**Documentation (for all Departments except CSE)**

* Page Layout: Top, Bottom, Right: 1” Left: 1.25” , Orientation : Portrait,

Page Layout- Size- A4, Alignment: Full page Justify, Line Spacing – 1.5

* Header: Domain- right, center - Logo and Technology - left (Cambria – 12)
* Footer: Address (Cambria – 12- center alignment)
* Chapter number( Font – Times new Roman – Size 16 Alignment: center), Bold

Chapter name (Font – Times new Roman – Size 14 Alignment: center), Bold

* Each Chapter should start in a fresh page
* Main headings in a Chapter must be of size 14, Bold, Times new Roman
* Sub headings in a Chapter must be of size 12, Bold, Times new Roman
* Figure names and table names should be of size 10, Bold, Times new Roman

Except Abstract the below mentioned are considered as chapters that should be present in the Document

**MODIFIED TITLE** (Times New Roman – 14 – Bold - ALL CAPS – center alignment)

**ABSTRACT**

**INTRODUCTION**

**LITERATURE REVIEW**

**EXISTING METHOD**

**PROPOSED METHOD**

**HARDWARE & SOFTWARE REQUIREMENTS**

**ADVANTAGES & APPLICATIONS**

**TESTING RESULTS**

**CONCLUSION**

**FUTURE SCOPE**

**REFERENCES**

Note: The Chapters are should be of Times new Roman with 14 Font size being bold and capitalized, the content should remain same without bold.

The document must have a minimum of 50 pages including tools and technologies.

**DESIGN AND DEVELOPMENT OF AN AGRI-BOT FOR AUTOMATIC SEEDING, AND WATERING APPLICATIONS**

**ABSTRACT:**

Agriculture is the backbone of rural India. Farmers face problems such as lack of timely availability of efficient workforce, as many have migrated from country side. Hence, to reduce the burden of farmers, automation in the field of farming is necessary. The robot developed in this work is semi-automatic and will be able to sow seeds based on seed spacing and depth while also regulating watering of plants. The applications of this can range from efficient and intensive commercial farming to using it for research and backyard gardening purposes. The main reason behind automation of farming processes are saving the time and energy required for performing repetitive farming tasks and increasing the productivity of yield by treating every crop individually using precision farming concept. The robot is able to automatically seed and water, spray pesticides according the path set by the user using the GUI that was developed. The amount of watering is based on the soil moisture sensor reading that is taken between the two plants

**Chapter-1**

**INTRODUCTION**

Food is Man's most basic need. Throughout the ages men have strived to produce more and better food for growing populations. More land, Better tools, improved techniques, and technological advances however, all this progress has had one common element that has not changed “Man”. Man has always been at the center of food production and has now become the limiting factor. For 10,000 years, the primary focus of technological advancement has always been increasing the productivity and safety of the operator. But these incremental improvements will soon be outpaced by the exponential population increase that we are experiencing. Sometime in the near future, the population will exceed our ability to provide food for all of earth's inhabitants. Automation Technology has proliferated in virtually every domain of human activity such as construction, manufacturing, communication, offices, households, transportation, warfare, exploration, and space travel. The investment in Automation Technology in agriculture is driven by various forces: The main motive for developing Agricultural Automation Technology is the decreasing labor force, a phenomenon common in the developed world. Other reasons are the need for improved food quality and security such as automated inspection of agricultural products for contaminants. Automation can also help solve problems with high volume seasonal labor such as harvesting of citrus fruits, grapes, and raisins. An important part of Automation is the use of robots.

Robotics in agriculture is not a new concept; in controlled environments (green houses), it has a history of over 20 years. Research has been performed to develop harvesters for cherry tomatoes, cucumbers, mushrooms, and other fruits. In horticulture, robots have been introduced to harvest citrus and apples. In dairy farming, milking robots are currently commonplace in the Netherlands. The pinnacle of highly automated crop production is without a doubt the Japanese ‘‘plant factory,’’ where vegetables are grown hydroponically under artificial lighting. Computers and robots control the process from out planting seedlings, to root cutting, packaging, and weighing, and the produce is free of any blemish, disease, or insect damage. The automation level in plant factories is so high that over time they may become completely autonomous production facilities. The most important current abilities of automatic agricultural vehicles can be grouped into four categories:

* Guidance (i.e., the way the vehicle navigates within the agricultural environment)
* Detection (the extraction of biological features from the environment)
* Action (the execution of the task for which the vehicle was designed.
* Mapping (the construction of a map of the agricultural field with its most relevant features).

However, those four cores are not independent. For safe and successful navigation, the vehicle has to know its position within the field and the elements from the surrounding environment (mapping); bad detection could lead to an incomplete or unreliable map. Furthermore, if the elements from the environment are not properly located within the map, an agricultural vehicle may not be able to execute its tasks successfully. In addition, an incomplete map should not be used for navigation purposes because of the risk of collision. As can be seen, the knowledge regarding the location of a vehicle within the environment and the location of the elements in an environment plays a crucial role in an automatic agricultural vehicle design. Slaughter et al. propose the main abilities for designing robotic vehicles for weed control only, without addressing the localization issues associated with such a design. Growth in the world population has led to the need for an increasing level of sophistication in precision agriculture for both environment preservation and production optimization. This need, in turn, has created a requirement for new methods, tools, and strategies for agricultural processes. Robotics and artificial intelligence achievements offer new solutions in precision agriculture to processes related to seeding, harvesting, weed control, grove supervision, chemical applications, etc, to improve productivity and efficiency. A service unit is an automatic vehicle for main or secondary tasks in the agricultural environment.

.

**Chapter-2**

**LITERATURE REVIEW**

**[2.1] “Agricultural Robot for Automatic Ploughing and Seeding” 2015 IEEE International Conference on Technological Innovations in ICT (TIAR 2015) (Amrita Sneha.A, Abirami.E, Ankita.A, Mrs. R. Praveen, Mrs. R. Srimeena).**

This paper strives to develop a robot capable of performing operations like automatic ploughing, seed dispensing. It also provides manual control when required and keeps tabs on the humidity with the help of humidity sensors .The main component here is the AVR At mega microcontroller that supervises the entire process. Initially the robot tills the entire field and proceeds to ploughing, simultaneously dispensing seeds side by side. On the field the robot operates on automated mode, but outside the field is strictly operated in manual mode.

**[2.2] “Design and Implementation of Seeding Agricultural Robot” (JIRAS) (P.Usha, V. Maheswari, Dr. V. Nandagopal)**

In this paper, the robot system is used to develop the process of cultivating agricultural land without the use of man power. The aim of the paper is to reduce the man power, time and increase the productivity rate.

**[2.3]. “Automated Farming Using Microcontroller and Sensors” (IJSRMS) ISSN: 23493371 (Abdullah Tanveer, Abhishek Choudhary, Divya Pal, Rajani Gupta, Farooq Husain) Farming can be done using new technologies to yield higher growth of the crops.**

In this project we are going to check temperature, light, humidity and soil moisture. The paper here is all about automatic control features with latest electronics technology using microcontroller and GSM phone line. The project works automatically and hence reduces the manpower.

**[2.4] “IOT Based Smart Agriculture” IJARCCE June 2016 (Nikesh Gondchawar1, Prof. Dr. R. S. Kawitkar2)**

In this paper a project model for agriculture robot is describe the newer scenario of decreasing water tables, drying up of rivers and tanks, unpredictable environment present an urgent need of proper utilization of water.

**Chapter-3**

**EXISTING METHOD**

We have many machines which are capable of seed sowing but they are hand operated machines, so we are designing a robot which will drill the soil and sow the seeds this robot has two modes of operations like auto mode and manual mode, in auto mode it moves in a particular grid by help of sensors. This robot is capable of receiving few sets of command instructions in the form of Bluetooth tones with help of android app and performs the necessary actions. Bluetooth module receives the commands we have an android app which will give signals to Bluetooth at robot for controlling the motion of the robot. This agriculture robot has two modes one is auto mode another is a manual mode in manual this set with help of switch button in manual it works with Bluetooth (HC -05) signals, in auto mode it works with Help of IR sensor in front of the robot for making a grid pattern seed sowing. In this agriculture robot project we are using Arduino UNO as motherboard which will control the robot driving motors and seed dispenser servo.

**Disadvantages:**

There are some drawbacks with Bluetooth communication as given below.

* It can lose connection in certain conditions.
* It has low bandwidth as compared to Wi-Fi.
* It allows only short range communication between devices.

**Chapter-4**

**PROPOSED METHOD**

We propose this “an agri-bot for automatic seeding, and watering applications” to overcome all the drawbacks in existing system. With this project we can do long range communication.by using this we control the robot from anywhere in world using internet. It reduce the burden of farmers, automation in the field of farming is necessary. automation of farming processes which saving the time and energy required for performing repetitive farming tasks and increasing the productivity of yield.in this system we have automatic seeding and watering function that makes very easy of seeding and watering to field. We can also make changes like pesticides spraying robot with small modifications. Camera that present in the robot sends images of field at any location in field to mail.dht 11 sensor and soil moisture sensors will updates the data to server at any location by clicking on data upload button. This data can help to understand the present situation on field.

MOTOR

Lcd

**Block diagram:**

L293D

raspberrypi

SEEDER

MOTOR

Cloud

CAMERA

MOBILE

dht11

MOTOR(PUMP)

soil moisture sensor

Power supply

**Fig.1. Block diagram**

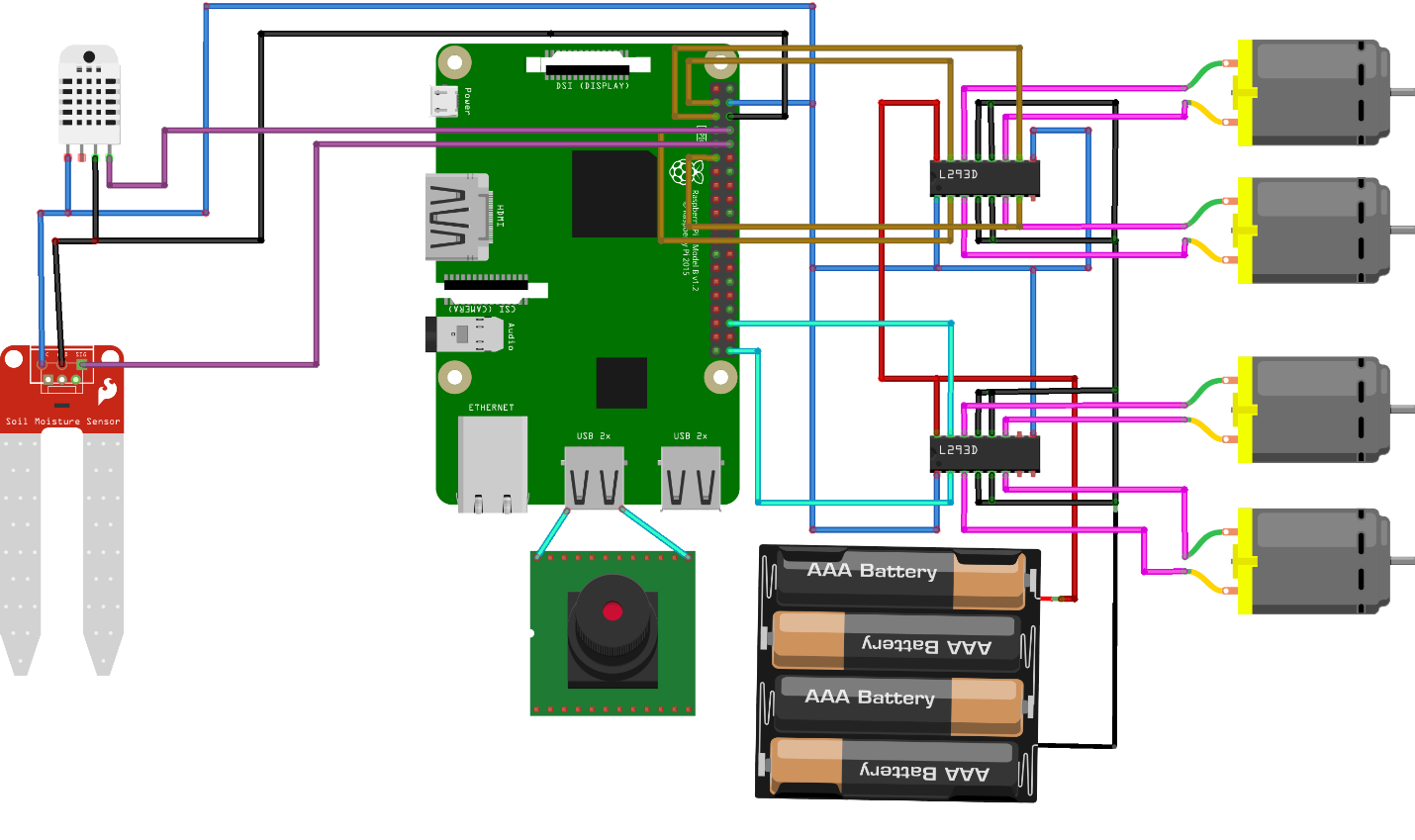
* 1. **Working:**

In Modern world, Automation robot is used in many of the fields such as defence, surveillance, medical field, industries and so on.the robot system is used to develop the process of cultivating agricultural land without the use of man power.All the basic automation robot works like weeding, harvesting and so on. Here the designing systems like sowing the seed, watering the plant or spraying the fertilizer and navigate the vehicle motion are preferred by this autonomous robot using internet of things with raspberry pi.movement of this robot in the land based on the controller app that from the mobile, thereby performs turning the position of robot either in left or right or forward direction.

IOT controlled robot is controlled by using Android mobile phone with blynk app. Here we use button in android phone to control the robot in forward, backward, left and right directions. So here android phone is used as transmitting device and raspberry pi is receiver module.so that robot can move in the required direction like moving forward, reverse, turning left, turning right and stop.in this system we placed a Camera that present in the robot sends images of field at any location in field to mail.DHT11 sensor and soil moisture sensors will updates the data to server at any location by clicking on data upload button. This data can help to understand the present situation on field.

Through web cam we can take photos for image analysis using matab.

* + 1. **Hardware architecture:**
  1. **Circuit diagram:**



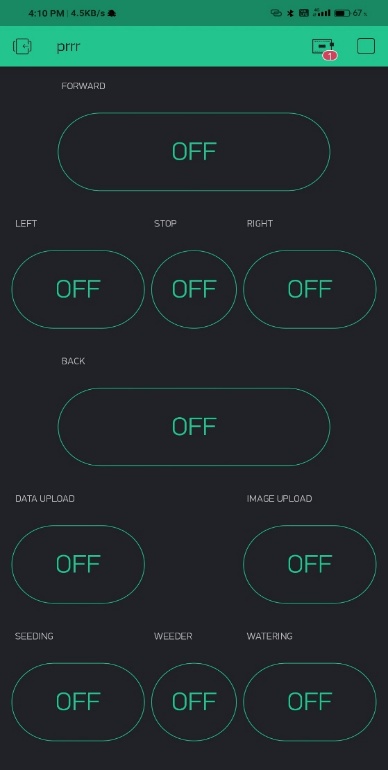
**Fig.3. circuit diagram**

Proposed project circuit diagram shown in fig.3. Here we connected first l293d motor drive to the pins 2,3,4,17 which supports the movement of the robot in multiple direction.gpio pins 16,21 is connected to second l293d motor drive for enabling dc motor pump and seeder mechanism for seeding and watering or spraying pesticides in the field.usb wed camera connected to usb port1 for image capture purpose.DHT11 and soil moisture sensor connected to pins 14 and 15 for knowing moisture level,temperature and humidity level.12v power supply connected to the motor drivers.5v power supply connected to enable pins of two motor drivers to enable the system.

**4.3.2 Software IMPLEMENTATION**

software system consists robot control mobile app and php web server.

**ROBOT CONTROL MOBILE APP:**

****

* Now, Controlling of robot is through iot,so we are using blynk app to control the robot
* We can send the image of particular place or plant image for processing using click buttons
* Seeding machine control,watrering motor and weeding can controller over blynk app

**PHP WEB SERVER**:

We use this server to check the values of temperature, humidity and soil moisture present in that area

Web server dashboard consist of home page, monitor page and register page

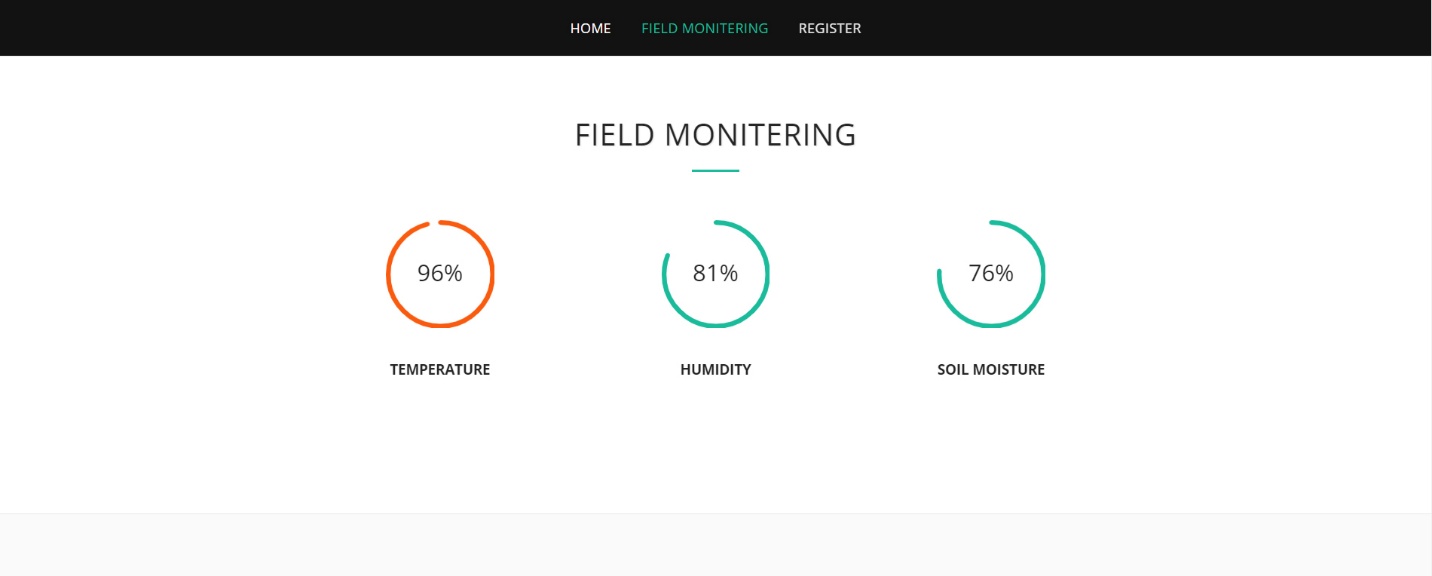


**Home page:**



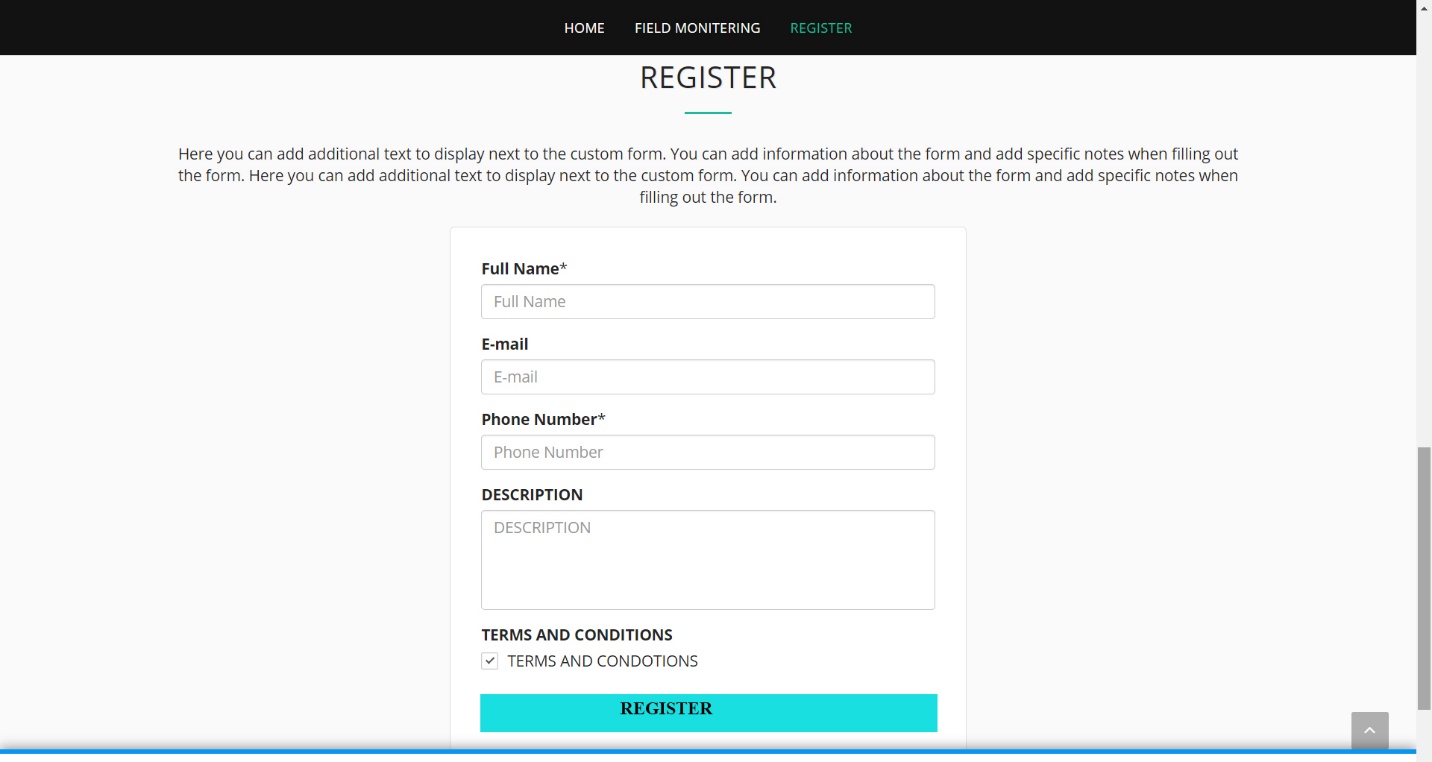
* Home page contains smart details about the project

**Monitoring page:**

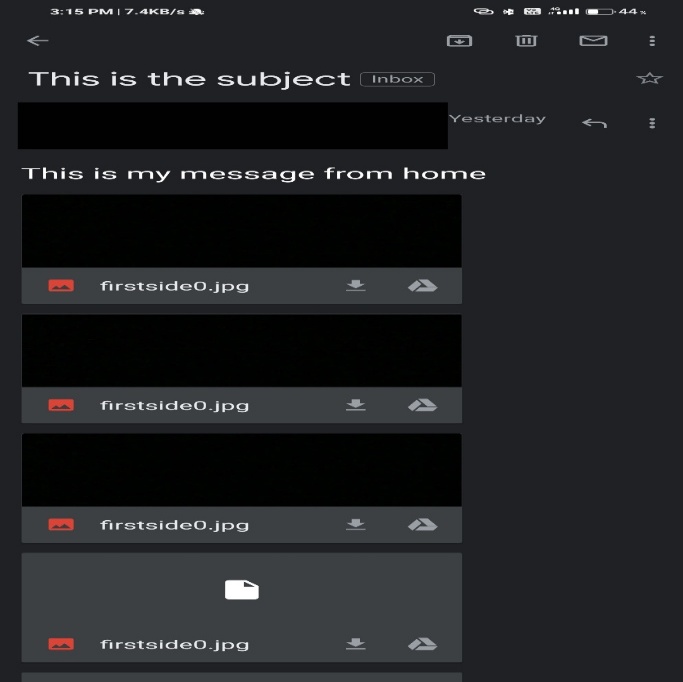


* Monitoring page have the details scales of the values from the sensors that are uploading continuously from the field when clicking button

**Register page:**

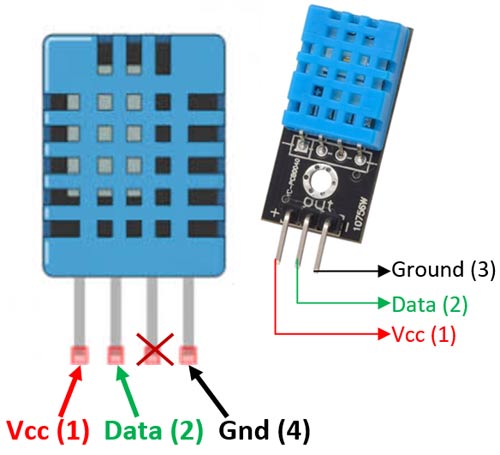


* In register page, we need register our details that we need to send. These data will be stored in cloud
* Data coming from sensors will store in cloud and also sends mail and sms to registered candidates



* 1. **Hardware components:**

**4.5.1 DHT11–Temperature and Humidity Sensor**

[](https://components101.com/sites/default/files/component_pin/DHT11%E2%80%93Temperature-Sensor-Pinout.jpg)

**DHT11–Temperature and Humidity Sensor Pinout**

**Pin Identification and Configuration:**

|  |  |  |
| --- | --- | --- |
| **No:** | **Pin Name** | **Description** |
| **For DHT11 Sensor** | | |
| 1 | Vcc | Power supply 3.5V to 5.5V |
| 2 | Data | Outputs both Temperature and Humidity through serial Data |
| 3 | NC | No Connection and hence not used |
| 4 | Ground | Connected to the ground of the circuit |
| **For DHT11 Sensor module** | | |
| 1 | Vcc | Power supply 3.5V to 5.5V |
| 2 | Data | Outputs both Temperature and Humidity through serial Data |
| 3 | Ground | Connected to the ground of the circuit |

**DHT11 Specifications:**

* Operating Voltage: 3.5V to 5.5V
* Operating current: 0.3mA (measuring) 60uA (standby)
* Output: Serial data
* Temperature Range: 0°C to 50°C
* Humidity Range: 20% to 90%
* Resolution: Temperature and Humidity both are 16-bit
* Accuracy: ±1°C and ±1%

**Difference between DHT11 Sensor and module:**

The **DHT11 sensor** can either be purchased as a sensor or as a module. Either way, the performance of the sensor is same. The sensor will come as a 4-pin package out of which only three pins will be used whereas the module will come with three pins as shown above.

The only difference between the sensor and module is that the module will have a filtering capacitor and pull-up resistor inbuilt, and for the sensor, you have to use them externally if required.

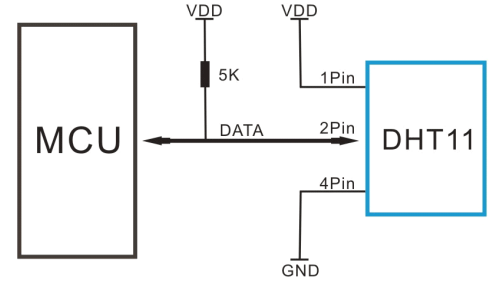
**Where to use DHT11:**

The **DHT11**is a commonly used **Temperature and humidity sensor.** The sensor comes with a dedicated NTC to measure temperature and an 8-bit microcontroller to output the values of temperature and humidity as serial data. The sensor is also factory calibrated and hence easy to interface with other microcontrollers.

The sensor can measure temperature from 0°C to 50°C and humidity from 20% to 90% with an accuracy of ±1°C and ±1%. So if you are looking to measure in this range then this sensor might be the right choice for you.

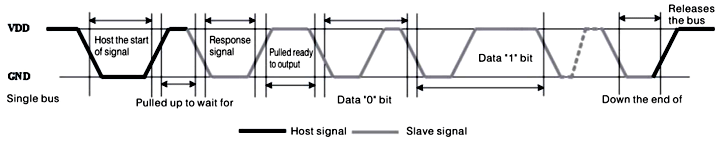
**How to use DHT11 Sensor:**

The DHT11 Sensor is factory calibrated and outputs serial data and hence it is highly easy to set it up. The connection diagram for this sensor is shown below.



As you can see the data pin is connected to an I/O pin of the MCU and a 5K pull-up resistor is used. This data pin outputs the value of both temperature and humidity as serial data. If you are trying to interface DHT11 with Arduino then there are ready-made libraries for it which will give you a quick start.

If you are trying to interface it with some other MCU then the datasheet given below will come in handy. The output given out by the data pin will be in the order of 8bit humidity integer data + 8bit the Humidity decimal data +8 bit temperature integer data + 8bit fractional temperature data +8 bit parity bit. To request the DHT11 module to send these data the I/O pin has to be momentarily made low and then held high as shown in the timing diagram below

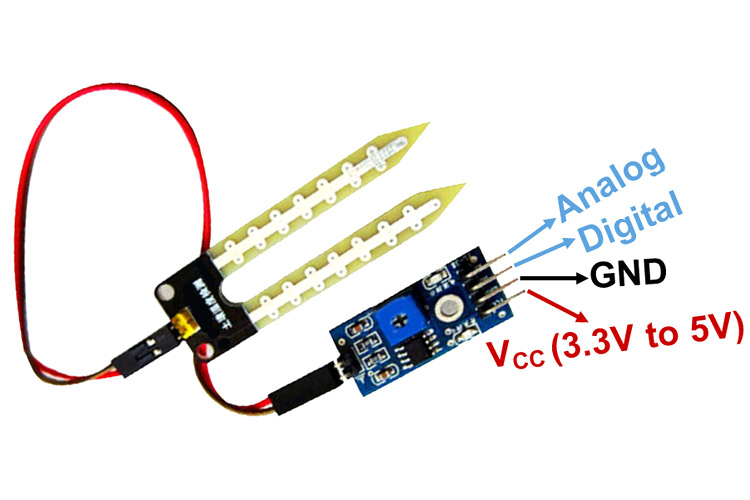


The duration of each host signal is explained in the DHT11 datasheet, with neat steps and illustrative timing diagrams

**Applications:**

* Measure temperature and humidity
* Local Weather station
* Automatic climate control
* Environment monitoring

**Soil Moisture Sensor Module**

[](https://components101.com/sites/default/files/component_pin/Moisture-Sensor-Module-Pinout.jpg)

**Soil Moisture Sensor Module**

This **soil moisture sensor module** is used to detect the moisture of the soil. It measures the volumetric content of water inside the soil and gives us the moisture level as output. The module has both digital and analog outputs and a potentiometer to adjust the threshold level.

**Moisture Sensor Module Pinout Configuration**

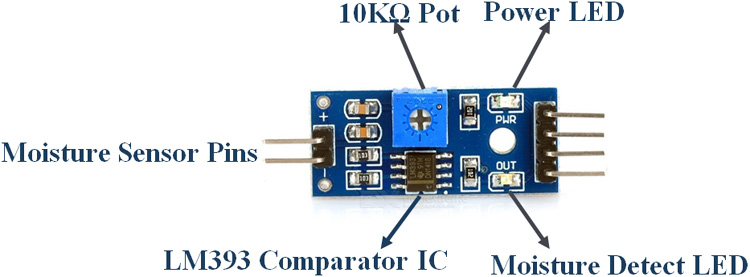
|  |  |
| --- | --- |
| **Pin Name** | **Description** |
| VCC | The Vcc pin powers the module, typically with +5V |
| GND | Power Supply Ground |
| DO | Digital Out Pin for Digital Output. |
| AO | Analog Out Pin for Analog Output |

**Moisture Sensor Module Features & Specifications**

* Operating Voltage: 3.3V to 5V DC
* Operating Current: 15mA
* Output Digital - 0V to 5V, Adjustable trigger level from preset
* Output Analog - 0V to 5V based on infrared radiation from fire flame falling on the sensor
* LEDs indicating output and power
* PCB Size: 3.2cm x 1.4cm
* LM393 based design
* Easy to use with Microcontrollers or even with normal Digital/Analog IC
* Small, cheap and easily available

**Brief about Soil Moisture Sensor Module**

This Moisture sensor module consists of a Moisture sensor, Resistors, Capacitor, Potentiometer, Comparator LM393 IC, Power and Status LED in an integrated circuit.



**LM393 IC**

[LM393 Comparator IC](https://components101.com/ics/lm393-low-offset-voltage-dual-comparators) is used as a voltage comparator in this Moisture sensor module. Pin 2 of LM393 is connected to Preset (10KΩ Pot) while pin 3 is connected to Moisture sensor pin. The comparator IC will compare the threshold voltage set using the preset (pin2) and the sensor pin (pin3).

**Moisture Sensor**

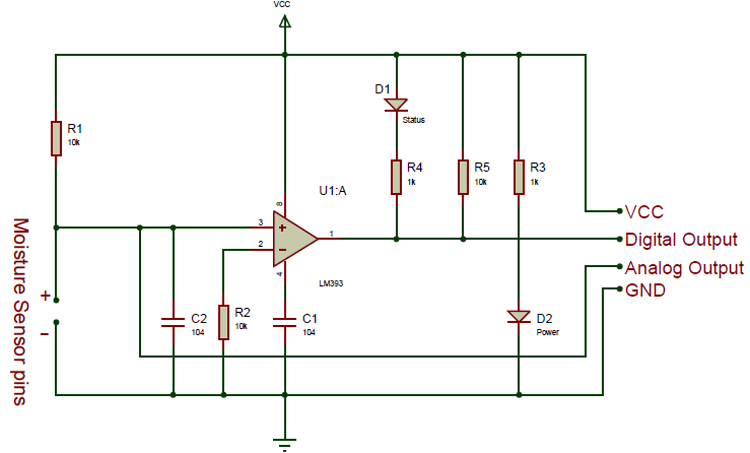
The moisture sensor consists of two probes that are used to detect the moisture of the soil**.**The moisture sensor probes are coated with immersion gold that protects Nickel from oxidation. These two probes are used to pass the current through the soil and then the sensor reads the resistance to get the moisture values.

**Preset (Trimmer pot)**

Using the onboard preset you can adjust the threshold (sensitivity) of the digital output.

**How to Use Soil Moisture Sensor Module**

Moisture sensor module consists of four pins i.e. VCC, GND, DO, AO. Digital out pin is connected to the output pin of LM393 comparator IC while the analog pin is connected to Moisture sensor. The internal Circuit diagram of the Moisture sensor module is given below.

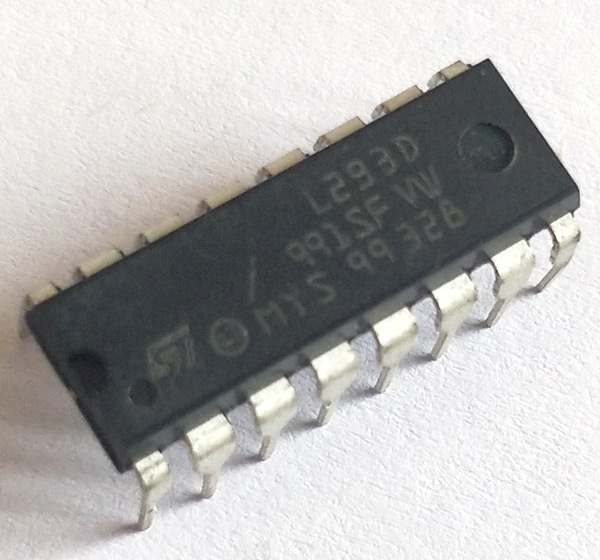


Using a Moisture sensor module with a microcontroller is very easy. Connect the Analog/Digital Output pin of the module to the Analog/Digital pin of Microcontroller. Connect VCC and GND pins to 5V and GND pins of Microcontroller. After that insert the probe inside the soil. When there is more water presented in the soil, it will conduct more electricity that means resistance will be low and the moisture level will be high.

**Applications of Soil Moisture Sensor**

* Gardening
* Irrigation Systems
* Used in Controlled Environments

**L293D Motor Driver IC**

[](https://components101.com/sites/default/files/components/L293D.jpg)

**L293D Motor Driver IC**

[](https://components101.com/sites/default/files/component_pin/L293D-Pinout.png)

**Motor Driver IC L293D Pinout**

**L293D Pin Configuration**

|  |  |  |
| --- | --- | --- |
| **Pin Number** | **Pin Name** | **Description** |
| 1 | Enable 1,2 | This pin enables the input pin Input 1(2) and Input 2(7) |
| 2 | Input 1 | Directly controls the Output 1 pin. Controlled by digital circuits |
| 3 | Output 1 | Connected to one end of  Motor 1 |
| 4 | Ground | Ground pins are connected to ground of circuit (0V) |
| 5 | Ground | Ground pins are connected to ground of circuit (0V) |
| 6 | Output 2 | Connected to another end of  Motor 1 |
| 7 | Input 2 | Directly controls the Output 2 pin. Controlled by digital circuits |
| 8 | Vcc2 (Vs) | Connected to Voltage pin for running motors (4.5V to 36V) |
| 9 | Enable 3,4 | This pin enables the input pin Input 3(10) and Input 4(15) |
| 10 | Input 3 | Directly controls the Output 3 pin. Controlled by digital circuits |
| 11 | Output 3 | Connected to one end of Motor 2 |
| 12 | Ground | Ground pins are connected to ground of circuit (0V) |
| 13 | Ground | Ground pins are connected to ground of circuit (0V) |
| 14 | Output 4 | Connected to another end of Motor 2 |
| 15 | Input 4 | Directly controls the Output 4 pin. Controlled by digital circuits |
| 16 | Vcc2 (Vss) | Connected to +5V to enable IC function |

Featurs:

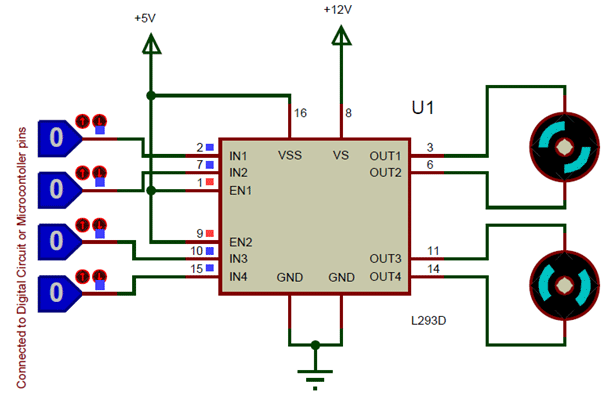
* Can be used to run Two DC motors with the same IC.
* Speed and Direction control is possible
* Motor voltage Vcc2 (Vs): 4.5V to 36V
* Maximum Peak motor current: 1.2A
* Maximum Continuous Motor Current: 600mA
* Supply Voltage to Vcc1(vss): 4.5V to 7V
* Transition time: 300ns (at 5Vand 24V)
* Automatic Thermal shutdown is available
* Available in 16-pin DIP, TSSOP, SOIC packages

**Where to use L293D IC**

The L293D is a popular 16-Pin **Motor Driver IC**. As the name suggests it is mainly used to drive motors. A single **L293D IC** is capable of running two [DC motors](https://components101.com/motors/toy-dc-motor) at the same time; also the direction of these two motors can be controlled independently. So if you have motors which has operating voltage less than 36V and operating current less than 600mA, which are to be controlled by digital circuits like Op-Amp, [555 timers](https://components101.com/555-timer-ic-pinout-datasheet), digital gates or even Micron rollers like Arduino, PIC, ARM etc.. this IC will be the right choice for you.

**How to use a L293D Motor Driver IC**

**Using this L293D motor driver IC** is very simple. The IC works on the principle of **Half H-Bridge**, let us not go too deep into what H-Bridge means, but for now just know that H bridge is a set up which is used to run motors both in clock wise and anti clockwise direction. As said earlier this IC is capable of running two motors at the any direction at the same time, the circuit to achieve the same is shown below.



All the Ground pins should be grounded. There are two power pins for this IC, one is the Vss(Vcc1) which provides the voltage for the IC to work, this must be connected to +5V. The other is Vs(Vcc2) which provides voltage for the motors to run, based on the specification of your motor you can connect this pin to anywhere between 4.5V to 36V, here I have connected to +12V.

The Enable pins (Enable 1,2 and Enable  3,4)  are used to Enable Input pins for Motor 1 and Motor 2 respectively. Since in most cases we will be using both the motors both the pins are held high by default by connecting to +5V supply. The input pins Input 1,2 are used to control the motor 1 and Input pins 3,4 are used to control the Motor 2. The input pins are connected to the any Digital circuit or microcontroller to control the speed and direction of the motor. You can toggle the input pins based on the following table to control your motor.

|  |  |  |
| --- | --- | --- |
| Input 1 = HIGH(5v) | Output 1 = HIGH | Motor 1 rotates in Clock wise Direction |
| Input 2 = LOW(0v) | Output 2 = LOW |
| Input 3 = HIGH(5v) | Output 1 = HIGH | Motor 2 rotates in Clock wise Direction |
| Input 4 = LOW(0v) | Output 2 = LOW |

|  |  |  |
| --- | --- | --- |
| Input 1 = LOW(0v) | Output 1 = LOW | Motor 1 rotates in Anti-Clock wise Direction |
| Input 2 = HIGH(5v) | Output 2 = HIGH |
| Input 3 = LOW(0v) | Output 1 = LOW | Motor 2 rotates in Anti -Clock wise Direction |
| Input 4 = HIGH(5v) | Output 2 = HIGH |

|  |  |  |
| --- | --- | --- |
| Input 1 = HIGH(5v) | Output 1 = HIGH | Motor 1 stays still |
| Input 2 = HIGH(5v) | Output 2 = HIGH |
| Input 3 = HIGH(5v) | Output 1 = LOW | Motor 2 stays still |
| Input 4 = HIGH(5v) | Output 2 = HIGH |

**Applications**

* Used to drive high current Motors using Digital Circuits
* Can be used to drive [Stepper motors](https://components101.com/motors/28byj-48-stepper-motor)
* High current LED’s can be driven
* [Relay](https://components101.com/5v-relay-pinout-working-datasheet) Driver module (Latching Relay is possible)

**A) Raspberry Pi 3 model B+:**

****

**Fig2: Raspberry Pi**

The Raspberry pi is a single computer board with credit card size that can be used for many tasks that your computer does, like games, word processing, spreadsheets and also to play HD video. It was established by the Raspberry pi foundation from the UK.

Simply we can say Raspberry Pi is a credit card sized computer which also serves as microcontroller. It is fast as compared to other controllers.

Raspberry Pi It is a small board computer, introduced by Raspberry Pi foundation in 14th March 2018 and is the most recent version of the Pi boards. It is a modified form of its predecessor Raspberry Pi 3 B that was introduced in 2016 and came with CPU, GPU, USP ports and I/O pins.

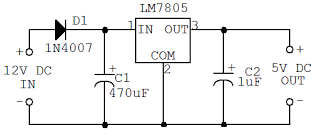
* Raspberry Pi 3 B+ was introduced by Raspberry Pi foundation on 14th March 2018. It is an advanced version of Raspberry Pi 3 B model that was introduced in 2016.
* It is a tiny computer board that comes the model B+ stays ahead in terms of processing speed and comes with an improved wireless capability.
* The dual-band Wi-Fi 802.11ac runs at 2.4GHz and 5GHz and provides a better range in wireless challenging environments and Bluetooth 4.2 is available with BLE support.
* The top side is painted with metal shielding, instead of plastic in the earlier models that acts as a heat sink and drains the excessive amount of heat if the board is subjected to the high temperature or pressure.
* This B+ model is three times faster than Pi 2 and 3 which is a major development in terms of speed, capable of executing different functions at a decent pace.
* The Ethernet port comes with 300 Mbit/s which is much faster than earlier version with 100 Mbit/s speed. It is known as gigabit Ethernet based on USB 2.0 interface.

**Technical specifications**:

* CPU is 64 bit with 1GB RAM (random access memory)
* Contains Broadcom BCM2837B0 chipset
* Comes with 1.4GHz Quad-Core ARM Cortex-A53, 4 cores
* Consists of 40 pin header (26 GPIOs)
* Stereo audio and composite video is supported by 3.5mm jack connector
* 4 USB 2.0 ports
* Gigabit Ethernet
* PoE (power over Ethernet) is a major feature incorporated in this device that lacks in B model
* 2-pin reset header
* Micro SD socket, used to enhance the memory capacity of the board
* MicroUSB power connector, used for transferring power to the device
* HDMI
* CSI camera interface
* Comes with Wi-Fi and Bluetooth facility that were not present in previous Raspberry Pi 1 and 2 versions
* DSI connector for official screen

**Applications**

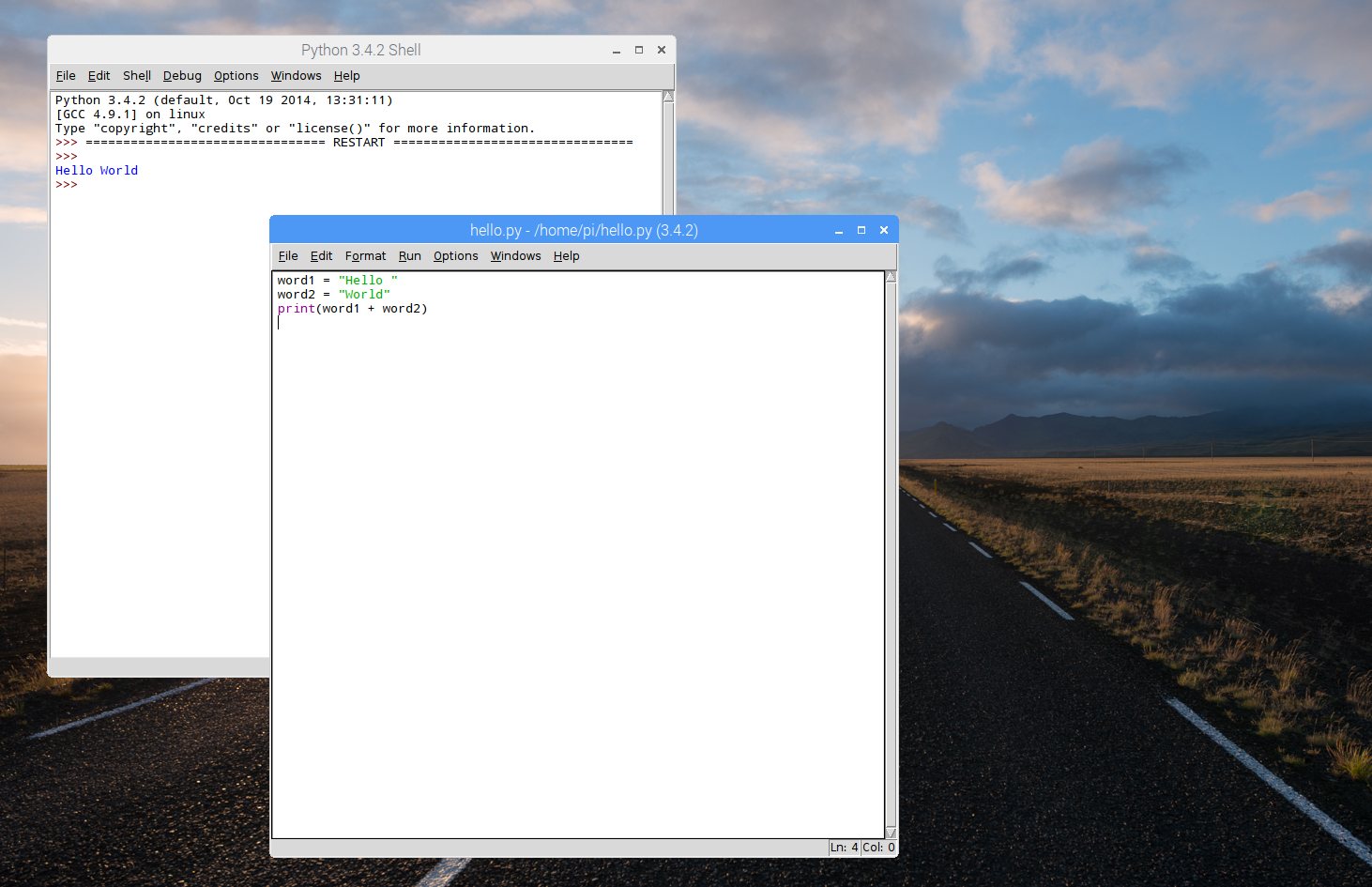
Raspberry Pi comes with a wide range of applications and works as a regular computer in some cases.

* Tablet Computer
* Home Automation
* Controlling Robots
* Coffee Projects
* Arcade Machine
* Media Streamer
* Internet Radio
* Cosmic Computer
* **4.5.10 Power supply:**
* 5V Voltage Regulator Converter AC/DC to DC Step down board. This is a general purpose power regulator and supply board. It basically takes an input of 7-20V AC or DC and outputs a constant voltage of 5V. It can supply a maximum of 1A of current at 5V. Comes with appropriate rectification, filtering circuits, an On/Off switch that controls the power at the output, a power indication LED and mounting holes for easy mounting. All inputs and outputs are provided through standard screw terminals, which allow easy connections. This board is great for generating constant voltages during prototyping, testing, experimenting, etc. Features- Outputs a constant 5V from a 7-20V AC or DC supply Maximum current capacity is 1A Comes with screw terminal connectors for easy connection.
* This is a circuit of a 12V to 5V converter using a 7805 regulated IC. In the previous article we have discussed about the circuit of 12V to 9V converter which can be used to convert any source of 12V DC to 9V DC using 7809 IC. The circuit mentioned here is also a step down DC to Dc converter like the pervious one but it is using LM7805 IC which can provide fixed 5 volt output from any 12V DC source. The circuit is ideal to use with a 12V car battery to step down the voltage to 5 volt DC. LM7805 is an IC of LM78xx series. It has many built in features like thermal shutdown, short circuit protection and safe operating area protection. These type of ICs are commonly used in regulated power supply circuits.
* 
* **Fig.20. power supply circuits.**

**Software Requirements**

**Python:**

Python is an interpreter, high level, interactive and general purpose programming language. It was developed by Guido van Rossum during 1985 – 1990. The source code is available under general public License. Python is named after a TV Show ‘Monty Python’s Flying Circus’ and not after Python-the snake. It supports Object Oriented programming approach for developing applications. We use python IDE to write the program.

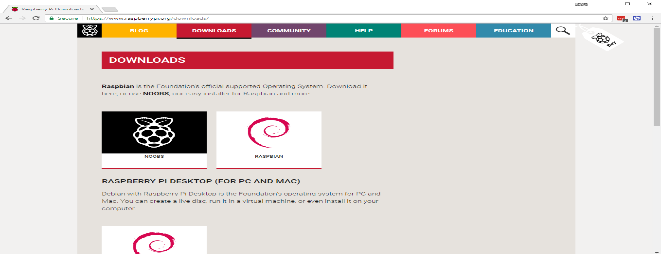


**Fig11: Python Shell**

**Raspbian OS:**

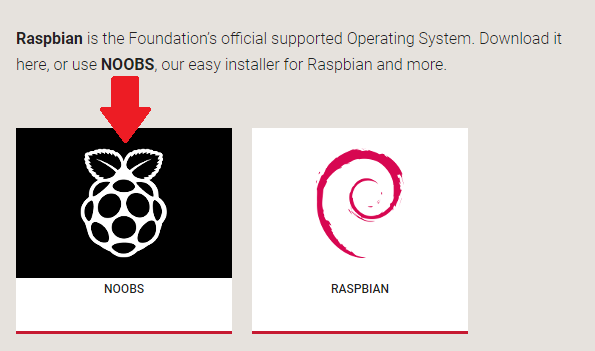
**Downloading NOOBS**

* Using NOOBS is the easiest way to install Raspbian on your SD card. To get hold of a copy of NOOBS



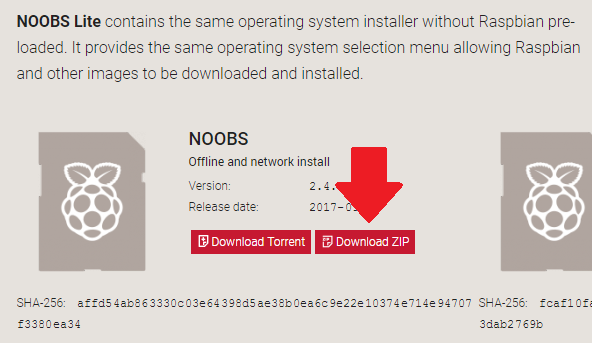
**Fig12: NOOBS software website**

* You should see a box with a link to the NOOBS files. Click on the link.



**Fig13: NOOBS software download**

* The simplest option is to download the zip archive of the files



**Fig14: NOOBS ZIP file to download**

**Formatting the SD Card**

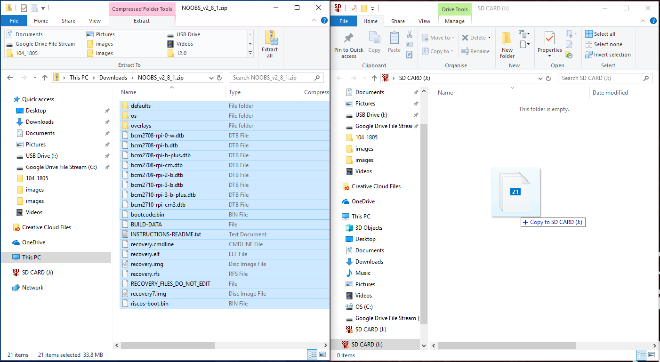
* If the SD card on which you wish to install Raspbian currently has an older version of Raspbian on it, you may wish to back up the files from the card first, as they will be overwritten during this process.
* Visit the SD Association’s website and download SD Formatter 4.0 for Windows or Mac.
* Follow the instructions to install the software.
* Insert your SD card into the computer or laptop’s SD card reader and make a note of the drive letter allocated to it, e.g. F:/.
* In SD Formatter, select the drive letter for your SD card, and format it.

**Extracting NOOBS from the zip archive**

* Next, you will need to extract the files from the NOOBS zip archive you downloaded from the Raspberry Pi website.
* Go to your Downloads folder and find the zip file you downloaded.
* Extract the files and keep the resulting Explorer/Finder window open.

**Copying the files**

* Now open another Explorer/Finder window and navigate to the SD card. It’s best to position the two windows side by side.
* Select all the files from the NOOBS folder and drag them onto the SD card.

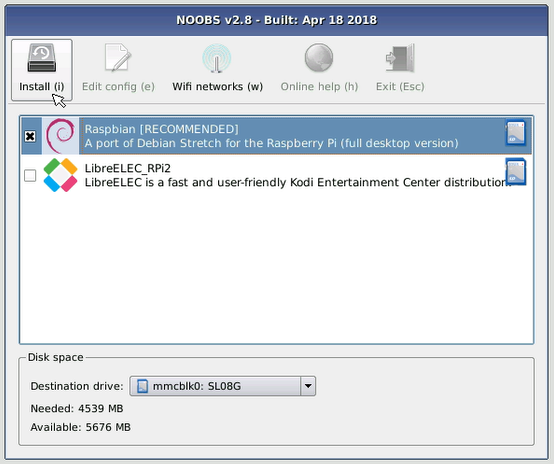


**Fig15: NOOBS folder coping to the SD card.**

* Eject the SD card.

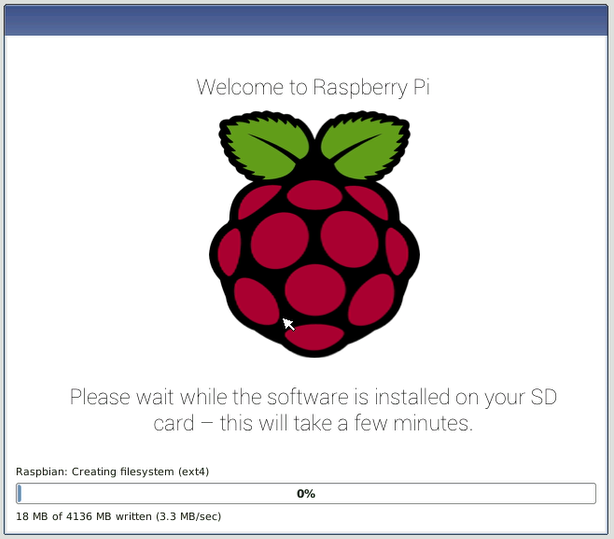
**Booting from NOOBS**

* Once the files have been copied over, insert the micro SD Card into your Raspberry Pi, and plug the Pi into a power source.
* You will be offered a choice when the installer has loaded. You should check the box for Raspbian, and then click Install.



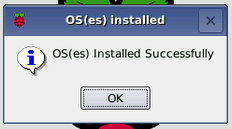
**Fig16: click on install button**

* Click yes at the warning dialog, and then sit back and relax. It will take a while, but Raspbian will install.



**Fig17: status bar of installation process**

* When Raspbian has been installed, click OK and your Raspberry Pi will restart and Raspbian will then boot up.



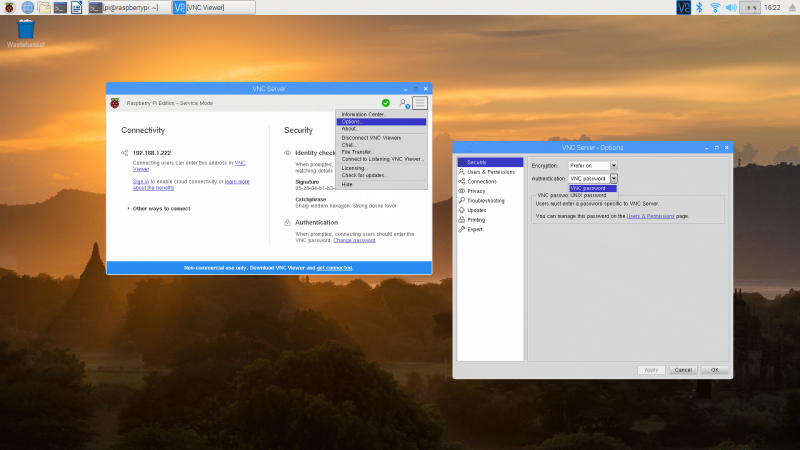
**Fig18: OS installed notification**

**VNC VIEWER:**

VNCstands for Virtual Network Computing. Sometimes it is not convenient to work directly on the Raspberry Pi. Maybe you would like to work on it from another device by remote control.

VNC is a graphical desktop sharing system that allows you to remotely control the desktop interface of one computer (running VNC Server) from another computer or mobile device (running VNC Viewer). VNC Viewer transmits the keyboard and either mouse or touch events to VNC Server, and receives updates to the screen in return.

You will see the desktop of the Raspberry Pi inside a window on your computer or mobile device. You'll be able to control it as though you were working on the Raspberry Pi itself.



**4.6.2 WHY CHOOSE “PHP”:**

Internet has joined the people living around the globe. It has, no doubt, gotten quite tough to sustain your identity in the cyber world as the competition has gone beyond the limits. To make it easier for you to compete and excel in the world of internet, PHP is among the best tools that can be used. PHP is abbreviation of “personal homepage” and sometimes is also known as Hypertext Preprocessor. The latter name is particularly used in the cyber circle. It is, in general, a HTML embedded scripting language being used widely for the web application development. The use of the language has increased in recent times due to the ease it offers to the developer. There are various benefits of using the language over the others developed for the same purpose. Some of the major pros pertinent to the language are discussed as under:

**Double end web development**

Some of the languages used for web development have limitation of purposes. PHP is one of its kinds because it may be used on both front-end and back-end web development. Due to this feature, a programmer may easily alter the present conditions of the website merely by changing a single code. Unlike PHP, other languages need to be uncoded to understand the correlation among the back-end and front-end languages making programming time taking and laborious.

**Why to pay when it’s free?**

Another reason why a programmer must prefer PHP over other languages is its legal free of cost availability. Some of the organizations having its similar programming languages in the market charge programmers against the language they offer. However, PHP may be downloaded and installed using any open source language house easily accessible from one’s computer. Therefore, the basic goal of earning more can be changed into reality by just taking a right step of using PHP as web application development.

**Simplicity and user friendliness**

It goes without saying that everyone wants ease out of the programming language. This is what PHP offers to the users. Unlike C++ and other similar languages, PHP is quite easy to be understood by the users. There is no need of any formal training prior to use the language for the required purpose. PHP programmed web applications are easy to scale and highly secure as compare to applications built in other language.

**Compatible to all the operating software**

Versatility in the available operating software, no doubt, has provided variety to the users but on the other hand has also caused some serious complexities for the programmers. However, you may make it easier for yourself by switching over to PHP as it is compatible to all the famous operating systems. Apple’s famous MAC and Microsoft’s Windows is among the top operating systems that are supported by PHP. Linux is also not out of the line of the compatible systems.

**ABOUT PHP:**

PHP started out as a small open source project that evolved as more and more people found out how useful it was. Rasmus Lerdorf unleashed the first version of PHP way back in 1994.

* PHP is a recursive acronym for "PHP: Hypertext Preprocessor".
* PHP is a server side scripting language that is embedded in HTML. It is used to manage dynamic content, databases, session tracking, even build entire e-commerce sites.
* It is integrated with a number of popular databases, including MySQL, PostgreSQL, Oracle, Sybase, Informix, and Microsoft SQL Server.
* PHP is pleasingly zippy in its execution, especially when compiled as an Apache module on the Unix side. The MySQL server, once started, executes even very complex queries with huge result sets in record-setting time.
* PHP supports a large number of major protocols such as POP3, IMAP, and LDAP. PHP4 added support for Java and distributed object architectures (COM and CORBA), making n-tier development a possibility for the first time.
* PHP is forgiving: PHP language tries to be as forgiving as possible.
* PHP Syntax is C-Like.

**4.6.3 SYSTEM REQUIREMENTS**

**Hardware Requirements**

# Processor : I3/Intel Processor

* RAM **:** 4GB (min)
* Hard Disk **:** 160GB
* Key Board **:** Standard Windows Keyboard
* Mouse **:** Two or Three Button Mouse
* Monitor **:** SVGA

**Software Requirements**

* OperatingSystem **:** Windows Family
* Programminglanguage **:** Core PHP
* Back**-**End **:** PHP
* Front-End **:** HTML, CSS, Bootstrap, Java script
* WebServer **:** Apache, MYSQL

**4.6.4 SYSTEM TESTING**

**Definition:**

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

**TYPES OF TESTS**

* **Unit testing**

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

Test strategy and approach

Field testing will be performed manually and functional tests will be written in detail.

Test objectives

* All field entries must work properly.
* Pages must be activated from the identified link.
* The entry screen, messages and responses must not be delayed.

Features to be tested

* Verify that the entries are of the correct format
* No duplicate entries should be allowed
* All links should take the user to the correct page
* **Integration testing**

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

Acceptance Testing

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

Test Results: All the test cases mentioned above passed successfully. No defects encountered

* **Functional Test**

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals. Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted.

Invalid Input : identified classes of invalid input must be rejected.

Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised.

Systems/Procedures: interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

* **System Test**

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

* **White Box Testing**

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

* **Black Box Testing**

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

**Chapter-5**

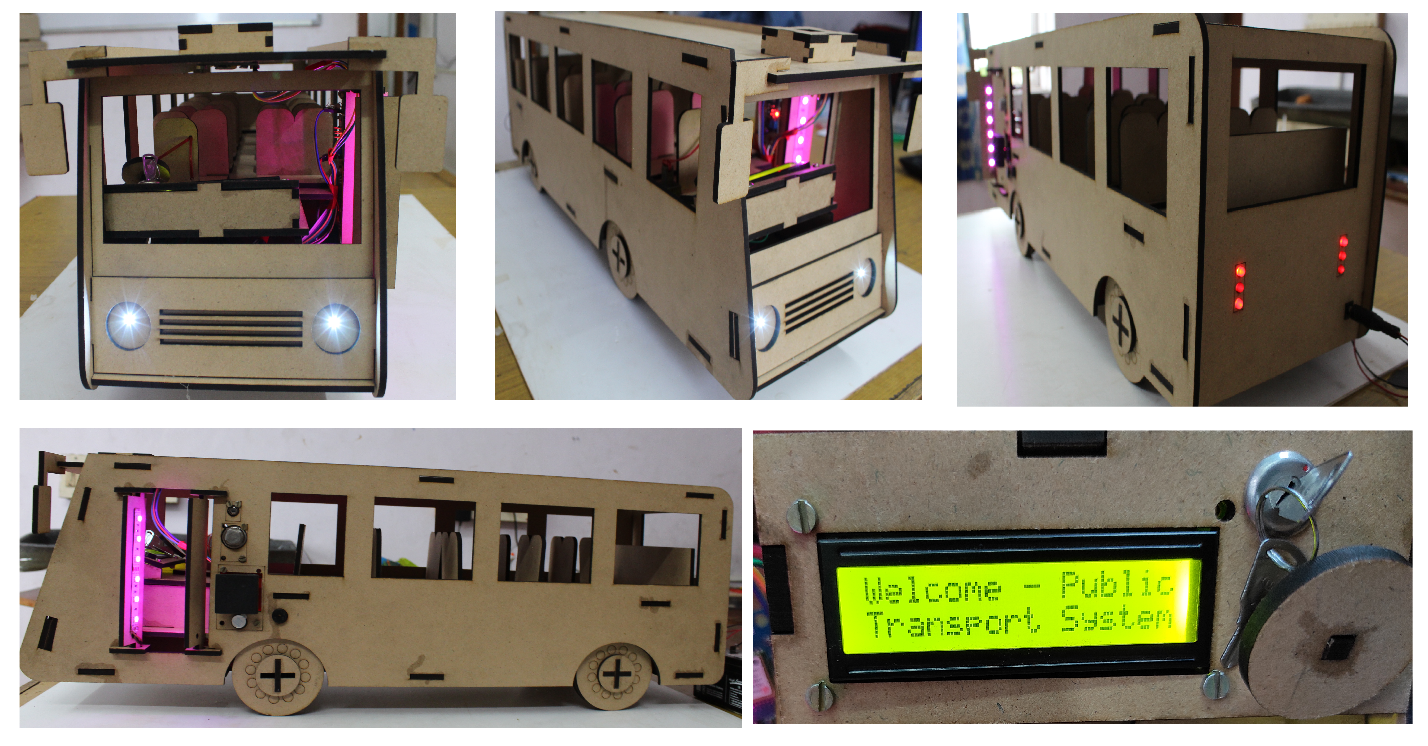
**ADVANTAGES AND APPLICATIONS RESULTS AND DISCUSSION**

* User can track the bus with bus number.
* User can check his travel history.
* User can deactivate his card in case of theft.
* User can get the next stop detailed in LCD.
* User can recharge his wallet.
* User can give the rating for his journey.
* Admin can view the bus travel history and ratings.
* Admin can view the repot like occupancy rate in trip/date wise.
* Admin can view the fuel consumption and km travel day wise/ month wise.
* Admin can view service alerts, fitness alert for the bus.
* Admin can view the overall fines against traffic violation.
* Safety of passengers and vehicles with active alerts and continuous surveillance
* Public address system and multi-way intercom between driver and in and out passengers
* Automatic passenger counting to improve scheduling, routes, and revenue
* Passenger information displays
* Aware of approximate time of arrival to destination.
* Knowing the capacity of vehicle used for transportation.
* Message alerts for every transaction.
* Bus, Logistic, School and collage transportations
* The data will be maintained in the Database.
* We can track the bus at any time.
* User can deactivate the Smart card in case of theft.
* User can view travel history in proposed system.
* Passenger no need to wait a long time for taking ticket. Because in this bus there is no conductor. We can easily travel in this bus.
* Passenger no need to pay the money by cash.
* Passenger can travel with the smart card.
* This proposed system decreases the manual work.

**Chapter-6**

**TESTING RESULTS**

Check all the connection are correctly before giving the power supply. Particularly on VCC and GND connection for all the modules and make sure your hotspot connection should be enable and your HOTSPOT credentials should be like SSID: project, PASSWORD: 123456789. Give the 12v DC supply from the back side sheet. If you do all the connections are correctly then system will on and the project will be look like as below figure. Initially all the LED’s will on IN LCD you can see the welcome message up to setup the Wi-Fi and GPS.



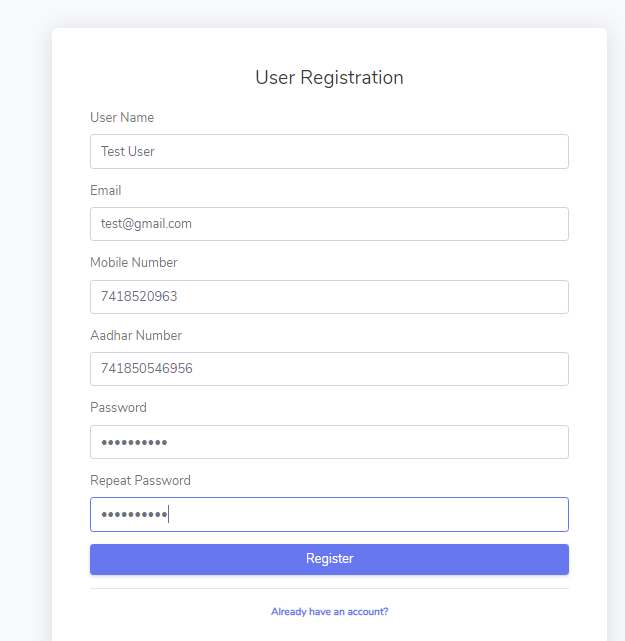
Once GPS will get the signal from satellite and NodeMCU will get the Wi-Fi from your mobile. Then you can start the testing of this project. But Before testing of your device this project you need to setup some software setting the Web server.

**Passenger or User:**

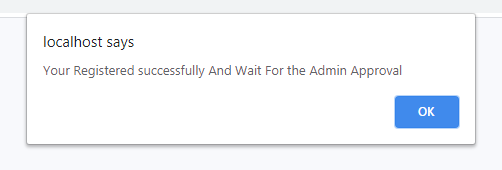
User or passenger need to follow these steps for using this site.

Registration:

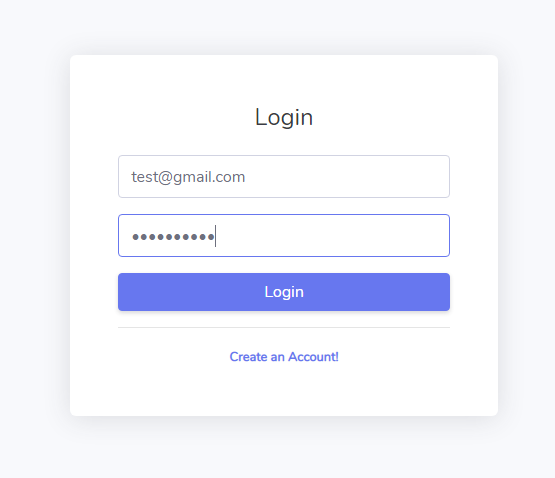
User (or) Passenger can Register by providing their details like name, email, Aadhar card, Mobile Number and password etc.. If the details are unique, user registered successfully to the application. If not user con not registered.



After Registration Completed User get this alert message in that site.

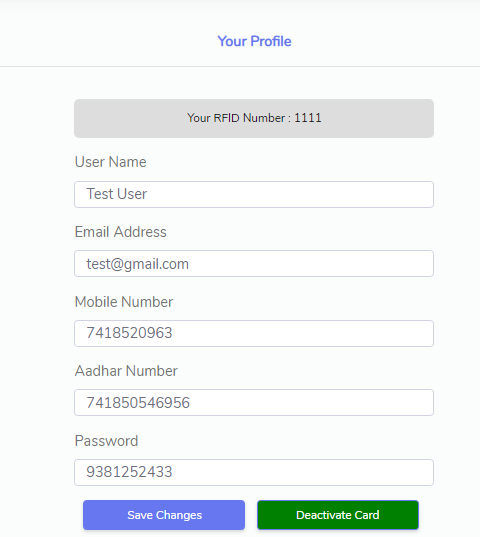


Once organization approved your request User can login to the system by using email id and password. And user will get the **RFID CARD** .

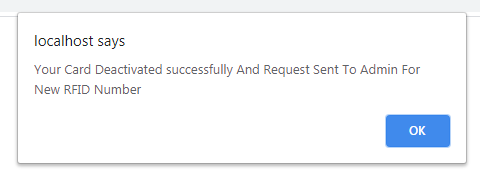


After Login user can do the following things:

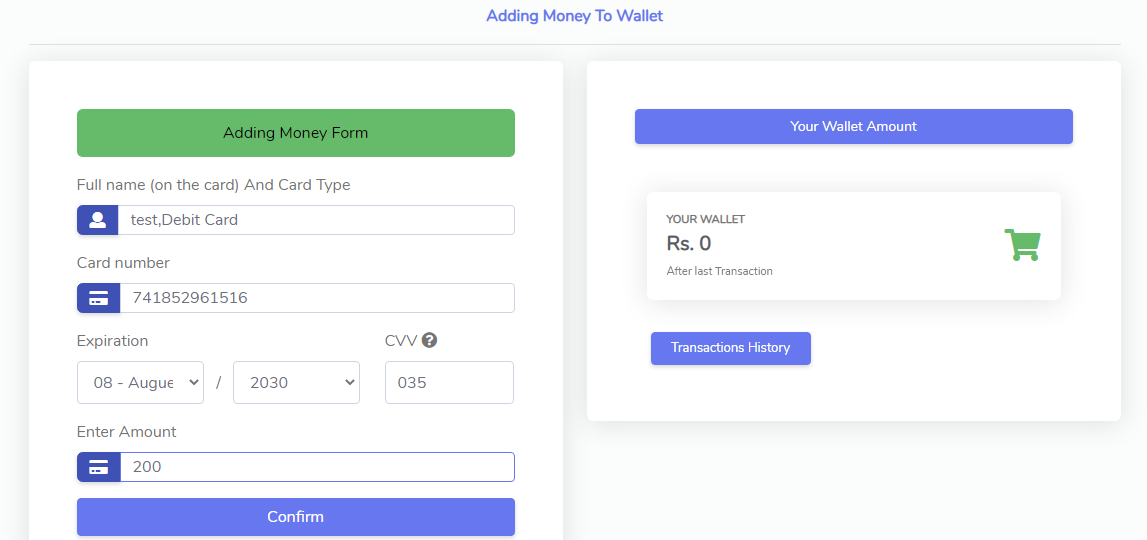
* User can view and edit profile and if he lose his RFID card then he will deactivate his card by clicking deactivate card button.



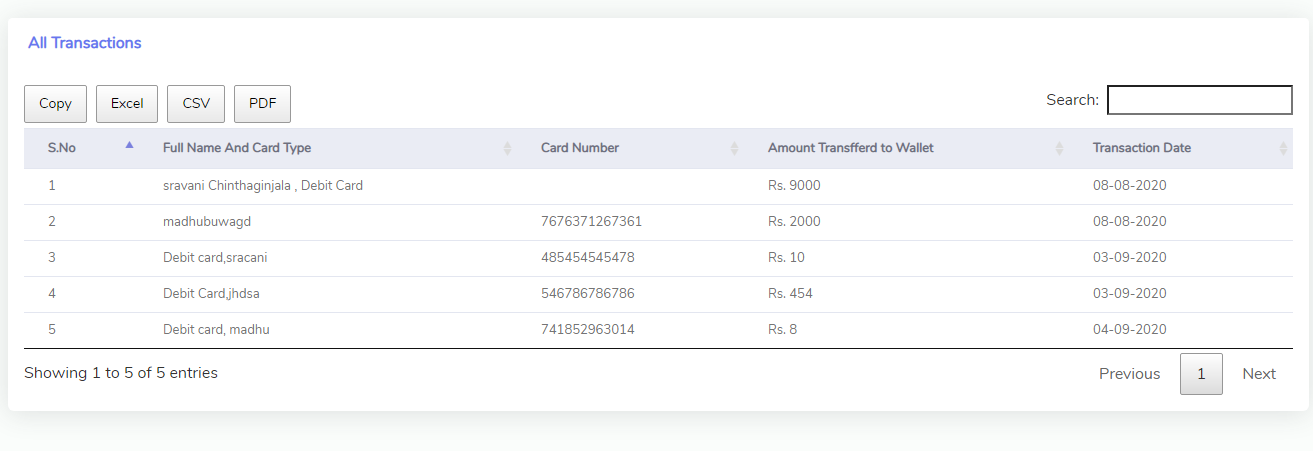
* Once user click the deactivate card button his will deactivate and request will sent to the admin for new RFID card . It shows the following picture.



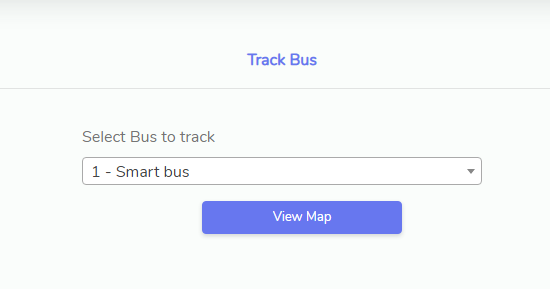
* User can Recharge his wallet by filling the following showed form and also view transactions history.

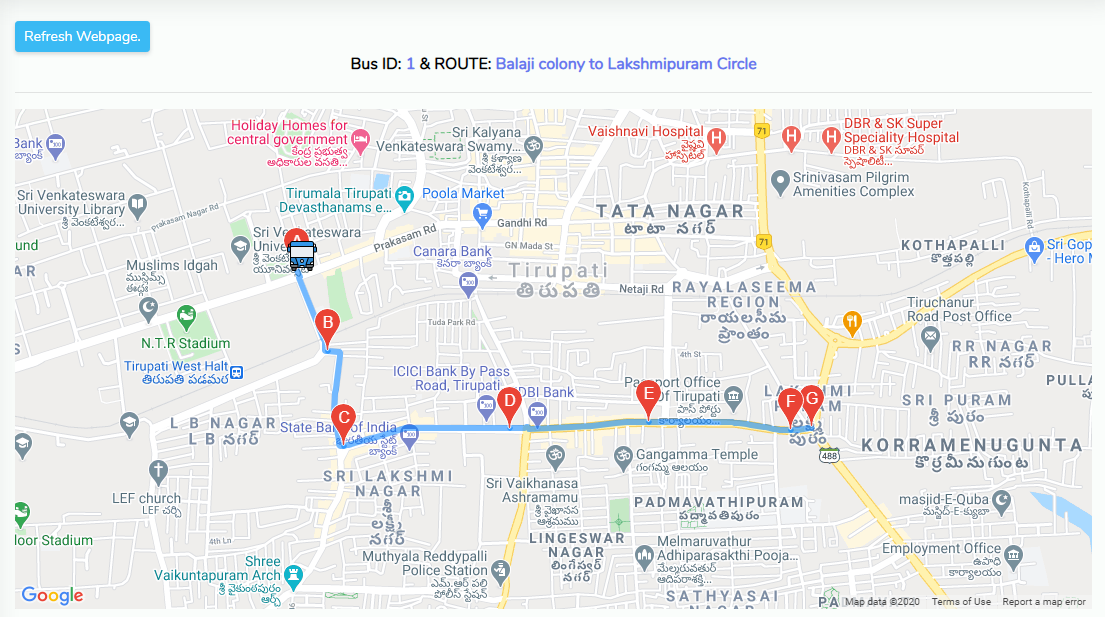


Transaction History

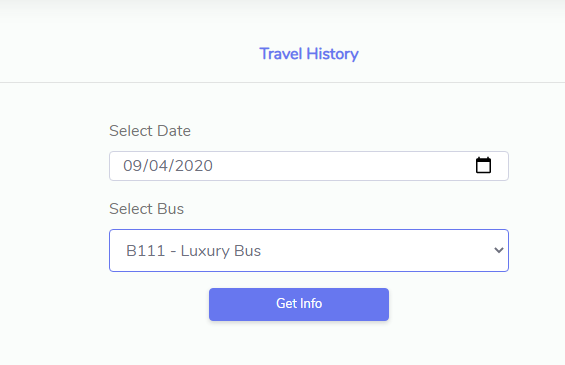


* User can track his bus by selecting bus id.





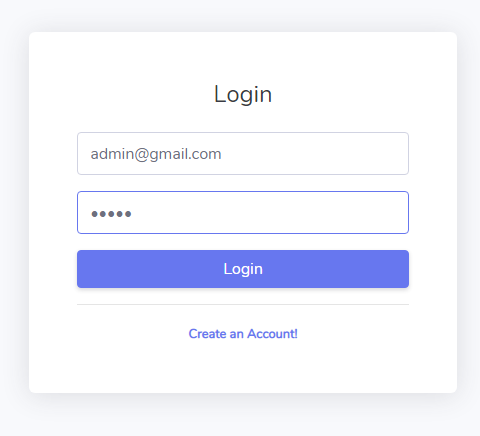
* User can view his travelling history date wise and bus wise



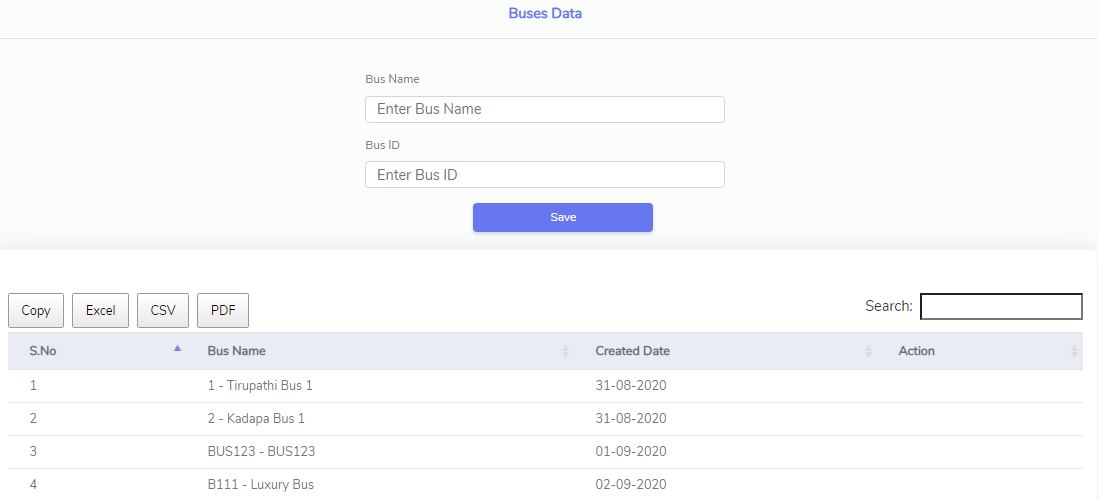


Admin or Organization

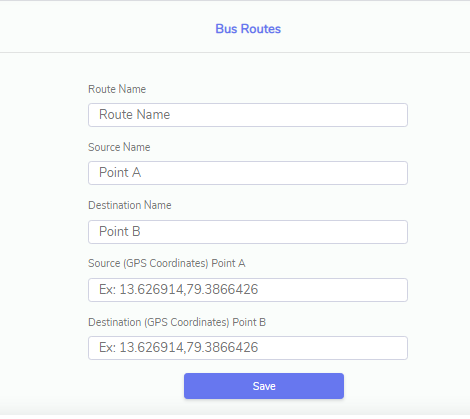
* Admin can login to the system by entering his credentials in the following form

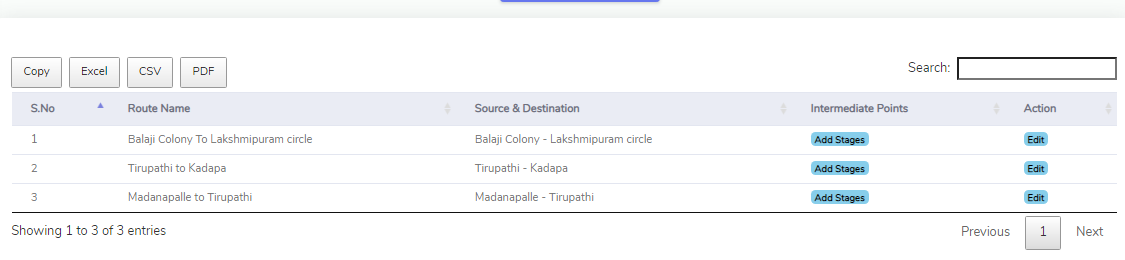


* Admin can add and view the bus data or details.

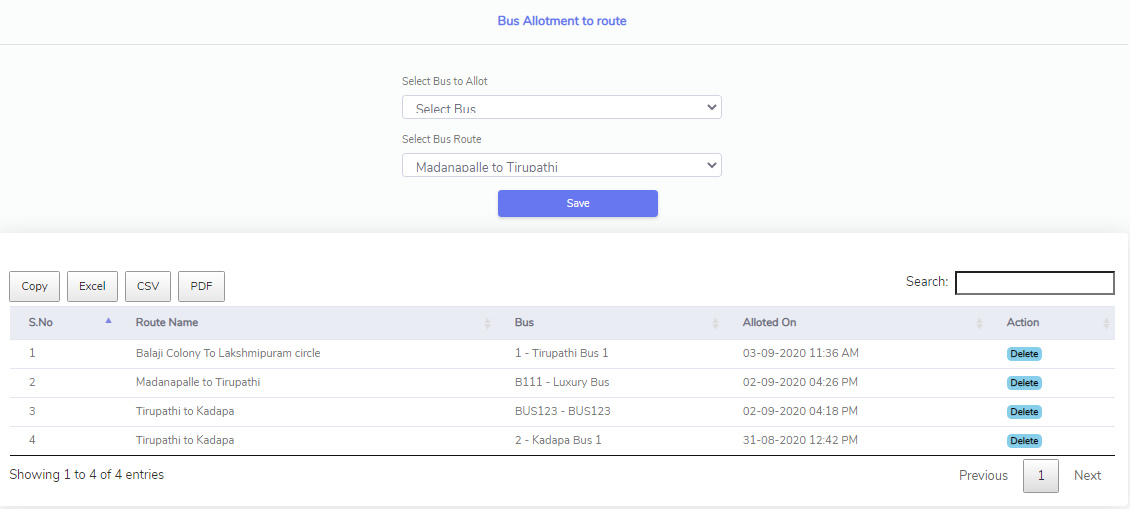


* Admin can add and view the routes and stages in between routes data.

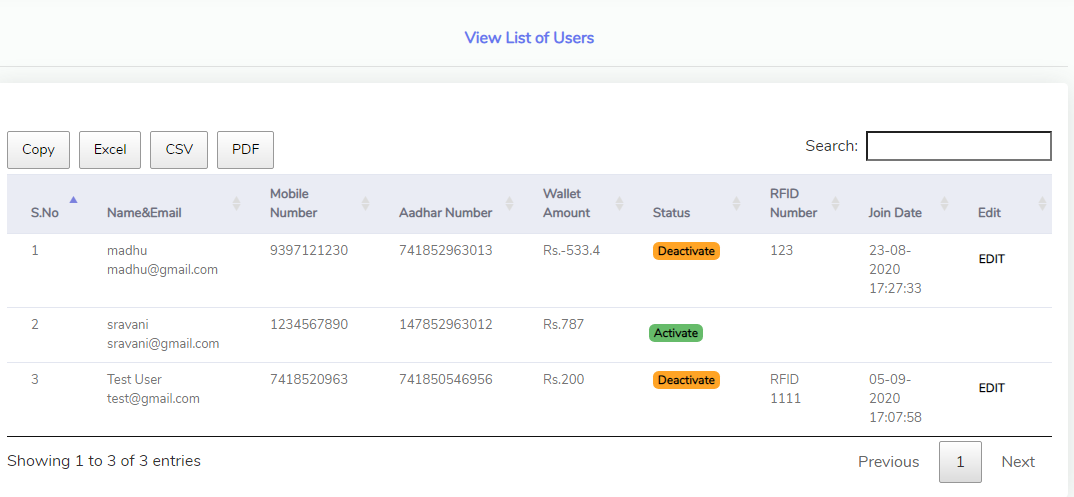




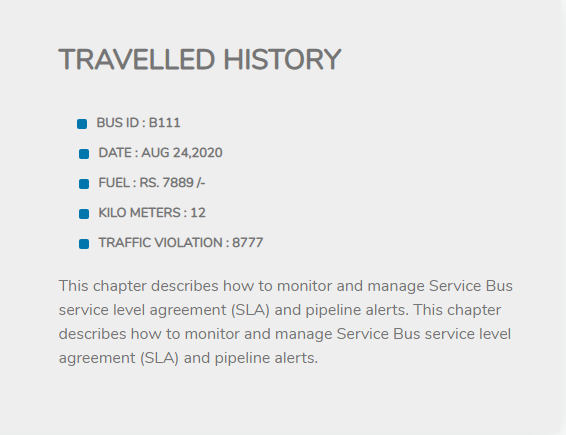
* Admin can add, view, delete the bus allotment to route.



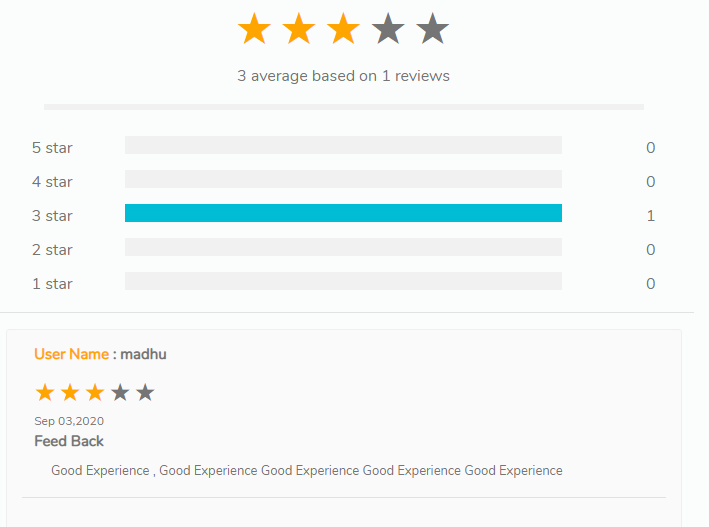
* Admin can activate and deactivate user RFID card .



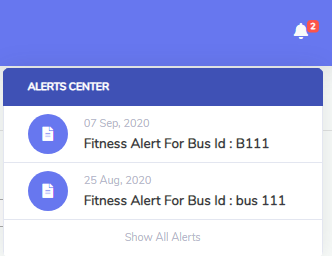
* Admin can view the daily travelling history of bus.



* Admin can view bus rating and feedback.

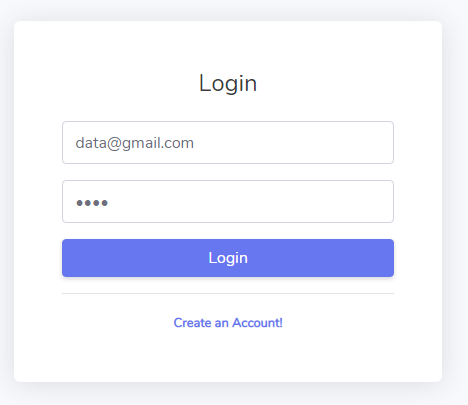


* Admin can view fitness and service alerts data .

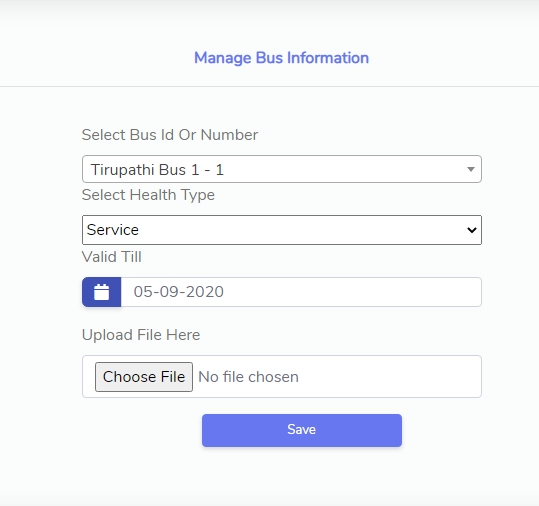


Data Entry Person :

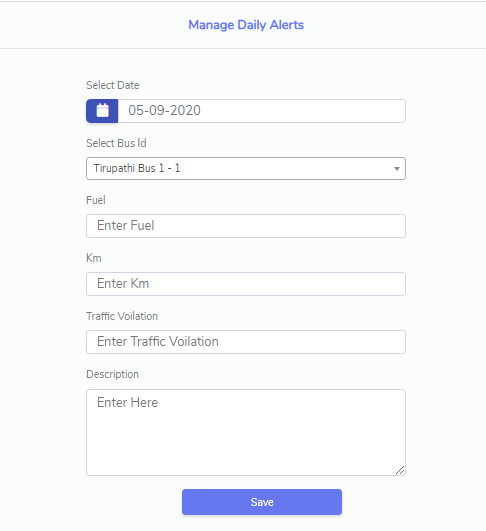
* Data Entry Person can login with email id and password given by organization or admin.



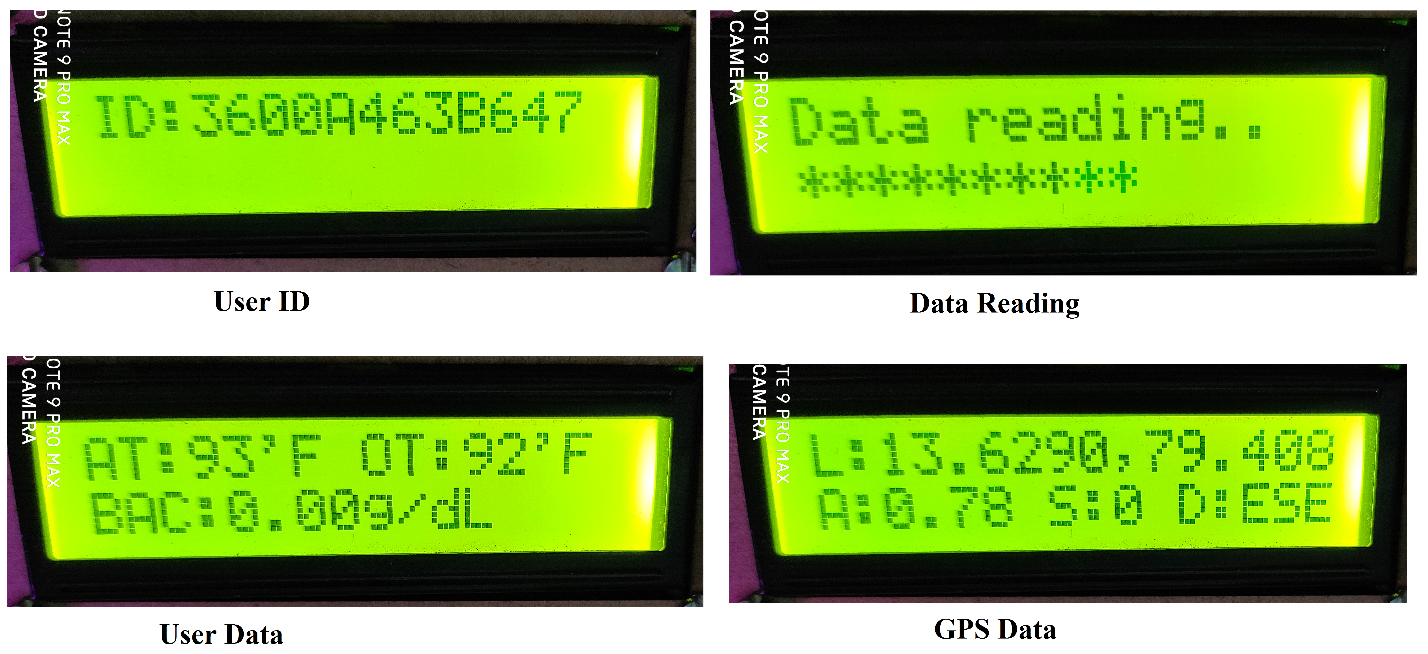
* Data Entry Person can enter the fitness and service alerts data with an attachment.



* Data Entry Person can enter the daily travelled history.



After finishing the website settings. You can start the testing. When you scan the RFID card then it will read the user data like ID, body temperature, BAC, and transfer the data to cloud and cloud server will verify all the parameters and transfer and echo string as valid/invalid. If NodeMCU will get the valid then it LCD will show transaction successful, and if invalid has received then LCD will show transaction failed and buzzer alert will come. As shown in below figure.



Above figure show the output data from LCD

User ID: when card swipe user card ID will show. After that sensor data will read and showed in LCD.

Description for LCD notation. I have explained below.

AT: ambient temperature (surrounding temperature)

OT: object temperature

BAC: Blood Alcohol status

L: longitude, latitude value

A: accelerations (vehicle speed)

S: No. of satellite connected

D: direction

we can check the seat belt alert when bus is in motion if driver didn’t ware the seat belt automatically buzzer and LCD alerts will come. And we can check the over speed alerts also whenever bus crossed its speed60 km/h automatically buzzer and LCD alerts will come.

Admin can monitor the bus and passenger status. If need more information on this you need to refer the software testing document

**CONCLUSION**

* This project introduces wireless technology in the field of agriculture.
* Exploits features of Android platform to help Farmers Significantly.
* Provides a flexible user interface to farmer to control the machine effectively.
* It reduces manual labour requirement which is a boon to the farmers as finding labourers is a very difficult job today.
* The Agribot can work in any sort of climatic condition as well as can work nonstop unlike humans.
* The time required to carry out the five functionalities reduces considerably in comparison with carrying out the same activities manually.
* It is a onetime investment which reduces the overall farming cost considerably.
* This Agribot acts as a gateway to automated smart farming.

**Future scope**

We need to add AI for this to improve the future performance of this project. We can add thermal camera for detecting the body temperature. We can use night vision camera for monitoring driver drowsiness. We need to add ML&AI concepts for calculating bus revenue and investment it will help for organization. And predating the rush hours and low occupancy Hours/trips.

**REFERENCES**

[1] “Agricultural Robot for Automatic Ploughing and Seeding” 2015 IEEE International Conference on Technological Innovations in ICT (TIAR 2015) (Amrita Sneha.A, Abirami.E, Ankita.A, Mrs. R. Praveen, Mrs. R. Srimeena).

[2] “Design and Implementation of Seeding Agricultural Robot” (JIRAS) (P.Usha, V. Maheswari, Dr. V. Nandagopal)

[3]. “Automated Farming Using Microcontroller and Sensors” (IJSRMS) ISSN: 23493371 (Abdullah Tanveer, Abhishek Choudhary, Divya Pal, Rajani Gupta, Farooq Husain) Farming can be done using new technologies to yield higher growth of the crops.

[4] “IOT Based Smart Agriculture” IJARCCE June 2016 (Nikesh Gondchawar1, Prof. Dr. R. S. Kawitkar2)

**Appendix-A**

Project source code:

#include <Wire.h>

#include <Adafruit\_MLX90614.h>

#include <ESP8266WiFi.h>

#include <WiFiClient.h>

#include <ESP8266WebServer.h>

#include <ESP8266HTTPClient.h>

#include <SoftwareSerial.h>

#include <TinyGPS++.h>

#include <LiquidCrystal.h>

const char \*ssid = "project"; //ENTER YOUR WIFI SETTINGS

const char \*password ="123456789";

Adafruit\_MLX90614 mlx = Adafruit\_MLX90614();

LiquidCrystal lcd(16,0,2,13,10,9);

//static const int RXPin = D5, TXPin = D6;

static const int RXPin = 14, TXPin = 13;

static const uint32\_t GPSBaud = 9600;

SoftwareSerial ss(RXPin, TXPin);

TinyGPSPlus gps;

int keyIndex = 0,temp\_a,temp\_o;

WiFiClient client;

float spd;

float sats;

String bearing;

String userid;

float longitude;

float latitude;

String latitude1,longitude1;

unsigned int move\_index = 1;

String ADCData, station, postData;

float sensor\_volt;

float RS\_gas;

float R0;

float ratio;

float BAC,alchol;

String gui="\*\*";

int R2 = 2000,count=0,seat;

void setup() {

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

Serial.begin(9600);

Serial.println();

mlx.begin();

ss.begin(GPSBaud);

lcd.begin(16,2);

lcd.clear();

lcd.setCursor(0,0);

lcd.print("Welcome - Public");

lcd.setCursor(0,1);

lcd.print("Transport System");

delay(2000);

WiFi.mode(WIFI\_OFF); //Prevents reconnection issue (taking too long to connect)

delay(1000);

WiFi.mode(WIFI\_STA); //This line hides the viewing of ESP as wifi hotspot

WiFi.begin(ssid, password); //Connect to your WiFi router

Serial.println("");

Serial.print("Connecting");

// Wait for connection

while (WiFi.status() != WL\_CONNECTED) {

delay(500);

Serial.print(".");

}

//If connection successful show IP address in serial monitor

Serial.println("");

Serial.print("Connected to ");

Serial.println(ssid);

Serial.print("IP address: ");

Serial.println(WiFi.localIP()); //IP address assigned to your ESP

pinMode(D6,OUTPUT);

pinMode(D8,INPUT\_PULLUP);

}

void loop() {

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

HTTPClient http;

if (ss.available() > 0)

{

// sketch displays information every time a new sentence is correctly encoded.

if (gps.encode(ss.read()))

displayInfo();

}

if(Serial.available()>0)

{

userid=" ";

userid=Serial.readString();

Serial.println("USER ID:"+userid);

//Serial.println("latitude: "+latitude1);

lcd.clear();

lcd.setCursor(0,0);

lcd.print("ID:");

lcd.setCursor(3,0);

lcd.print(userid);

delay(1000);

lcd.clear();

lcd.setCursor(0,0);

lcd.print("Data reading..");

delay(100);

for(int k=0; k<8; k++)

{

lcd.setCursor(0,1);

lcd.print(gui);

delay(100);

int sensorValue = analogRead(A0);

sensor\_volt=(float)sensorValue/1024\*5.0;

RS\_gas = ((5.0 \* R2)/sensor\_volt) - R2;

/\*-Replace the value of R0 with the value of R0 in your test -\*/

R0 = 16000;

ratio = RS\_gas/R0;// ratio = RS/R0

double x = 0.4\*ratio;

BAC = pow(x,-1.431); //BAC in mg/L

BAC=BAC\*0.001;

Serial.print("BAC = ");

Serial.print(BAC); //convert to g/dL

Serial.print(" g/dL\n\n");

delay(100);

alchol= alchol+BAC;

gui=gui+"\*\*";

}

BAC=alchol/8;

alchol=0;

Serial.print("F BAC = ");

Serial.print(BAC); //convert to g/dL

Serial.print(" g/dL\n\n");

delay(100);

delay(500);

B: temp\_a=mlx.readAmbientTempF();

temp\_o=mlx.readObjectTempF();

Serial.print("Ambient = "); Serial.print(mlx.readAmbientTempF());

Serial.print("\*F\tObject = "); Serial.print(mlx.readObjectTempF()); Serial.println("\*F");

delay(500);

gui="\*\*";

if (temp\_a > 200)

{

goto B;

}

String line1data= "AT:"+String(temp\_a)+"'F OT:"+String(temp\_o)+"'F";

String line2data= "BAC:"+String (BAC)+"g/dL";

lcd.clear();

lcd.setCursor(0,0);

lcd.print(line1data);

lcd.setCursor(0,1);

lcd.print(line2data);

delay(1000);

if((BAC <0.03)&&(temp\_o < 99))

{

postData = "?bus\_id=1&+&rfid="+userid+"&lat="+latitude1+"long="+longitude1+"&tmp="+temp\_o+"&ah="+String(BAC);

String Link = "http://smartbus.takeoffprojects.com/scan\_users.php" + postData; //http://smartbus.takeoffprojects.com/scan\_users.php?bus\_id=&long=&lat=&card\_number=&tmp

// http://smartbus.takeoffprojects.com/scan\_users.php?bus\_id=&rfid=&lat=&long=&tmp=&ah

http.begin(Link);

int httpCode = http.GET();

String payload = http.getString();

Serial.println(payload);

http.end();

if( payload== "valid")

{

lcd.clear();

lcd.setCursor(0,0);

lcd.print(" Transaction ");

lcd.setCursor(0,1);

lcd.print(" sucessful");

delay(1000);

digitalWrite(D6,LOW);

}

if(payload == "invalid" )

{

lcd.clear();

lcd.setCursor(0,0);

lcd.print(" Transaction ");

lcd.setCursor(0,1);

lcd.print("Faild");

delay(1000);

for(int i=0; i<3;i++)

{

digitalWrite(D6,HIGH);

delay(500);

digitalWrite(D6,LOW);

delay(500);

}

}

}

if(BAC>0.03)

{

lcd.clear();

lcd.setCursor(0,0);

lcd.print("High BAC:"+String (BAC)+"g/dL");

lcd.setCursor(0,1);

lcd.print("for passenger");

for(int i=0; i<3;i++)

{

digitalWrite(D6,HIGH);

delay(200);

digitalWrite(D6,LOW);

delay(100);

}

delay(1000);

}

if(temp\_o >99)

{

lcd.clear();

lcd.setCursor(0,0);

lcd.print("High T:"+String(temp\_o)+"'F");

lcd.setCursor(0,1);

lcd.print("for passenger");

for(int i=0; i<3;i++)

{

digitalWrite(D6,HIGH);

delay(200);

digitalWrite(D6,LOW);

delay(100);

}

delay(1000);

}

}

}

void displayInfo()

{

if (gps.location.isValid() )

{

latitude = (gps.location.lat()); //Storing the Lat. and Lon.

longitude = (gps.location.lng());

latitude1=String(latitude, 6);

longitude1=String(longitude, 6);

spd = gps.speed.kmph(); //get speed

sats = gps.satellites.value(); //get number of satellites

bearing = TinyGPSPlus::cardinal(gps.course.value()); // get the direction

seat= digitalRead(D8);

Serial.println("latitude: "+latitude1+"longitude:"+longitude1+ "Speed:"+String(spd)+"Satellite:"+String(sats)+"Direction:"+String(bearing));

lcd.clear();

delay(500);

lcd.setCursor(0,0);

lcd.print("L:");

lcd.setCursor(2,0);

lcd.print(latitude1);

lcd.setCursor(9,0);

lcd.print(",");

lcd.setCursor(10,0);

lcd.print(longitude1);

lcd.setCursor(0,1);

lcd.print("A:");

lcd.setCursor(2,1);

lcd.print(spd);

lcd.setCursor(6,1);

lcd.print(" S:");

lcd.setCursor(9,1);

lcd.print(int(sats));

lcd.setCursor(11,1);

lcd.print("D:");

lcd.setCursor(13,1);

lcd.print(bearing);

delay(1000);

count= count+1;

Serial.println("count:"+String(count));

if((seat == 1)&&(spd > 10))

{

lcd.clear();

lcd.setCursor(0,0);

lcd.print("Hello! Driver");

lcd.setCursor(0,1);

lcd.print("Ware Seat belt");

delay(1000);

for(int i=0; i<3;i++)

{

digitalWrite(D6,HIGH);

delay(500);

digitalWrite(D6,LOW);

delay(500);

}

}

if(spd > 60)

{

lcd.clear();

lcd.setCursor(0,0);

lcd.print("Hello! Driver");

lcd.setCursor(0,1);

lcd.print("Over Speed Alert!");

delay(1000);

for(int i=0; i<3;i++)

{

digitalWrite(D6,HIGH);

delay(500);

digitalWrite(D6,LOW);

delay(500);

}

}

if (count == 10)

{

HTTPClient http;

postData = "?b\_id=1&lat="+latitude1+"&long="+longitude1+"&speed="+spd;

String Link = "http://smartbus.takeoffprojects.com/save\_track.php" + postData; //http://smartbus.takeoffprojects.com/save\_track.php?bus\_id=1&long=79.157719&lat=13.653105&speed=0

//http://smartbus.takeoffprojects.com/save\_track.php?b\_id=&lat=&long=&speed

Serial.println(Link);

delay(1000);

http.begin(Link);

int httpCode = http.GET();

String payload = http.getString();

Serial.println(payload);

http.end();

count=0;

}

delay(2000);

}

}