A PROJECT REPORT

ON

"A SMART GREEN HOUSE SYSTEM BASED ON IoT"

Submitted to

UNIVERSITY OF MUMBAI

In Partial Fulfilment of the Requirement for the Award of

BACHELOR'S DEGREE IN
COMPUTER ENGINEERING

BY

PAWLE MUSA ASMAT AYSHA	14DCO60
PEERZADE NILOFER NASARUDDIN SHAINAZ	14DCO62
THAKUR ATIF UMAR FAHMIDA	14DCO69
GIGANI MEENAZ ASHRAF ASMA	15DCO47

UNDER THE GUIDANCE OF
PROF. MUBASHIR KHAN

DEPARTMENT OF COMPUTER ENGINEERING
Anjuman-I-Islam's Kalsekar Technical Campus
SCHOOL OF ENGINEERING & TECHNOLOGY
Plot No. 2 3, Sector - 16, Near Thana Naka,
Khandagaon, New Panvel - 410206

2017-2018

AFFILIATED TO

UNIVERSITY OF MUMBAI

A PROJECT II REPORT
ON

"A SMART GREENHOUSE SYSTEM BASED ON IoT"

Submitted to
UNIVERSITY OF MUMBAI

In Partial Fulfilment of the Requirement for the Award of

**BACHELOR'S DEGREE IN
COMPUTER ENGINEERING**

BY

PAWLE MUSA ASMAT AYSHA	14DCO60
PEERZADE NILOFER NASARUDDIN SHAINAZ	14DCO62
THAKUR ATIF UMAR FAHMIDA	14DCO69
GIGANI MEENAZ ASHRAF ASMA	15DCO47

UNDER THE GUIDANCE OF
PROF. MUBASHIR KHAN

DEPARTMENT OF COMPUTER ENGINEERING
Anjuman-I-Islam's Kalsekar Technical Campus
SCHOOL OF ENGINEERING & TECHNOLOGY
Plot No. 2 3, Sector - 16, Near Thana Naka,
Khandagaon, New Panvel - 410206

2017-2018
AFFILIATED TO

UNIVERSITY OF MUMBAI

Anjuman-i-Islam’s Kalsekar Technical Campus

Department of Computer Engineering

SCHOOL OF ENGINEERING & TECHNOLOGY

Plot No. 2 3, Sector - 16, Near Thana Naka,
Khandagaon, New Panvel - 410206

CERTIFICATE

This is certify that the project entitled

“A SMART GREENHOUSE SYSTEM BASED ON IOT“

submitted by

PAWLE MUSA ASMAT AYSHA	**14DCO60**
PEERZADE NILOFER NASARUDDIN SHAINAZ	**14DCO62**
THAKUR ATIF UMAR FAHMIDA	**14DCO69**
GIGANI MEENAZ ASHRAF ASMA	**15DCO47**

is a record of bonafide work carried out by them, in the partial fulfilment of the requirement for the award of Degree of Bachelor of Engineering (Computer Engineering) at ***Anjuman-I-Islam’s Kalsekar Technical Campus, Navi Mumbai*** under the University of MUMBAI. This work is done during year 2017-2018, under our guidance.

Date: / /

Prof. Khan Mubashir
Project Supervisor

Prof. Kalpana R.Bodke
Project Coordinator

Prof. Tabrez Khan
HOD, Computer Department

DR. ABDUL RAZAK HONNUTAGI
Director

External Examiner

Acknowledgements

We would like to take the opportunity to express our sincere thanks to Our guide **PROF. MUBASHIR KHAN**, Assistant Professor, Department of Computer Engineering, AIKTC, School of Engineering, Panvel for his invaluable support and guidance throughout Our project research work. Without his kind guidance & support this was not possible.

We are grateful to him for his timely feedback which helped us track and schedule the process effectively. His time, ideas and encouragement that he gave has helped us to complete our project efficiently.

We would like to express deepest appreciation towards **DR. ABDUL RAZAK HONNUTAGI**, Director, AIKTC, Navi Mumbai, **Prof. Khan Tabrez**, Head of Department of Computer Engineering and **Prof. Kalpana R.Bodke**, Project Coordinator whose invaluable guidance supported us in completing this project.

At last we must express our sincere heartfelt gratitude to all the staff members of Computer Engineering Department who helped us directly or indirectly during this course of work.

PAWLE MUSA ASMAT AYSHA

PEERZADE NILOFER NASARUDDIN SHAINAZ

THAKUR ATIF UMAR FAHMIDA

GIGANI MEENAZ ASHRAF ASMA

Project I Approval for Bachelor of Engineering

This project entitled ***A SMART GREENHOUSE SYSTEM BASED ON IoT*** by ***PAWLE MUSA ASMAT AYSHA, PEERZADE NILOFER NASARUDDIN SHAINAZ, THAKUR ATIF UMAR FAHMIDA, GIGANI MEENAZ ASHRAF ASMA*** is approved for the degree of ***Bachelor of Engineering in Department of Computer Engineering.***

Examiners

1. ....
2. ....

Supervisors

1. ....
2. ....

Chairman

....

Declaration

We declare that this written submission represents our ideas in our own words and where others ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

PAWLE MUSA ASMAT AYSHA
14DCO60

PEERZADE NILOFER NASARUDDIN SHAINAZ
14DCO62

THAKUR ATIF UMAR FAHMIDA
14DCO69

GIGANI MEENAZ ASHRAF ASMA
15DCO47

ABSTRACT

Today, convection agricultural is now improving to emphasise the productivity of the field. The agriculture, production becomes nowadays important in two terms quality and quantity. Under the Bureau of Indian Standards IS 15930(Part 1): 2010 requirements for good agricultural practices have been prescribed. From the research work of Mr S.K. Jadhav el at 2016, information like crop period, whether pruning is the foundation or forward pruning and appearance of infection of downy mildew etc, his team suggests various preventive measures and different pesticide treatments[5]. They conclude that the knowledge-based system will be helpful to an agricultural professional to take decision-related to the management of crop. Today‘s scenario was facing the problem of less space and more output. Overall expenditure of agricultural field reduced to half of its value.

Use IOT based smart agricultural system give new research area in field. India's food deficient was changed to leading agricultural status. 21st-century the market is facing the main problem of "smart customer". In this plight, technology is playing a vital role to uplift the agricultural production. Due to boom explosion in population, there is a vast improvement in agricultural machines over last century. As humans are making more relevant them self with monitoring systems, GPS locators, maps and an electronic sensor, these technologies start taking the stand in the agricultural field. Agricultural engineers work on planning, supervising, and managing the building of dairy effluent schemes, irrigation, drainage, and flood and water control systems. They aim to conserve soil and water and to improve the processing of agricultural products. In this paper, there is a survey on the technical aspect of agriculture.

Keywords: agricultural, smart customer, technology etc.

Glossary:

- **IoT:-**Internet of things is the network of physical devices, vehicles and other items embedded with electronics, software, sensors, actuators, and network connectivity which enable these objects to collect and exchange data.

- **GPD-**Gross domestic product is a monetary measure of the market value of all final goods and services produced in a period of time.

- **GSM-**Global System for Mobile Communications is used to describe the protocols for second-generation digital cellular networks used by mobile devices such as tablets.

- **Green Revolution-**The Green Revolution refers to a set of research and the development of technology that increased agricultural production worldwide.

- **UTRA-**Universal asynchronous receiver transmitter is a computer hardware device for asynchronous serial communication in which the data format and transmission speeds are configurable.

- **AWS-**Amazon Web Services is a subsidiary of Amazon.com that provides on-demand cloud computing platforms to individuals, companies and governments, on a paid subscription basis.

- **RFID-**Radio frequency identification uses electromagnetic fields to automatically identify and track tags attached to objects. The tags contain electronically stored information

Contents

Acknowledgement iii
Project I Approval for Bachelor of Engineering iv
Declaration v
Abstract vi
Table of Contents xiii

1 Introduction 2
1.1 Purpose 3
1.2 Project Scope 3
1.3 Project Goals and Objectives 4
1.3.1 Goals 4
1.3.2 Objectives 5
1.4 Organisation of Report 5

2 Literature Survey 8
2.1 Design and Implementation a Smart Greenhouse . . . 8
2.1.1 Advantages of Paper 9
2.1.2 Disadvantages of Paper 9

2.1.3 How to overcome the problems mentioned in Paper 10
2.2 IoT based Smart Greenhouse 10
2.2.1 Advantages of Paper 12
2.2.2 Disadvantages of Paper 12
2.2.3 How to overcome the problems mentioned in Paper 13
2.3 Intelligent Agriculture Greenhouse Environment Monitoring System Based on IOT Technology 13
2.3.1 Advantages of Paper 15
2.3.2 Disadvantages of Paper 15
2.3.3 How to overcome the problems mentioned in Paper 15
2.4 Green House by using IOT and Cloud computing . . . 15
2.4.1 Advantages of Paper 16
2.4.2 Disadvantages of Paper 17
2.4.3 How to overcome the problems mentioned in Paper 17
2.5 IOT Based Smart Greenhouse Automation Using Arduino 17
2.5.1 Advantages of Paper 18
2.5.2 Disadvantages of Paper 18

2.5.3 How to overcome the problems mentioned in Paper 19
2.6 Technical Review 19
2.6.1 Advantages of Technology 22
2.6.2 Reasons to use this Technology 23
2.7 Comparative Study 24

3 Project Planning 25
3.1 Members and Capabilities 25
3.2 Roles and Responsibilities 25
3.3 Assumptions and Constraints 26
3.4 Project Management Approach 26
3.5 Ground Rules for the Project 27
3.6 Project Budget 27
3.7 Project Timeline 28

4 Software Requirements Specification 29
4.1 Overall Description 29
4.1.1 Product Perspective 29
4.1.2 Product Features 30
4.1.3 User Classes and Characteristics 31
4.1.4 Operating Environment 32
4.1.5 Design and Implementation Constraints 32
4.2 System Features 33

4.2.1 System Feature 33

4.3 External Interface Requirements 36

4.3.1 User Interfaces 36

4.3.2 Hardware Interfaces 36

4.3.3 Software Interfaces 39

4.3.4 Communications Interfaces 41

4.4 Nonfunctional Requirements 41

4.4.1 Performance Requirements 41

4.4.2 Safety Requirements 42

4.4.3 Security Requirements 43

5 System Design 44

5.1 System Requirements Definition 44

5.1.1 Functional requirements 46

5.1.2 System requirements (non-functional requirements) 50

5.2 System Architecture Design 50

5.3 Sub-system Development 51

5.3.1 Sensor testing 51

5.3.2 Wifi Connectivity 52

5.3.3 Status Shown 53

5.3.4 Joining all together 53

5.4 Systems Integration 53

5.5 Class Diagram 54
5.5.1 Component Diagram 55
5.5.2 Deployment Diagram 56
5.5.3 Activity Diagram 57
5.5.4 State Chart 58

6 Implementation 59
6.1 Sensors Module 59
6.2 Wifi connectivity 64
6.3 Connectivity to thingspeak server 68
6.4 Communication between hardware and software 72

7 System Testing 79
7.1 Test Cases and Test Results 79
7.2 Sample of a Test Case for sensor 80
7.2.1 Software Quality Attributes 83

8 Screenshots of Project 84
8.1 A smart greenhouse system based on IoT 84

9 Conclusion and Future Scope 89
9.1 Future Scope 89

References 90

Achievements 93

Achievements

List of Figures

2.1 Structure of the system[15] 13

3.1 Spiral Model 26

3.2 Gantt chart 28

4.1 Class diagram for A Smart Green House System Based on IoT 31

4.2 Sequence Diagram for Smart Green House System Based on IoT 35

4.3 DHT11 36

4.4 KG003 37

4.5 LDR 38

4.6 ESP8266 38

5.1 Arduino function block 45

5.2 Usecase diagram for A Smart Green House System Based on IoT 47

5.3 DFD Diagram Level 0 48

5.4 DFD Diagram Level 1 49

5.5 DFD Diagram Level 2 49

5.6 System architecture for A Smart Green House System Based on IoT 50

5.7 Class Diagram for A Smart Green House System Based on IoT 54

5.8 Sequence Diagram for A Smart Green House System Based on IoT 55

5.9 Component Diagram for A Smart Green House System Based on IoT 55

5.10 Deployment Diagram for A Smart Green House System Based on IoT 56

5.11 Activity Diagram for A Smart Green House System Based on IoT 57

5.12 State Chart Diagram for A Smart Green House System Based on IoT 58

6.1 Screenshot on serial monitor for DHT11 62

6.2 Screenshot on serial monitor for LDR 62

6.3 Screenshot on serial monitor for Gas 63

6.4 Screenshot on serial monitor for Soil sensor 63

6.5 Screenshot of Temperature CSV file 67

6.6 Screenshot of Humidity CSV file 67

6.7 Screenshot of Dewpoint CSV file 68

6.8 System picture1 72

7.1 Result 83

8.1 System picture2 84

8.2 System picture3 85

8.3 System picture1 85

8.4 Picture of Serial monitor 86

8.5 Picture of ThinkSpeak 86

8.6 Loing for A Smart Green House System Based on IoT 87

8.7 App1 for A Smart Green House System Based on IoT . 87

8.8 App2 for A Smart Green House System Based on IoT . 88

8.9 App3 for A Smart Green House System Based on IoT . 88

List of Tables

2.1 Comparison in between current system 24

3.1 Table of Capabilities 25

3.2 Table of Responsibilities 25

3.3 Table of Budget 28

4.1 Table of features 34

6.1 DHT11 Pin configuration 60

6.2 ESP8266 Pin configuration 64

6.3 LDR Pin configuration 68

6.4 LDR Pin configuration 68

Chapter 1

Introduction

The agricultural field has a very vast history in the human race. India is well known as Golden Sparrow due to its production of agriculture in each era of the races. Three different season of the nation gives a very important position in the agricultural field in India.

R Gopalakrishnan and Dr. YSP Thorat in collaboration with Tata companies have surveyed and concluded that agricultural growth was excellent till 2000 and then it has been slowly declined[3]. To which an urgent attention is required from the youth of the nation. As we know, the agricultural field provides employment to many youths.

Even though the share of agriculture in GDP has declined to one fifth from one-half at the time of independence, agriculture remains the predominant sector in terms of employment and livelihood provision for more than half of India's workforce engaged in it as the principal occupation[5].

1.1 Purpose

Today every market in the world is facing the main problem with the concept of the Smart customer. To keep up with the requirement of this new Smart customer, agricultural field requires new technique to improve quality and production. In this aspect, science and technology provide hands to this field. Scientific research provided data to improve the quality whereas technology approach provides control on the quality and production.

The only users are going to be farmers as our main purpose to build this system is to help them. They will log-in and will push few buttons to control environment. Arduino will be smart and will do the actuation on it's own if the environment is over the threshold environment.

1.2 Project Scope

The proposed project aims at monitoring the various analogue parameters such as Humidity, temperature, soil moisture etc. The system should control the environment according to the requirement for a plant to grow healthily.

The user will also be able to control the environment if desired. The processed data will be sent over a wireless network to the owner to remotely access from anywhere in the world. To tackle problems farmers face in natural agriculture process we have proposed a design which will determine present environmental conditions.

According to the requirement of the crops the threshold will be set, if any environmental condition goes below or above the threshold value,then the proposed project will sense the changing in parameters are monitored simultaneously.

Entire data will be transmitted to farmers, according to that farmer will take the controlling decision and send to the system. With the help of actuator, the system will control the environment(parameter).

1.3 Project Goals and Objectives

1.3.1 Goals

Our goal is to control the environment according to the requirement for a plant to grow healthily. The smart system will do the desired changes on it's own, but the user has also the authority to modify the environment.

As indicated by the prerequisite of the products the edge will be set, if any natural condition goes beneath or over the limit value,then the proposed venture will detect the changing in parameters are observed at the same time

1.3.2 Objectives

The objective of the proposed project is that with the help of IoT concept we can implement an automated smart system for greenhouses.The processed data will be sent over a wireless network to the owner to remotely access from anywhere in the world.

To tackle problems farmers face in natural agriculture process we have proposed a design which will determine present environmental conditions. Entire data will be transmitted to farmers, according to that farmer will take the controlling decision and send to the system. With the help of actuator, the system will control the environment(parameter)

1.4 Organisation of Report

In Chapter 1: We have considered Project overview under which we have explained various important terminologies like Introduction of the project. Motivation (what exactly motivated us to create priority based cab search),problem definition,About current system ,Advantages over current system,Goals and Objectives,Scope and Application.

In Chapter 2: We have discussed about various paper that we have referred for our project. We have mentioned the description , pros and cons and how the overcome the problem under every paper. a total of three paper have been referred.

In Chapter 3: We have discussed about the requirement analysis under it we have consider about the requirement the platform requirement supporting the os of the software and hardware requirement along with the feasible study.

In Chapter 4: We can see the system design and architecture various diagram can be seen in this chapter which represent the software , diagram including our system architecture usecase diagram dfd diagram class diagram and component diagram.

In Chapter 5: We have seen the methodology here we have explain the project in detail by dividing into module.various module of priority based cab search are explained with the help of few diagram.

In Chapter 6: We have discussed about the implementation details the assumption and dependencies this part contains details of the implementation of methodology that we discuss earlier.

In Chapter 7: We have shown the test cases and result along with analytic discussion this part contained the result of the output of the project.

In Chapter 8: We have concluded the whole project and future scope along with the limitation followed up by reference and chapter 9 with Appendix.

Chapter 2

Literature Survey

2.1 Design and Implementation a Smart Greenhouse

This paper proposes an efficient automatic irrigation system based on computing various changes necessary in green house using wireless sensor network and using server and client web service for control and monitoring. Our model has two main factors which are reduces the power and controlling and monitoring over long distances.

The condition of supplying diverse environmental situations for each plant has drove researchers to use different monitoring systems[6]. The results demonstrate that the control model is measure the sensing data and accurate tool for calculating values of adapted sensors as well as the self-control the output plugged devices.

It is intended to variety of monitoring for automation by using input parameter (temperature, light intensity, soil moisture, and the amount of carbon dioxide) in intended system based on Android devices[7].

Also, results depict that our system has several advantageous characteristics, such as: ease of network management and control motors and valves. The input and output parameters used in greenhouse control were displayed within Android device.

Finally, process of the input and output parameters of the system were measured and information was given about the rule base[8]. There are five sensors adapted in the proposed design system, soil moisture, humidity, temperature, CO2, and light sensor, each of these sensors has measure changes in environment inside the greenhouse.

2.1.1 Advantages of Paper

a. Ease of network management

b. Good control on motors

c. Four Sensors are used.

2.1.2 Disadvantages of Paper

a. This system doesn't include RFID.

b. This system needs a computer to control the system.

c. The system is only automate we can not actuate the system manually.

2.1.3 How to overcome the problems mentioned in Paper

a. We can install an RFID tagging.

b. An android application should be developed to access it from anywhere.

c. With the help of an Android application we can actuate the system manually.

2.2 IoT based Smart Greenhouse

The irrigation of agriculture field is carried out using automatic drip irrigation, which operates according to the soil moisture threshold set accordingly so as optimal amount of water is applied to the plants. Based on data from soil health card, proper amount of nitrogen, phosphorus, potassium and other minerals can be applied by using drip for irrigation techniques.

Proper water management tanks are constructed and they are filled with water after measuring the current water level using an ultrasonic sensor. Plants are also provided the requisite wavelength light during the night using growing lights. Temperature and air humidity are controlled by humidity and temperature sensors and a foggier is used to control the same.

This work provides a model of a smart greenhouse,which helps the farmers to carry out the work in a farm automatically without the use of much manual inspection. Greenhouse[9].

A tube well is controlled using GSM module (missed call or sms). Bee-hive boxes are deployed for pollination and boxes are monitored using ultrasonic sensors to measure honey and send mails to the buyers when they are filled. Being a closed structure protects the plants from extreme weather conditions namely: wind,hailstorm, ultraviolet radiations, and insect and pest attacks [10]. Further, the readings collected from storage containers are uploaded to cloud service (Google drive) and can be forwarded to an e-commerce company.

A tube well is controlled using GSM[11] module (missed call or sms). Bee-hive boxes aredeployed for pollination and boxes are monitored using ultrasonic sensors to measure honeyand send mails to the buyers when they are filled. This system can be installed by any individual in his house (Rooftop greenhouse), who does't have knowledge about farming. Since one can maintain any climatic condition in this type of Greenhouse, it is possible to cultivate any type of crop.

Additionally, this framework can be introduced by any person in his home (Rooftop nursery), who doesn't know about cultivating. Since one can keep up any climatic condition in this kind of Greenhouse, it is conceivable to develop any sort of product.

Hardware-Software Resources used in this paper are[12]

1. Intel Galileo Gen 2 and Arduino
2. Regrowing LED lights
3. Fan
4. Ultrasonic Sensor
5. GSM sim900a
6. RFID tag and sensor
7. LCD for display
8. Relays for connecting pumps, peltier, lights, fogger
9. Arduino IDE (Software)

2.2.1 Advantages of Paper

a. Able to produce insecticide and pesticide free crops

b. Create a climate for the proper growth of plants

c. Provides alternative source of income through actuators

2.2.2 Disadvantages of Paper

a. The system is only automate we can not actuate the system manually.

b. Not accessible from anywhere.

2.2.3 How to overcome the problems mentioned in Paper

a. An android application should be developed to access it from anywhere.

b. With the help of an Android application we can actuate the system manually.

2.3 Intelligent Agriculture Greenhouse Environment Moni-toring System Based on IOT Technology

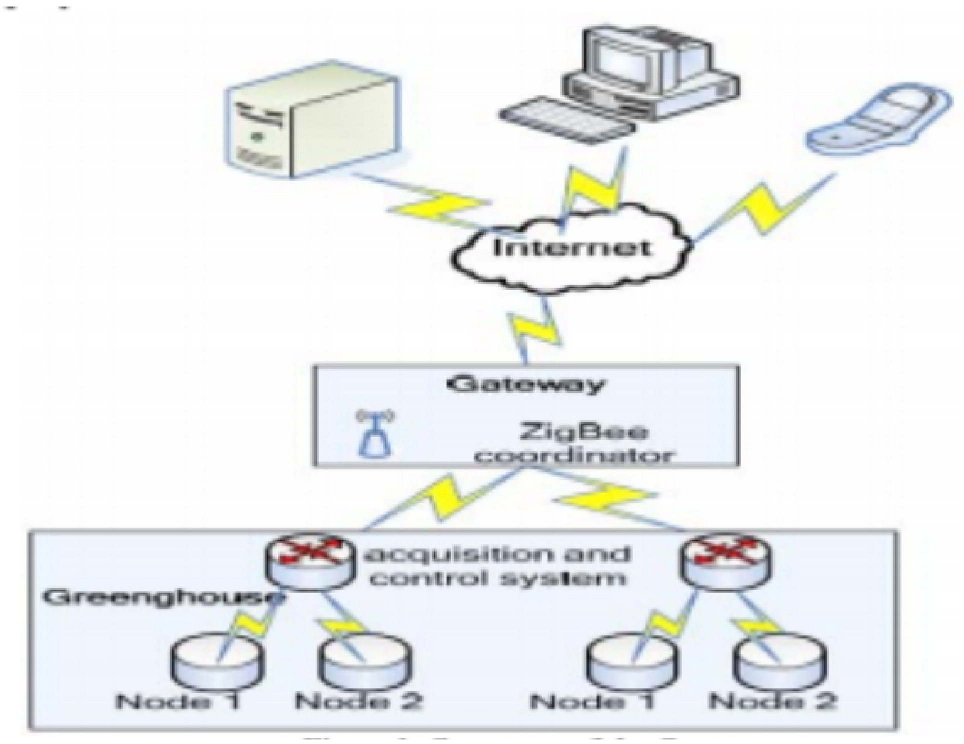


Figure 2.1: Structure of the system[15]

This paper takes CC2530 chip as the core, presents the design and implementation of agriculture Greenhouse Environment monitoring system based on Zigbee technology, the wireless sensor and control nodes takes CC2530F256 as core to control the environment data.

This system is made up of front end data acquisition, data processing, data transmission and data reception. The ambient temperature is real time process by the temperature sensor of the data terminal node.

Processed data is send to the intermediate node through a wireless network. Intermediate node aggregates all the data, and then sends the data to the PC through a serial port, at the same time, staff may view, analysis and storage the data by the C that provide the real time data for agricultural greenhouse, fans and other temperature equipment, and achieve automatic temperature control.

In this paper the low cost and power Zigbee technology applies in greenhouse system. The System realises the remote intelligent control to the room equipment through internet.

It Improves the operational efficiency and system application flexibility by using the wireless sensor network and at the same time Design and Implementation a Smart Greenhouse reduce the manpower cost.

They used a microprocessor (Samsung S5VP210) where 512mb ram, sensors, camera, Zigbee coordinator cc2530, ID card reader and GPRS module was connected. Zigbee is connecting to stm8s003f3 which is used as to take input in AD form.

2.3.1 Advantages of Paper

a. Remote intelligent control.

b. Less use of protective chemicals.

c. Low cost and power Zigbee technology

2.3.2 Disadvantages of Paper

a. Expensive microprocessor (Samsung S5VP210)

b. Old Zigbee technology

2.3.3 How to overcome the problems mentioned in Paper

a. Use Arduino Uno

b. Use ESP8266 insted of Zigbee technology

2.4 Green House by using IOT and Cloud computing

In this paper environment builds up by using two technologies ioT and cloud computing. By using IOT(Internet on things) we control devices or any environmental needs anytime, anywhere and the cloud which provides storage and computing resources to implement a web page.

Internet on things and cloud computing collectively makes a system that control green house effectively. This system will sense all the environmental parameter and sends that data to the user via cloud.

User will take controlling action according to that this will done by using actuator by using this system. IoT means connecting, transferring data of devices via the Internet. By using IoT we control appliance anytime, anywhere and the cloud which provides storage and computing resources to implement a web application[14]. This asset allows the farmer to improve the cultivation in a way the plants need. It leads to higher crop yield, prolonged production period, better quality, and less use of protective chemicals.computer.

2.4.1 Advantages of Paper

a. Improved system

b. Helpful in rational planting

2.4.2 Disadvantages of Paper

a. Computer or a Laptop is needed to control the environment remotely

b. Old Zigbee technology

c. very costly micro-controller is used (Samsung S5PV210)

2.4.3 How to overcome the problems mentioned in Paper

a. Develop an Android app so it can be done with help of any android mobile phone.

b. Use ESP8266 for Wifi module

c. Arduino uno can be used instead which is less costly

2.5 IOT Based Smart Greenhouse Automation Using Arduino

In this paper, the Design had been aimed data asset in greenhouse for multiple sensors to use data for simulation or processing to achieve the better enhancement of growth in greenhouse, this data has effect on the climate of greenhouse. Graphical User Interfaces (GUI) had been used through LabVIEW, firmware of Arduino as software and Arduino board and sensors as hardware. by using Arduino mega board provides multiple inputs analogs and I/O digitals to made read data sensor easy to take temperature, humidity, CO2 gas.

Also measuring the soil moisture that needed for irrigation plants and the intensity of lights that applied for greenhouse.The arduino Uno is a microcontroller board based on the ATmega328, It has 14 digital input/output pins, 6 analog input, a 16 MHZ crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button[4]. These factors has the major effect on increase in growth of plants. Greenhouse environments monitoring different changes to parameters, the system for this purpose had been provided and given ability to control on climate of greenhouse.

2.5.1 Advantages of Paper

a. Improved system

b. Helpful in rational planting

2.5.2 Disadvantages of Paper

a. Computer or a Laptop is needed to control the environment remotely

b. Old Zigbee technology

c. very costly micro-controller is used (Samsung S5PV210)

2.5.3 How to overcome the problems mentioned in Paper

a. Develop an Android app so it can be done with help of any android mobile phone.

b. Use ESP8266 for Wifi module

c. Arduino uno can be used instead which is less costly

2.6 Technical Review

Arudino Arduino is an open-source PC equipment and programming organisation, venture, and client group that outlines and produces. The Arduino Uno is a microcontroller board in light of the ATmega328, It has 14 advanced info/yield pins, 6 simple input, a 16 MHZ precious stone oscillator, a USB association, a control jack, an ICSP header, and a reset catch. The Uno contrasts from every single going before board in that it doesn't utilise them FTDI USB to serial driver chip."UNO" implies one in Italian what's more, is named to stamp the up and coming arrival of Arduino 1.0. The Uno is the most recent in a progression of USB Arduino sheets and reference demonstrate for Arduino stage. The Arduino Uno can control by means of the USB association or with outer power supply. Outer power can come either from an AC to DC connector or battery.

Soil Sensor(KG003) The two copper drives go about as the sensor tests. They are submerged into the example soil whose dampness content is under test. The conductivity of soil relies on the sum of dampness display in it. It increments with increment in the water substance of the dirt that structures a conductive way between two sensor tests prompting a nearby way to permit current moving through.

Temperature Sensor(DHT11) This DHT11 Temperature and Humidity Sensor includes an adjusted advanced flag yield with the temperature and stickiness sensor complex. Its innovation guarantees the high unwavering quality and superb long haul solidness. A superior 8-bit microcontroller is associated. This sensor incorporates a resisting component and a feeling of wet NTC temperature estimating gadgets. It has magnificent quality, quick reaction, hostile to obstruction capacity and high-cost execution favourable circumstances.

Light Sensor(LDR) The light sensor is to a great degree touchy in obvious light range. With the light sensor connected to the framework when the encompassing characteristic lights are low, it shows the advanced values.

Wifi Module(ESP8266) The ESP8266 is a minimal effort Wi-Fi microchip with full TCP/IP stack and microcontroller capacity created by a Shanghai-based Chinese maker, Espressif Systems.

1. 802.11 b/g/n protocol
2. Wi-Fi 2.4 GHz, support WPA/WPA2
3. Super small module size (11.5mm x 11.5mm)
4. Integrated 10-bit ADC
5. Integrated TCP/IP protocol stack (ipv4 only at the moment)
6. Integrated TR switch, balun, LNA, power amplifier and matching network Integrated PLL, regulators, and power management units

Design is the very first step in the development phase for any techniques and principles for the purpose of defining a device, a process or system in sufficient detail to permit its physical realisation. Once the software requirements have been analysed and specified the software design,coding, implementation and testing that are required to build and verify the software.

The design activities are of main importance in this phase, because in this activity decision ultimately affecting the success of the software implementation and it ease of maintenance are made. These decision have the final bearing upon reliability and maintainability of the system.Design is the place where quality is fostered in development.

Software design is the process through which requirements are translated into a representation of software. Software design is conducted in two steps. Preliminary design is concerned with transformation of requirement into data.

The tremendous volumes of new information from IoT sensors and gadgets just adds to the enormous pool of huge information. We now have on the planet and shrewd organisations utilises that information to educate key and operational basic leadership.

Strategic decision is the place the senior authority group recognises the basic inquiries it needs replying. Operational basic leadership is the place information and investigative are made accessible to everybody in the association, frequently by means of a self-benefit apparatus, to illuminate information driven choice at all levels.

2.6.1 Advantages of Technology

a. Machine to Machine communication

b. Extensive amount of automation and control

c. Monitoring and access of information.

d. Eliminating risk of greenhouse not being maintained at specified environmental conditionals due to human error.

e. Minimizes labour costs involved in maintaining a greenhouse.

f. Customer able to define specific greenhouse conditions.

g. Digitization in Agriculture sector

h. To overcome the demands and increase productivity

2.6.2 Reasons to use this Technology

a. Improve decision-making

b. Improve and optimize operations

c. To have information and access from anywhere.

d. Automatically control environmental conditions within greenhouse allowing any type of plants to be grown all year round.

2.7 Comparative Study

Parameters	IoT based Smart Greenhouse	Green House by using IOT and Cloud computing	Intelligent Agriculture Greenhouse Environment Monitoring System Based on IOT Technology	Design and Implementation a Smart Greenhouse	A Smart Green House System Based nn IoT
Software neede for Implementation	-	HTML	LABVIEW	Microsoft Visual Studio 2012 .NET	Android SDK
Cloud services	Google Drive Server	Google Drive Server	Embedded web server	Google Drive	Amazon Web Services
Wifi connectivity through	GSM sim900a	Zigbee	Zigbee	-	ESP8266
Portability	No	No	No	No	Yes
LCD display	Yes	No	No	No	Yes
Board Used	Arduino	Arduino	Microprocessor Samsung S5PV210	Arduino	Arduino

Table 2.1: Comparison in between current system

Chapter 3

Project Planning

3.1 Members and Capabilities

Table 3.1: Table of Capabilities

SR. No	Name of Member	Capabilities
1	Pawle Musa	Arduino Programming and wifi connectivity
2	Peerzade Nilofar	Android Programming
3	Thakur Atif	Android Programming
4	Gigani Meenaz	Android Programming and Database

3.2 Roles and Responsibilities

Table 3.2: Table of Responsibilities

SR. No	Name of Member	Role	Responsibilities
1	Pawle Musa	Leader	connectivity between Arduino and server, Working of various sensors, Working of actuator on inhibited environment.
2	Peerzade Nilofar	Android developer	Working of Android application
3	Thakur Atif	Android developer	Working and connection for Android application
4	Gigani Meenaz	Android developer	Working of android application, connecting server with the application

3.3 Assumptions and Constraints

Assumption is that the data that is going to upload on server and the mobile application will receive that data from the ever. Similarly for controlling the system with it.

3.4 Project Management Approach

Spiral Model is a combination of a waterfall model and iterative model. Each phase in spiral model begins with a design goal and ends with the client reviewing the progress. We have use this model for version control.

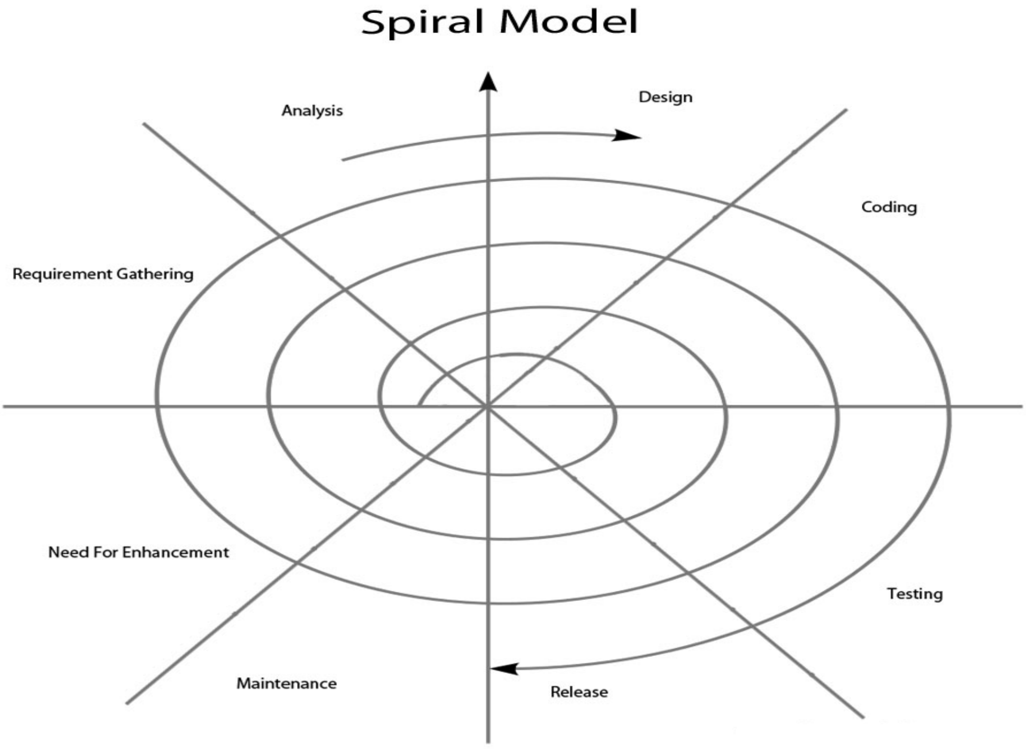


Figure 3.1: Spiral Model

3.5 Ground Rules for the Project

1. We treat each other with respect.
2. We intend to develop personal relationships to enhance trust and open communication.
3. We value constructive feedback. We will avoid being defensive and give feedback in a constructive manner.
4. As team members, we will pitch in to help where necessary to help solve problems and catch-up on behind schedule work.
5. Additional meetings can be scheduled to discuss critical issues or tabled We should create and adopt written notes for help No responsibilities to be assigned unless the person who is being assigned the responsibility accepts it. If a person to be given a responsibility is not at the meeting, the team leader must review that assignment or action item with the person before the responsibility is designated.

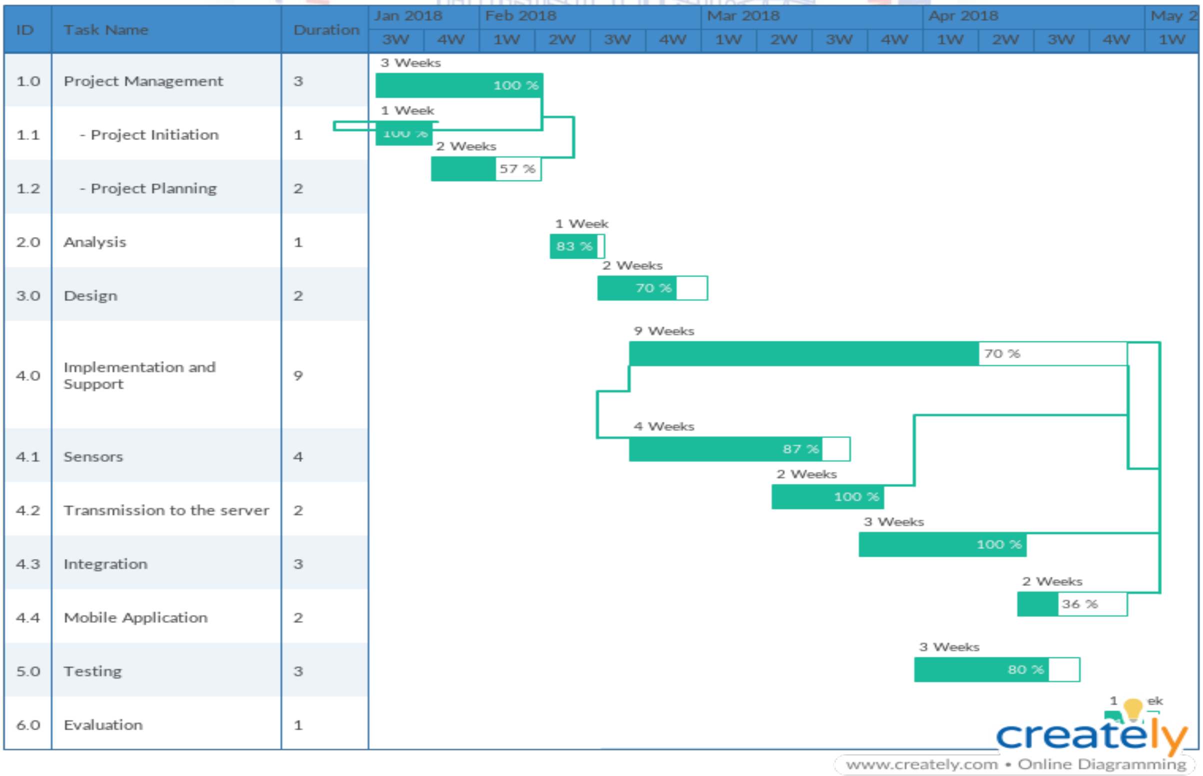
3.6 Project Budget

The budget for this project is very moderate.Following are the budget for the project

Table 3.3: Table of Budget

SR. No	Name of product	Quantity	Cost
1	Arduino	1	538
2	Wires Male to Male	20	100
3	Wires Male to Female	20	100
4	ESP8266 module	1	996
5	Fan	1	150
6	Water pump	1	150
7	Shutter motor	1	100
8	Fan	1	150
9	DHT11	1	100
10	MQ135	1	100
11	ldr	1	10
12	KG003	1	100
13	Environment	1	1300
14	Arudino IDE	-	-
15	Android SDK	-	-
-	Total	-	3934

3.7 Project Timeline



ID	Task Name	Duration
1.0	Project Management	3
1.1	- Project Initiation	1
1.2	- Project Planning	2
2.0	Analysis	1
3.0	Design	2
4.0	Implementation and Support	9
4.1	Sensors	4
4.2	Transmission to the server	2
4.3	Integration	3
4.4	Mobile Application	2
5.0	Testing	3
6.0	Evaluation	1

Figure 3.2: Gantt chart

Chapter 4

Software Requirements Specification

4.1 Overall Description

4.1.1 Product Perspective

The primary point of these Project is to enable ranchers to even in inverse ecological conditions like in overheat and less dampness conditions. A greenhouse is an important part of the agriculture and horticulture sectors in our country. It is used for growing plants faster at any season whether the climate is according to plant or not

Automatic monitoring is done, through which plants will fulfill their requirements and also there will be a great help to our farmers. At those places, Greenhouse is used as an artificial environment to create required environmental conditions.

The main aim of these Project is to help farmers even in opposite environmental conditions like in overheat and fewer humidity conditions. Here controlling part will be handled by Arduino, And the monitoring part is handled by IOT. We proposed greenhouse system having four main sensors needed to check the climatic change in environment.

The sensors we proposed in the project are light sensor, humidity sensor, soil sensors and temperature sensor. An Arduino Uno is used to operate on these sensors and to detect availability of light, water in the environment.

Internet of things makes monitoring of system accessible everywhere having net access on any Android device.The product is supposed to be an open source, it is hardware and software based implementation, The following main feature provided by an application are.

4.1.2 Product Features

Internet of things makes observing of framework available wherever having net access on any Android device.The item should be an open source, it is equipment and programming based execution, The accompanying fundamental component gave by an application are. In our product part will be taken care of by Arduino, And the checking part is dealt with by IoT.

- Low in cost
- Android application to get information and control the environment
- Efficient framework

4.1.3 User Classes and Characteristics

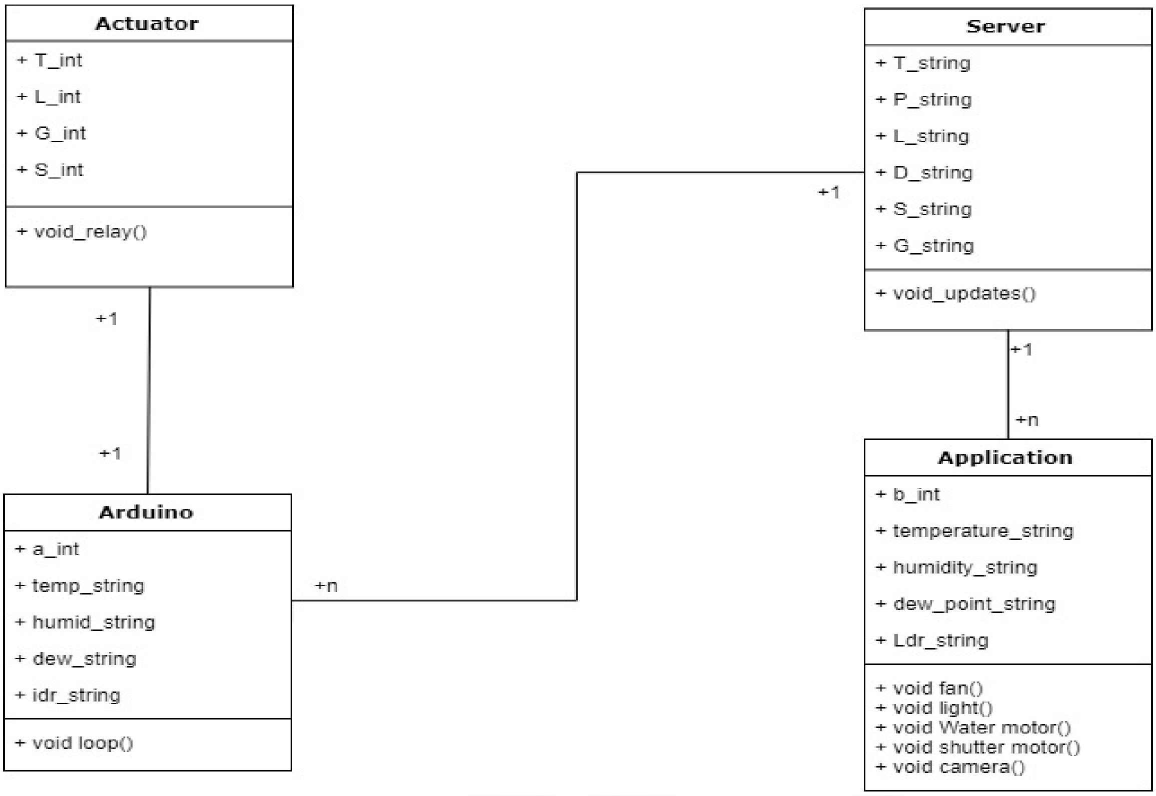


Figure 4.1: Class diagram for A Smart Green House System Based on IoT

We have Arduino uno to control the environment. We have used ESP8266 to send and receive the data over internet. We have used application class for android application and similarly server class to send and receive data to our server thinkspeak server. We have utilised ESP8266 to send and get the information over web.We have utilised application class for android application and correspondingly server class to send and get information to our server thinkspeak server.

4.1.4 Operating Environment

This is an android based system and hence will require the operating environment for a client and server GUI. The product has an Android based framework and subsequently will require the working condition for a customer and server GUI. This will work in the accompanying working condition.

- This software highly pivot on type and version of android being installed in the android application.
- Our software mainly dependent on the Internet, without wifi use the application can't work.
- Our project mainly rely on Hardware.

4.1.5 Design and Implementation Constraints

1. Lower speed compare to wired network.

2. More complex to configure than wired network.

3. Gets distracted by various elements like Bluetooth.

4. It does not make sensing quantities in buildings easier.

5. It does not reduce costs for installation of sensors.

4.2 System Features

We have Arduino uno to control the environment. We have used ESP8266 to send and receive the data over internet. We have used application class for android application and similarly server class to send and receive data to our server thinkspeak server. We have utilised ESP8266 to send and get the information over web. We have utilised application class for android application and correspondingly server class to send and get information to our server thinkspeak server.

4.2.1 System Feature

- Low in cost

- Android application to get information and control the environment

- Efficient framework

- Extensive amount of automation and control

- Monitoring and access of information.
- Eliminating risk of greenhouse not being maintained at specified environmental conditionals due to human error.
- Minimises labour costs involved in maintaining a greenhouse.
- Customer able to define specific greenhouse conditions.
- Digitisation in Agriculture sector

Description and Priority

Table 4.1: Table of features

SR. No	Feature	Priority	Risk
1	Farmers are able to monitoring and access of our system .	10	10
2	With the help of IoT technology farmers can minimizes labour cost	7	2
3	Our system Eliminates risk of greenhouse not being maintained at specified environmental conditionals due to human error.	8	5
4	Farmers are able to define specific greenhouse conditions.	9	4

Stimulus/Response Sequences

This is the sequence diagram of the system in which modules are sensor, gateway, middle-ware, server. The smart greenhouse using IoT, the sensors data will be transmitted through a set of different devices,to the cloud where it get processed, if a measurement crossed the threshold, the cloud sends a command to the specified greenhouse in order to perform the suitable action.The diagram below (Fig.4.2)describes greenhouse

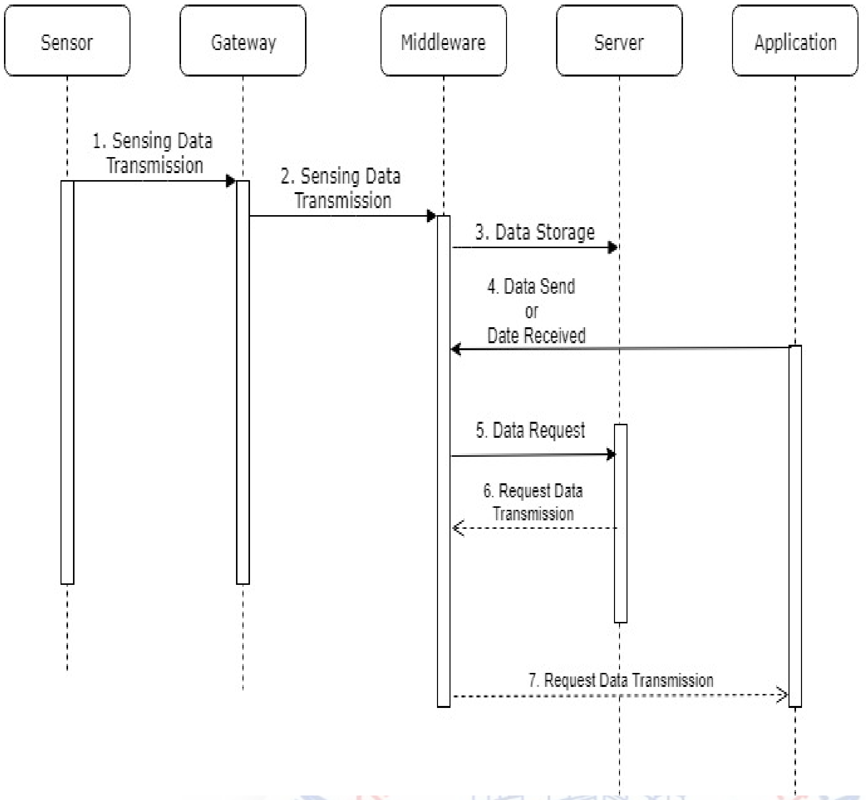

Sensor
Gateway
Middleware
Server
Application
1. Sensing Data Transmission
2. Sensing Data Transmission
3. Data Storage
4. Data Send or Date Received
5. Data Request
6. Request Data Transmission
7. Request Data Transmission

Figure 4.2: Sequence Diagram for Smart Green House System Based on IoT

Functional Requirements

1. Sensor should sense the temperature in the given environment of greenhouse System.
2. The correct details of the sensors should be fetched by the server.
3. The server should response quickly.
4. Immediate output on the display screen.

4.3 External Interface Requirements

4.3.1 User Interfaces

Farmer are the main audience of our system. This interface should be very user friendly and easily understandable to the farmer. All the transactions and requirements are merged up in this interface. The whole system focuses on this user and tried to provide the best utility of this application.

4.3.2 Hardware Interfaces

Sensors

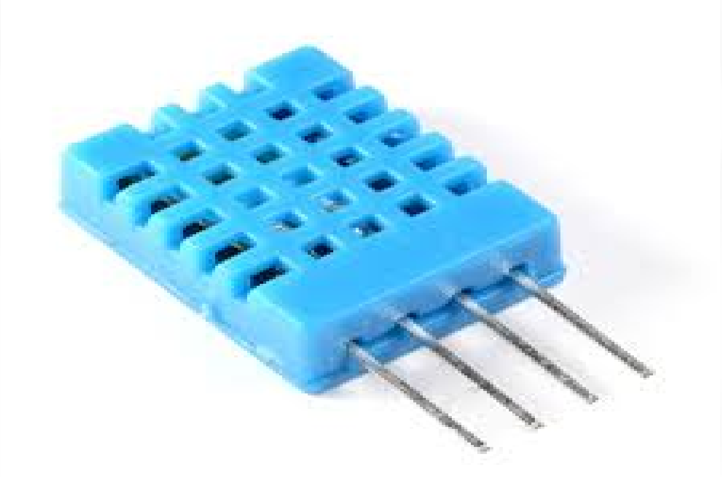


Figure 4.3: DHT11

1. DHT11 is a Humidity and Temperature Sensor, which generates calibrated digital output. DHT11 can be interface with any micro-controller like Arduino, Raspberry Pi, etc. and get instantaneous results. DHT11 is a low cost humidity and temperature sensor which provides high reliability and long term stability.

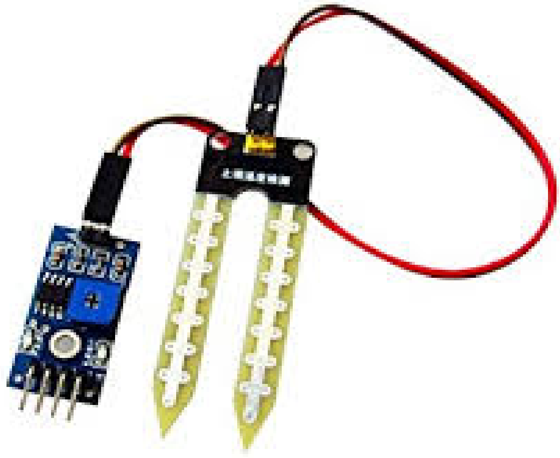


Figure 4.4: KG003

2. Soil sensor(KG003) this is an easy to use digital soil moisture sensor. Just insert the sensor in the soil and it can can measure moisture or water level content in it. It gives a digital output of 5V when moisture level is high and 0V when the moisture level is low in the soil.

3. Gas sensor(mq135) a gas sensor is a gadget that distinguishes the nearness of gases in a territory, regularly as a major aspect of a well being framework. This sort of gear is utilized to identify a gas spill or different discharges and would interface be able to with a control framework so a procedure can be naturally closed down. A gas indicator can sound an alert to administrators in the region where the break is happening, giving them the chance to clear out.

4. An LDR is a light-controlled variable resistor. The resistance of a photo resistor decreases with increasing incident light intensity; in

other words, it exhibits photo conductivity. To put it more simply it detects presence of light in the environment.

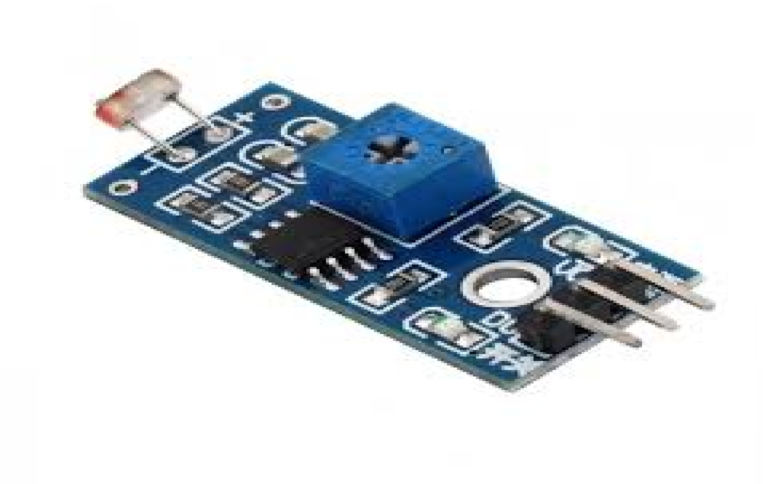


Figure 4.5: LDR

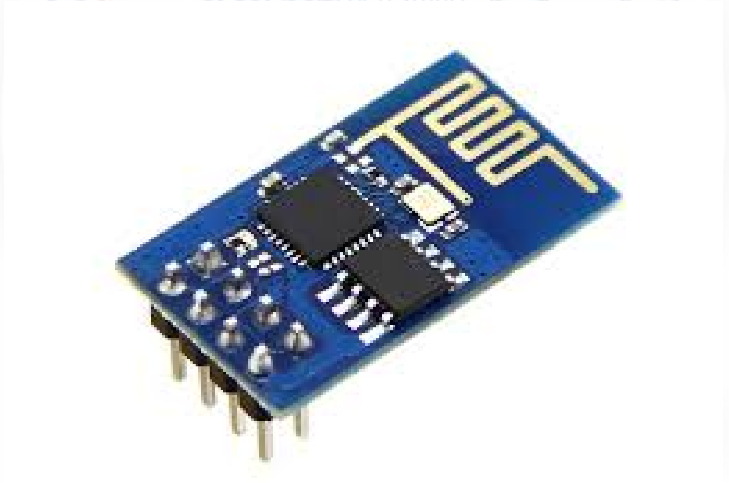


Figure 4.6: ESP8266

WI-Fi module(ESP8266) It connects the sensor and Arduino with the internet to mobile application. It updates the whether information on the both side updates the thinkspeak server and the also update information in database. We are using ESP8266 to connect our Arduino Uno to the server.

Arduino Arduino is controller which is used for particular small application such as GREENHOUSE automation, etc. In this project the sensors are sensing for available resources from the environment such as Humanity and Temperature, Carbon Dioxide in PPM, Moisture of Air and Soil etc. These sensed values are passed to Arduino which decides the activation of actuators. The Arduino Uno is a micro controller board based on the ATmega328 (datasheet). It has 14 digital i/o pins in which six are analog inputs.

For example, if there is no wind in the environment, the sensors sense it and passes to the Arduino. Arduino activates fan to provide wind in agriculture. Even though Actuators are activated or not continue data is being stored on Internet Cloud connected via WiFi.

If due to unavoidable circumstances, WiFi is unable to establish connection with the cloud or if the server is down, a Server Down Message is printed else data will be auto-updated in Mobile App.

4.3.3 Software Interfaces

Android Studio: Windows and Linux both support android studio, the current version we are using is 2.2 with lollipop and marshmallow operating system. Currently the craze of android language make us a decision to choose this language. It is mobile application Arduino simulator to code the program for sensor to connect them to hardware. Arduino programming language can be divided in three main parts: structure, values (variables and constants), and functions. A versatile application is a PC program intended to keep running on a cell phone, for example, a telephone/tablet or watch.

We are utilising Android to get to the earth remotely. Most cell phones are sold with a few applications packaged as pre-introduced programming, for example, a web program, email customer, datebook, mapping program, and an application for purchasing music, other media, or more applications. Some pre-introduced applications can be expelled by a customary uninstall process, hence leaving more storage room for wanted ones. Where the product does not permit this, a few gadgets can be attached to dispense with the undesired applications. Versatile application improvement requires the utilisation of specific coordinated advancement environments.Android is a portable working framework created by Google, in view of an adjusted form of the Linux piece and other open source programming and planned essentially for touchscreen cell phones, for example, cell phones and tablets.

FUNCTIONS: For controlling the Arduino board and performing computations. eg: digitalwrite(),digitalread()

VARIABLES: Arduino data types and constants are used to hold the value in a variable. eg: string(),int

STRUCTURE: The elements of Arduino (C++) code. For eg loop(),setup()

4.3.4 Communications Interfaces

L2CAP: logical link control and adaption protocol. Wi-Fi Security: WPA-PSK, WEP-64, open. Routing ADOV: Ad hoc on-Demand Distance vector DSR: Dynamic source routing B.A.T.M.A.N.: Better approach to mobile application network. (IOT) The inter-networking of physical devices, vehicles, etc. Embedded and electronics, s/w, sensor, actuators, and network connectivity that enable devices to communicate to other and/or exchange of data.

4.4 Nonfunctional Requirements

4.4.1 Performance Requirements

1. Data retrieving speed: the speed of retrieving data from sensor should be faster as possible, more delay may cause the wrong solution.

2. Data manipulation speed: Manipulating the fetched data and provided database must be faster, it may leads to miss

management and may not give the accurate solution according to current parameters.

3. Giving notification: Notification time should be normal, that farmer can get the whole the situation and fix the problem.
4. Display timing: After completing all the processes, farmer should know that what crop he has to choose next, if display not work properly then the farmer could not get the information.

4.4.2 Safety Requirements

- Use case :Use-case is used to prevent the circuit of the system form external circumstances, the circuit include Arduino, Wi-Fi module, sensors, wires, etc. they all should be protected from external affairs. Maintenance of the core circuit is done by use-case.
- IOT: IOT- Internet of things, it includes all the safety parameters regarding to the Internet such as http, https and other security parameters such as communication protocols, Wi-Fi protocols ,etc.
- Database: Database must be secure, because it contains all the information related to farmer, farm manager , crops details and much more. It should be prevent from external users, otherwise a

small change in database may leads dangerous failure of the system.

- Mobile app: Mobile application code must be private to developer, otherwise any one can change the environment related to application functionalists, features, components ,etc. So make the coding part private as possible.

4.4.3 Security Requirements

- **Availability:** The system should be available on any part of earth with internet connection as it supports IoT.
- **Correctness:** The data send and revice should be always accurate and correct
- **Flexibility:** The system should work with any android application.
- **Reliability:** The system should always be able to work according to the request to the farmer.
- **Maintainability:** The system doesn't need much maintenance thus it can be done easily

Chapter 5

System Design

5.1 System Requirements Definition

Arduino is an open source Hardware and Software based development platform.The Arduino microcontroller is easy to use yet powerful single board computer.It's combination of microcontroller based Arduino boards, Arduino programming Language and the Arduino software for development and compilation.

All the Sensors and actuators will be connected to the Arduino. A wifi module will be used to connect the Arduino to the Internet. Cloud server will provide an interface between the user and the database. Examples for cloud server are AWS, Thinkspeak etc. There are Arduino shields, which are fix on board. We can use them externally as well through ESP8266, GSM module and Zigbee.

To reach long distance over internet, Zigbee devices use mesh network of intermediate devices. Zigbee has a defined rate of 250 kbit/s, best suited for intermittent data transmissions from a sensor or input device.Where as GSM modules are used to send a message or miscall over a network. The ESP8266 is a low-cost Wi-Fi chip with on load full TCP/IP stack.

It's a transceiver that can be added to any existing microcontroller based setups via UART (serial link) to enable the system to communicate over the Internet via Wi-Fi. Commands will be sent from an android application through the internet and to Wi-Fi network to ESP8266 with Arduino Mega. For data storing we can save it on Google drive.

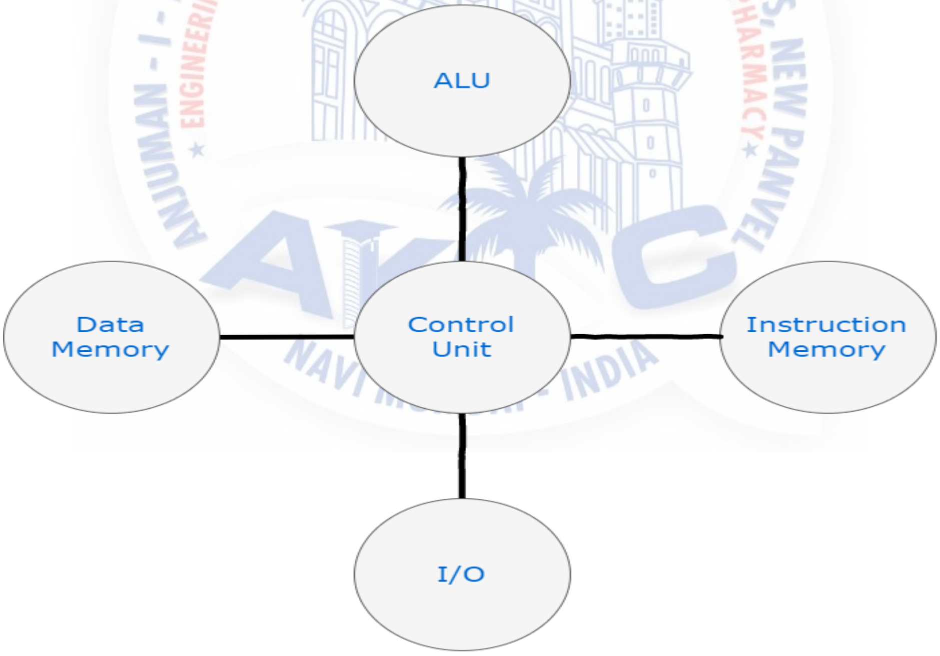


Figure 5.1: Arduino function block

5.1.1 Functional requirements

Every one of the Sensors and actuators will be associated with the Arduino. A wifi module will be utilised to associate the Arduino to the Internet. Cloud server will give an interface between the client and the database. Cases for cloud server are AWS, Thinkspeak and so forth. There are Arduino shields, which are settle on board. We can utilise them remotely too through ESP8266, GSM module and Zigbee.

Use-case Diagram

By using the Use case diagram, the user can monitor the greenhouse (view the environmental factors current values and history, actuators state, and weather information), perform a manual action (turn on or off an actuator), and view the notifications, the following diagram describe the functional requirement of the smart green house based on Iot.

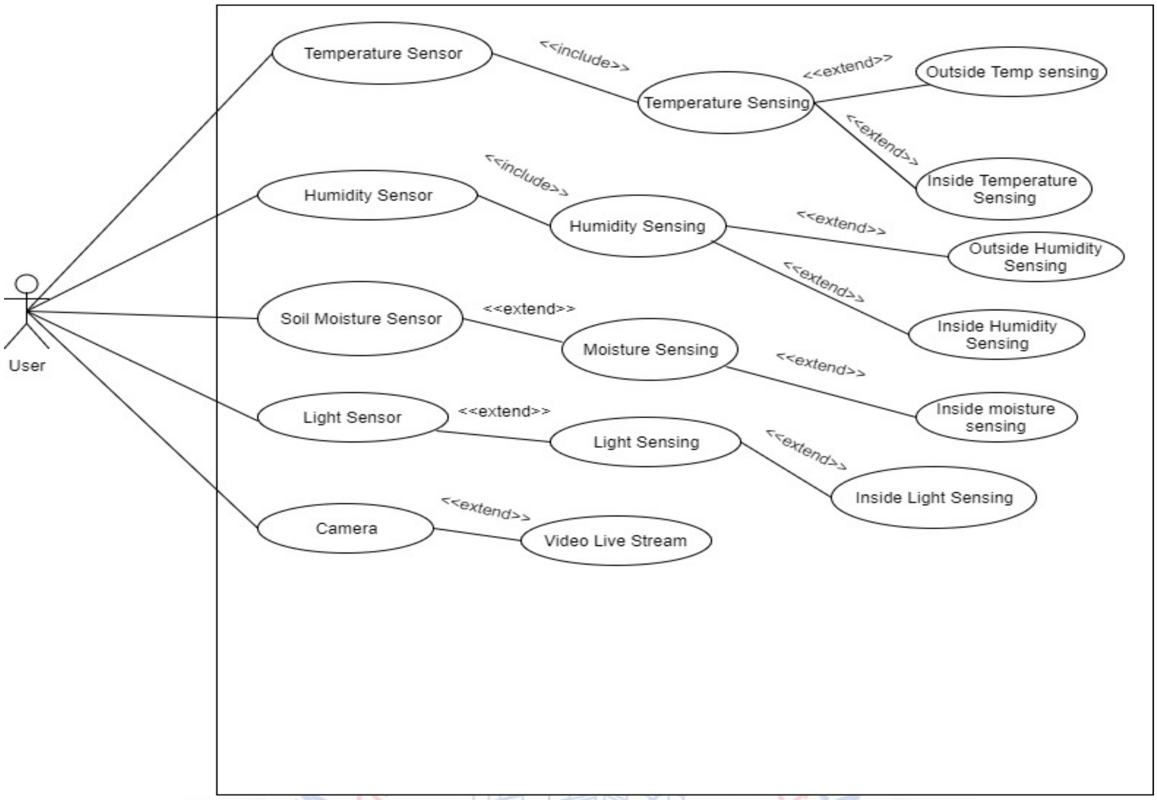


Figure 5.2: Usecase diagram for A Smart Green House System Based on IoT

Data-flow Diagram

A data flow diagram (DFD) maps out the flow of information for any process or system. It uses defined symbols like rectangles, circles and arrows, plus short text labels, to show data inputs, outputs, storage points and the routes between each destination. Data flowcharts can range from simple, even hand-drawn process overviews, to in-depth, multilevel DFDs that dig progressively deeper into how the data is handled. They can be used to analyse an existing system or model a new one.

Like all the best diagrams and charts, a DFD can often visually “say” things that would be hard to explain in words, and they work for both technical and nontechnical audiences, from developer to CEO. That's why DFDs remain so popular after all these years. While they work well for data flow software and systems, they are less applicable nowadays to visualising interactive, real-time or database-oriented software or systems.

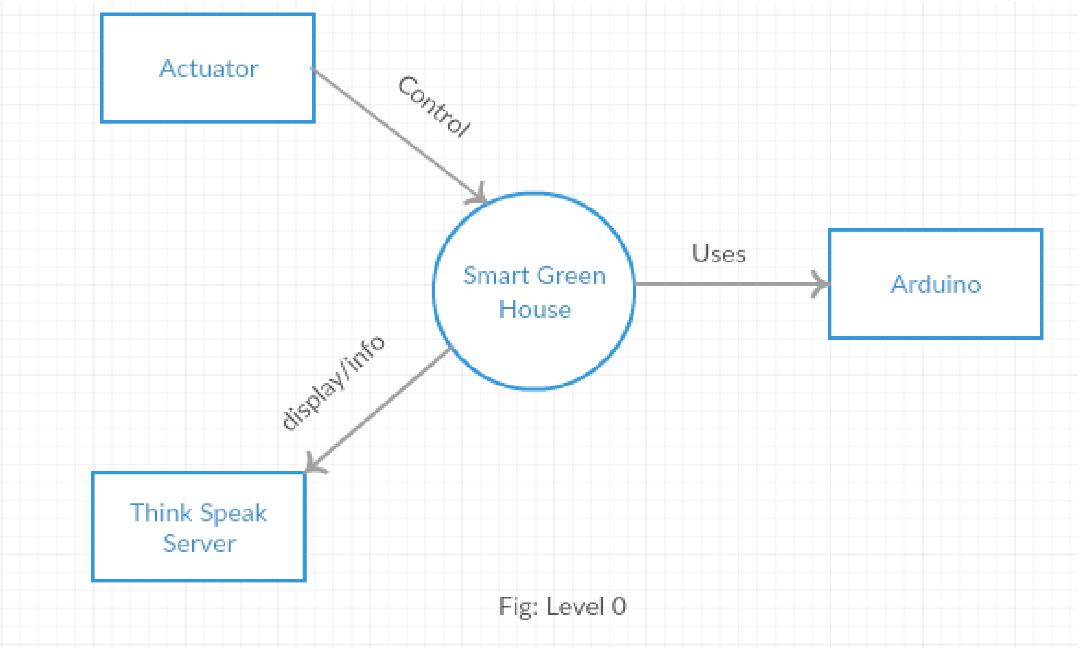


Figure 5.3: DFD Diagram Level 0

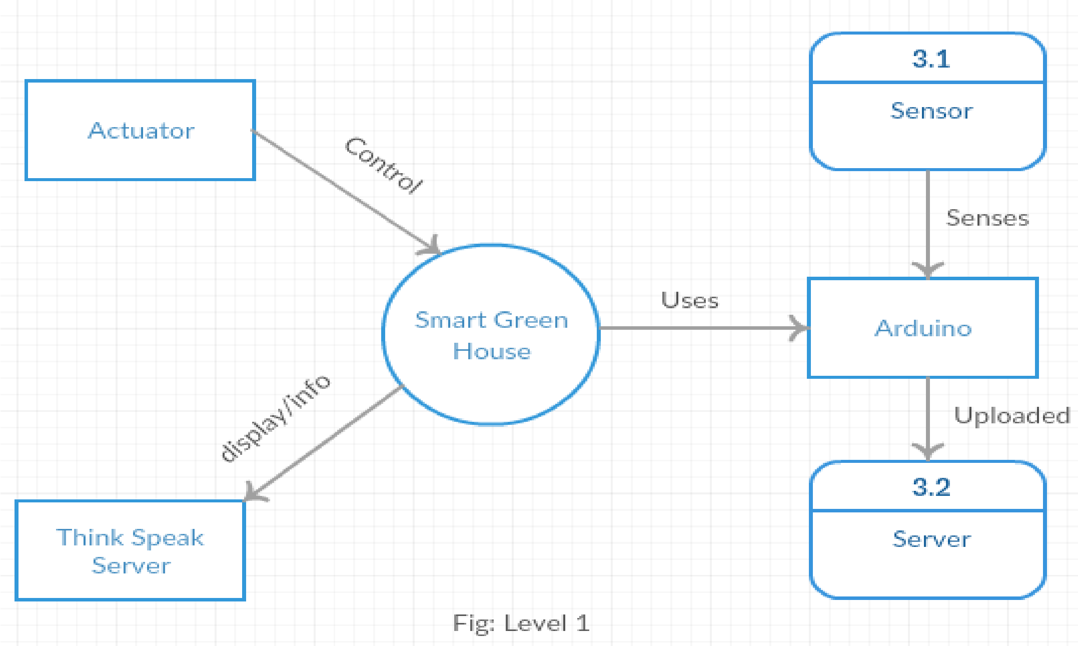


Figure 5.4: DFD Diagram Level 1

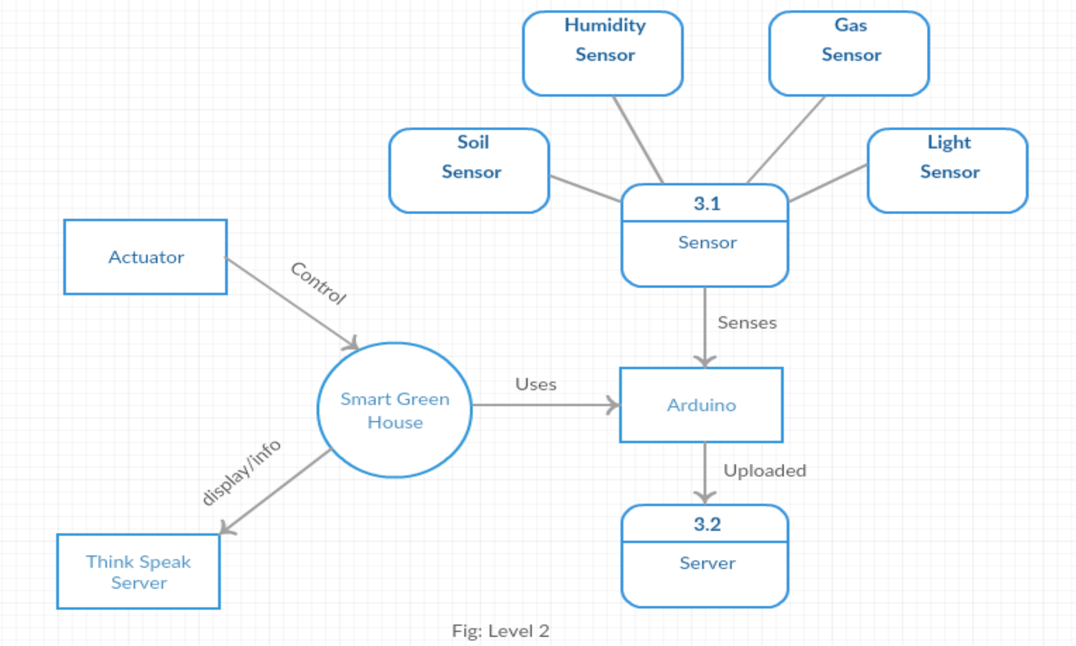


Figure 5.5: DFD Diagram Level 2

5.1.2 System requirements (non-functional requirements)

These are non-functional system properties such as availability, performance and safety etc. They define functions of a system, services and operational constraints in detail.

5.2 System Architecture Design

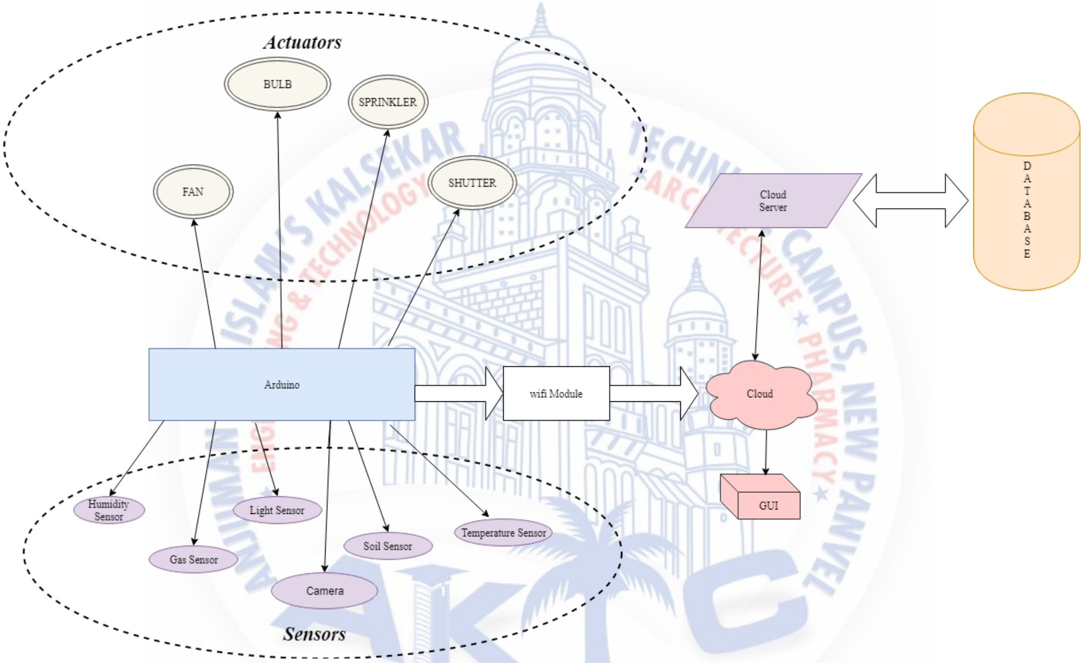


Figure 5.6: System architecture for A Smart Green House System Based on IoT

Arduino is an open source Hardware and Software based advancement platform.The Arduino microcontroller is anything but difficult to utilize yet capable single board computer.It’s mix of microcontroller based Arduino sheets, Arduino programming Language and the Arduino programming for improvement and gathering.

Every last one of the Sensors and actuators will be connected with the Arduino. A wifi module will be used to relate the Arduino to the Internet. Cloud server will give an interface between the client and the database. Cases for cloud server are AWS, Thinkspeak and whatnot.

It's a handset that can be added to any current microcontroller based setups through UART (serial connection) to empower the framework to convey over the Internet by means of Wi-Fi. Charges will be sent from an android application through the web and to Wi-Fi system to ESP8266 with Arduino Mega. For information putting away we can spare it on Google drive.

5.3 Sub-system Development

Our system is basically divided in to many parts but the main two are hardware and software. Hardware is further divided into sensors and transmission of data. We have explained in modules as follows:

5.3.1 Sensor testing

We tried to test all the sensors. We connected each sensor individually to the Arduino. For eg we connected DHT11 for temperature, humidity and dew point in the atmosphere of the environment. We understood the pin configuration of Arduino Uno and Dht11.

We associated every sensor independently to the Arduino. We associated Dht11 for temperature, stickiness and dew point in the climate of the habitat. We comprehended the pin layout of Arduino Uno and Dht11. We related each sensor autonomously to the Arduino.

We also had to download special library files for Dht11 to run. We also need to set com port for Arduino so that our laptop can identify it/We also need Arduino Software Library like SOFTWARESERIAL.H.

5.3.2 Wifi Connectivity

In this module tried to test ESP8266 by sending sensors value to our server thinkspeak. We used ESP8266 wifi module to connect our Arduino to the internet so that it can send data to our server. We tried to connect every sensor individually to the server seprately in this module.

Second module is sending the sensors values to our server ie thingspeak. We used ESP8266 wifi module to connect our Arduino to the internet so that it can send data to our server.

5.3.3 Status Shown

Third module of the system is data fetching in which the phpmyadmin data about the sensors is fetch from the respective servers. all this module are integrated in to one whole app module using the android functionality of integrating.

5.3.4 Joining all together

We integrated all the sensors while the sensors uploading the value to our server. We used thinkspeak for our server and we used shields for sensors as they were getting lose.

5.4 Systems Integration

First module of the system is connecting each sensors to the arduino individually and its pin configuration of dht11 and arduino Uno. Second module is sending the sensors values to our server ie thingspeak. We used ESP8266 wifi module to connect our Arduino to the internet so that it can send data to our server.Third module of the system is data fetching in which the phpmyadmin data about the sensors is fetch from the respective servers. all this module are integrated in to one whole app module using the android functionality of integrating.

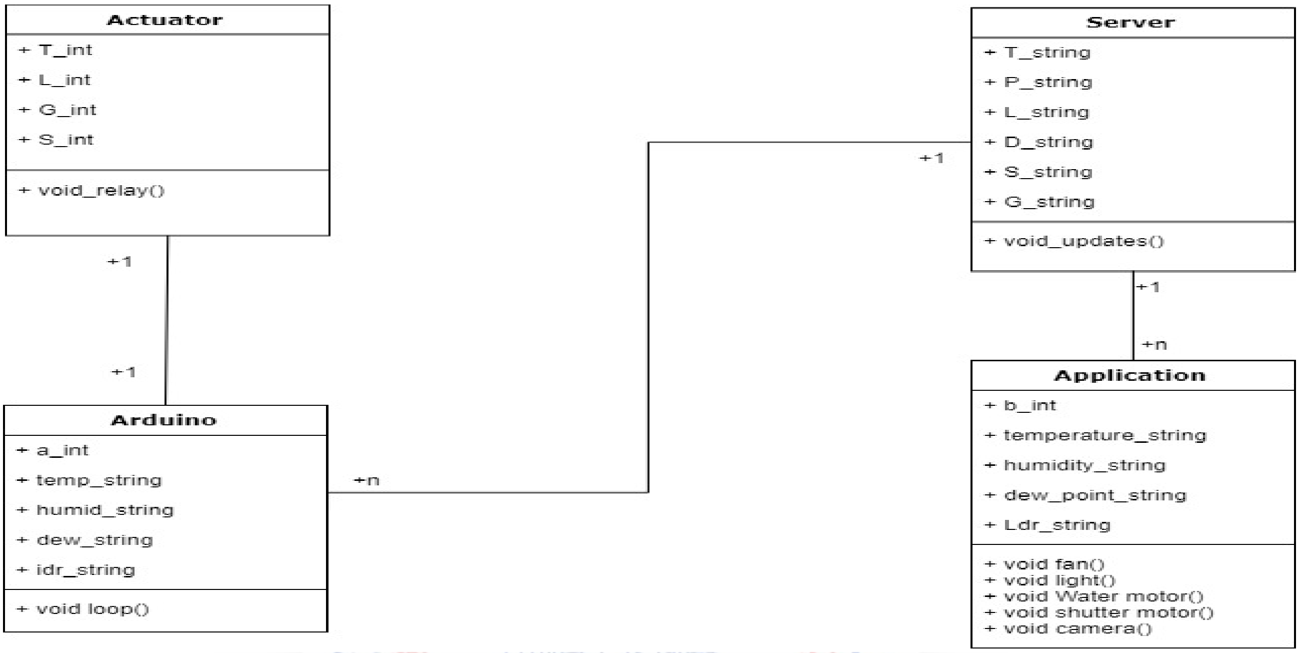


Figure 5.7: Class Diagram for A Smart Green House System Based on IoT

5.5 Class Diagram

This is the Class diagram of the system in which the modules which will be there are shown .The thingspeak server as well as the actuator, arduino and android application is shown for easy understanding of the project.

Sequence Diagram

The smart greenhouse using IoT, the sensors data will be transmitted through a set of different devices,to the cloud where it get processed, if a measurement crossed the threshold, the cloud sends a command

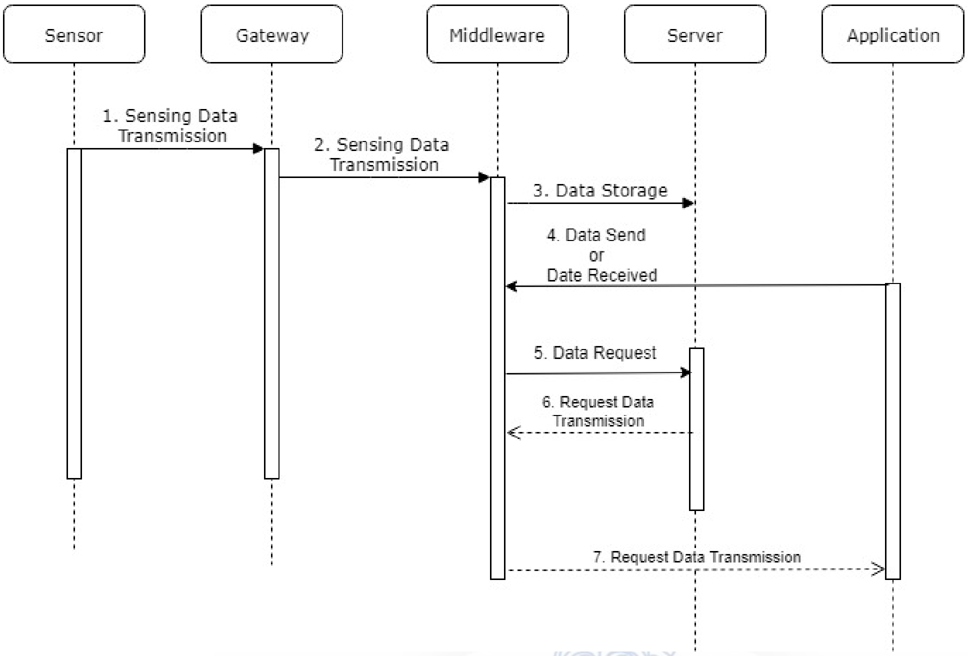

Sensor
Gateway
Middleware
Server
Application
1. Sensing Data Transmission
2. Sensing Data Transmission
3. Data Storage
4. Data Send or Date Received
5. Data Request
6. Request Data Transmission
7. Request Data Transmission

Figure 5.8: Sequence Diagram for A Smart Green House System Based on IoT

to the specified greenhouse in order to perform the suitable action.The diagram below (Fig. 9)describes greenhouse.

5.5.1 Component Diagram

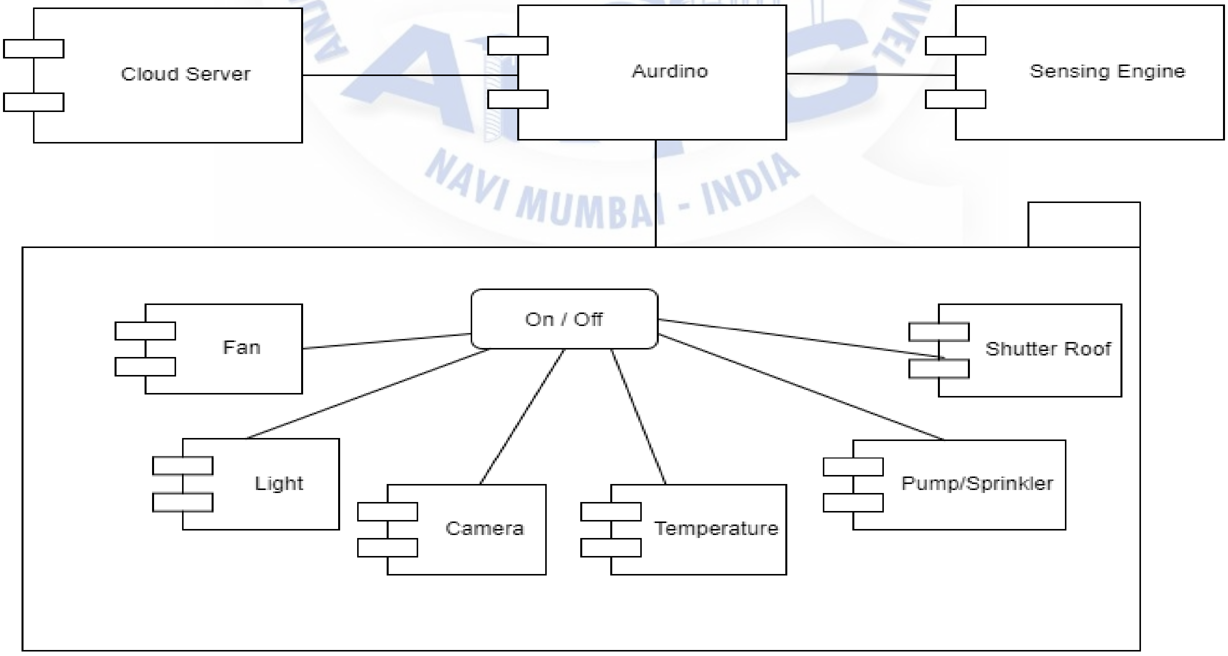


Figure 5.9: Component Diagram for A Smart Green House System Based on IoT

In component diagram it helps to understand the each module in better way forming its integrated sub module likewise shown in the diagram. Iot gateway allows a device to report data using its sensors to a remote location . Every one of these greenhouses is connected to the same gateway forming a network, this gateway collects the data from all the greenhouses, filters it (depending on message priority) and sends it periodically to the cloud, which leads to a low cost as IoT cloud platforms use messages count as a payment unit (1 message instead of N messages, where N is greenhouses count)

5.5.2 Deployment Diagram

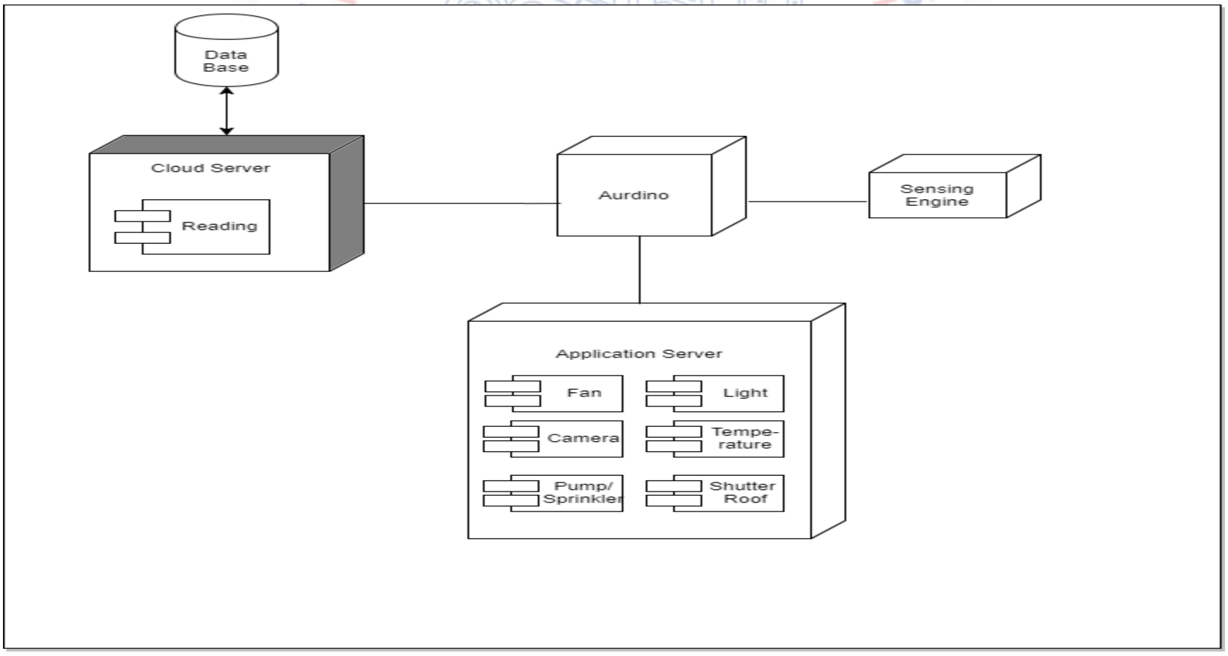


Figure 5.10: Deployment Diagram for A Smart Green House System Based on IoT

Because the application has three major module like arduino,sensing engine,cloud server each one for a specific goal. The following wireframe represent the result and explains the different parts that the application will haveLike the mobile application, the android application should offer a remote monitoring ability, information about the weatheroutside the greenhouses, display the running actuators and the ability to control greenhouse. This wireframe shows its functionalities:

5.5.3 Activity Diagram

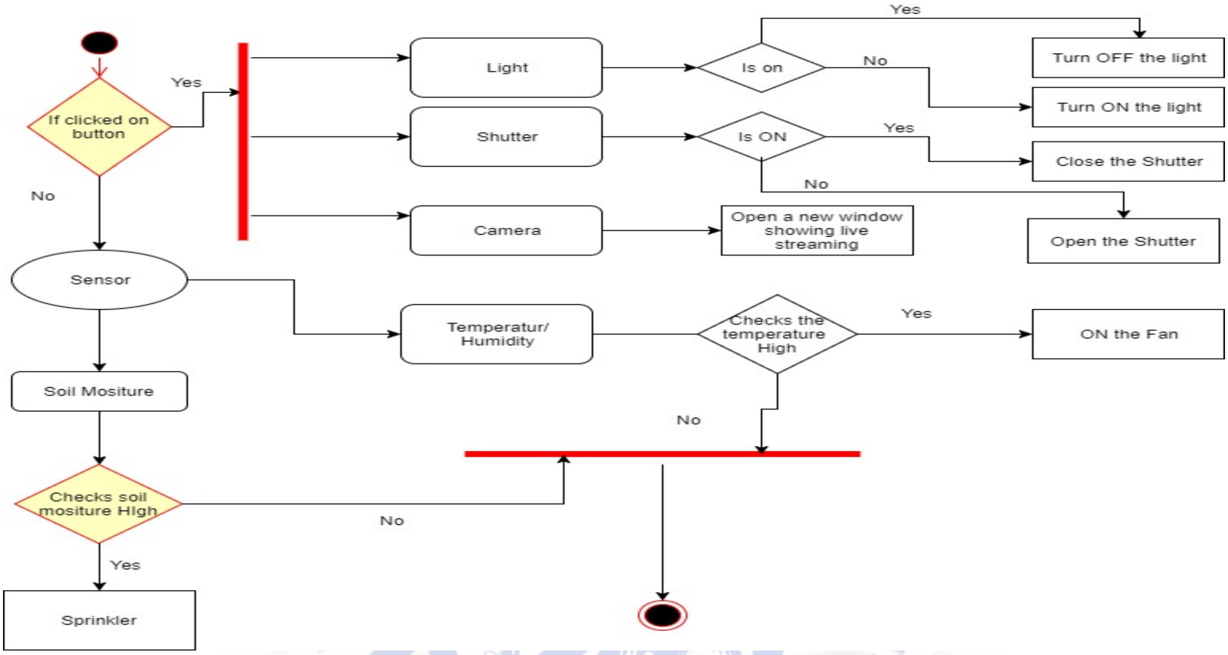


Figure 5.11: Activity Diagram for A Smart Green House System Based on IoT

Once authentication is successfully done, the application gets the data history and start sensing and displays it as a chart. When the user performs a manual action (enable/disable actuator of his choice), the application will send the command to the cloud, which will process it and send it to the gateway, then the gateway sends it to the specified greenhouse. The next sequence diagram shows the monitoring and controlling scenario (Fig. 5.7).

5.5.4 State Chart

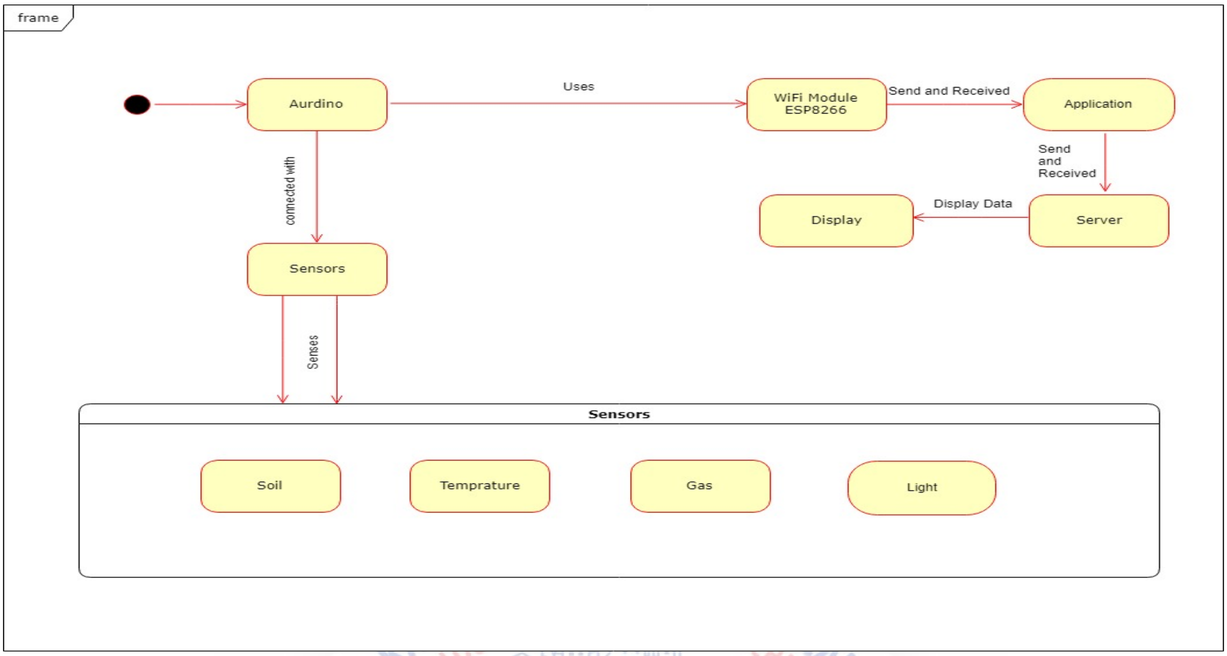


Figure 5.12: State Chart Diagram for A Smart Green House System Based on IoT

The state chart diagram below represents the overall greenhouse scenario.The idea behind the smart greenhouse based on Iot is to allow the farmer to consult the greenhouse status or history, and controlling it from anywhere with IoT, using a mobile application or a website. By using the android application, the user can monitor the greenhouse (view the environmental factors current values and history, actuators state, and weather information), perform a manual action (turn on or off an actuator), and view the notifications, the following diagram describe the use of it.

Chapter 6

Implementation

6.1 Sensors Module

In this module we tried to test all the sensors. We connected each sensor individually to the Arduino. For eg we connected DHT11 for temperature, humidity and dew point in the atmosphere of the environment. We understood the pin configuration of Arduino Uno and Dht11.

We associated every sensor independently to the Arduino. We associated Dht11 for temperature, stickiness and dew point in the climate of the habitat. We comprehended the pin layout of Arduino Uno and Dht11. We related each sensor autonomously to the Arduino.

We also had to download special library files for Dht11 to run. We also need to set com port for Arduino so that our laptop can identify it/We also need Arduino Software Library like SOFTWARESERIAL.H.

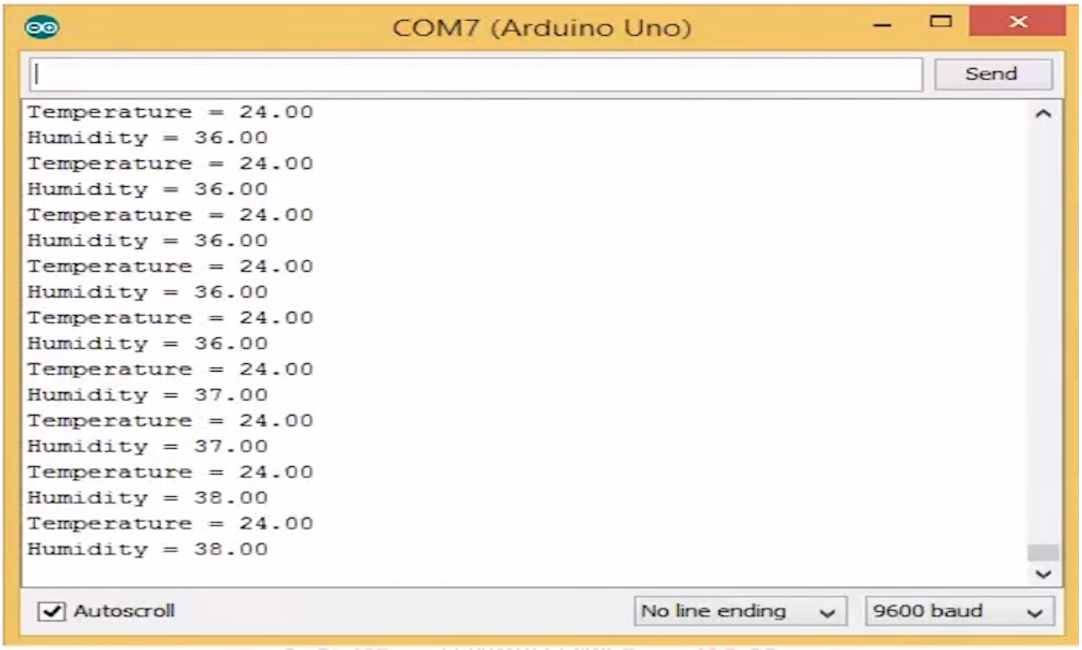
1. It is single wire digital interface
2. Low in cost
3. Ultra-small size around 12X15.5X5.5 mm
4. High reliability optimised long-term stability

Table 6.1: DHT11 Pin configuration

SR. No	Connection
1	VCC
2	Data
3	No connection
4	GND

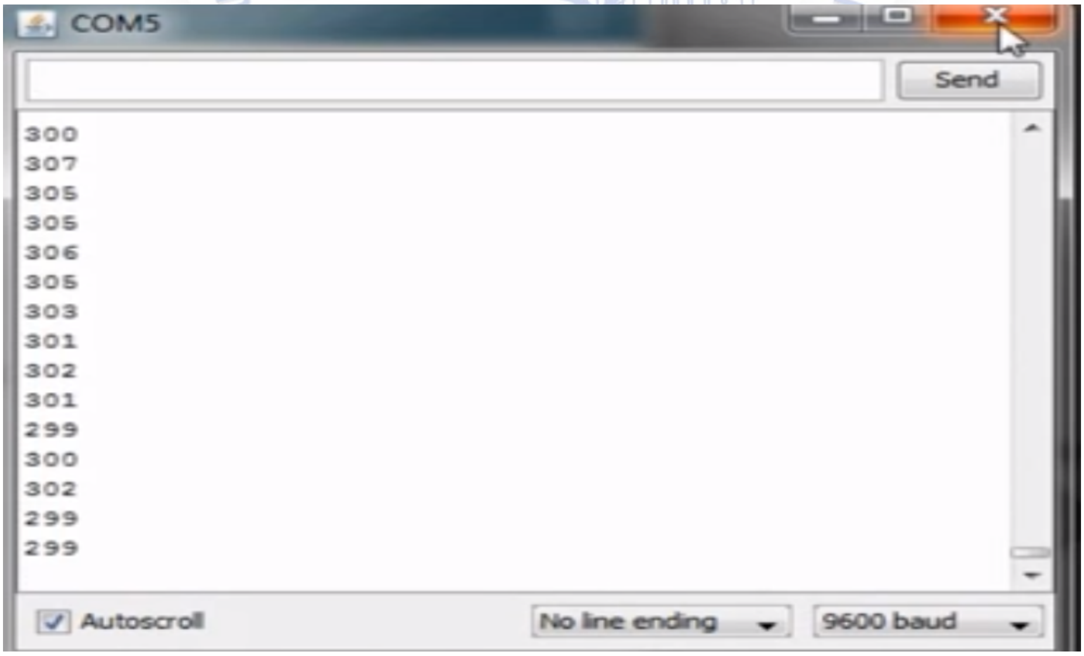
```
For Dht11
#include <SoftwareSerial.h> //include the software serial library
#include<dht.h> // include the dht library
void setup(){
  Serial.begin(9600);
  delay(500);//Delay to let system boot
  Serial.println(”DHT11 Humidity & temperature Sensor\n\n”);
  delay(1000);//Wait before accessing Sensor
}//end ”setup()”
void loop(){
  //Start of Program
    DHT.read11(dht_apin);
    Serial.print(”Current humidity = ”);
    Serial.print(DHT.humidity);
    Serial.print(”%  ”);
    Serial.print(”temperature = ”);
    Serial.print(DHT.temperature);
    Serial.println(”C  ”);
    delay(5000);//Wait 5 seconds before accessing sensor again.

}// end loop(

For soil sensor
void setup() {
  // initialize serial communication at 9600 bits per second:
```

```
  Serial.begin(9600);
}
void loop() {
  // read the input on analog pin 0:
  int sensorValue = analogRead(A0);

  Serial.println(sensorValue);
  delay(100);
}


Code for Ldr
#include <SoftwareSerial.h> //include the software serial library
int LDR = 0;      //analog pin to which LDR is connected , here we set it to 0 so it means A0
int LDRValue = 0;       // thats a variable to store LDR values
int light_sensitivity = 500;     //This is the approx value of light surrounding your LDR
void setup()
  {
    Serial.begin(9600);           //start the serial monitor with 9600 buad
    pinMode(13, OUTPUT);      //we mostly use 13 because there is already a built in yellow LED in arduino which shows output when 13 pin is enabled
  }
void loop()
  {
    LDRValue = analogRead(LDR);       //reads the ldrs value through LDR
    Serial.println(LDRValue);        //prints the LDR values to serial monitor
    delay(50);         //This is the speed by which LDR sends value to arduino

    if (LDRValue < light_sensitivity)
      {
        digitalWrite(13, HIGH);
      }
    else
      {
        digitalWrite(13, LOW);
      }
  }
```



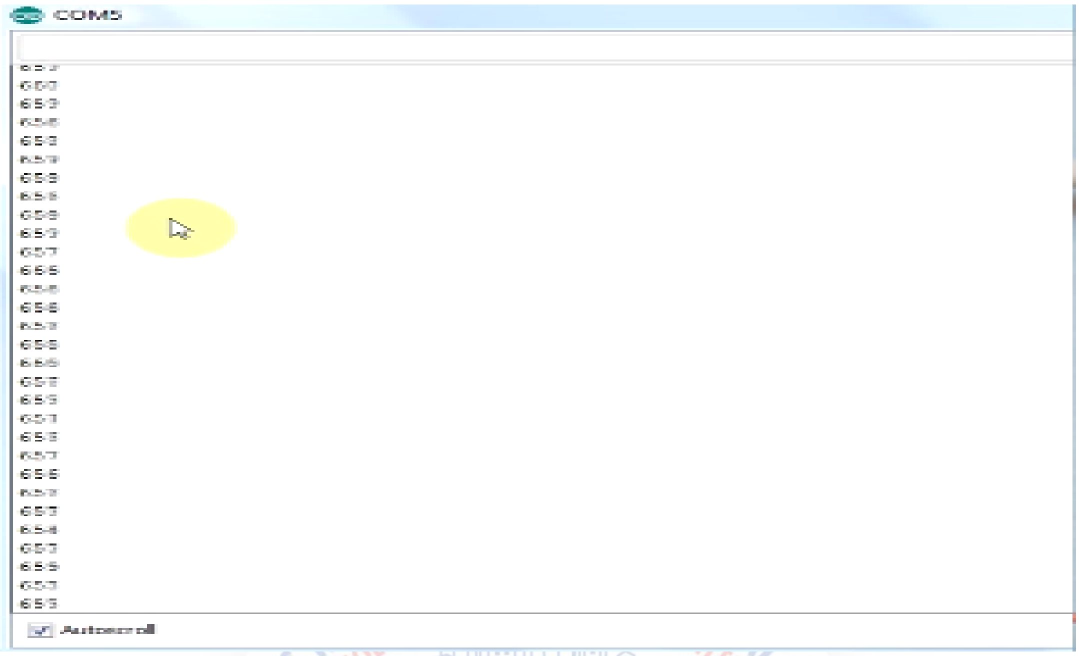
```
Temperature = 24.00
Humidity = 36.00
Temperature = 24.00
Humidity = 36.00
Temperature = 24.00
Humidity = 36.00
Temperature = 24.00
Humidity = 36.00
Temperature = 24.00
Humidity = 36.00
Temperature = 24.00
Humidity = 37.00
Temperature = 24.00
Humidity = 37.00
Temperature = 24.00
Humidity = 38.00
Temperature = 24.00
Humidity = 38.00
```

Figure 6.1: Screenshot on serial monitor for DHT11



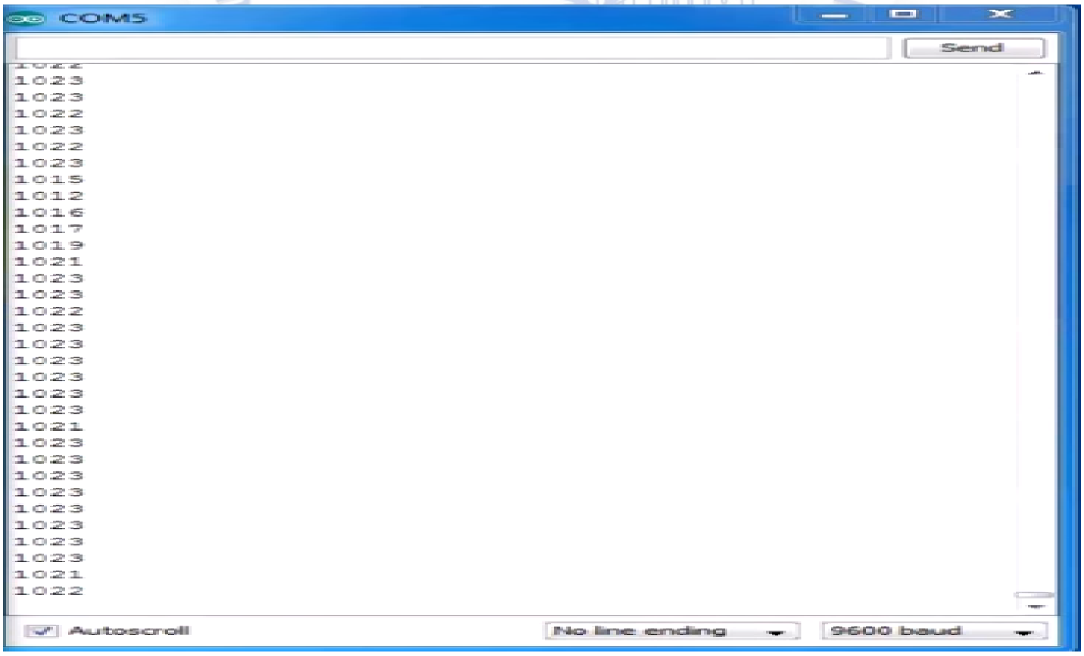
```
300
307
305
305
306
305
303
301
302
301
299
300
302
299
299
```

Figure 6.2: Screenshot on serial monitor for LDR



```
650
653
656
653
657
659
658
659
653
657
655
656
658
657
655
656
657
655
657
653
657
656
657
657
654
657
655
657
655
```

Figure 6.3: Screenshot on serial monitor for Gas



```
1023
1023
1022
1023
1022
1023
1015
1012
1016
1017
1019
1021
1023
1023
1022
1023
1023
1023
1023
1023
1023
1021
1023
1023
1023
1023
1023
1023
1023
1023
1021
1022
```

Figure 6.4: Screenshot on serial monitor for Soil sensor

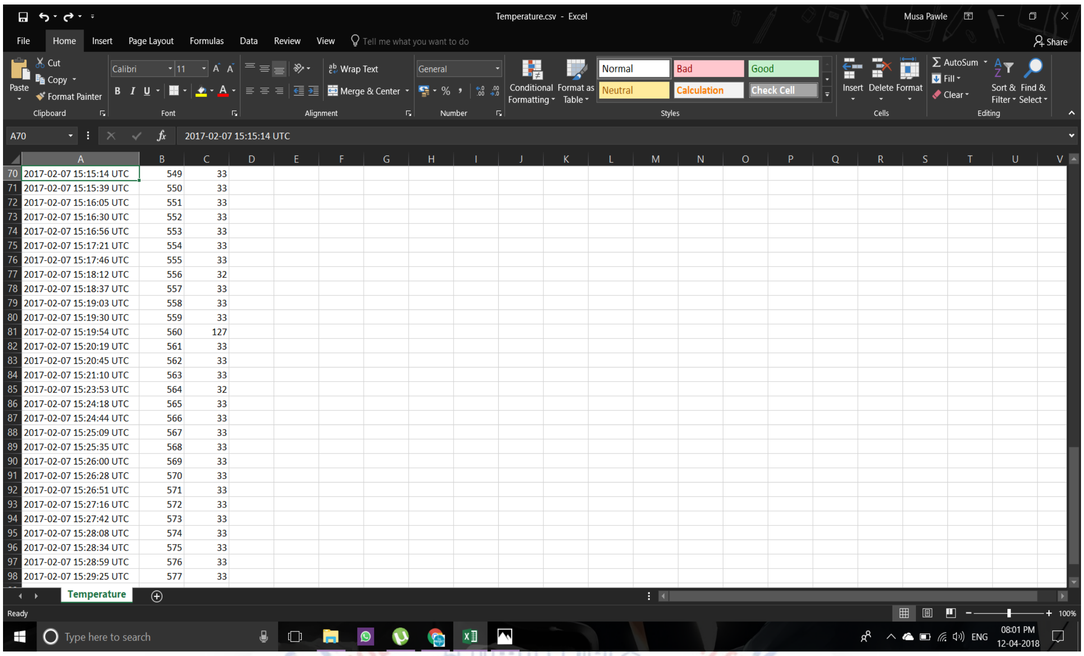
6.2 Wifi connectivity

In this module we tried to test ESP8266 by sending sensors value to our server thinkspeak. We used ESP8266 wifi module to connect our Arduino to the internet so that it can send data to our server. We tried to connect every sensor individually to the server separately in this module.

Table 6.2: ESP8266 Pin configuration

SR. No	Connection
1	GND
2	GPIO1
3	GPIO2
4	CH$_PD$
5	GPIO0
6	RESET
7	GPIO3
8	VCC

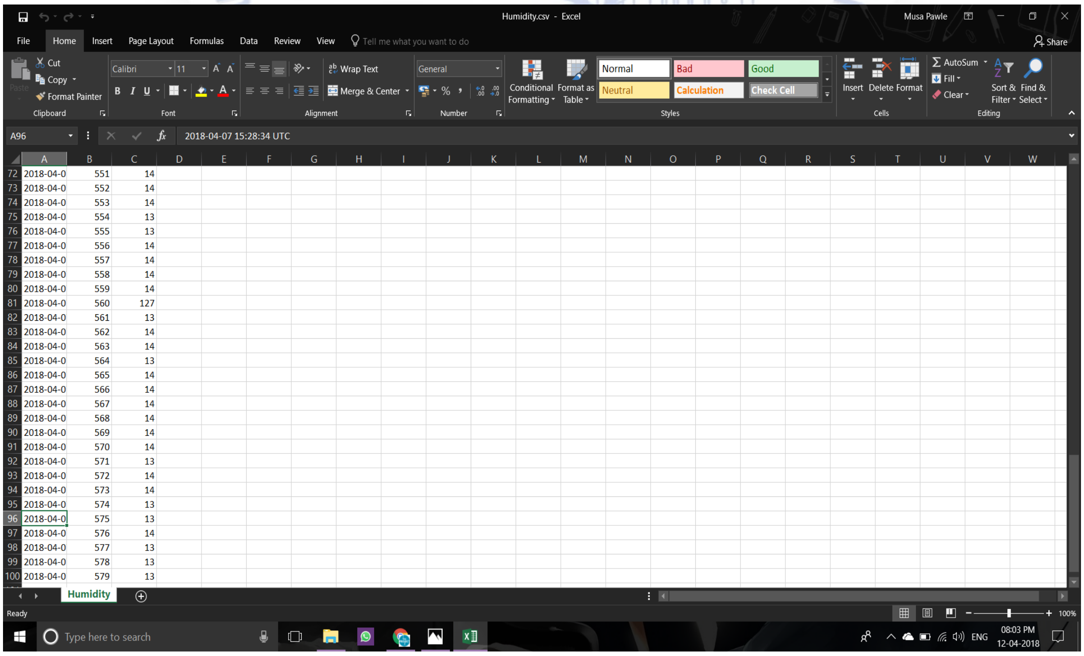
```
#include <SoftwareSerial.h> //include the software serial library
#include<dht.h> // include the dht library
dht DHT; // initiate the DHT obeject
int a; // a variable to store the raw sensor value
String temp; // Variable to store the temperature
String humid; // Variable to store the humidity
String dew; // Variable to store the dew point
SoftwareSerial esp8266(3, 4); //set the software serial pins RX pin = 3, TX pin = 4
//definition of variables
#define DEBUG true //show messages between ESP8266 and Arduino in serial port, when set to true
#define SSID ”Arzaan” //replace x with your wifi network name
#define PASS ”alohomora” //replace x with your wifi network password
String sendAT(String command, const int timeout, boolean debug)
{
  String response = ””;
  esp8266.print(command);
  long int time = millis();
  while ( (time + timeout) > millis())
  {
    while (esp8266.available())
    {
      char c = esp8266.read();
      response += c;
    }
  }
  if (debug)
  {
    Serial.print(response);
  }
  return response;
}
void setup()
{
  Serial.begin(115200);// begin the serial communication with baud of 9600
  esp8266.begin(115200);// begin the software serial communication with baud rate 9600
  sendAT(”AT+RST\r\n”, 2000, DEBUG); // call sendAT function to send reset AT command
  sendAT(”AT\r\n”, 1000, DEBUG);
  sendAT(”AT+CWMODE=1\r\n”, 1000, DEBUG); //call sendAT function to set ESP8266 to station mode
  sendAT(”AT+CWJAP=\””SSID”\”,\””PASS”\”\r\n”, 2000, DEBUG); //AT command to connect wit the wifi network
  while(!esp8266.find(”OK”)) { //wait for connection
  }
  sendAT(”AT+CIFSR\r\n”, 1000, DEBUG); //AT command to print IP address on serial monitor
  sendAT(”AT+CIPMUX=0\r\n”, 1000, DEBUG); //AT command to set ESP8266 to multiple connections
}
void loop(){
  /// put your main code here, to run repeatedly:
  a=DHT.read11(5); // read the sensor data and store it in variable a
  temp=DHT.temperature; // store the temperature values in temp variable
  humid=DHT.humidity;// store the humidity values in humid variable
  double gamma = log(DHT.humidity / 100) + ((17.62 * DHT.temperature) / (243.5 + DHT.temperature));
   double dp = 243.5 * gamma / (17.62 - gamma);
   dew= (String) dp;
  Serial.print("Temperature :"); // print the temperature on serial monitor
  Serial.println(temp); // Print the temperature value
  Serial.print("Humidity : ");//Print Humidity on serial monitor
  Serial.print(humid); // Print the humidity value on serial monitor
  Serial.print("Dew : ");//Print Humidity on serial monitor
  Serial.print(dew); // Print the humidity value on serial monitor
  Serial.print(" \n ");
  updateTS(temp,humid,dew); // call the function to update Thingspeak channel
  delay(3000);
}
void updateTS(String T,String P, String D){
  Serial.println("");
  sendAT("AT+CIPSTART=\"TCP\",\"api.thingspeak.com\",80\r\n", 1000, DEBUG);
  delay(2000);
  String cmdlen;
  String cmd="GET /update?key=YSBBBRDTOBT9XQG9&field1="+T+"&field2="+P+"&field3="+D+"\r\n"; // update the temprature and humidity values on thingspeak url,replace xxxxxxx with your write api key
   cmdlen = cmd.length();
  sendAT("AT+CIPSEND="+cmdlen+"\r\n", 2000, DEBUG);
   esp8266.print(cmd);
   Serial.println("");
   sendAT("AT+CIPCLOSE\r\n", 2000, DEBUG);
   Serial.println("");
   delay(15000);
   }
```



A70 | 2017-02-07 15:15:14 UTC

	A	B	C
70	2017-02-07 15:15:14 UTC	549	33
71	2017-02-07 15:15:39 UTC	550	33
72	2017-02-07 15:16:05 UTC	551	33
73	2017-02-07 15:16:30 UTC	552	33
74	2017-02-07 15:16:56 UTC	553	33
75	2017-02-07 15:17:21 UTC	554	33
76	2017-02-07 15:17:46 UTC	555	33
77	2017-02-07 15:18:12 UTC	556	32
78	2017-02-07 15:18:37 UTC	557	33
79	2017-02-07 15:19:03 UTC	558	33
80	2017-02-07 15:19:30 UTC	559	33
81	2017-02-07 15:19:54 UTC	560	127
82	2017-02-07 15:20:19 UTC	561	33
83	2017-02-07 15:20:45 UTC	562	33
84	2017-02-07 15:21:10 UTC	563	33
85	2017-02-07 15:23:53 UTC	564	32
86	2017-02-07 15:24:18 UTC	565	33
87	2017-02-07 15:24:44 UTC	566	33
88	2017-02-07 15:25:09 UTC	567	33
89	2017-02-07 15:25:35 UTC	568	33
90	2017-02-07 15:26:00 UTC	569	33
91	2017-02-07 15:26:28 UTC	570	33
92	2017-02-07 15:26:51 UTC	571	33
93	2017-02-07 15:27:16 UTC	572	33
94	2017-02-07 15:27:42 UTC	573	33
95	2017-02-07 15:28:08 UTC	574	33
96	2017-02-07 15:28:34 UTC	575	33
97	2017-02-07 15:28:59 UTC	576	33
98	2017-02-07 15:29:25 UTC	577	33

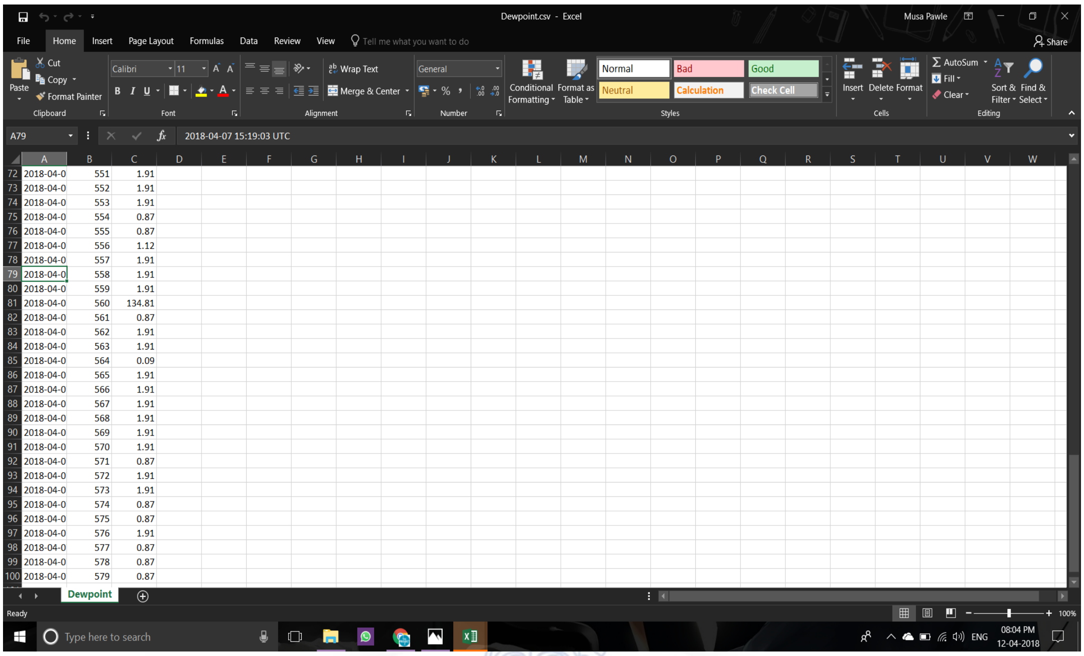
Figure 6.5: Screenshot of Temperature CSV file



A96 | 2018-04-07 15:28:34 UTC

	A	B	C
72	2018-04-0	551	14
73	2018-04-0	552	14
74	2018-04-0	553	14
75	2018-04-0	554	13
76	2018-04-0	555	13
77	2018-04-0	556	14
78	2018-04-0	557	14
79	2018-04-0	558	14
80	2018-04-0	559	14
81	2018-04-0	560	127
82	2018-04-0	561	13
83	2018-04-0	562	14
84	2018-04-0	563	14
85	2018-04-0	564	13
86	2018-04-0	565	14
87	2018-04-0	566	14
88	2018-04-0	567	14
89	2018-04-0	568	14
90	2018-04-0	569	14
91	2018-04-0	570	14
92	2018-04-0	571	13
93	2018-04-0	572	14
94	2018-04-0	573	14
95	2018-04-0	574	13
96	2018-04-0	575	13
97	2018-04-0	576	14
98	2018-04-0	577	13
99	2018-04-0	578	13
100	2018-04-0	579	13

Figure 6.6: Screenshot of Humidity CSV file



A79 2018-04-07 15:19:03 UTC

	A	B	C
72	2018-04-0	551	1.91
73	2018-04-0	552	1.91
74	2018-04-0	553	1.91
75	2018-04-0	554	0.87
76	2018-04-0	555	0.87
77	2018-04-0	556	1.12
78	2018-04-0	557	1.91
79	2018-04-0	558	1.91
80	2018-04-0	559	1.91
81	2018-04-0	560	134.81
82	2018-04-0	561	0.87
83	2018-04-0	562	1.91
84	2018-04-0	563	1.91
85	2018-04-0	564	0.09
86	2018-04-0	565	1.91
87	2018-04-0	566	1.91
88	2018-04-0	567	1.91
89	2018-04-0	568	1.91
90	2018-04-0	569	1.91
91	2018-04-0	570	1.91
92	2018-04-0	571	0.87
93	2018-04-0	572	1.91
94	2018-04-0	573	1.91
95	2018-04-0	574	0.87
96	2018-04-0	575	0.87
97	2018-04-0	576	1.91
98	2018-04-0	577	0.87
99	2018-04-0	578	0.87
100	2018-04-0	579	0.87

Figure 6.7: Screenshot of Dewpoint CSV file

6.3 Connectivity to thingspeak server

We integrated all the sensors while the sensors uploading the value to our server. We used thinkspeak for our server and we used shields for sensors as they were getting lose.

Table 6.3: LDR Pin configuration

SR. No	Connection
1	VCC
2	GND
3	DO
4	AO

Table 6.4: LDR Pin configuration

SR. No	Connection
1	AO
2	GND
3	VCC

```
\begin{figure}[htbp!]
\centering
    \includegraphics[height=09cm,width=15cm]{project/images/ScreenShorts/allsen1
        .png}
```

```
  \caption{Screenshot of Thinkspeak graph 1}
  \label{fig:Thinkspeak graph 1}
\end{figure}
\begin{figure}[htbp!]
\centering
    \includegraphics[height=09cm,width=15cm]{project/images/ScreenShorts/Allsens2.png}
  \caption{Screenshot of Thinkspeak graph 2}
  \label{fig:Thinkspeak graph 2}
\end{figure}
\begin{figure}[htbp!]
\centering
    \includegraphics[height=09cm,width=15cm]{project/images/ScreenShorts/Allsen3.png}
  \caption{Screenshot of Thinkspeak graph 3}
  \label{fig:Thinkspeak graph 3}
\end{figure}


\begin{lstlisting}
#include <SoftwareSerial.h> //include the software serial library
#include<dht.h> // include the dht library
dht DHT; // initiate the DHT obeject
int a; // a variable to store the raw sensor value
String temp; // Variable to store the temperature
String humid; // Variable to store the humidity
String dew; // Variable to store the dew point
String ldr;//Variable to store the ldr value
String soil;//Variable to store the soil sensor value
String gas;//Variable to store the gas sensor value
SoftwareSerial esp8266(3,4); //set the software serial pins RX pin = 3, TX pin = 4
//definition of variables
#define DEBUG true //show messages between ESP8266 and Arduino in serial port, when set to true
#define SSID ”Arzaan” //replace x with your wifi network name
#define PASS ”alohomora” //replace x with your wifi network password
String sendAT(String command, const int timeout, boolean debug)
{
  String response = ””;
  esp8266.print(command);
  long int time = millis();
  while ( (time + timeout) > millis())
  {
    while (esp8266.available())
    {
      char c = esp8266.read();
      response += c;
    }
  }
  if (debug)
  {
    Serial.print(response);
  }
  return response;
}
void setup()
{
  Serial.begin(115200);// begin the serial communication with baud of 9600
  esp8266.begin(115200);// begin the software serial communication with baud rate 9600
  sendAT(”AT+RST\r\n”, 2000, DEBUG); // call sendAT function to send reset AT command
  sendAT(”AT\r\n”, 1000, DEBUG);
  sendAT(”AT+CWMODE=1\r\n”, 1000, DEBUG); //call sendAT function to set ESP8266 to station mode
  sendAT(”AT+CWJAP=\””SSID”\”,\””PASS”\”\r\n”, 2000, DEBUG); //AT command to connect wit the wifi network
  while(!esp8266.find(”OK”)) { //wait for connection
  }
  sendAT(”AT+CIFSR\r\n”, 1000, DEBUG); //AT command to print IP address on serial monitor
  sendAT(”AT+CIPMUX=0\r\n”, 1000, DEBUG); //AT command to set ESP8266 to multiple connections
}
void loop(){
  /// put your main code here, to run repeatedly:
  a=DHT.read11(5); // read the sensor data and store it in variable a
  temp=DHT.temperature; // store the temperature values in temp variable
  humid=DHT.humidity;// store the humidity values in humid variable
  double gamma = log(DHT.humidity / 100) + ((17.62 * DHT.temperature) / (243.5 + DHT.temperature));
```

```
   double dp = 243.5 * gamma / (17.62 - gamma);
   d/ew= (String) dp;
   ldr=digitalRead(6);
   soil= digitalRead(7);
   gas= digitalRead(8);
  Serial.print("Temperature :"); // print the temperature on serial monitor
  Serial.println(temp); // Print the temperature value
  Serial.print(" Humidity : ");//Print Humidity on serial monitor
  Serial.print(humid); // Print the humidity value on serial monitor
  Serial.print(" Dew : ");//Print Dew :  on serial monitor
  Serial.print(dew); // Print the dew point value on serial monitor
  Serial.print(" LDR :");
  Serial.print(ldr);// Print the LDR value on serial monitor
  Serial.print(" soil sensor :");
  Serial.print(soil);// Print the Soil sensor value on serial monitor
    Serial.print(" gas sensor :");
  Serial.print(gas);// Print the gas sensor value on serial monitor
  Serial.print(" \n ");
  updateTS(temp,humid,dew,ldr,soil,gas); // call the function to update Thingspeak channel
  delay(3000);
}
void updateTS(String T,String P, String D, String L, String S, String G){
  Serial.println("");
  sendAT("AT+CIPSTART=\"TCP\",\"api.thingspeak.com\",80\r\n", 500, DEBUG);
  delay(1000);
  String cmdlen;
  String cmd="GET /update?key=YSBBBRDTOBT9XQG9&field1="+T+"&field2="+P+"&field3="+D+"&field4="+L+"&field5="+S+"&field6="+G+"\r\n"; // update the temprature and humidity values on thingspeak url,replace xxxxxxx with your write api key
   cmdlen = cmd.length();
  sendAT("AT+CIPSEND="+cmdlen+"\r\n", 1000, DEBUG);
   esp8266.print(cmd);
   Serial.println("");
   sendAT("AT+CIPCLOSE\r\n", 1000, DEBUG);
   Serial.println("");
   delay(1000);
   }
```

6.4 Communication between hardware and software

Arduino will send a signal to relay accordingly. The responsible relay will be set to activate the actuator. As discussed in the above eg if light is not present then ldr will inform Arduino and then the responsible actuator i.e a light panel will be turned on.

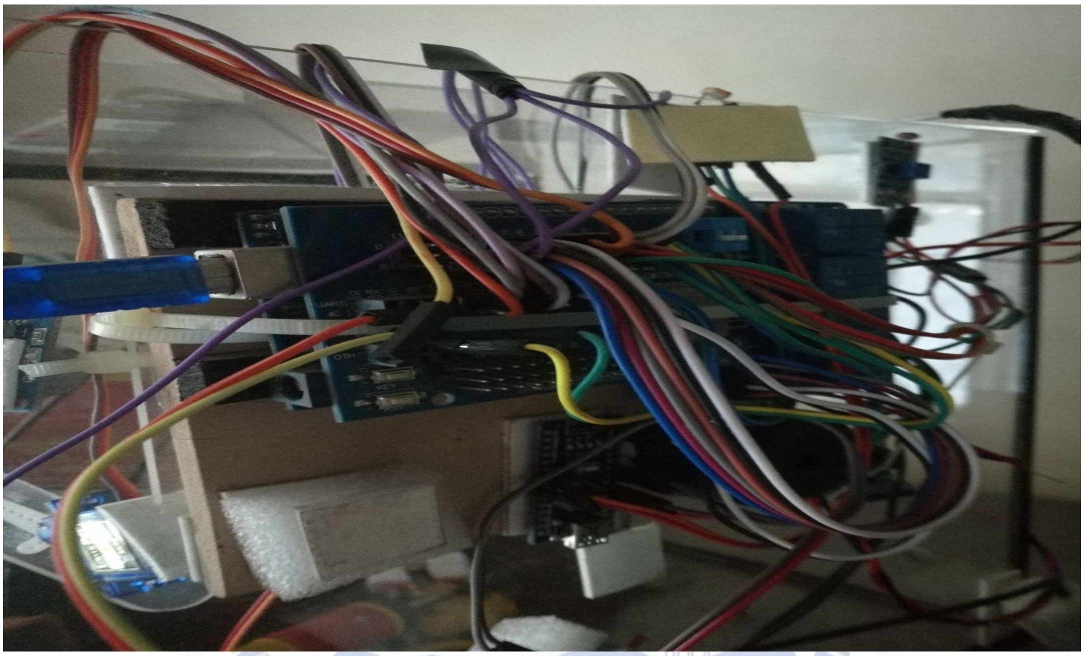


Figure 6.8: System picture1

```
#include <SoftwareSerial.h> //include the software serial library
#include<Servo.h>
#include<dht.h> // include the dht library
dht DHT; // initiate the DHT obeject
int a; // a variable to store the raw sensor value
String temp; // Variable to store the temperature
String humid; // Variable to store the humidity
String dew; // Variable to store the dew point
String ldr;//Variable to store the ldr value
String soil;//Variable to store the soil sensor value
String gas;//Variable to store the gas sensor value
SoftwareSerial esp8266(3,4); //set the software serial pins RX pin = 3, TX pin = 4
//definition of variables
#define DEBUG true //show messages between ESP8266 and Arduino in serial port, when set to true
#define SSID ”Arzaan” //replace x with your wifi network name
#define PASS ”alohomora” //replace x with your wifi network password
String sendAT(String command, const int timeout, boolean debug)
{
  String response = ””;
  esp8266.print(command);
  long int time = millis();
  while ( (time + timeout) > millis())
  {
    while (esp8266.available())
    {
      char c = esp8266.read();
      response += c;
    }
  }
  if (debug)
  {
    Serial.print(response);
  }
  return response;
}
//——————————————————Relay initialization
int tempR;// integer for temp relay
int ldrR;// integer for ldr relay
int soilR;// integer for soil relay
int gasR;// integer for gas relay
//--------------------------------------------END
Servo left;
Servo right;
int pos = 0;
/*
//----------------------------------------------OLED Display
#include <Wire.h>
#include <Adafruit_SSD1306.h>
#include <Adafruit_GFX.h>
Adafruit_SSD1306 display(-1);
#if (SSD1306_LCDHEIGHT != 64)
#error(”Height incorrect, please fix Adafruit_SSD1306.h!”);
#endif
//----------------------------------------------------END8*/
void setup()
{
  Serial.begin(115200);// begin the serial communication with baud of 9600
  esp8266.begin(115200);// begin the software serial communication with baud rate 9600
  sendAT(”AT+RST\r\n”, 2000, DEBUG); // call sendAT function to send reset AT command
  sendAT(”AT\r\n”, 1000, DEBUG);
  sendAT(”AT+CWMODE=1\r\n”, 1000, DEBUG); //call sendAT function to set ESP8266 to station mode
  sendAT(”AT+CWJAP=\””SSID”\”,\””PASS”\”\r\n”, 2000, DEBUG); //AT command to connect wit the wifi network
  while(!esp8266.find(”OK”)) { //wait for connection
  }
  sendAT(”AT+CIFSR\r\n”, 1000, DEBUG); //AT command to print IP address on serial monitor
  sendAT(”AT+CIPMUX=0\r\n”, 1000, DEBUG); //AT command to set ESP8266 to multiple connections
//------------------------Relay setup
  pinMode(10, OUTPUT);
  digitalWrite(10, HIGH);
    pinMode(11, OUTPUT);
  digitalWrite(11, HIGH);
   pinMode(12, OUTPUT);
  digitalWrite(12, HIGH);
  pinMode(9, OUTPUT);
  digitalWrite(9, HIGH);
   left.attach(7);
   right.attach(A3);
   left.write(0);
    right.write(0);
 /* //------------------------------------OLED display setup
// initialize and clear display
  display.begin(SSD1306_SWITCHCAPVCC, 0x3C);
  display.clearDisplay();
  display.display();
  // display a line of text
  display.setTextSize(1);
  display.setTextColor(WHITE);*/
}
void loop(){
  /// put your main code here, to run repeatedly:
  a=DHT.read11(5); // read the sensor data and store it in variable a
  temp=DHT.temperature; // store the temperature values in temp variable
  tempR= DHT.temperature;// store the temperature value in int
  humid=DHT.humidity;// store the humidity values in humid variable
  double gamma = log(DHT.humidity / 100) + ((17.62 * DHT.temperature) / (243.5 +
      DHT.temperature));
   double dp = 243.5 * gamma / (17.62 - gamma);
   dew= (String) dp;
   ldr=digitalRead(6);
   ldrR=digitalRead(6);
   soil= analogRead(A2);
   soilR=analogRead(A2);
   gas= digitalRead(A0);
   gasR= digitalRead(A0);
  Serial.print(”Temperature :”); // print the temperature on serial monitor
  Serial.println(temp); // Print the temperature value
  Serial.print(” Humidity : ”);//Print Humidity on serial monitor
  Serial.print(humid); // Print the humidity value on serial monitor
  Serial.print(” Dew : ”);//Print Dew : on serial monitor
  Serial.print(dew); // Print the dew point value on serial monitor
  Serial.print(” LDR :”);
  Serial.print(ldr);// Print the LDR value on serial monitor
  Serial.print(” soil sensor :”);
  Serial.print(soil);// Print the Soil sensor value on serial monitor
  Serial.print(” gas sensor :”);
  Serial.print(gas);// Print the gas sensor value on serial monitor
  Serial.print(” \n ”);
  updateTS(temp,humid,dew,ldr,soil,gas); // call the function to update Thingspeak channel
 Relay(tempR, ldrR,  soilR, gasR);
 // oled(temp,humid,dew,ldr,soil,gas);
  delay(3000);
}
void updateTS(String T,String P, String D, String L, String S, String G){
  Serial.println(””);
  sendAT(”AT+CIPSTART=\”TCP\”,\”api.thingspeak.com\”,80\r\n”, 1000, DEBUG);
  delay(2000);
  String cmdlen;
  String cmd=”GET /update?key=YSBBBRDTOBT9XQG9&field1=”+T+”&field2=”+P+”&field3=”+D+”&field4=”+L+”&field5=”+S+”&field6=”+G+”\r\n”; // update the temprature and humidity values on thingspeak url,replace xxxxxxx with your write api key
   cmdlen = cmd.length();
  sendAT(”AT+CIPSEND=”+cmdlen+”\r\n”, 2000, DEBUG);
   esp8266.print(cmd);
   Serial.println(””);
   sendAT(”AT+CIPCLOSE\r\n”, 2000, DEBUG);
   Serial.println(””);
   delay(15000);
   }
void Relay(int T, int L, int S, int G){
digitalWrite(10, HIGH);
digitalWrite(11, HIGH);
digitalWrite(12, HIGH);
digitalWrite(2, HIGH);
   if(T>=30)
  {
    digitalWrite(10, LOW);        // sets the digital pin 10 off to make Relay on for fan
  delay(100);
  }
  else
  {
    digitalWrite(10, HIGH);        // sets the digital pin 10 off to make Relay off for fan
  delay(100);
  }
if(L==0)
  {
    digitalWrite(11, HIGH);        // sets the digital pin 10 off to make Relay on for fan
  delay(100);
  }
  else
  {
    digitalWrite(11, LOW);        // sets the digital pin 10 off to make Relay off for fan
  delay(100);
  }
 if(S>650)
  {
    digitalWrite(12, LOW);        // sets the digital pin 10 off to make Relay on for fan
  delay(100);
  }
  else
  {
    digitalWrite(12, HIGH);        // sets the digital pin 10 off to make Relay off for fan
  delay(100);
  }
   if(G==0)
  {
    digitalWrite(9, LOW);        // sets the digital pin 10 off to make Relay on for fan
    for (pos = 0; pos <= 90; pos += 1) {
    left.write(pos);
    right.write(pos);
    delay(15);
    }
  delay(50);
  }
  else
  {
    digitalWrite(9, HIGH);        // sets the digital pin 10 off to make Relay off for fan
```

```
          left.write(0);
        right.write(0);
    delay(100);
    }
}
```

Chapter 7

System Testing

We tried to test sensors individually 1st and then we tried to upload the data to the server individually and then integrated the system for the hardware part.

7.1 Test Cases and Test Results

Test ID	Test Case Title	Test Condition	System Behavior	Expected Result
T01	Individual Sensors	All the sensors should work fine indivually	Gas sensor mq135 wasn't working properly	All sensors working fine

T02	Server uploading	Arduino should send data to the server	All the sensors were sending data to the internet but connection was loose so it use to get disconnected	All the sensors were sending data to the internet
T03	Integration	Sensors integrated to Arduino should sens and send data over internet to the server	All the sensors were sensing and sending data over internet	All the sensors should sensing and sending data over internet
T04	Android	control system with Android application	The Android application should control the environment	Android wasn't anle to send data properly

7.2 Sample of a Test Case for sensor

Working of sensors: Sensors should sens the environment and send the data to Arduino.

Description: All the sensors should sens the data and fetch the environment condition and provide it to Arduino. For eg DHT 11 will send data of temperature, Humidity and dew point.

Precondition: Arduino was idle

Assumption: Every one of the sensors should sens the information and give it to Arduino

Test Steps:

1. Connect all the sensors to Arduino board.
2. Embed programmed with the same pin you have used to connect on board.
3. Provide power supply to the Arduino board.
4. Use Serial monitor to see the output of the programme

Expected Result: Every one of the sensors should sens the information and get the earth condition and give it to Arduino.

Actual Result: Every one of the sensors should sens the information and get the earth condition and give it to Arduino.

Integration of sensors: Sensors should sens the environment and send the data to thinkspeak.

Description: Reconciliation of sensors – Sensors should sens nature and send the information to thinkspeak. Thinkspeak will get the information and make a graphical portrayal. Arduino will use ESP8266 wifi module to send the data over internet to the server. In other words

Arduino will utilise ESP8266 wifi module to send the information over web to the server. Thinkspeak will receive the data and create a graphical representation. *Precondition:* The wifi should be on so that ESP8266 can access it and the code must have the name and password of the hot-spot. *Assumption:* Data is represented in a graphical format by the server

Test Steps:

1. Connect all the sensors to Arduino board.
2. Remove the connection for Tx and Rx to the Arduino board.
3. Embed programmed with the same pin you have used to connect on board.
4. Connect the Tx and Rx pin back to Arduino.
5. Provide power supply to the Arduino board.
6. Log-in to the thinkspeak to check the working.

Expected Result: The data sensed by the sensors is getting updated on the thinkspeak server. **Actual Result:** The data sensed by the sensors should be updated on the thinkspeak server.

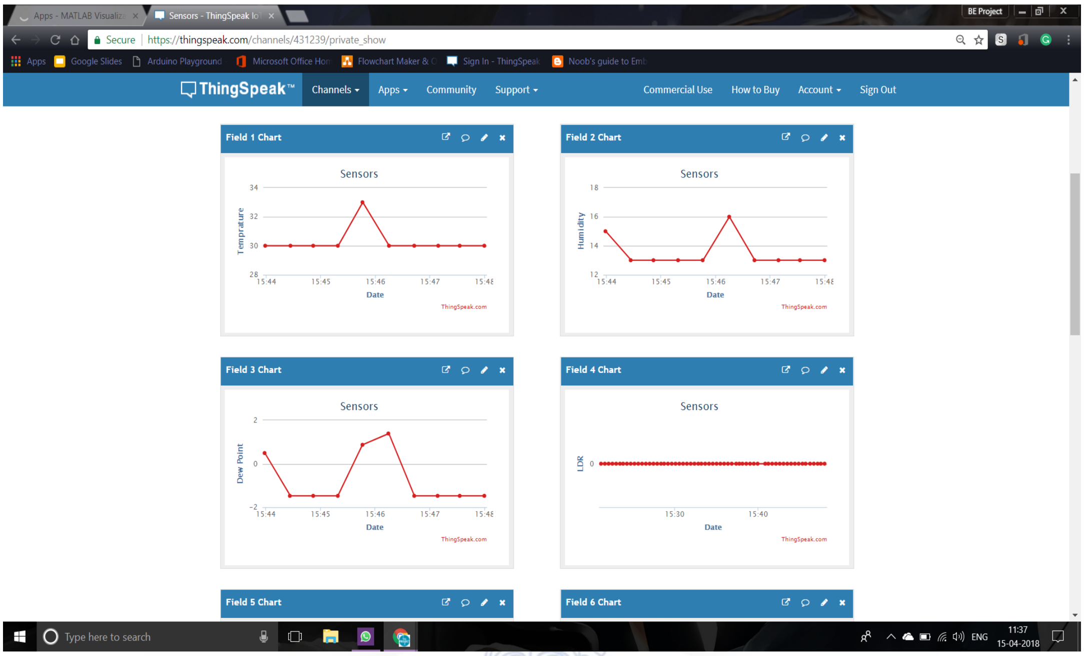


Figure 7.1: Result

7.2.1 Software Quality Attributes

1. Availability:The system should not be down,whenever the user use the system the specific data should be available to the user.
2. Correctness:As per as the user search correct data should be shown to user.
3. Maintainability:The administrator of the system should maintain the system.
4. Reliability: The system should be reliable for producing correct output so that user can reliable on system.
5. Extensibility: The system is capable to be modified by changing some modules or by adding some features to the existing system.

Chapter 8

Screenshots of Project

8.1 A smart greenhouse system based on IoT

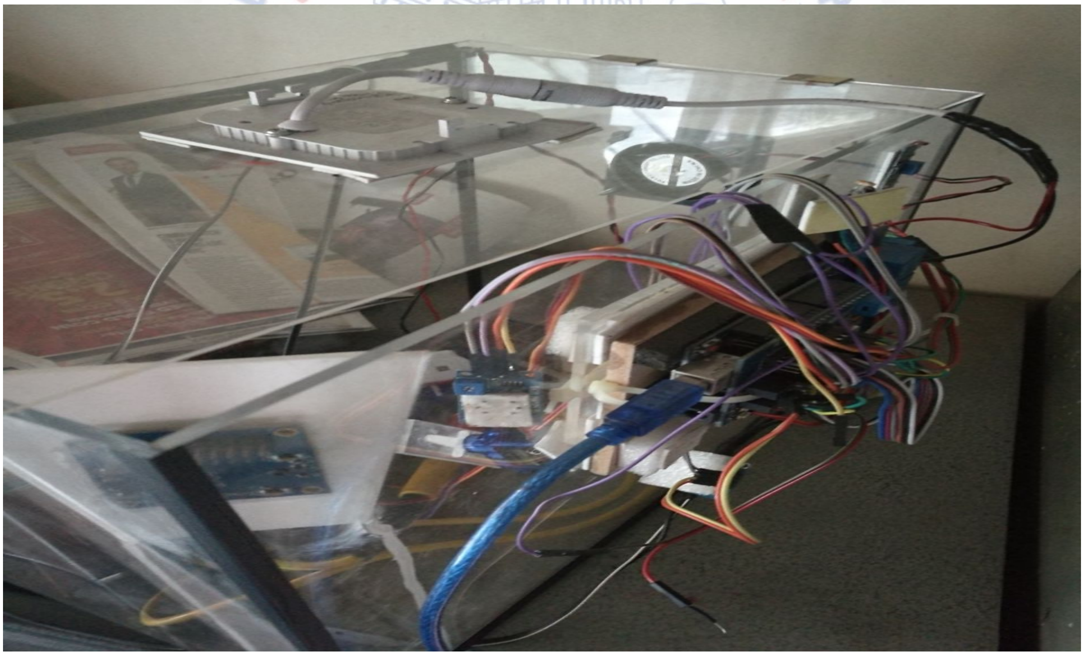


Figure 8.1: System picture2

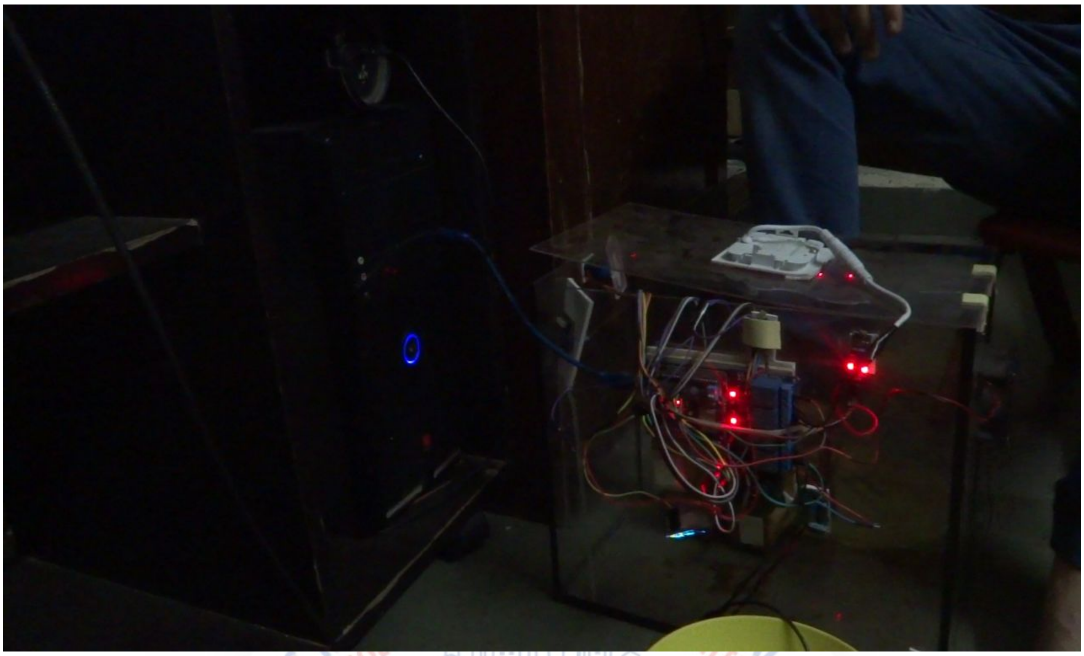


Figure 8.2: System picture3

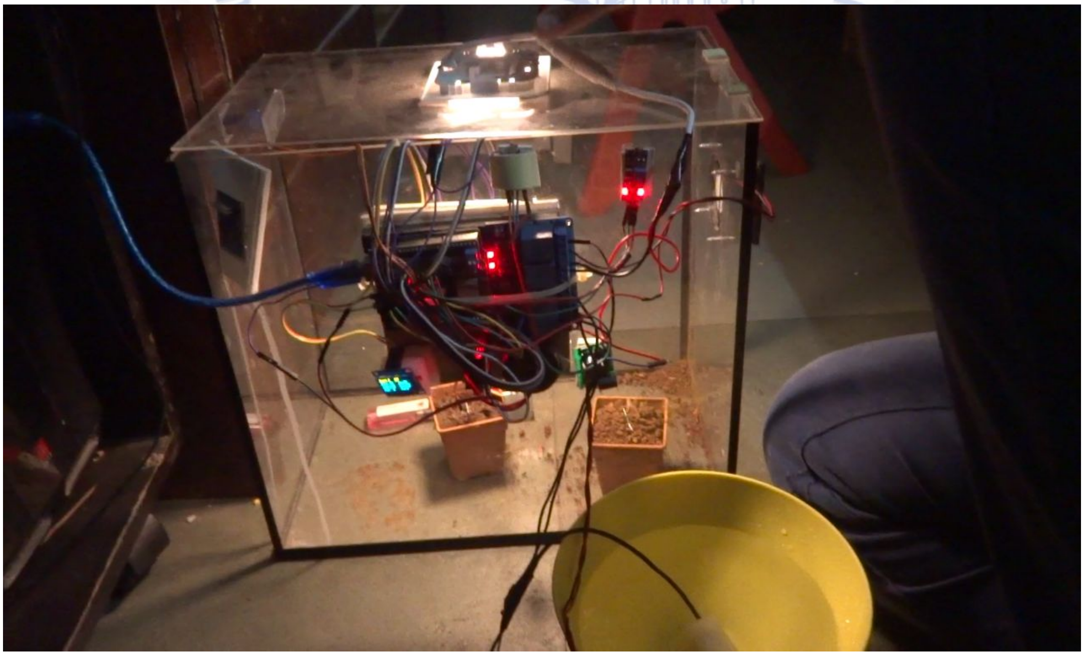
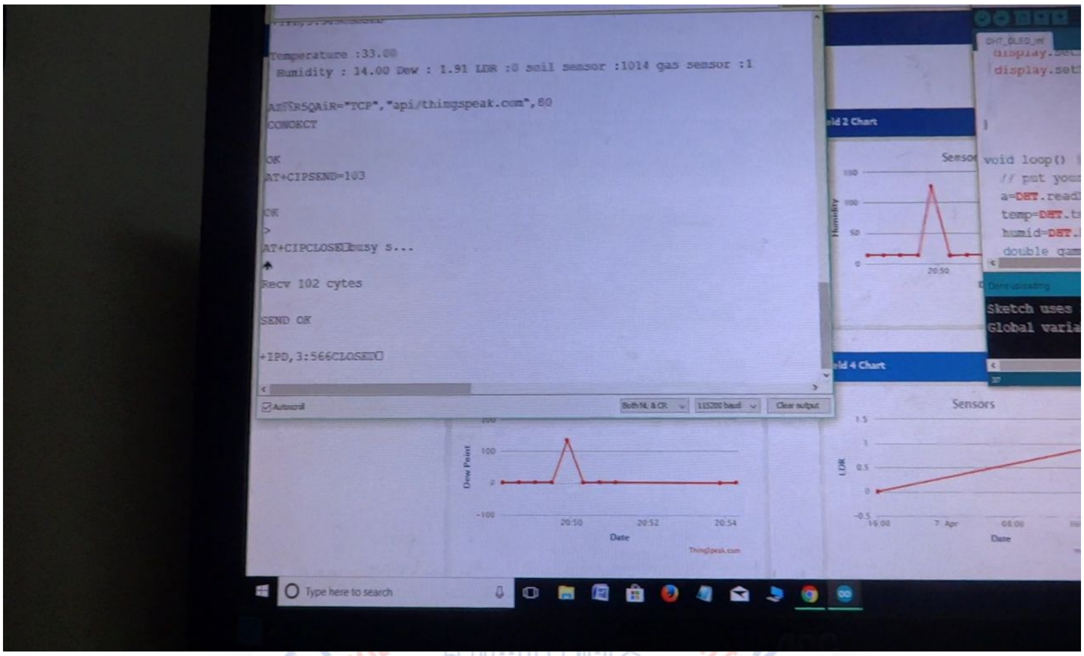


Figure 8.3: System picture1



```
Temperature :33.00
 Humidity : 14.00 Dew : 1.91 LDR :0 soil sensor :1014 gas sensor :1

AT+CIPSTART="TCP","api.thingspeak.com",80
CONNECT

OK
AT+CIPSEND=103

OK
>
AT+CIPCLOSE busy s...

Recv 102 bytes

SEND OK

+IPD,3:566CLOSED
```

Figure 8.4: Picture of Serial monitor

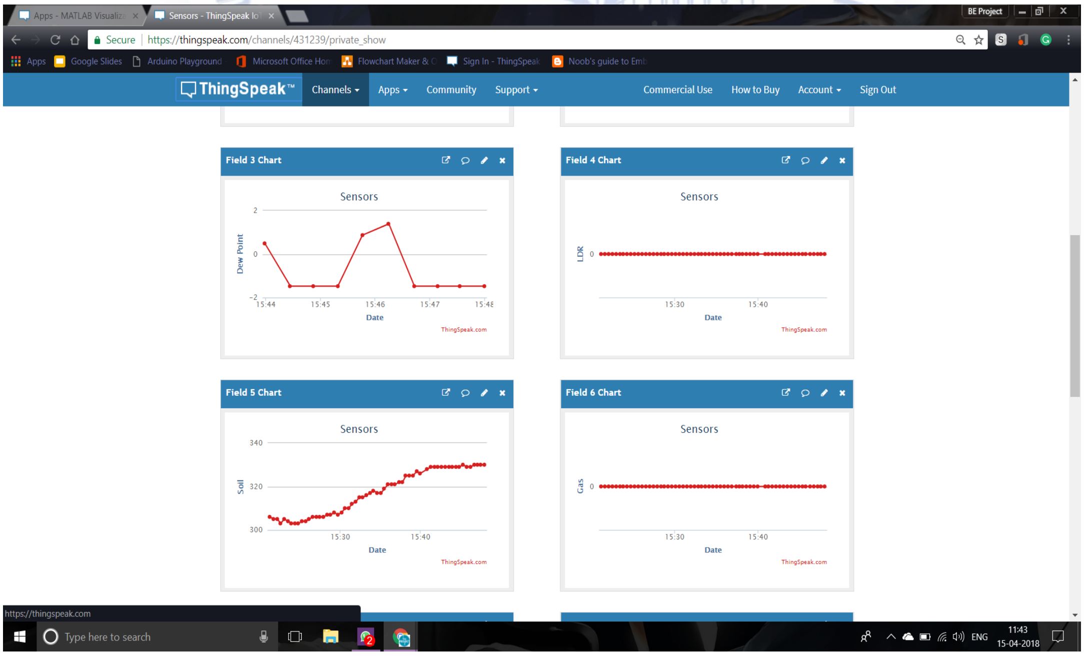
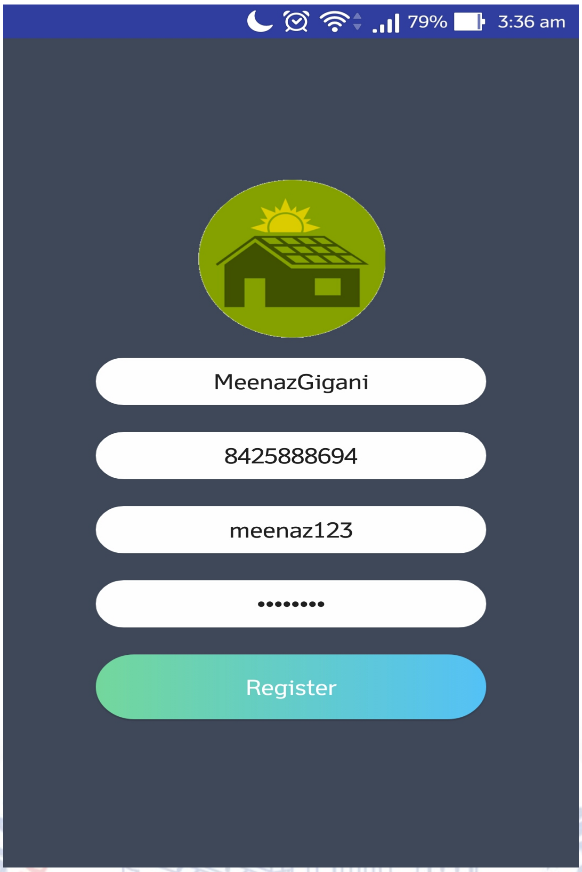


Figure 8.5: Picture of ThinkSpeak



MeenazGigani

8425888694

meenaz123

••••••••

Figure 8.6: Loing for A Smart Green House System Based on IoT

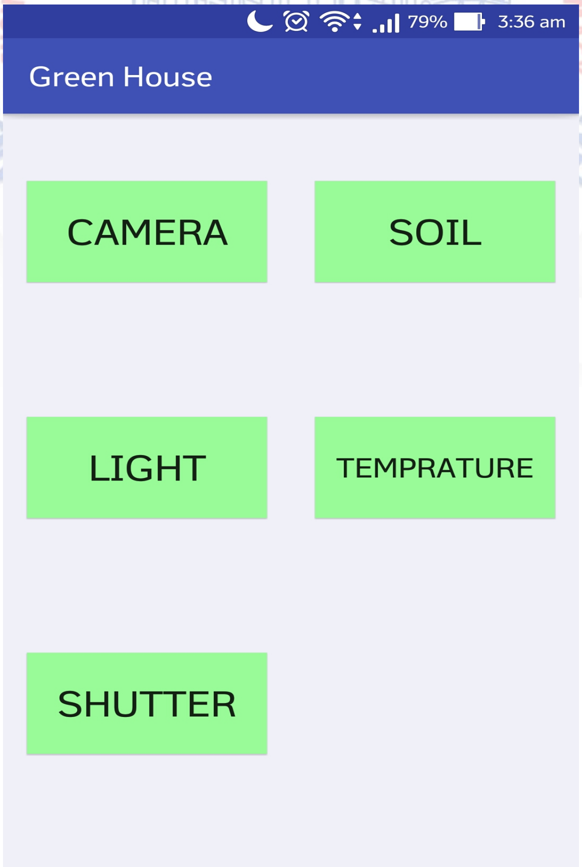
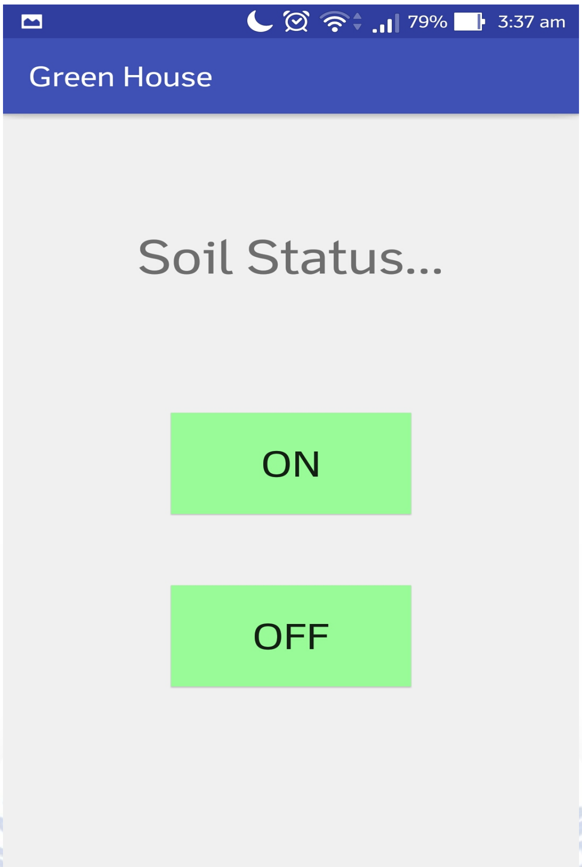
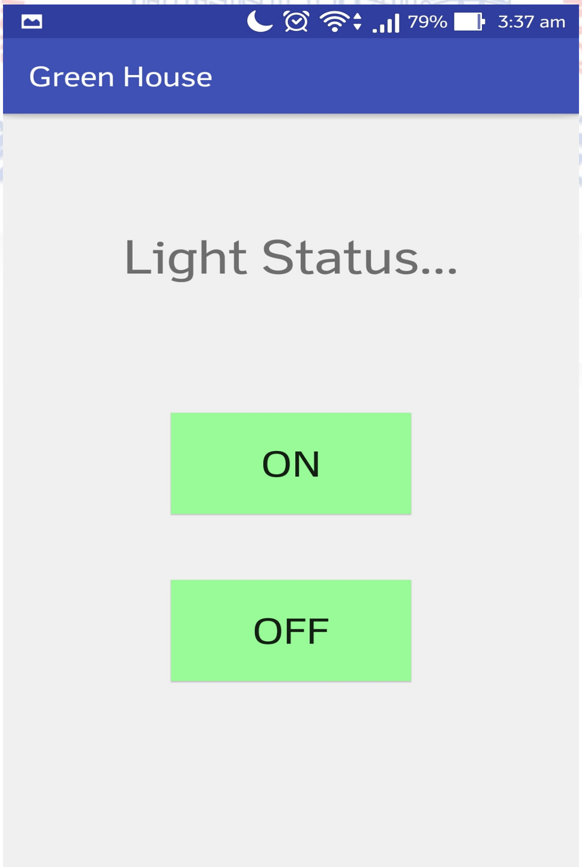


Figure 8.7: App1 for A Smart Green House System Based on IoT



Soil Status...

Figure 8.8: App2 for A Smart Green House System Based on IoT



Light Status...

Figure 8.9: App3 for A Smart Green House System Based on IoT

Chapter 9

Conclusion and Future Scope

There are the ways to make a smart move toward the need of agriculture field. Technology of today's era is providing hand in improvement. Agricultural professional and the management have to look in this matter. In conventional farming, we have to face lots of problems. With help of technology we can able to produce insecticide and pesticide free crops and create a suitable environment for the proper growth of plants. Most important the Smart customers of today's era are directly connected to the farmer. So, farmer knows the requirement of end users that improve the quality of crops.

9.1 Future Scope

- System's performance can be modified by providing the power supply with the help of battery source which would be rechargeable.
- In addition to measure the conditions that have been mentioned

, other conditions may be included like shade and fire detection. Triggered alarm can be respond via voice command.

- We can use RFID tagging for crops growth data maintenance.

References

[1] *A Study of Prototype in the Rule Based Expert System for the Management of Downy Mildew Disease in Grape Crop, Agricultural Situation in India* Mr. S.K. Jadhav, Dr. R.D. Kumbhar 2016, pp.42-48.

[2] *Arduino website* `http://www.arduino.org/products/boards/arduino-uno`

[3] *What India can do differently in agriculture TATA companies* R Gopalakrishnan Dr. YSP Thorat 2015.

[4] *IOT Based Smart Greenhouse Automation Using Arduino*; Prof. D.O.Shirsath, Punam Kamble, Rohini Mane, Ashwini Kolap, Prof.R.S.More,International Journal of Innovative Research in Computer Science Technology (IJIRCST), 2017

[5] *Planning Commission 1. Inclusive Growth, Vol.1, (Chapter 1, p.1), and 2. Agriculture, Rural Development, Industries, Services and Physical Infrastructure, Vol. 4 (Chapter 1, p.1 and Chapter 4, p.79* Oxford University Press, New Delhi.

[6] *Gene Giacomelli; Designing The Greenhouse to Meet Your Expectations: What's Your Technology Level?, Controlled Environment Agriculture Center CEAC*, Tucson, 2004.:The University of Arizona.

[7] *Coordination Control of Greenhouse Environmental Factors*; Feng Chen, Yong-Ning Tang, and Ming-Yu Shen, International Journal of Automation and Computing, 2011.

[8] *Fuzzy Logic Based Approach for Integrated Control of Protected Cultivation*Oliver L. Iliev, Ahmad Zakeri, Pavle Sazdov and A.M. Baytelieva, University of Wolverhampton, School of Technology, United Kingdom:World Applied Sciences Journal 24 (5): 561-569, 2013.

[9] *Decision support tool for greenhouse farmers in low-resource settings*; A. Bseiso, B. Abele, S. Ferguson, P. Lusch, and K. Mehta, Global Humanitarian Technology Conference (GHTC), Oct 2015, pp. 292–297.

[10] *Quantifying water savings with greenhouse farming*; K. A. Czyzyk, S. T. Bement, W. F. Dawson, and K. Mehta, Global Humanitarian Technology Conference (GHTC), Oct 2014, pp. 325–332.

[11] *The design and implementation of the greenhouse monitoring system based on gsm and rf technologies*; Z. Xiaoyan, Z. Xiangyang, D. Chen, C. Zhaohui, S. Shangming, and Z. Zhaohui, Computational Problem-solving (ICCP), 2013 International Conference on, Oct 2013, pp. 32–35.

[12] *Iot based Smart Greenhouse*; Ravi Kishore Kodali, Vishal Jain and Sumit Karagwal, Department of Electronics and Communication Engineering National Institute of Technology: Warangal.2014

[13] *The working principle of an arduino*; Y. A. Badamasi, in Electron- ics, Computer and Computation (ICECCO), 11th International Conference, Sept 2014, pp. 1–4.

[14] *Green House by using IOT and Cloud computing*; Sheetal Vatari, Aarti Bakshi, Tanvi Thakur, IEEE International Conference On Recent Trends In Electronics Information Communication Technology, May 20-21, 2016, India

[15] *Agriculture Greenhouse Environment Monitoring System Based on IOT Technology* , LIU Dan, Cao Xin, Huang Chongwei and JI Liangliang: Dep of Computer Science and Technology, Neusoft Institute of Information, Dalian, China pg-487:Figure1-”Structure of the system”.

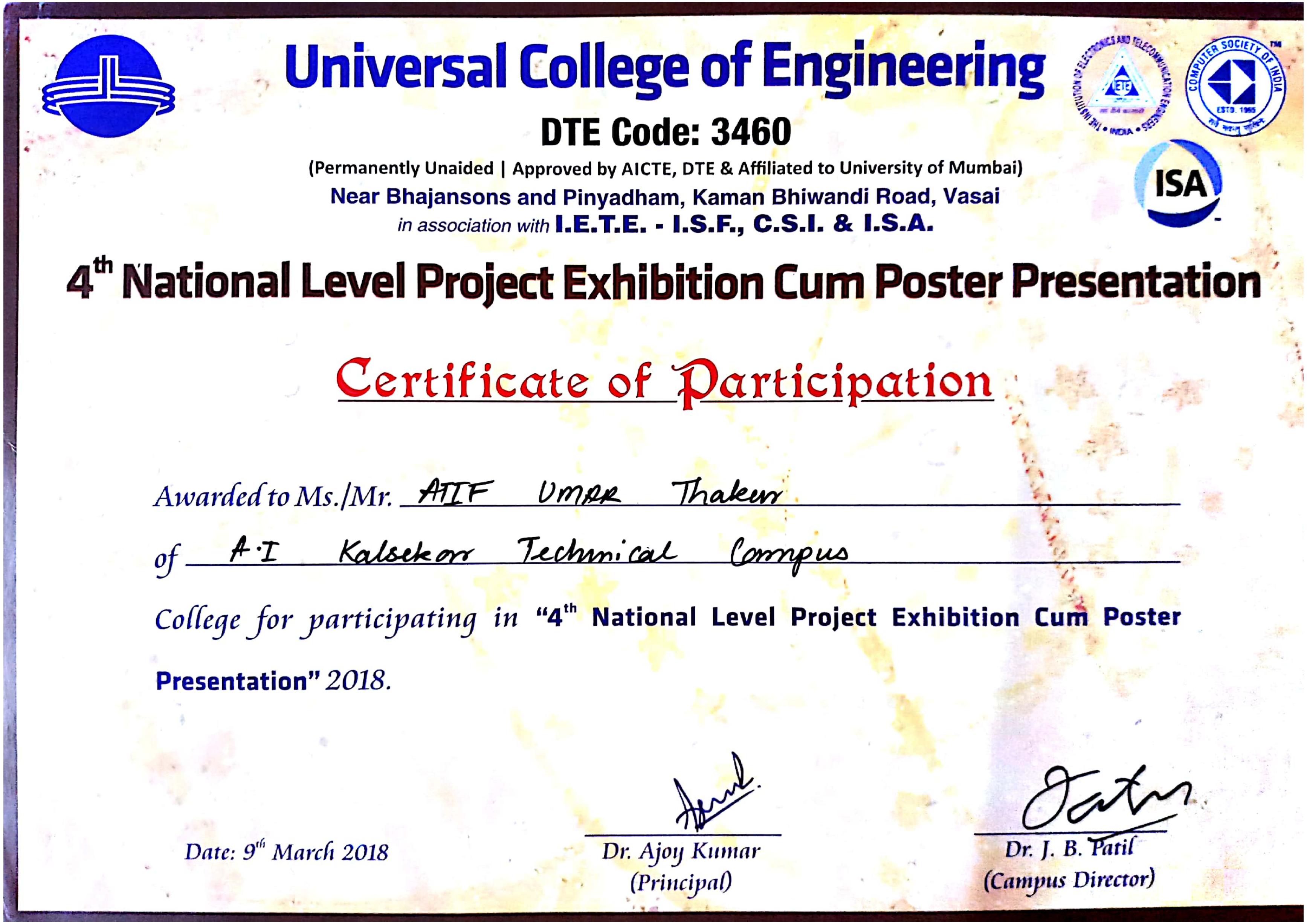
Achievements

1. Conferences

   (a) *Technical approach for agriculture*; PAWLE MUSA ASMAT, Name of conference , Feb and 2018 of attend(Venue :Thakur College of Engineering Technology, MULTICON-W 201)

2. Project Competitions

   (a) *A Smart Green House System Based on IoT*; Pawle Musa, Peerzade Nilofar, Thakur Atif, Gigani Minaz, 4th National Level Project Exhibition cum Poster Presentation , March-2018(Venue : Universal College of Engineering)



Universal College of Engineering

DTE Code: 3460

(Permanently Unaided | Approved by AICTE, DTE & Affiliated to University of Mumbai)

Near Bhajansons and Pinyadham, Kaman Bhiwandi Road, Vasai

in association with **I.E.T.E. - I.S.F., C.S.I. & I.S.A.**

4th National Level Project Exhibition Cum Poster Presentation

Certificate of Participation

Awarded to Ms./Mr. ATIF UMAR Thaker

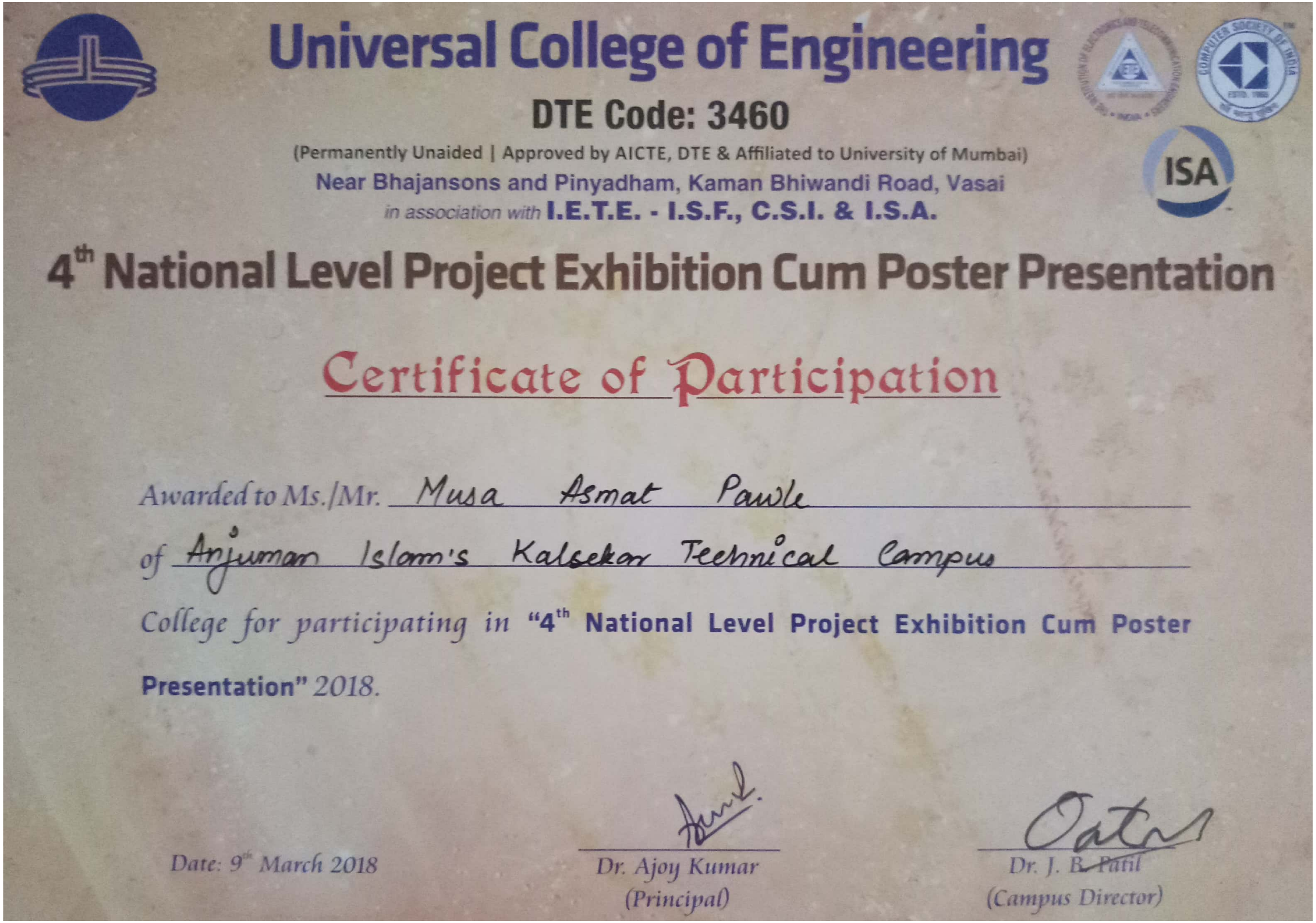
of A.I Kalsekar Technical Campus

College for participating in **"4th National Level Project Exhibition Cum Poster Presentation"** 2018.

Date: 9th March 2018

Dr. Ajoy Kumar
(Principal)

Dr. J. B. Patil
(Campus Director)



Universal College of Engineering

DTE Code: 3460

(Permanently Unaided | Approved by AICTE, DTE & Affiliated to University of Mumbai)
Near Bhajansons and Pinyadham, Kaman Bhiwandi Road, Vasai
in association with I.E.T.E. - I.S.F., C.S.I. & I.S.A.

4th National Level Project Exhibition Cum Poster Presentation

Certificate of Participation

Awarded to Ms./Mr. Musa Asmat Pawle

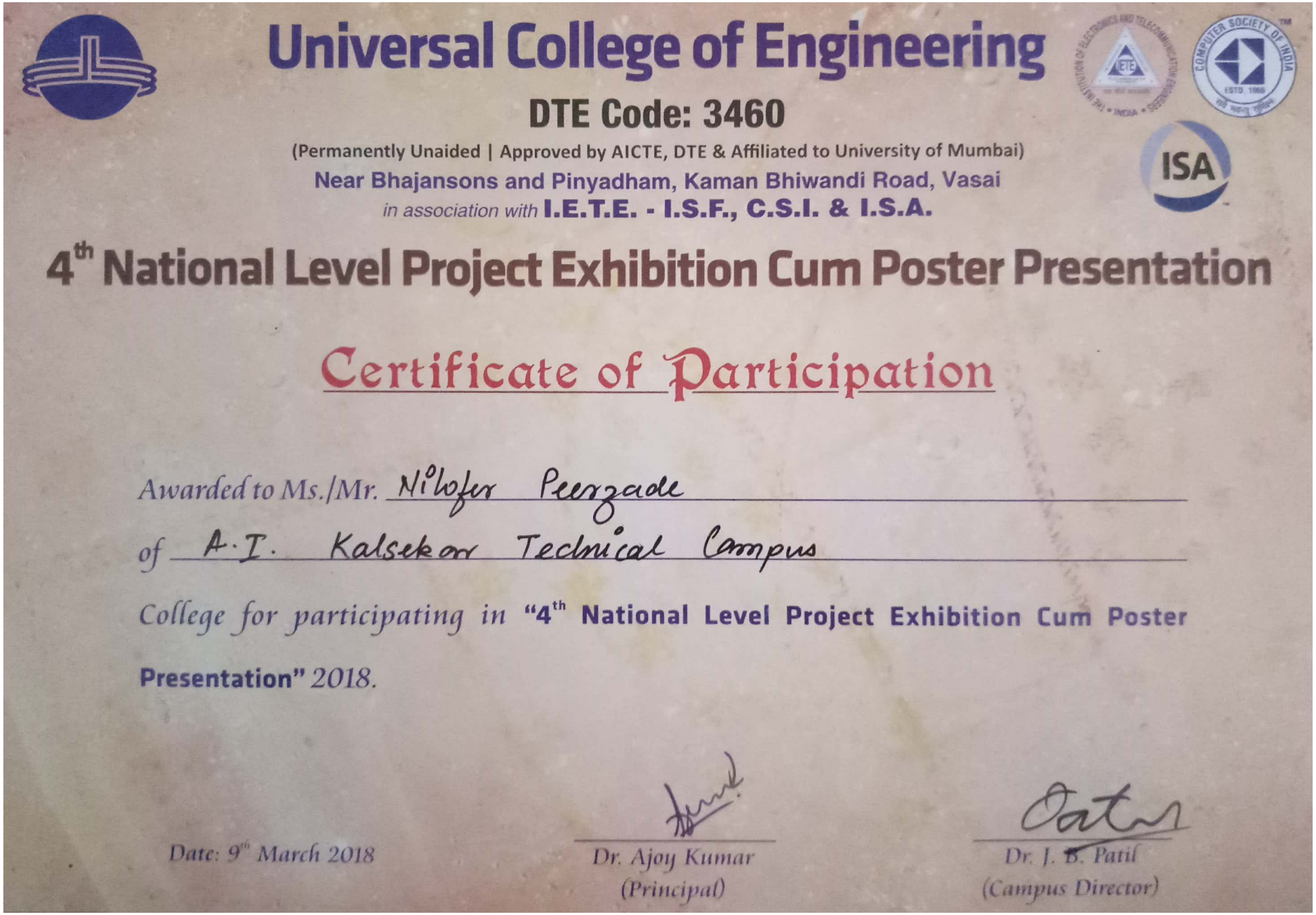
of Anjuman Islam's Kalsekar Technical Campus

College for participating in "4th National Level Project Exhibition Cum Poster Presentation" 2018.

Date: 9th March 2018

Dr. Ajoy Kumar
(Principal)

Dr. J. B. Patil
(Campus Director)



Universal College of Engineering

DTE Code: 3460

(Permanently Unaided | Approved by AICTE, DTE & Affiliated to University of Mumbai)
Near Bhajansons and Pinyadham, Kaman Bhiwandi Road, Vasai
in association with **I.E.T.E. - I.S.F., C.S.I. & I.S.A.**

ISA

4th National Level Project Exhibition Cum Poster Presentation

Certificate of Participation

Awarded to Ms./Mr. Nilofer Peerzade

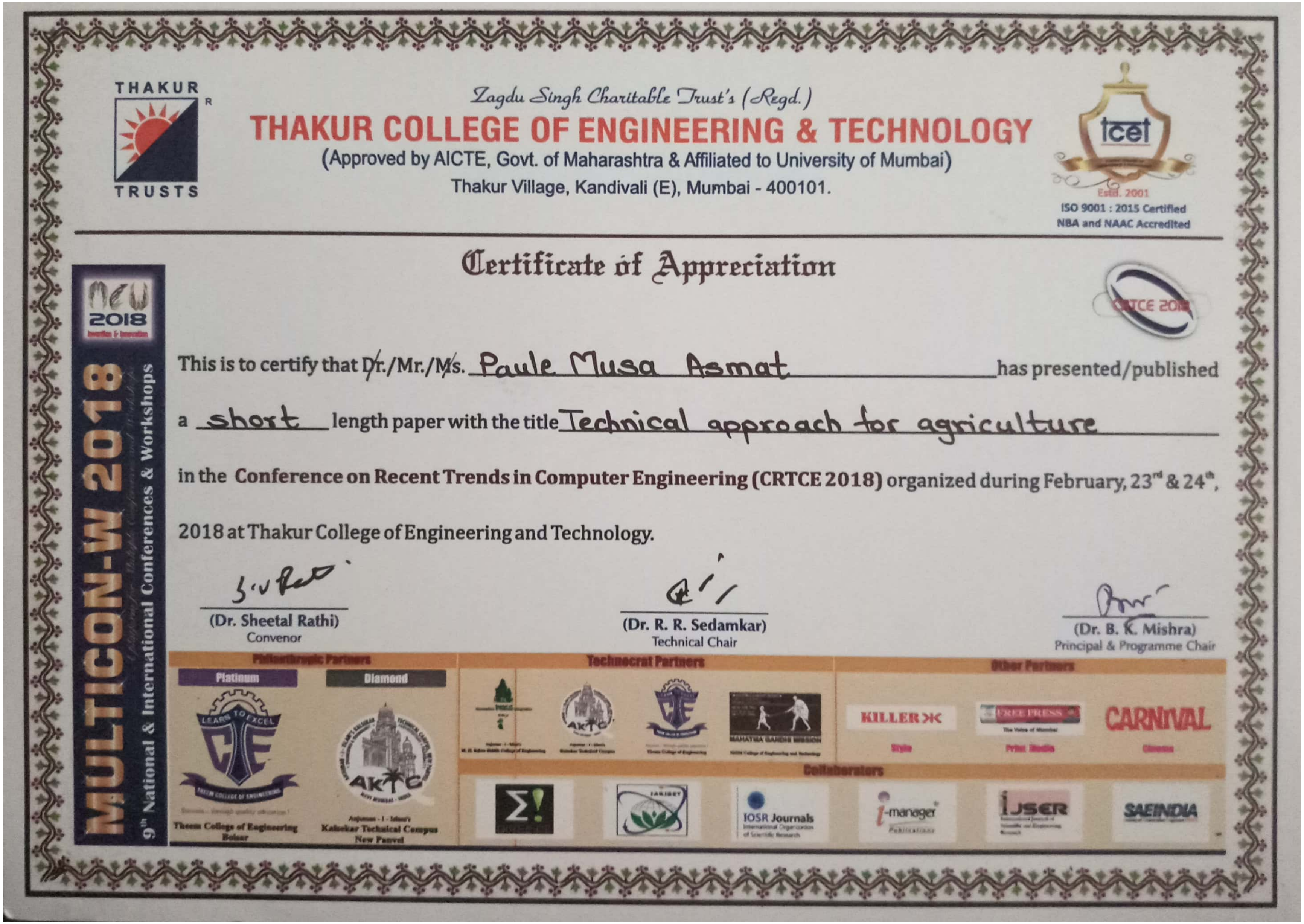
of A.I. Kalsekar Technical Campus

College for participating in **"4th National Level Project Exhibition Cum Poster Presentation"** 2018.

Date: 9th March 2018

Dr. Ajoy Kumar
(Principal)

Dr. J. B. Patil
(Campus Director)



THAKUR TRUSTS

Zagdu Singh Charitable Trust's (Regd.)

THAKUR COLLEGE OF ENGINEERING & TECHNOLOGY

(Approved by AICTE, Govt. of Maharashtra & Affiliated to University of Mumbai)

Thakur Village, Kandivali (E), Mumbai - 400101.

tcet

Estd. 2001

ISO 9001 : 2015 Certified

NBA and NAAC Accredited

MULTICON-W 2018

9th National & International Conferences & Workshops

NEU 2018

Certificate of Appreciation

CRTCE 2018

This is to certify that ~~Dr.~~/Mr./~~Ms.~~ Paule Musa Asmat has presented/published a short length paper with the title Technical approach for agriculture in the Conference on Recent Trends in Computer Engineering (CRTCE 2018) organized during February, 23rd & 24th, 2018 at Thakur College of Engineering and Technology.

(Dr. Sheetal Rathi)
Convenor

(Dr. R. R. Sedamkar)
Technical Chair

(Dr. B. K. Mishra)
Principal & Programme Chair

Philanthropic Partners

Platinum

Theem College of Engineering
Boisar

Diamond

Anjuman - I - Islam's
Kalsekar Technical Campus
New Panvel

Technocrat Partners

Mahatma Gandhi Mission

Other Partners

KILLER
Style

FREE PRESS
Print Media

CARNIVAL
Cinema

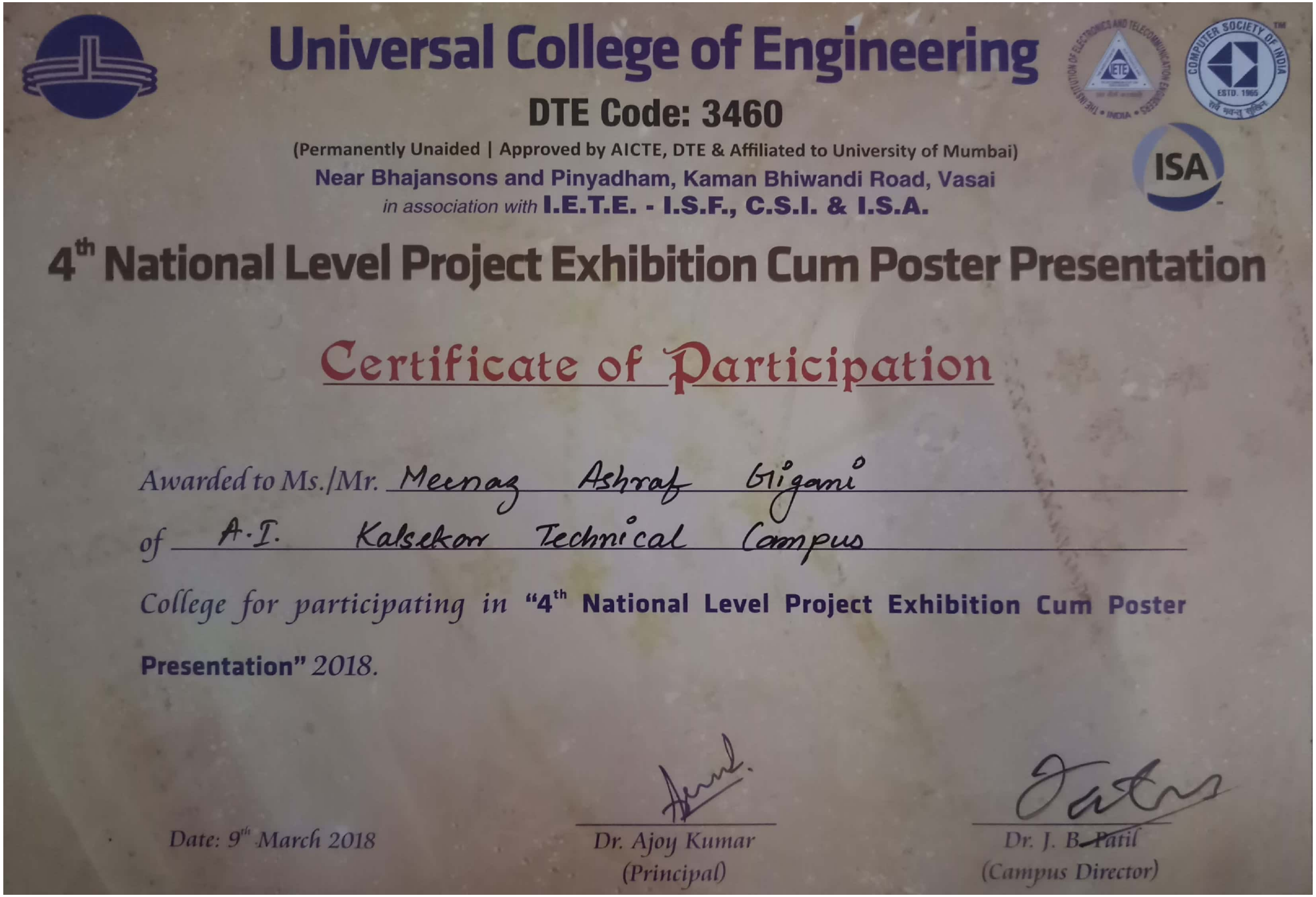
Collaborators

IOSR Journals

i-manager

IJSER

SAEINDIA



Universal College of Engineering

DTE Code: 3460

(Permanently Unaided | Approved by AICTE, DTE & Affiliated to University of Mumbai)
Near Bhajansons and Pinyadham, Kaman Bhiwandi Road, Vasai
in association with **I.E.T.E. - I.S.F., C.S.I. & I.S.A.**

4th National Level Project Exhibition Cum Poster Presentation

Certificate of Participation

Awarded to Ms./Mr. Meenaz Ashraf Gigani

of A.I. Kalsekar Technical Campus

College for participating in **"4th National Level Project Exhibition Cum Poster Presentation"** 2018.

Date: 9th March 2018

Dr. Ajoy Kumar
(Principal)

Dr. J. B. Patil
(Campus Director)