatically controlled by modern technology. IoT or Internet of Things is now the buzzword of modern world. Every developed countries are now focused on developing IoT in their home and business area. In this project I am using the technology of IoT or Internet of Things and a Mobile Application. The mobile application will be the remote of the project which will control every systems of the project. With a smart phone like android mobile anybody who is in the system can control every electronic devices connected to this system. A real time database in cloud server will help the user in milliseconds to know if there is any change in the system.

To ensure a safe journey or a short travelling this application will be a very useful assistant. Every members registered with this system can track by this application where they are going or their current location. So if a member go out for travelling then the other members can know where he/she is right now and they reached their destination or not. In case of any emergency situation they can call the nearby police or hospital according to the situation. Members can do send short messages with the application without any cost.

My creativity in this project is to make a Smart Home System based on an Android Application that would be an efficient app for smart phone and also a very useful app for user. So I started working to create an android application for making a Smart Home System which can also become an assistant of every family members when they are outside of their home.

1.2 Motivations

The motivation for developing a Smart Home System [2] comes from many reasons as like security, modern lifestyle, smart communication etc. This is a new area of research and development in modern world. Developed countries are spending a lot of money and resources in this area. For our country a smart home system is a totally new system. It will save the time and also make the use of energy more efficiently. It will also make the life of user more safe and secure. A user who is registered with this system can easily monitor his home from outside that everything is going ok or not. Any accident like fire accident will notify the user immediately which may become a very important reason to save the properties. As this is an android based system so there is also some useful features which can help when the user is outside of home. A member in the system can easily know the other members location when they are outside of the home. This will make the life of a family tension free and easier.

1.3 Objectives and Methodology

1.3.1 Objectives

The main objectives of my application are make the life of a family with a smart home [6] system using a smart phone and to ensure the security of family members when they are in home of outsides of the home.

1.3.2 Methodology

To build this project both hardware and software combination will be needed. Different sensors will be used to detect different condition or situation of a different places in a home and if there is any change those sensors will send signals to a real time server using Arduino. A mobile application will catch the signal as a message and the user of the application will know what is happening to his home. A location monitoring system will be implemented in the mobile application and that will heap a family member to keep knowledge about the others.
The major components of the project is-

- Mobile Application (Android)
- Real Time Cloud Server (Firebase)
- Arduino Uno
- Different sensors.

1.4 Contribution

The way I followed to reach my goals –

- Collected the necessary information about Android and
- Learned Android Application Development.
- Collected necessary information about Firebase Cloud Technology.
- Learned Firebase cloud messaging and push notification technique.
- Collected information about IoT.
- Collected different devices, sensors and modules of IoT.
- Collected requirements for my project.
- Used PHP MYSQL database for web server database.
- Made all necessary diagrams of my project Data flow diagram.
- Tested my application and it passed in all the method I applied.

Chapter 2

Background and Related Work

2.1 What is Internet of Things (IoT)?

The Internet of Things (IoT) is a system of interrelated computing devices, mechanical and digital machines, objects, animals or people that are provided with unique identifiers and the ability to transfer data over a network without requiring human-to-human or human-to-computer interaction [3].

A thing, in the Internet of Things, can be a man with a heart screen embed, a homestead creature with a biochip transponder, a car that has worked in sensors to alarm the driver when tire weight is low - or some other normal or man-made protest that can be relegated an IP address and gave the capacity to exchange information over a system.

IoT has developed from the merging of remote advances, small scale electromechanical frameworks (MEMS), microservices and the internet. The merging has helped tear down the storehouse dividers between operational innovation (OT) and data innovation (IT), permitting unstructured machine-created information to be broke down for bits of knowledge that will drive changes.

Down to earth uses of IoT innovation can be found in numerous businesses today, including exactness horticulture, building administration, medicinal services, vitality and transportation. Availability alternatives for hardware architects and application designers dealing with items and frameworks for the Internet of Things include:

In spite of the fact that the idea wasn't named until 1999, the Internet of Things has been being developed for a considerable length of time. The principal internet apparatus, for instance, was a Coke machine at Carnegie Melon University in the mid 1980s. The developers could interface with the machine over the internet, check the status of the machine and decide if there would be a cool drink anticipating them, should they choose to make the trek down to the machine.

2.2 What is an IoT platform?

Internet of Things (IoT) stage [4] is the help software that interfaces edge equipment, passageways, and information systems to different parts of the esteem chain (which are by and large the end-client applications). IoT stages commonly handle continuous administration errands and information perception, which enable clients to robotize their condition. You can think about these stages as the agent between the information gathered at the edge and the client confronting SaaS or portable application.

IoT stages are often alluded to as middleware arrangements, which are the "pipes" of the IoT. For the most part, an IoT or M2M arrangement is a mashup of capacities from different merchants, which include:

- Sensors or controllers.
- A portal gadget to total and transmit information forward and backward to the information organize.
- A correspondences system to send information.
- Software for breaking down and deciphering information.
- The end application benefit, which makes a great part of the esteem.

These arrangements can all things considered be alluded to as the esteem chain of IoT.

2.3 What is Android?

Android is a mobile operating system developed by Google, based on a modified version of the Linux kernel and other open source software and designed primarily for touchscreen mobile devices such as smartphones and tablets [5].

At first Android is produced by Android Inc., which Google purchased in 2005, Android was disclosed in 2007, with the principal business Android gadget propelled in September 2008. The working framework has since experienced numerous significant discharges, with the present variant being 8.1 "Oreo", discharged in December 2017. The center Android source code is known as Android Open Source Project (AOSP), and is basically authorized under the Apache License.

Android is likewise connected with a suite of restrictive software created by Google, including center applications for administrations, for example, Gmail and Google Search, and in addition the application store and advanced conveyance stage Google Play, and related improvement stage. These applications are authorized by producers of Android gadgets affirmed under measures forced by Google, however AOSP has been utilized as the premise of contending Android biological systems, for example, Amazon.com's Fire OS, which use their own particular counterparts to the Google Mobile Services.

Android has been the raving success OS worldwide on phones since 2011 and on tablets since 2013. As of May 2017, it has more than two billion month to month dynamic clients, the biggest introduced base of any working framework, and starting at 2017, the Google Play store includes more than 3.5 million applications.

2.4 Version history

The version history [7] of the Android mobile operating system began with the public release of the Android beta on November 5, 2007. The first version, Android 1.0, was released on September 23, 2008. Android is developed by Google and the Open Handset Alliance, and it has seen a number of updates to its base operating system since the initial release.
Versions 1.0 and 1.1 were not released under specific code names. Android code names are dessert shop themed and have been in sequential order arrange since 2009's Android 1.5 Cupcake, with the latest significant rendition being Android 8.1 Oreo, discharged in December 2017.

List of Android version names [6]:

1. Cupcake
2. Donut
3. Eclair
4. Froyo
5. Gingerbread
6. Honeycomb
7. Ice Cream Sandwich
8. Android 4.2 Jelly Bean (API level 17)
9. Android 4.3 Jelly Bean (API level 18)

10. Android 4.4 Kit Kat (API level 19-20)
11. Android 5 Lollipop (API level 21-22)
12. Android 6 Marshmallow (API level 23)
13. Android 7 Nougat (API level 24-25)
14. Android 8 Oreo (API level 26)

2.5 Architecture of Android

Android is actualized as a product stack design comprising of a Linux piece, a runtime domain and relating libraries, an application structure and an arrangement of uses [8].

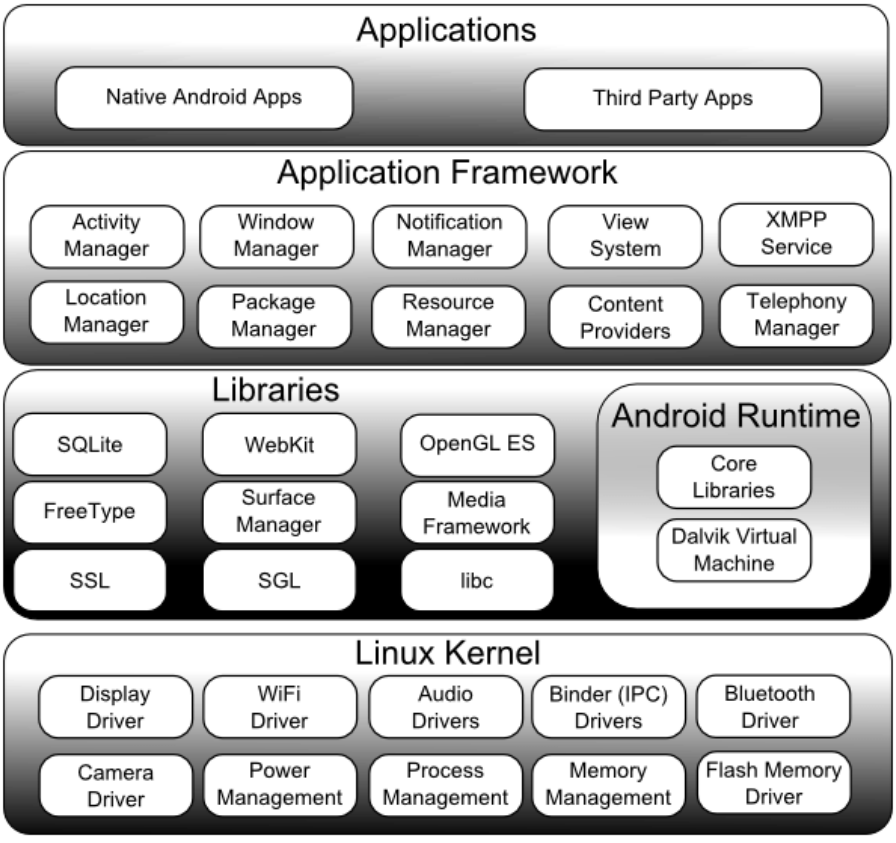

Figure 2.1: Android Architecture

The five components of Android architecture made:

There are five components [9] of Android architecture. They are -

1) Linux kernel

It is the core of android design that exists at the base of android engineering. Linux kernel is responsible for power management, device drivers, memory management, device management and resource access.

2) Native Libraries

On the top of linux kernel, there are Native libraries such as WebKit, OpenGL, Free Type, SQLite, Media, C runtime library (libc) etc.

The WebKit library is in charge of program bolster. SQLite is for database, Free Type for textual style bolster, Media for playing and recording sound and video groups.

3) Android Runtime

In android runtime, there are center libraries and DVM (Dalvik Virtual Machine) which is mindful to run android application. DVM resembles JVM however it is improved for cell phones. It devours less memory and gives quick execution.

4) Android Framework

On the highest point of Native libraries and android runtime, there is android system. Android structure incorporates Android API's, for example, UI (User Interface), communication, assets, areas, Content Providers (information) and bundle administrators. It gives a considerable measure of classes and interfaces for android application advancement.

5) Applications

On the highest point of android system, there are applications. All applications, for example, home, contact, settings, diversions, programs are utilizing android structure that utilizations android runtime and libraries. Android runtime and local libraries are utilizing linux kernal.

2.6 Application Fundamentals

Android applications are written in the Java programming language. The Android SDK tools compile the code-along with any data and resource files-into an Android package, an archive file with an .apk suffix [9].

All the code in a single.apk record is thought to be one application and is the document that Android-fueled gadgets use to introduce the application. Once introduced on a gadget, every Android application lives in its own particular security sandbox:

The Android working framework is a multi-client Linux framework in which every application is an alternate client.

Naturally, the framework allocates every application a one of a kind Linux client ID (the ID is utilized just by the framework and is obscure to the application). The framework sets consents for every one of the records in an application with the goal that exclusive the client ID alloted to that application can get to them.

Each procedure has its own particular virtual machine (VM), so an application's code keeps running in separation from different applications.

As a matter of course, every application keeps running in its own Linux procedure. Android begins the procedure when any of the application's segments should be executed, at that point close down the procedure when it's never again required or when the framework must recoup memory for different applications.

Along these lines, the Android framework executes the standard of minimum benefit. That is, every application, naturally, approaches just to the segments that it requires to do its work and no more. This makes an exceptionally secure condition in which an application can't get to parts of the framework for which it isn't given authorization. Be that as it may, there are routes for an application to impart information to different applications and for an application to get to framework administrations:

It's conceivable to orchestrate two applications to have a similar Linux client ID, in which case they can get to each other's records. To ration framework assets, applications with a similar client ID can likewise mastermind to keep running in a similar Linux process and offer the same VM (the applications should likewise be marked with a similar declaration). An application can ask for consent to get to gadget information, for example, the client's contacts, SMS messages, the mountable

capacity (SD card), camera, Bluetooth, and that's only the tip of the iceberg. All application consents must be allowed by the client at introduce time.

That covers the nuts and bolts in regards to how an Android application exists inside the framework. Whatever is left of this report acquaints you with:

The center system segments that characterize your application.

The show record in which you pronounce parts and required gadget highlights for your application.

Assets that are separate from the application code and enable your application to nimbly enhance its conduct for an assortment of gadget arrangements.

Chapter 3

Requirement Analysis

3.1 Requirements for implement the project

Below I discussed for the implementation of projects.

3.1.1 Functional Requirements:

The project smart home system [10] can be based on android, iOS and web, for end user it is implementing using android and iOS apps but in server side can be implement using Windows server, Linux Server. Additionally, we need a high configuration pc for android development and mobile phone for testing purpose also web server, domain, hosting for APIs. For hardware implementation we can use Raspberry Pi or Arduino and different sensors, modules and devices. I am implementing this project based on Android Operating System. I am using hosting server as Linux server. PHP is used for implementing API and MySql is used for database. To implement mobile application I am using Android Studio 3.0 and Java as programming language. To design different logo and icons I am using Adobe Photoshop CS6 version as photo editing tool. For sending and receiving data through API, I am using Volley library. With a domain mobile application can call the API and the API changes the value in the MySql database. For hardware implementation I am using Arduino and different sensors like Bluetooth sensor, Smoke sensor, Reed sensor, Buzzer, Ultrasonic sensor etc.

3.1.2 Nonfunctional Requirements:

Every system some weakness and strength my system has some weakness and strength below I am discussing this:

A. Weakness

The system is depending on android phone and some hardware devices. Without smart phone this project can’t run. For GPS information phone GPS has been used. It is server client application so it need internet connection for performing operation. Sometimes hardware devices can give wrong output due to device environmental fact. The hardware devices needs electric powers. So they always need to connected with electricity and internet.

B. Strength:

Nowadays almost everyone has a smart phone. So this project will be easy to use. Every user has his/her username and password so the apps is free from an authorize access. Also, the password is encrypted so user information is save in the server side. Server can handle multiple user at a time. IoT devices are very cheap in costs nowadays. This project can make the life easier than before.

Chapter 4

System Design and Architecture

4.1 Use Case Diagram

Use case diagram for a user, server and IoT environment. Here I state what is user can able to do and how server act with system, and users response. The use case diagram of the app is shown in Figure 4.1.

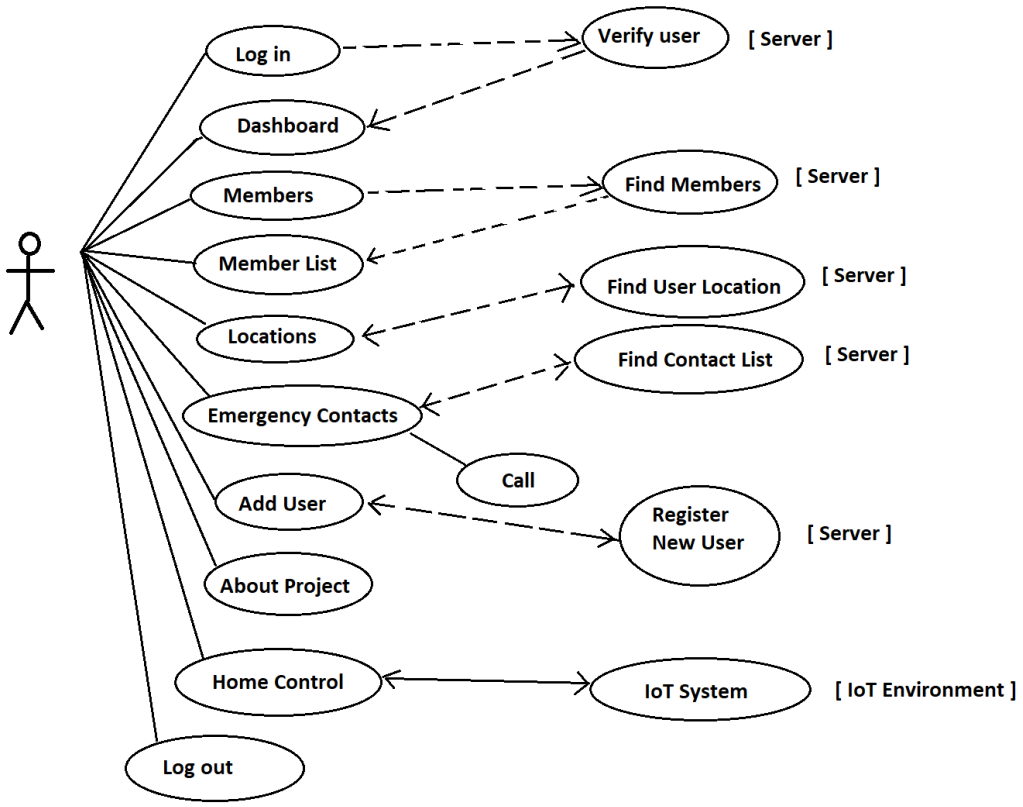

Figure 4.1: Use case diagram.

When the user open the mobile application, he/she will see the splash screen. The splash screen will check the application information and user status in the application. If the user is previously logged in, than the Home Screen will open after the splash screen. But, if the user not logged in, than the Login Screen will come. From the Login Screen the user will input the login credentials and send it to the server. The server will verify the request from the user. If the request is valid then the server will approve the user to login into the mobile application. After

successful login the user will see the Home Screen. From the home screen the user can any action using the application. For this the user need to send a request to the server. The server will verify the request and will response to the user according to the user request and action. For example if the user is in an emergency state and he/she needs to call a doctor than the user will open the Emergency contact list, and from there the user will find the nearest hospital's contact number and can call them immediately. Similarly the user can see the member list, can add new members, can call or send email a member from the member list, can find the locations of the members etc. The main service of the app is to control the electronic devices of the house. For this the user need to connect the mobile application with the IoT environment System. After connecting successfully the user can find the device list of the house which are connected in the IoT environment. The user can turn on or turn off any device from the mobile application. For example, if the user want to turn on the fan of his/her room than the user need to switch on the button of the fan from the mobile application. Then a signal will generated and send from the mobile to the Arduino. After getting the signal from the mobile application the Arduino will take action according to the signal.

4.2 Activity Diagram

The activity diagram of the project has shown in Figure 4.2 and in the Figure 4.2 every state of activity in the project is given there. A user can start his activity in the project by login into the mobile application with proper credentials (user id and password). If the credentials are valid than the user can access the application and will see the dashboard or home screen of the project. From the dashboard screen the user can take any action include in the application and those actions will change the activity state. If the user close the application from dashboard than the activity diagram will be finished there. If the user log out from dashboard page than the login screen will come again and ask for credentials. The user can view others members information by going to member list activity state. From this state the user can contact any member using phone call, sms or email. The user can also find the current locations of other members from the location activity state. The user can also add new members. Emergency contacts and emails of different institutions like hospitals, police stations etc. are accessible from emergency contact state. The user can call or email to the emergency contacts if it is necessary. The main functionalities of the project is home automation system. This part is implemented inside the home control activity. From home control activity the user have to connect with the IoT environment as

well as Arduino with the mobile app. Using Bluetooth connectivity the user can easily connect to the system. The Bluetooth module from Arduino will build a communication state between the mobile application and Arduino.

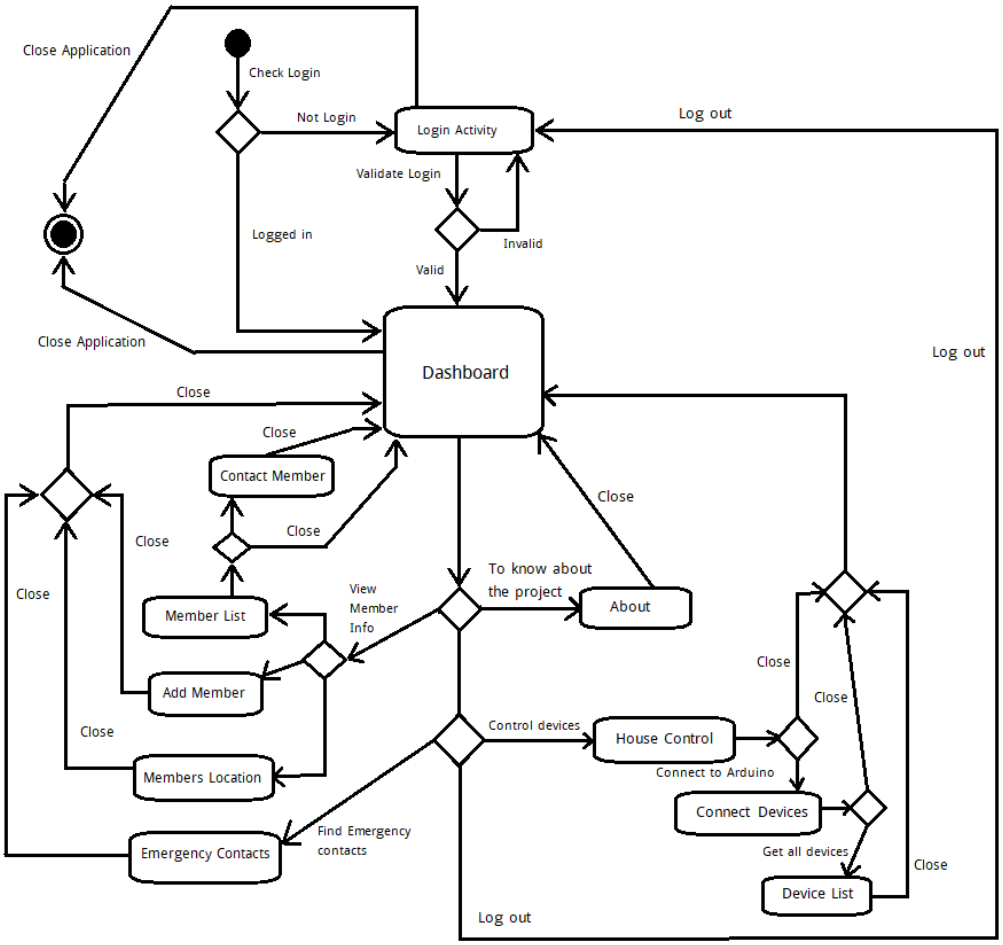

Figure 4.2: Activity Diagram.

After connecting with Arduino the user can find the devices which are connected with the IoT environment. From the list the user can turn on or can turn off any electronic device using the mobile application. If a user take any action to change the current state of an electronic device the Bluetooth sensor in the Arduino will get a signal and the Arduino will analyze the signal and will take the action. After completing any action the user close any activity then he/she will be back into the dashboard activity. From dashboard activity the user can close the application by clicking back button of the mobile phone.

4.3 Deployment Diagram

In this project the user send request to the server. The server analysis the user request and give a response to the user. Here is the deployment diagram of the project shown in Figure 4.3.

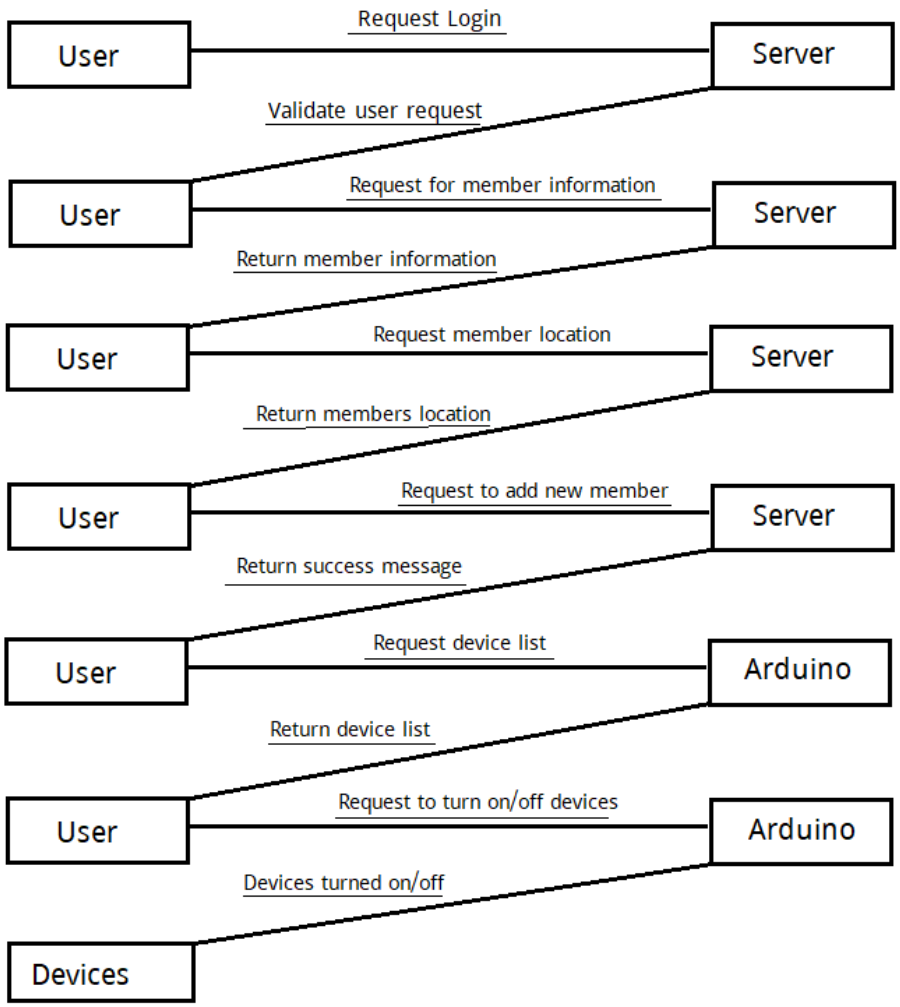

Figure 4.3: Deployment Diagram.

After opening the application the user will see the login screen activity. From the login screen the user can request for login using his/her user id and password. The request will send to the server and the will check the user is valid or not. If the user is not valid then the server will not approved the login. If the user credential is valid then the server will let the user to enter into the app. After successful log in a user can send different request and will return response according to those request. The user can view the list of all the members in the application. For this the app will send a request to the server through API and the server will return the list of

members and their primary information. If the user want to add more members than the user need to fill the necessary data and send a request to the server. The data will be saved inside the database of the server and the server will return success message as reply. The user can find out the other members location from the server. The user can send email, sms or call other users using the application. To control the electronic devices using the application the user need to connect the device with the Arduino. After connecting with Arduino the user can see the device list in the mobile application screen. Then the user can send request to turn on or to turn of the device. The targeted device will change according to the request.

4.4 Design and Architecture

In this project the user send request to the server. The server analysis the user request and give a response to the user. According to the server response the user can find different status and take actions.

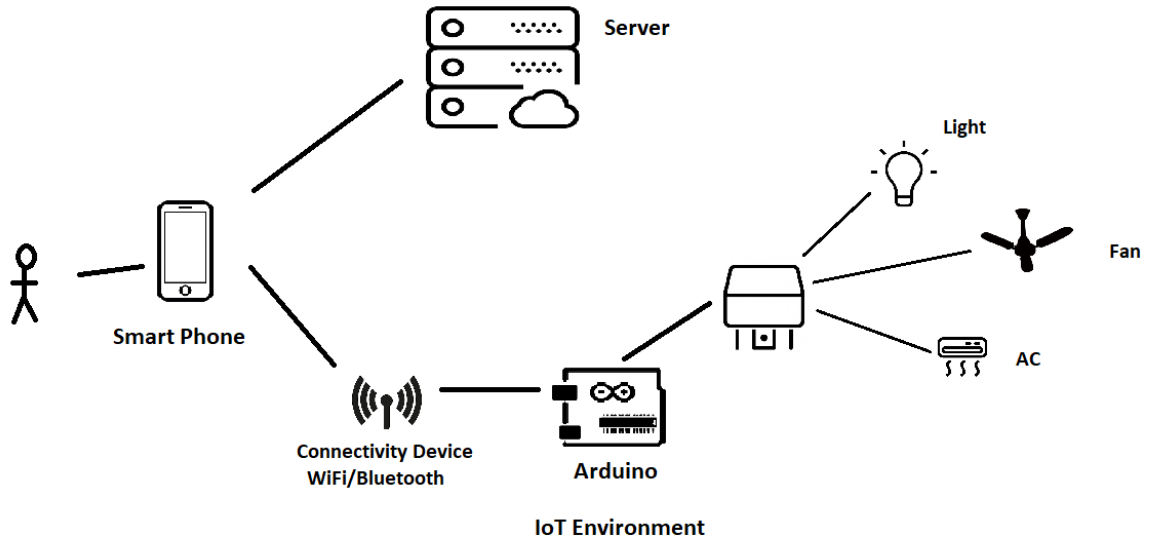

Figure 4.4: Design and Architecture.

In figure 4.4 the full architecture is shown. Here the user need to connect both server and IoT environment. As this project is just a demo version, I am using Bluetooth connection to connect with IoT environment. Using the mobile application the user can turn on/off any electronic device connected with the system through the Arduino. All the state and actions will updated in the server.

Chapter 5

System Implementation and Evaluation

5.1 Implementation

This project is both hardware and software based project. The hardware parts are the sensors and other devices which are implemented inside the house. After successful implementation of the hardware devices an online database server need to be implemented. Than the mobile Application part will be implemented.

5.2 Implementation Procedure

Creating a fair environment for android development and IoT implementation.
Here are some simple pre-requisites one must have to develop an android app and to implement the part of IoT.

5.3 Mobile Application Developer Requirement:

- Basic to advanced knowledge of java
- Basic knowledge of XML
- Basic to advance knowledge on Android Application development.
- Basic knowledge on Firebase Cloud Messaging.
- Knowledge on Android Studio.

5.4 Hardware Requirement for Mobile Application:

Development PC must be a faster one with minimum 8GB of RAM. I used an Intel Core i5 processor CPU with 16GB of RAM. A big monitor or two is also helpful.

5.5 IoT Development Requirement:

- Basic to advanced knowledge on Internet of Things
- Basic to advance knowledge on Arduino.
- Basic knowledge on Electronica Circuits.
- Basic knowledge on different sensors.

5.6 IoT Devices and Sensors:

Here are some devices and sensors I used for implementation of the project.

5.6.1 Arduino Mega 2560

Description:

The Arduino Mega 2560 [11] is a microcontroller board in view of the ATmega2560 . It has 54 digital input/output pins (of which 15 can be utilized as PWM yields), 16 simple sources of info, 4 UARTs (equipment serial ports), a 16 MHz precious stone oscillator, a USB association, a power jack, an ICSP header, and a reset catch. It contains everything expected to help the microcontroller, just interface it to a Computer with a USB link or power it with an AC-to-DC connector or battery to begin. The Mega is perfect with most shields intended for the Arduino Duemilanove or Diecimila. The Mega2560 highlights the ATmega8U2 customized as a USB-to-serial converter.

Figure 5.1: Arduino Mega 2560.

Specification:

- Microcontroller: ATmega2560
- Operating Voltage: 5V
- Input Voltage (recommended): 7-12V
- Input Voltage (limits): 6-20V
- Digital I/O Pins: 54 (of which 15 provide PWM output)
- Analog Input Pins: 16
- DC Current per I/O Pin: 40 mA
- DC Current for 3.3V Pin: 50 mA

- Flash Memory: 256 KB of which 8 KB used by bootloader
- SRAM: 8 KB
- EEPROM: 4 KB
- Clock Speed: 16 MHz

Purpose of use:

The Arduino Mega 2560 is the mainboard of the project. Each and every hard ware devices, sensors and modules will be connected with this and it will send the reports to the server.

5.6.2 Relay Module 5V

Description:

Multiple 5V Relayes [12] together in a single board to control multiple loads simultaneously. The relays can be easily switched by microcontroller or Arduino.

Figure 5.2: Multiple 5V Relayes.

Specification:

- Rated coil Voltage: 5V DC
- Operating Frequency:50/60HZ
- Nominal Current:10A AC, 10A at 28V DC
- Maximum Switching Voltage:250V AC, 28V DC.
- Power Required by the board:Vcc=5V DC(For relay coils),5V DC(For energizing every relay individually)

Purpose of use:

Relayes are using for controlling multiple electronic devices like light, fan, ac etc. It is use as like a remote controlled switches.

5.6.3 Reed Switch

Description:

Reed Switch [13] is a small device. When the device is exposed to a magnetic field, the two ferrous materials inside the switch pull together and the switch closes. When the magnetic field is removed, the reeds separate and the switch opens. This makes for a great non-contact switch. This switch can carry up to 1.2A.

Figure 5.3: Reed Switch.

Purpose of use:

Reed Switchs will be used for tracking door and windows opening and closing. It will send a signal when the door is opening or closed.

5.6.4 Battery 9V

Description:

Light portable 9V battery [14] come handy in lots of project where mobility is main concern.

Figure 5.4: Light portable 9V battery.

Specification:

- Length 48mm and width 26mm
- Height 17mm and weight 35gm

Purpose of use:

To power the Arduino and other devices.

5.6.5 ESP8266 ESP-01 Wifi Module

Description:

ESP8266 [15] is a small powerful wifi chip that has so many features.

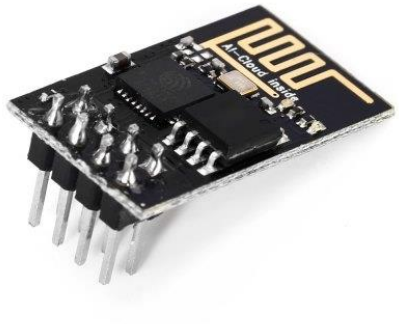

Figure 5.5: ESP8266 wifi module.

Specifications:

- 802.11 b/g/n
- Wi-Fi Direct (P2P), soft-AP
- Integrated TCP/IP protocol stack
- Integrated PLLs, regulators, DCXO and power management units
- +19.5dBm output power in 802.11b mode
- Power down leakage current of <10uA
- Integrated TR switch, balun, LNA, power amplifier and matching network
- 1MB Flash Memory
- Integrated low power 32-bit CPU could be used as application processor
- SDIO 1.1 / 2.0, SPI, UART
- STBC, 1×1 MIMO, 2×1 MIMO
- A-MPDU & A-MSDU aggregation & 0.4ms guard interval
- Wake up and transmit packets in < 2ms
- Standby power consumption of < 1.0mW (DTIM3)

Purpose of use:

ESP8266 is using to send information to the server through internet with a wifi connection.

5.6.6 DT-9205A Digital Multimeter

Description:

9205A Digital Multimeter [16] with measures AC and DC voltage, DC current, resistance, electrical capacitance, diode/transistor check etc.

Figure 5.6: 9205A Digital Multimeter.

Specification:

Function	Range	Accuracy
DC Voltage	200mV-1000V	±(0.5%+1dgt)
AC Voltage	200mV-700V	±(0.8%+3dgt)
DC Current	2mA-20A	±(0.5%+1dgt)
AC Current	2mA-20A	±(1.0%+3dgt)
Resistance	200Ω-200MΩ	±(0.8%+1dgt)
Capacitance	2000pF-200μF	±(4.0%+3dgt)
Continuity Buzzer	Yes	
Auto Power Off	Yes	
Data Hold	No	
Power Supply	9V Battery	
Maximum Display	1999	
Product Net Weight	300g	
Product Size	191*89*35mm	

Purpose of use:

9205A Digital Multimeter is using to measure the voltage and current.

5.6.7 Buzzer 5V To 12V

Description:

Buzzer [17] generates a continuous beep usually when supplied with power. It can generate any tone by interfacing it with a micrcontroller with proper coding.

Figure 5.7: Buzzer.

Specification:

- Type: Magnetic Transducer, Self Drive (External Drive with Feedback)
- Sound Pressure Level:82dB
- Mounting Type: Through Hole
- Frequency: 2.3kHz
- Packaging: Tray
- Operating Mode: Continuous
- Voltage Range: 2 ~ 5VDC
- Voltage - Rated: 3VDC
- Current Rating: 30mA
- Size / Dimension: Circular - 12mm Dia x 7.5mm H
- Termination Solder: Pins/Posts
- Lead Free Status: Lead Free
- RoHS Status: RoHS Compliant

Purpose of use:

Buzzer is using for giving a simple alarm if it needed.

5.6.8 HC-SR04 Ultrasonic Sonar Sensor

Description:

This is a ultrasonic [18] distance sensor provides a non-contact distance measurement with a fine accuracy of 3mm. The module is capable of measuring distance from obstacles in range of 2 cm to 400 cm or 1 inch to 3 Ft. Unlike IR sensors, the performance of this sensor is no dependent on the sunlight or color of the object.

Figure 5.8: HC-SR04 Ultrasonic Sonar Sensor.

Specification:

- Power Supply :+5V DC
- Working Current: 15mA
- Effectual Angle: < 15 degree
- Ranging Distance : 2cm - 400 cm/1 inch - 13ft
- Resolution : 0.3 cm
- Measuring Angle: 30 degree
- Trigger Input Pulse width: 10uS

Purpose of use:

This sensor is using for measuring the distance of an object. In this project it is using to measure the water level inside the water tank.

5.6.9 Jumper Wires

Description:

These jumper wires [19] are 7" long. They have 0.1" male header on both end. They can be installed next to each other on standard pitch 0.1" header. The wires can all be used together or easily peeled apart individually or in sections. They fit snug next to each other on a breadboard, when placed with the silver side out because the side with the silver showing through the black connector is standard breadboard spacing (2.54mm) compared to 2.25mm on the other side.

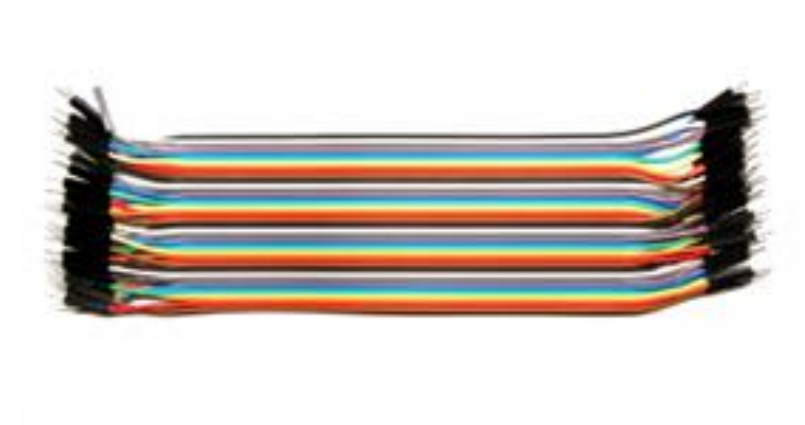

Figure 5.9: Jumper Wires – male to male.

Specification:

- Male header contacts on both ends
- Male to female header contacts only one end.
- Standard 0.1" (2.54mm) spacing when placed next to each other
- Length: 200mm (7.87")
- Wire Colors: brown, red, orange, yellow, green, blue, purple, grey, white, black (each cable includes 4 of each color)
- Fits breadboard
- Weight: 31g (1.1oz)

Purpose of use:

To connect sensors and devices with each others.

5.6.10 MQ-2 Smoke/LPG/CO Gas Sensor

Description:

The Analog Smoke/LPG/CO Gas Sensor(MQ2) [20] module utilizes an MQ-2 as the sensitive component and has a protection resistor and an adjustable resistor on board. The MQ-2 gas sensor is sensitive to LPG, i-butane, propane, methane, alcohol, Hydrogen and smoke. It could be used in gas leakage detecting equipment in family and industry. The resistance of the sensitive component changes as the concentration of the target gas changes.

Figure 5.10: MQ-2 Smoke/LPG/CO Gas Sensor.

Specification:

- Power supply needs: 5V
- Fast response and High sensitivity
- Wide detecting scope
- Simple drive circuit,Stable and long life

Purpose of use:

To protect home from fire this sensor will give a signal if there is smoke in house..

5.7 Android Application Development

For developing application, "Android Studio" and it is the official IDE. There are also other IDEs. But Android Studio is best for development because it is the latest IDE and supported by Google. It is more comfortable to work with both Java classes and GUI for android in Android Studio than other IDEs. In this project I am using Android Studio because it is regularly updated with new tools and features and can develop applications for all the devices supported with android operating system like handsets, tablets, TV, and wear devices.

5.8 Project Setup

Android Studio makes it easy to create Android apps. Some beginning steps are described here.

5.8.1 Start a new project

The New Project wizard lets you choose the form factors for your app and populates the project structure with everything you need to get started.

Step 1: Start and configure the project

If you don't have a project opened, Android Studio shows the Welcome screen, where you can click Start a new Android Studio project.
If you do have a project opened, click **File > New > New Project**.
In the Configure your new project screen that appears, you can set the name of your app, the package name, and the location of your project. If you want to use C++ and/or Kotlin code in your project, check the corresponding checkboxes to set up your project accordingly. You can always add C++ code and add Kotlin code later.
Enter the values for your project then click **Next**.

Step 2: Select form factors and API level

The next window lets you select the device form factors you want to build for, and the minimum version you want to support for each. For each device you select, the wizard adds a corresponding module to your project.
Each module contains all the code and resources that will be built into an Android app package (APK) for the corresponding device. If you later decide to add support for a new device, you

can add a module at that time. And you can share code and resources between modules using an Android library.

For more information about the Android project structure and module types, read Projects Overview.

To see more information about the different Android versions, click Help me choose. This opens a new window that shows the distribution of devices running each version of Android. Click on an API level to see a summary of top features introduced in that version. To return to the wizard, click **OK**. Once you've selected your form factors and API versions, click **Next**.

Step 3: Add an activity

The next screen lets you select an activity type to add to your app, as shown in figure 3. This screen displays a different set of activities for each of the form factors you selected earlier.

Choose an activity type then click **Next**.

Note: If you choose "Add No Activity," click Finish to create the project.

Step 4: Configure your activity

The next screen lets you configure the activity to add to your app. Enter the activity name, the layout name, and the activity title. Then click Finish. Android Studio now sets up your project and opens the IDE.

Now you are ready to develop your app. If you're new to Android development, start with Getting Started on Android.

5.9 System Evaluation

This application is been developed in android studio. Some screen shoots of the application is given and described here.

5.9.1 Splash Screen

After opening the mobile app by clicking the app icon then the Splash Screen opened. This page used to check the status of a user weather the user is already logged in or not.

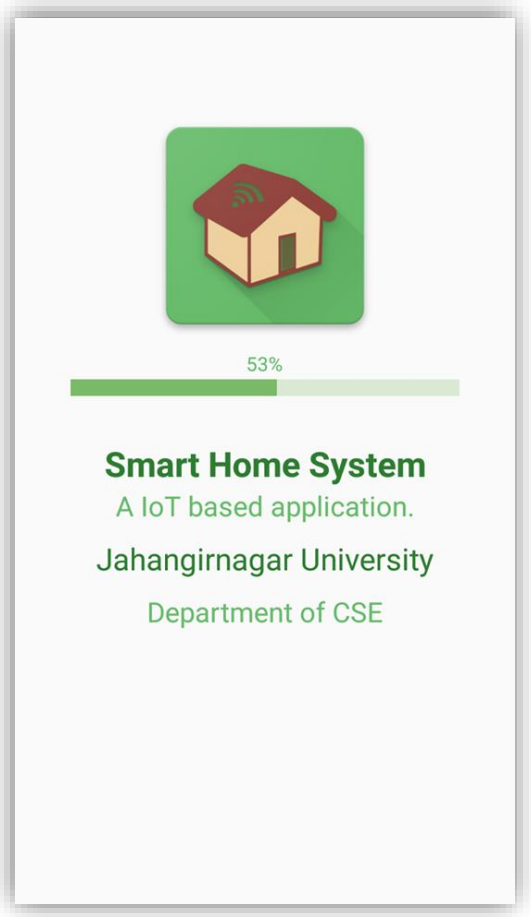

Figure 5.11: Splash Screen.

If the user is logged in then the next page screen be the Home Screen. If the user is not logged in then the Screen will be the Login Screen.

5.9.2 Login Screen

If the user is not logged in than the Login Screen will opened.

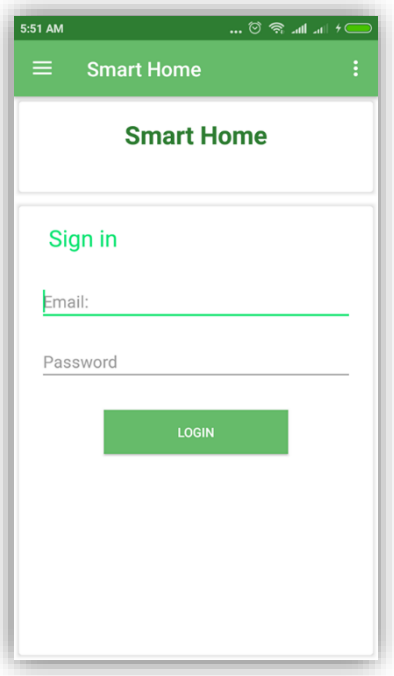

Figure 5.12: Login Screen

5.9.3 Home Screen

After successfully login the user will find the Home Screen. In Home Screen the user see the main dashboard of the application.

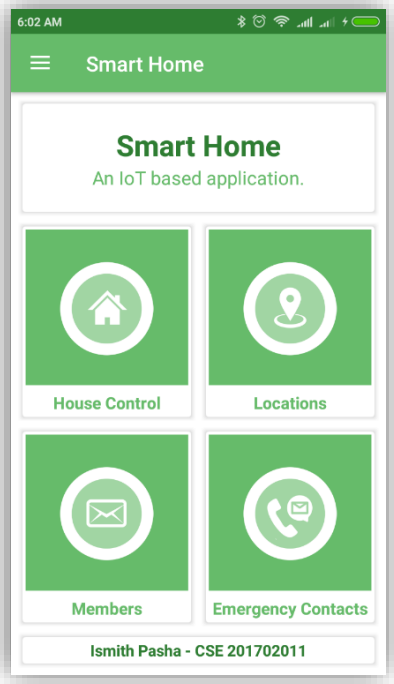

Figure 5.13: Home Screen.

5.9.4 Navigation Menu

The Navigation Screen is used for finding the other Screens of the application.

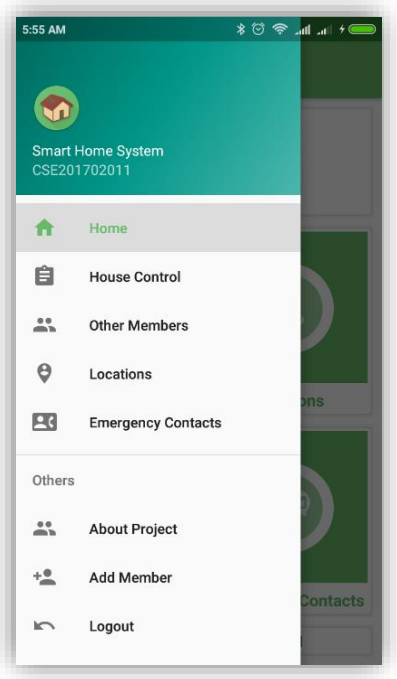

Figure 5.14: Navigation Screen.

5.9.5 Home Control Connection

This stage works for connecting IoT device with mobile phone.

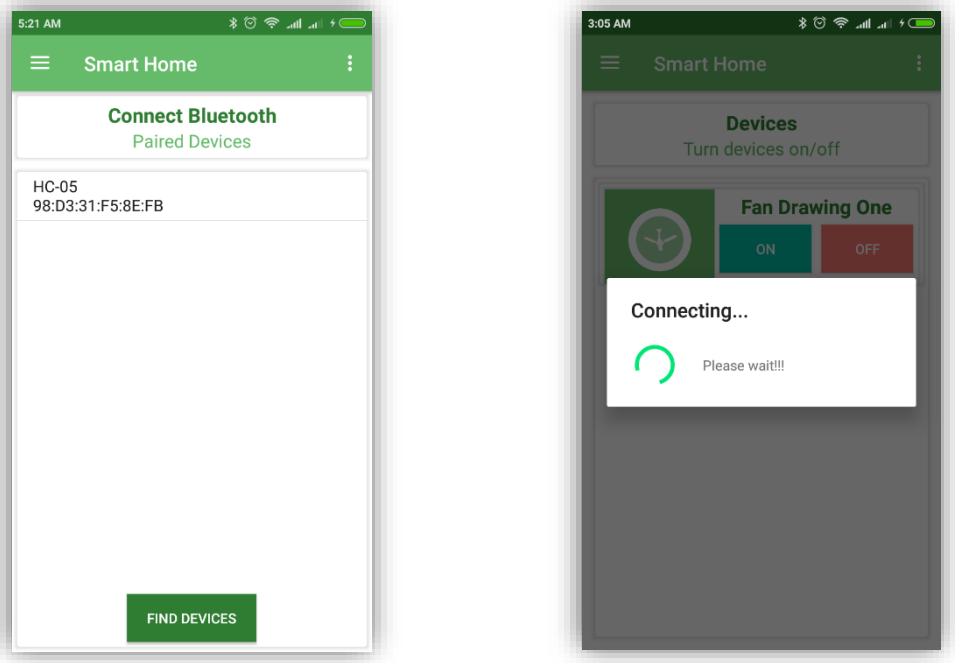

Figure 5.15: Connectivity State.

5.9.6 Home Control

This is the page to control the Home from mobile app.

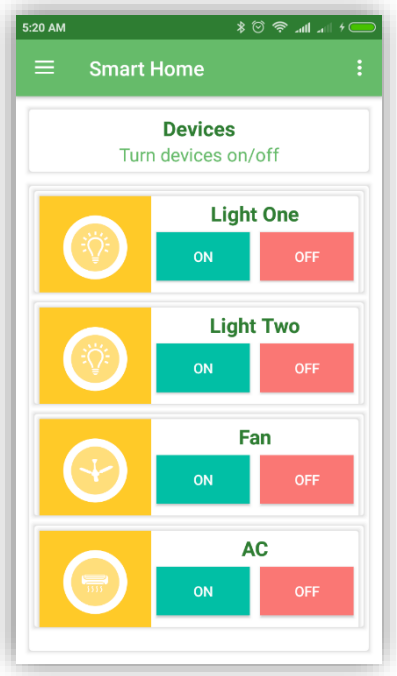

Figure 5.16: Home control.

5.9.7 Add Members

To add a new member admin user have to go to the page.

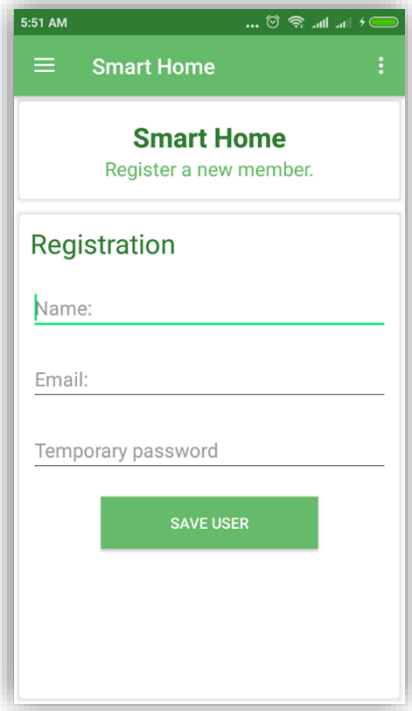

Figure 5.17: Add new member.

5.9.8 Members

To see all the members and their information. And to connect with members.

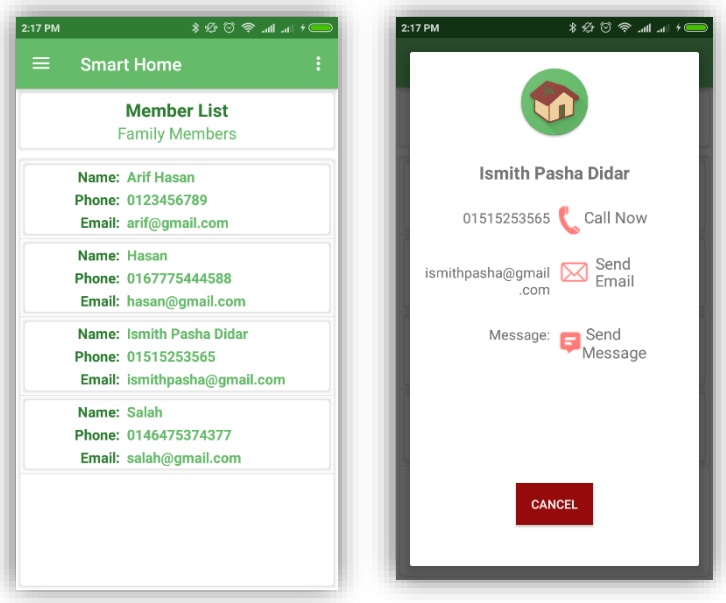

Figure 5.18: Members.

5.9.9 Location

To know the members current location.

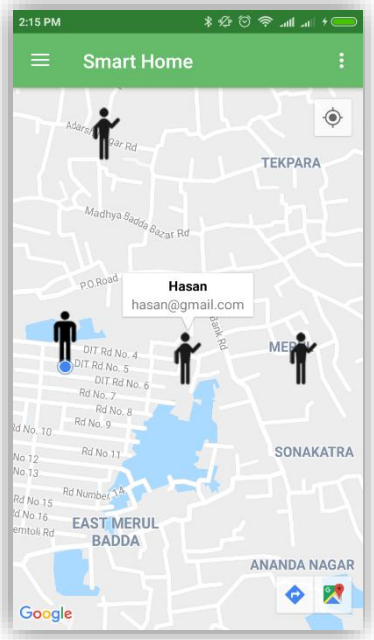

Figure 5.19: Member locations.

5.9.10 Emergency Contacts

To find emergency contact list when necessary.

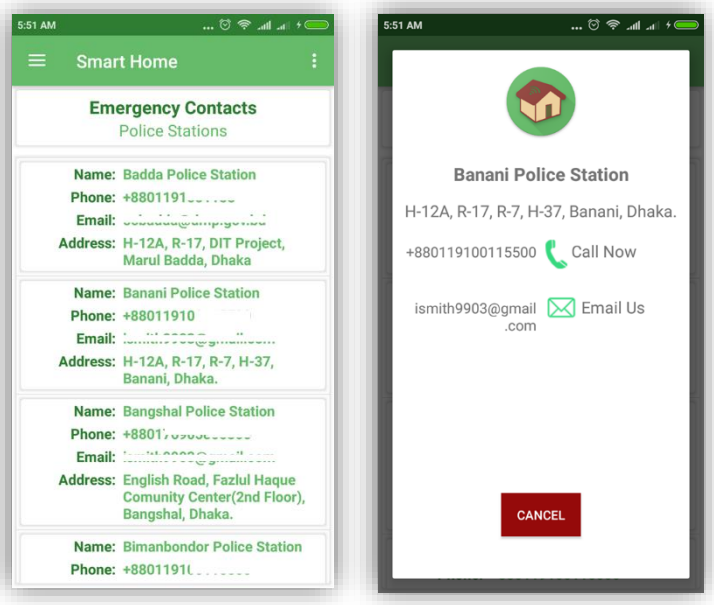

Figure 5.20: Emergency contacts.

5.9.11 About

To know about the project.

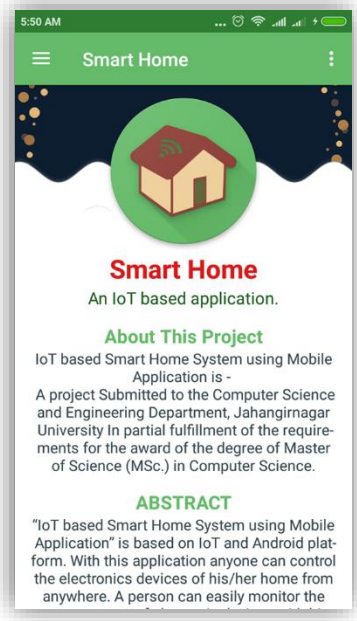

Figure 5.21: About.

5.9.12 Log out

To signing out from the project.

5.10 Hardware Implementation

The hardware part is the most important part of the project. First different sensors are connected and implemented with Arduino. Then it is powered with battery and the project works successfully.

5.10.1 Setting up Arduino and Other sensors

First different sensors has been connected with Arduino using jumper wires and breadboard. Every sensors are implemented with appropriate circuit connections. Different sensors like Bluetooth, Buzzer, Gas sensors, Relay etc. are connected properly according to the planning.

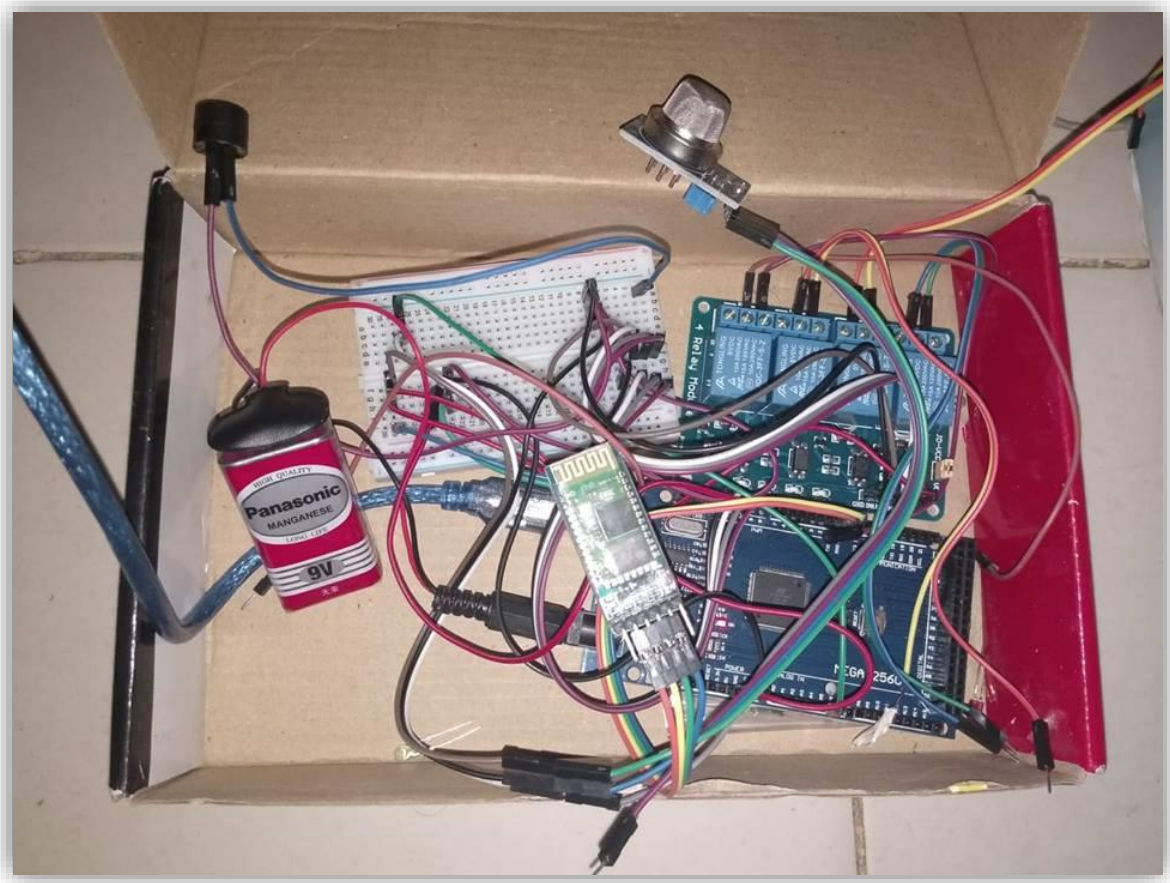

Figure 5.22: Hardware Implementation.

When the hardware has been successfully implemented the Arduino is powered with 9V battery. Then it was connected with mobile application and the project works successfully. In Figure 5.22 the hardware implementation has been shown.

5.10.1 Result of the Project

Here is a mini version of the project. A light and a fan inside a small room is constructed which is shown in Figure 5.23. The hardware part is beside the demo room. When the Arduino is connected with the mobile application and a user turn on or turn off light or fan the light and fan inside the room is also turned on and off.

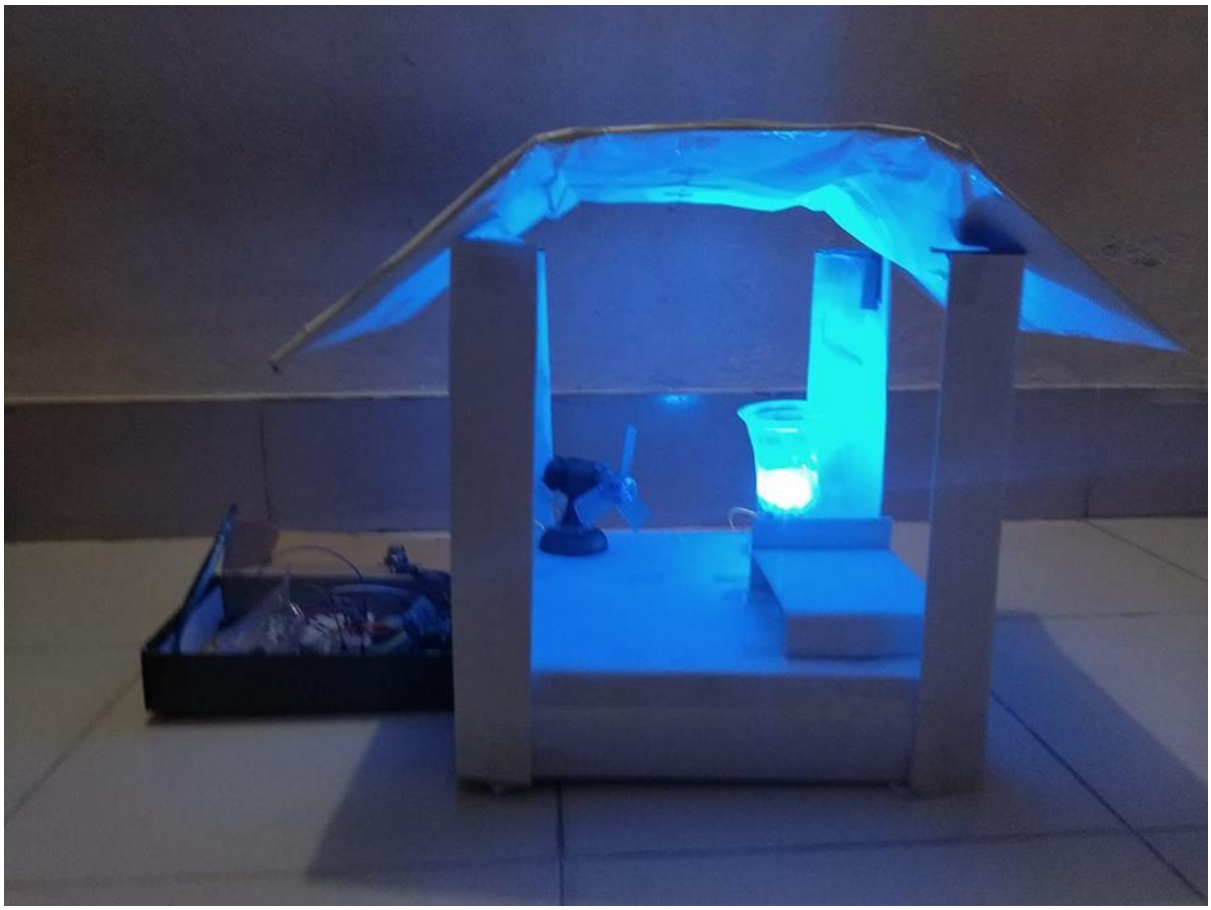

Figure 5.23: Mini project.

Chapter 6

Conclusion and Future work

6.1 Conclusion

Internet of Things IoT is bringing a huge revolution in modern world. Everyone in modern world are welcoming this new technology. Smart Home System or home automation is one of the most important applications of Internet of Things. It makes the daily life easier and more comfortable. In this Smart Home System project, IoT is implemented based on an android mobile app. This app make the system more comfortable and easy to control. This project is mainly developed with Arduino because it has less cost than others.

This project also has the features of tracking the location of family members during their travelling or if they are outside from home. It will help the family members to ensure they safely reached or not. There will be all emergency contact information with this application. So, if a user feels something wrong he/she can directly call or email to the emergency contacts. This project is not just a Home automation project, this is also covering some safety during travelling or outside from home.

6.2 Future Work:

Nowadays IoT is having a tremendous attention to the people and its various projects are changing our lifestyle continuously. The main focus of the project is to build a smart and comfortable home automation and to ensure the security safety. At this moment this is a very small project. But in future there will be added more features in this project. Some of the future features are –

- Fire alarm, which can detect if there is any fire or smoke in the home.
- Anti-theft system to prevent thief.
- Automatic device turn on/off using the GPS.
- Road and travelling safety.

Appendix

References:

[1] Muhammad Asadullah and Ahsan Raza, "An Overview of Home Automation Systems", *Department of Electrical Engineering National University of Computer and Emerging Sciences Peshawar, Pakistan, October 06, 2017,* accessed: June 06, 2018.

[2] Satish Palaniappan, Naveen Hariharan, Naren T Kesh, Vidhyalakshimi S, Angel Deborah S, "Home Automation Systems - A Study", *CSE, SSN College of Engineering, Anna University, Chennai, India, April 2015,* accessed: June 07, 2018.

[3] Internet of things (IoT), Internet: https://internetofthingsagenda.techtarget.com/definition/Internet-of-Things-IoT, June 2018, accessed: June 07 2018.

[4] What is an IoT platform?, Internet: https://www.link-labs.com/blog/what-is-an-iot-platform , August 03, 2016, accessed: June 07 2018.

[5] Android (operating system), Internet: https://en.wikipedia.org/wiki/Android_(operating_system) , June 06, 2018, accessed: June 07 2018.

[6] Martin Lärka, "Smart homes with smartphones", *Department of Applied Physics and Electronics, UMEA UNIVERSITY, June 24, 2015,* accessed: June 11, 2018.

[7] Android version history, Internet: https://en.wikipedia.org/wiki/Android_version_history, June 12, 2018, accessed: June 12 2018.

[8] An Overview of the Android Architecture, Internet: https://www.techotopia.com/index.php/An_Overview_of_the_Android_Architecture, October 27, 2017, accessed: June 19, 2018.

[9] What are the five components of Android architecture made up?, Internet: https://www.quora.com/What-are-the-five-components-of-Android-architecture-made-up, September 20, 2016, accessed: June 19, 2018.

[10] Kaniz Fatema Tuly, "A Survey on Novel Services in Smart Home (Optimized for Smart Electricity Grid)", *Norwegian University of Science and Technology, Department of Computer and Information Science, June 12, 2017,* accessed: June 22, 2018.

[11] Arduino Mega 2560?, Internet: https://www.techshopbd.com/product-categories/boards/1452/arduino-mega-2560-china-techshop-bangladesh, June 2018, accessed: June 25, 2018.

[12] 4 Channel 5V Relay Module, Internet: https://www.techshopbd.com/product-categories/modules-98775/1447/4-channel-5v-relay-module-techshop-bangladesh, June 2018, accessed: June 25, 2018.

[13] Reed Switch, Internet: https://www.techshopbd.com/product-categories/switch/852/reed-switch-techshop-bangladesh, June 2018, accessed: June 26, 2018.

[14] Battery 9V, Internet: https://www.techshopbd.com/product-categories/battery/646/battery-9v-techshop-bangladesh, June 2018, accessed: June 26, 2018.

[15] ESP8266 Al Cloud Inside, Internet: http://bdspeedytech.com/index.php?route=product/product&product_id=1979, June 2018, accessed: June 27, 2018.

[16] DT-9205A Digital Multimeter, Internet: https://www.techshopbd.com/product-categories/meters/1063/dt-9205a-digital-multimeter-techshop-bangladesh, June 2018, accessed: June 27, 2018.

[17] Buzzer, Internet: https://www.techshopbd.com/product-categories/buzzers/177/buzzer-techshop-bangladesh, June 2018, accessed: June 28, 2018.

[18] HC-SR04 Ultrasonic Sonar Sensor, Internet: http://bdspeedytech.com/index.php?route=product/product&product_id=1380, June 2018, accessed: June 28, 2018.

[19] Male-Male Jumper Wires, Internet: http://bdspeedytech.com/index.php?route=product/product&product_id=1729, June 2018, accessed: June 30, 2018.

[20] MQ2 Smoke/LPG/CO Gas Sensor, Internet: http://bdspeedytech.com/index.php?route=product/product&product_id=228, June 2018, accessed: June 30, 2018.

[21] Mohamed Hisham Moubarak, Dr. Amr Desouky, “Internet of Things for HomeAutomation”, *Media Engineering and Technology FacultyGerman University in Cairo, Mar 10, 2017,* accessed: July 03, 2018.