TABLE OF CONTENTS

Chapter 1: Introduction to the project domain

The domain of our project is Home Automation. Home automation is building automation for a home, called a smart home or smart house. A home automation system will control lighting, climate, entertainment systems, and appliances and hardware. It may also include home security such as access control and alarm systems. When connected with the Internet, home devices are an important constituent of the Internet of Things. A home automation system typically connects controlled devices to a central hub or "gateway". The user interface for control of the system uses either wall-mounted terminals, tablet or desktop computers, a mobile phone application, or a Web interface, that may also be accessible off-site through the Internet.

Problem Definition

This is a simple project made with the arduino uno,the aim is to open or close a door lock without keys and using a smart phone and a bluetooth module to accomplish this. The communication medium will be the internet i.e wifi module-ESP8266 or NODE MCU. IN this the admin will receive a notification from the services and will accordingly given the response to open or close the door. The automated door locking and unlocking system is really important because through an automated system, you can lock your doors with the tap of your finger. This quickly eases your mind, so you can focus on your day's work rather than who may or may not be entering your home. This is also a great benefit for you if you have to leave to work before your children leave for school. Often, children run out the door to catch the bus and forgot to

lock the door. Not only this you can even allow your friends ,relatives and family members to enter your house even when you are not there.

Project Requirements

Hardware Components Used:

- wifi module-ESP8266
- Servo Motor
- Bluetooth Module HC06
- Jumper Wires
- Push Button
- Bread Board
- Ardino board

Software Requirement:

- Ardino IDE

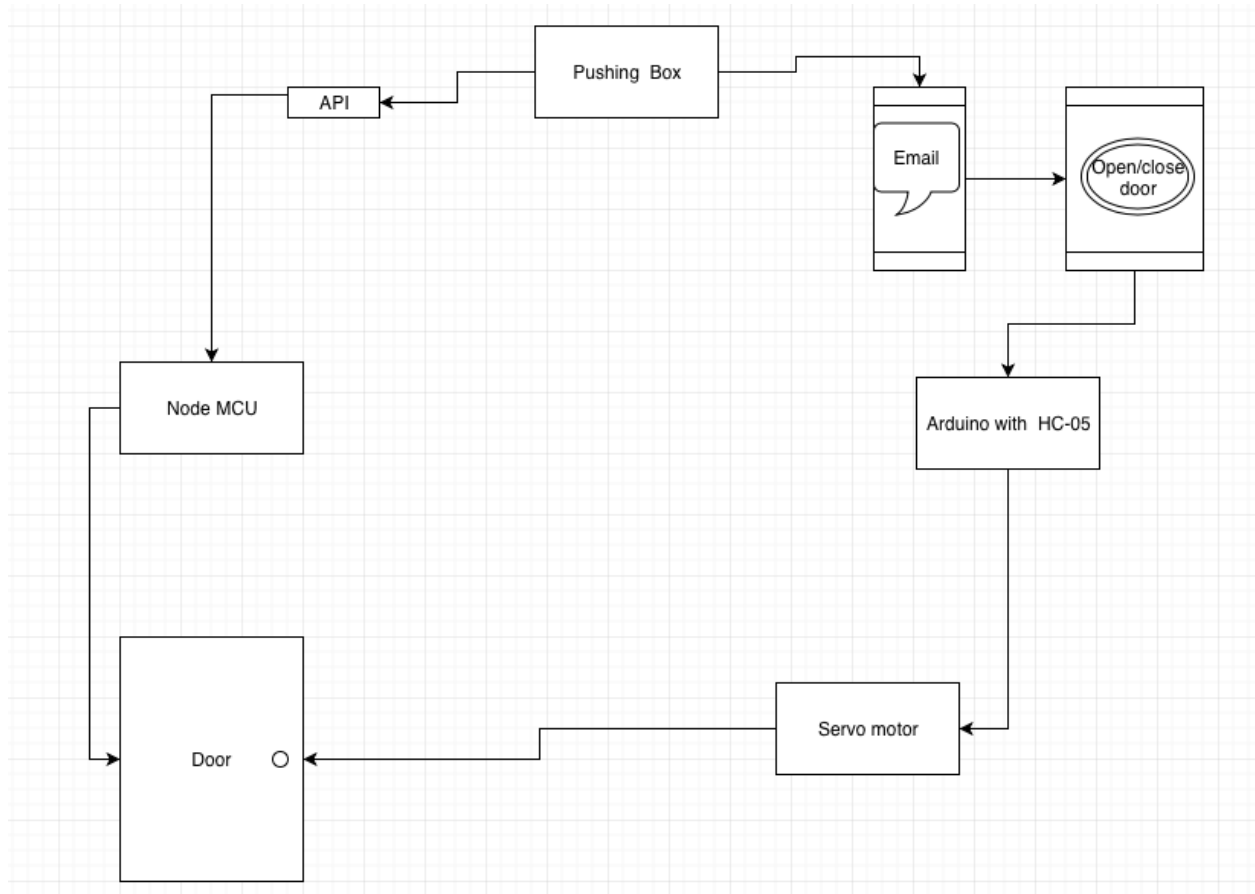# Chapter 2:Design and Implementation

Architectural Diagram:

Fig1: Schematic Architectural Diagram
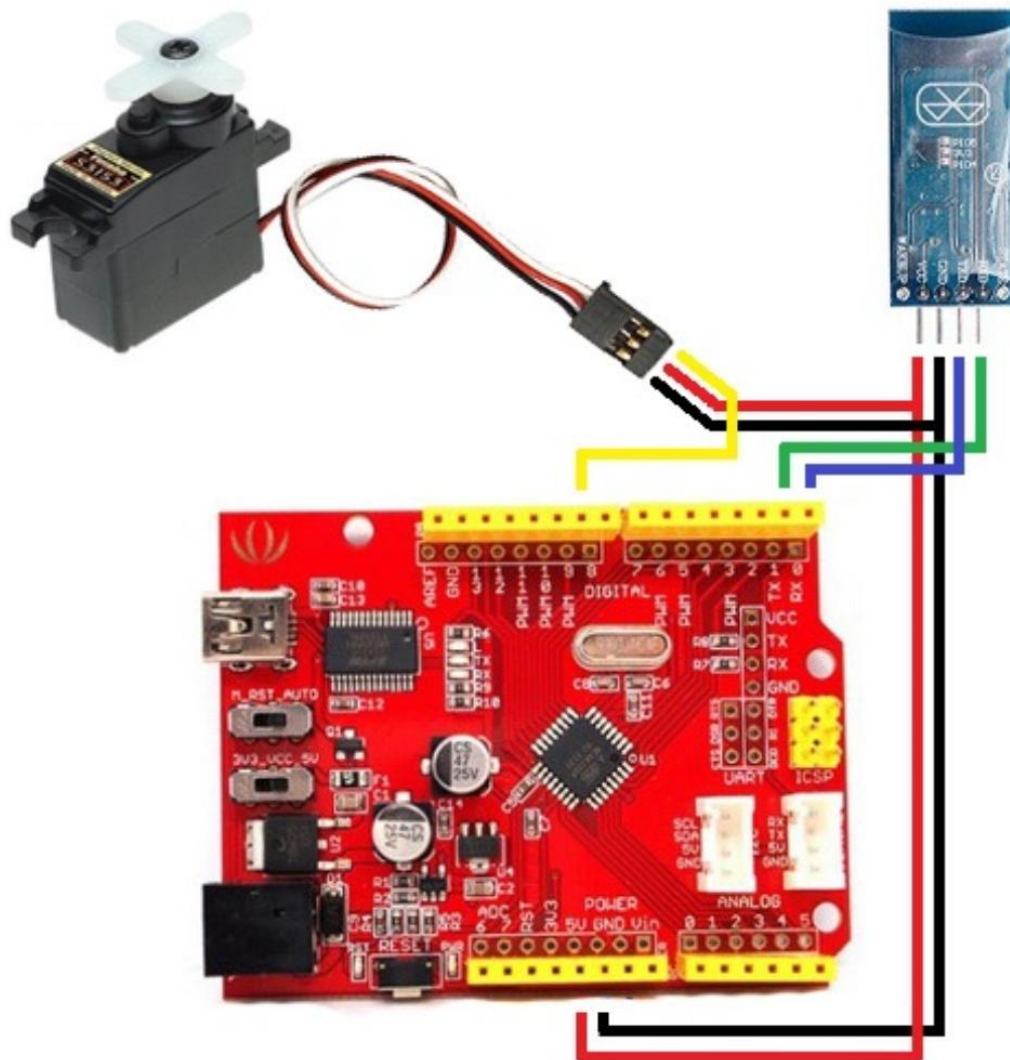
533P Internet of Things

Pin Diagrams:

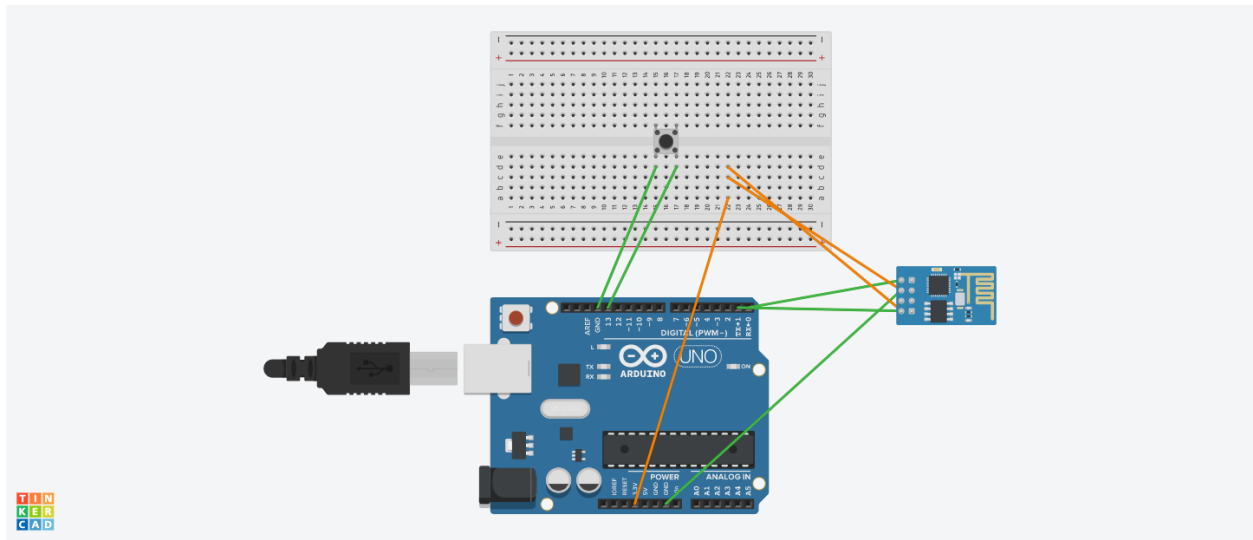Fig2: Bluetooth module and servo motor configuration

Fig3: wifi module and push button used as door bell

Overall Design

The circuit uses the following components:

1.Arduino UNO

The Arduino Uno is an open-source microcontroller board based on the Microchip ATmega328P microcontroller and developed byArduino.cc.The board is equipped with sets of digital and analoginput/output(I/O) pins that may be interfaced to various expansion boards(shields) and other circuits. The board has 14 Digital pins, 6 Analog pins, and programmable with theArduino IDE(Integrated Development Environment) via a type BUSB cable. It can be powered by the USB cable or by an external9-volt battery

2.Breadboard

A breadboard is a solderless device for temporary prototype with electronics and test circuit designs. Most electronic components in electronic circuits can be interconnected by inserting their leads or terminals into the holes and then making connections through wires where appropriate.

3.Connecting wires

Connecting wires allows an electrical current to travel from one point on a circuit to another because electricity needs a medium through which it can move. Most of the connecting wires are made up of copper or aluminum.

4.Bluetooth Module

The HC-06 is a class 2 slave Bluetooth module designed for transparent wireless serial communication. Once it is paired to a master Bluetooth device such as PC, smart phones and tablet, its operation becomes transparent to the user. All data received through the serial input is immediately transmitted over the air. When the module receives wireless data, it is sent out through the serial interface exactly at it is received. No user code specific to the Bluetooth module is needed at all in the user microcontroller program.

5.ESP8266 WiFi Module

ESP8266 is Wi-Fi enabled system on chip (SoC) module developed by Espressif system. It is mostly used for development of IoT (Internet of Things) embedded applications.It employs a 32-bit RISC CPU based on the Tensilica Xtensa L106 running at 80 MHz (or overclocked to 160 MHz). It has a 64 KB boot ROM, 64 KB instruction RAM and 96 KB data RAM. External flash memory can be accessed through SPI.

6.Servo Motor

A servo motor is an electrical device which can push or rotate an object with great precision. If you want to rotate and object at some specific angles or distance, then you use servo motor. It is just made up of simple motor which run through servo mechanism. If motor is used is DC powered then it is called DC servo motor, and if it is AC powered motor then it is called AC servo motor. We can get a very high torque servo motor in a small and light weight packages. Doe to these features they are being used in many applications like toy car, RC helicopters and planes, Robotics, Machine etc.

7.Push Button

A push button is a simple type of switch that controls an action in a machine or some type of process. Most of the time, the buttons are plastic or metal. The shape of the push button may conform to fingers or hands for easy use, or they may simply be flat. It all depends on the individual design. The push button can be normally open or normally closed.

DRAWBACKS:

- Additional features like webcam or voice recognition can be added.

- Range of bluetooth is less.

Chapter 3 :Testing/Result and Analysis

The system has been tested and found to be consistent.

Test Scenarios:

Test Scenario 1: Check if the Arduino board is functional.

Test Scenario 2: Check if the Bluetooth module is functioning.

Test Scenario 3: Check if the Wifi module is functioning.

Test Scenario 4: Check if the Servo Motor is Working.

Test Scenario -1: Check if the Arduino board is functional.

Test Case	Description	Test Steps	Test Result
1	Check if the Arduino board is functional.	Connect the board to USB port of the computer.	LED light on the arduino board should glow.

Test Scenario-2: Check if the Bluetooth module is functioning.

Test Case	Description	Test Steps	Test Result
1	Check if the Bluetooth module is functioning.	Connect Bluetooth according to the circuit.	LED light on the Bluetooth should glow.

Test Scenario-3: Check if the wifi module is functioning.

Test Case	Description	Test Steps	Test Result
1	Check if the Wifi module is functioning.	Connect Wifi according to the circuit.	LED light on the wifi module should glow.

Test Scenario 4: Check if the Servo Motor is Working.

Test Case	Description	Test Steps	Test Result
1	Check if the Servo Motor is Working.	Connect the servo motor to the Arduino Board	The motor shaft should rotate to an angle.

Chapter 4: Conclusions and Future Scope

CONCLUSION:

In this project, a digital door lock with enhanced security functions was designed to work with the Internet of Things. The designed digital door lock senses the physical impact of an invalid visitor and notifies the users mobile device. Another efficient system function is that when a user approaches the door, the door lock system opens or closes the door after receiving the instructions from the admin. We expect that if the some problems mentioned previously are resolved, the proposed system can be commercialized into a useful product, such as a secure security system with enhanced convenience, especially when compared to existing digital door lock systems.

FUTURE SCOPE:

Growth in Automation Market in India (2018-22) in the next phase for the home automation market will occur based on a few key improvements in the technology available in Automation, such as improvement in Wireless Automation solutions as well as lowering of price points as the market begins to accept Home automation usage in larger volumes. Some trends that we foresee for this phase of the industry are Solution offerings will slowly move to a more user friendly design, where aside from a few key components, users will be able to buy and use the Automation products themselves without the aid of any technical expert. Commodification of Automation market will come into play.The products themselves will reach a plug and play type of usability, where users can simply purchase pieces from the store and use it themselves without any support from professionals or even make them easily.

References

- GitHub.com

- Youtube.com

- (2014) The Edwards Robotics website. [Online]. Available: http://www.edwardsrobotics.com/

- (2017) The DroneBotWorkshop website. [Online]. Available: https://dbot.ws/dbwrkshp

- (2018) The Tinker Cad Website. [Online] . Available: https://tinkercad.com

- Margolis, Michael. Arduino Cookbook: Recipes to Begin, Expand, and Enhance Your Projects. " O'Reilly Media, Inc.", 2011.

- Instructables circuits: https://www.instructables.com/

- Journal: https://www.ijedr.org

- Journal: http://troindia.in

ANNEXURE 1

CODE FOR WIFI MODULE:

```
// PushingBox scenario DeviceId code and API

String deviceId = "v49675F92FA316E3";

const char* logServer = "api.pushingbox.com";


const char* ssid = "I'm Cheating on my Wifi";

const char* password = "*****";


void setup() {

  Serial.begin(74880);

  // Sending a notification Mto your mobile phone

  // function takes the message as a parameter

  sendNotification("Somebody at the door!!!");

}


void sendNotification(String message){


  Serial.println("- connecting to Home Router SID: " + String(ssid));


  WiFi.begin(ssid, password);

  while (WiFi.status() != WL_CONNECTED) {
```

```
  delay(500);

 Serial.print(".");

}


Serial.println();

Serial.println("- succesfully connected");

Serial.println("- starting client");


WiFiClient client;


Serial.println("- connecting to pushing server: " + String(logServer));

if (client.connect(logServer, 80)) {

 Serial.println("- succesfully connected");


 String postStr = "devid=";

 postStr += String(deviceId);

 postStr += "&message_parameter=";

 postStr += String(message);

 postStr += "\r\n\r\n";


 Serial.println("- sending data...");


 client.print("POST /pushingbox HTTP/1.1\n");

 client.print("Host: api.pushingbox.com\n");

 client.print("Connection: close\n");
```

```
        client.print("Content-Type: application/x-www-form-urlencoded\n");

        client.print("Content-Length: ");

        client.print(postStr.length());

        client.print("\n\n");

        client.print(postStr);

      }

     client.stop();

     Serial.println("- stopping the client");

    }


     void loop() {}

     Attachments area
```

CODE FOR SERVOMOTOR MODULE:

```
#include<Servo.h>

int data = 0;

Servo myservo;

int pos = 0;


void setup() {

  // put your setup code here, to run once:

myservo.attach(9);

Serial.begin(9600);

}
```

```
void loop() {

  // put your main code here, to run repeatedly:

if(Serial.available() >0)

{

  data=Serial.read();

  Serial.print(data);

  if(data =='1')

  {

    myservo.write(0);

  }

  if (data =='0')

  {

    myservo.write(90);

  }

}

}
```

ANNEXURE 2

Screenshots
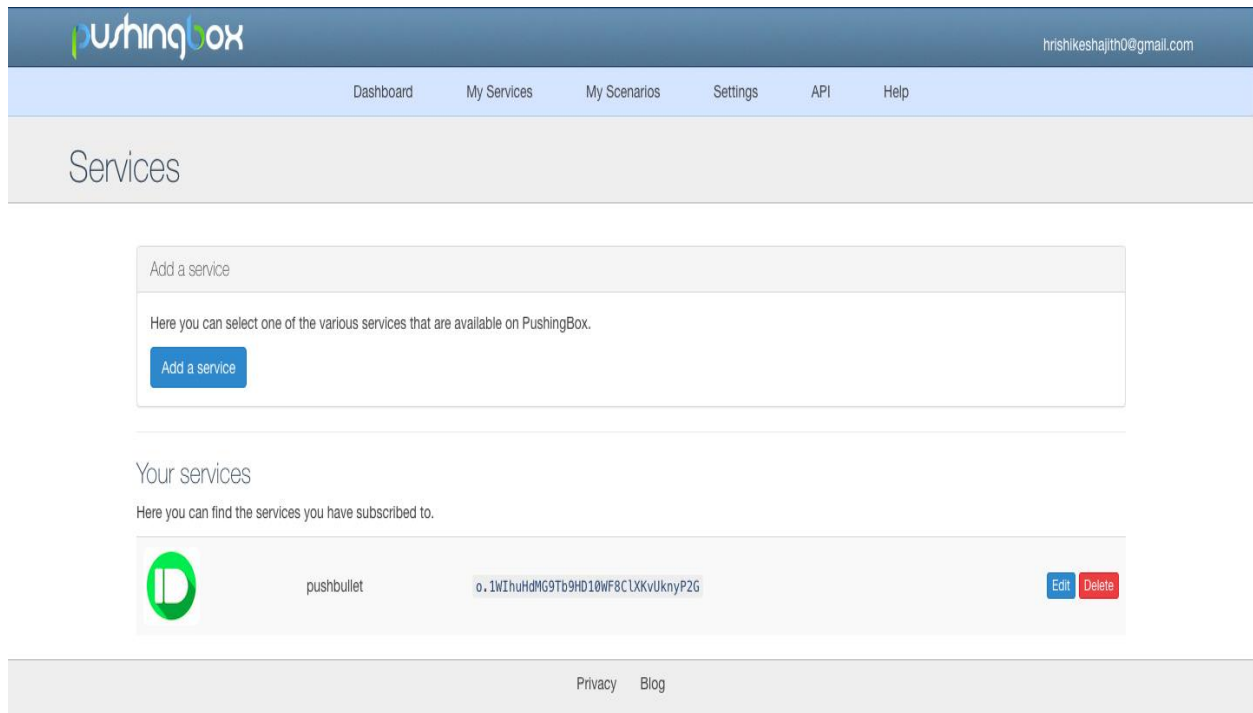
Fig6,7 Configuring the Pushing box service
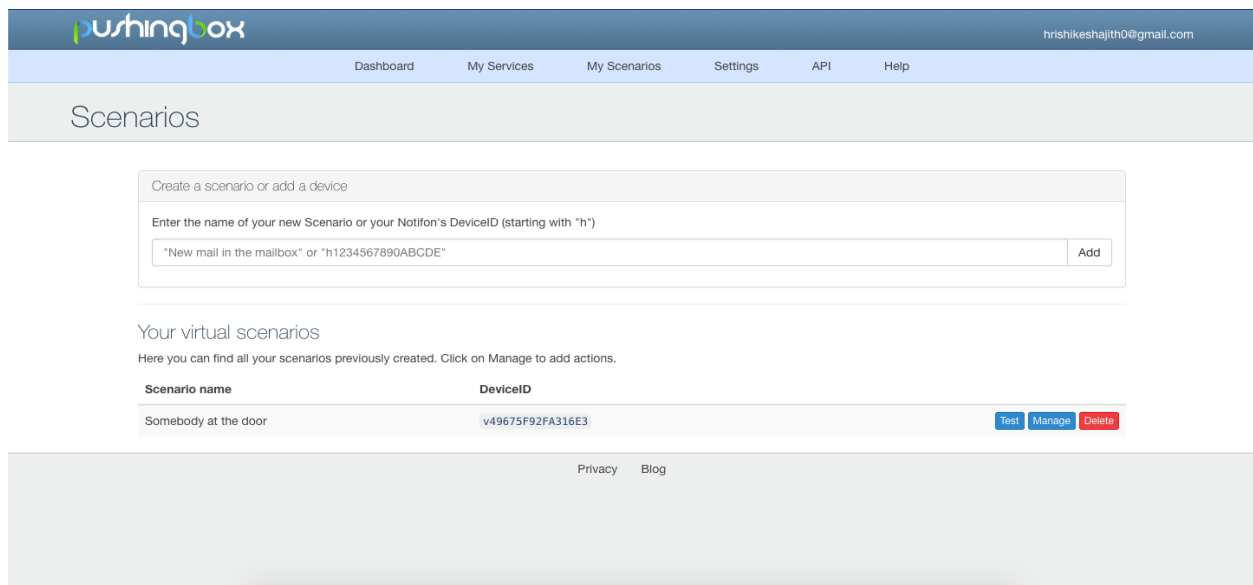
533P Internet of Things

Fig8,9 Adding the push bullet service

533P Internet of Things