

ALGORITHM ASSIGNMENT

NAME: OLADOKUN DAVIDWAY

MATRIC NO: 18/ENG01/016

DEPARTMENT: CHEMICAL ENGINEERING

QUESTION.

One of the major challenges of ABUAD farm, Ado Ekiti during the dry season is the irrigation system of the farm. The board of the company decided the best way to resolve the problem is to automate the system, as a software developer for ABUAD farm, you are mandated to develop software that interacts with the machine. The software through the machine must be able to:

- Read the temperature of the soil.
- Determine the moisture content of the soil.
- Configure time interval for the water system based on the above.
- Triggered an alarm if there is no sufficient water in the tank for the irrigation.
- Enabled password for the system.

A. Discuss the application development following the software development cycle.

B. Critically discuss the hardware and software features.

C. Support your answer with a flowchart and an algorithm.

Draw the Top-down or Bottom-up design approach of the application

INTRODUCTION.

OBJECTIVES.

The main objective of this project is to develop an automated system that solves most problems related to irrigation and agriculture. such as controlling and saving both the water and electricity, Increasing agricultural production using small quantities of water, Minimize manual intervention in watering operations with increasing watering speed, Preserving plants from fungi, and finally. All these features make the automated system sustainable option to be considered to improve the agriculture and irrigation efficiency. The goals of this study are to discover the excellent automation technique for irrigation system automatically controlled through software in a way that allows the user to monitor all information and manage the device immediately from mobile. Researchers in last decade have discussed the issue of the smart irrigation system. Many solutions are resolved from these studies (e.g. level of water in the land, control the system from mobile by SMS)

The objectives to consider are:

- Simplify the irrigation system by installing and designing the whole irrigation system.
- Save energy, which allows the application of smart irrigation system used more other application.
- Optimize water consumption.
- Automated system fully.
- Decrease the cost of operation.
- Make system easy to use by farmers

SOFTWARE DEVELOPMENT PROCESS

The Software Development Process otherwise known as the Software Development Cycle. They are steps involved in building an application. There are six steps involved in the software development Process of the application;

CONCEPTUALIZATION

SPECIFICATION

DESIGN

IMPLEMENTATION

TESTING & DEBUGGING

RELEASE & UPDATE

For the *Irrigation System for ABUAD Farm, Ado ekiti*, the steps listed above would be followed and explain.

CONCEPTUALIZATION.

Conceptualization involves having an idea or concept to solve a problem or develop an application to help solve the problem.

The issue about the Dry season in Ado Ekiti, is that Crops and Plants have hard time getting sufficient water for them to use in order to grow due to the lack rainfall. This problem can be solved by using a method called Irrigation System. Irrigation is the application of controlled amounts of water to plants at needed intervals. Irrigation helps to grow agricultural crops, maintain landscapes, and revegetate disturbed soils in dry areas and during periods of less than average rainfall. As an

Engineer, I have been given a task to create an automated irrigation system software for ABUAD Farm. The software would be able to, read temperature, determine moisture content for the soil, configure time interval for the water system, etc.

SPECIFICATION.

Specification involves the dividing the application into two different Modules, the Software and the Hardware.

Hardware

- Tensiometer (Moisture Sensor)
- Soil temperature Sensor
- Sprinklers
- Pipe System
- Tank / Reservoir
- Source of Water
- Pump
- Water level Sensor
- Rain Sensor
- Relative Humidity Sensor
- Light Sensor
- Bluetooth Module
- Arduino
- Arduino LCD

Software

- GUI (Graphical User Interface)
- Data Base Management
- Timer

- GPS
- Arduino
- Notification System
- Error Detection System
- Weather Forecast
- Access Control
- MIT App Inventor

DESIGN.

Design involves breaking the application into step by step into which the application will flow. Tools used in designing an application i.e. algorithm, Flow chart, Top-Bottom design.

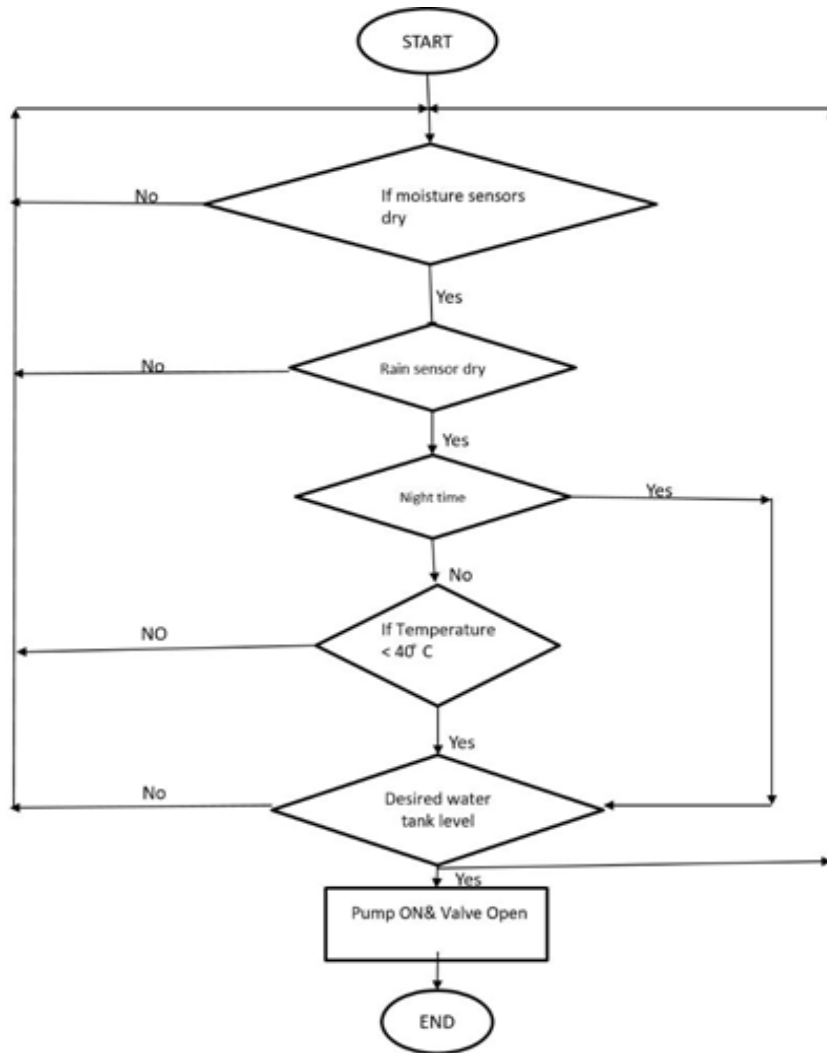
After the application has being installed on user's device, the application will be opened. First of all, a screen of the application will be loaded, where the user has a create an account first using an email. This account would require a password which the user would provide when logging, also the email used would where notifications would be sent to, notification like error detection, when the sprinklers rise to wet the plants and when they are done wetting. After creating this account, the main screen would then load, on the main screen, the user would need to connect the device to the system's Bluetooth. The system will check if the device is connected, if it is connected then it notifies the user that the device is connected otherwise notify the user device is not connected. In the main screen, the user can directly switch On/Off the entire system. Also, on the Main screen, there's a detail button, if tapped on, the crops planted would be displayed and if the any of crops are tapped on, information and details of the crops would show, details like temperature, soil moisture, amount of water, etc. Then the main screen also has GPS, the GPS would be used when moving around to

know your current location and also the location of crops, (this would be useful for large farm lands). On the main screen, there's also a button for Analysis, when this tapped on, Analysis on the crops, how long it will take to mature (time before it reaches harvest period) also, the weather of the Farm if it would be sunny, rainy, cloudy, etc. And the last but not the least, a settings button, where settings for system is located. Where the timer would be set, to operate the sprinklers, and other important settings.

Asides from the application which would be used, there's also going to be a LCD displays for the information which would be displayed on the application at the Farm, close to the system.

IMPLEMENTATION.

Implementation is when the code is written for the application. This is performed by using either a High level language or Low level language. The Arduino code is the code for the system, this code basically is how the system would work.



Arduino Code

```

#include <LiquidCrystal.h>

LiquidCrystal lcd(7, 8, 9, 10, 11, 12);

int Flow1 = 11;

int Flow2 = 12;

int Flow3 = 13;

int Flow4 = 14;

int Received = 0;

const int mo1 = 1; // analog

```



```
const int mo2 = 2; // analog
const int mo3 = 3; // analog
const int mo4 = 4; // analog
const int mo5 = 5; // analog
const int mo6 = 6; // analog
const int mo7 = 7; // analog
const int mo8 = 8; // analog
const int mo9 = 9; // analog
int pump = 2;
const int vlv = 3; // digital
const int vlv2 = 4; // digital
const int vlv3 = 6; // digital
const int water_level = 10; // analog
const int Temp = 15; // analog
const int light = 28; // digital
const int rain = 26; // digital
void setup() {
//      put your setup code here, to run once: analogWrite (water_level, LOW); pinMode
(vlv, OUTPUT);
digitalWrite (vlv, LOW); pinMode (vlv2, OUTPUT); digitalWrite (vlv2, LOW); pinMode (vlv3,
OUTPUT); digitalWrite (vlv3, LOW); pinMode (pump, OUTPUT); digitalWrite (pump, LOW);
pinMode (light, INPUT); digitalWrite (light, LOW); pinMode (rain, INPUT); digitalWrite (rain,
LOW);
lcd.begin(16, 4);
Serial.begin(9600);
}
void loop() {
// put your main code here, to run repeatedly:
//Temperature Sensor
//      int reading = analogRead(Temp);
```

```
//      float voltage = reading * 5.0;
//      voltage /= 1024.0;
//      Serial.print(voltage); Serial.println(" volts");
//      float temperatureC = (voltage - 0.5) * 10 ;
//      Serial.print(temperatureC); Serial.println(" degrees C"); //delay(1000);
int moisture1 = analogRead (mo1); Serial.println("mo 1");
Serial.println(moisture1);
int moisture2 = analogRead (mo2);
//      Serial.println("mo 2");
//      Serial.println(moisture2);
int moisture3 = analogRead (mo3);
//      Serial.println("mo 3");
//      Serial.println(moisture3);
int moisture4 = analogRead (mo4);
//      Serial.println("mo4");
//      Serial.println(moisture4);
int moisture5 = analogRead (mo5);
//      Serial.println("mo 5");
//      Serial.println(moisture5);
int moisture6 = analogRead (mo6);
//      Serial.println("mo 6");
//      Serial.println(moisture6);
int moisture7 = analogRead (mo7);
//      Serial.println("mo 7");
//      Serial.println(moisture7);
int moisture8 = analogRead (mo8);
//      Serial.println("mo 8");
//      Serial.println(moisture8);
```

```

int moisture9 = analogRead (mo9);
//      Serial.println("mo 9");
//      Serial.println(moisture9);
int level = analogRead (water_level);
//      Serial.print("LEVEL:");
//      Serial.println(level);
int light1 = digitalRead (light);
//      Serial.print("light:");
//      Serial.println(light1);
int rain1 = digitalRead (rain);
//      Serial.print("rain:");
//      Serial.println(rain1);
//      1,2,3

if (moisture1 > 800 && moisture2 > 800 || moisture1 > 800 && moisture3 > 800 || moisture2 >
800 && moisture3 > 800)
{
digitalWrite(vlv, HIGH);
digitalWrite (pump, HIGH);
}

if (moisture1 > 800 && moisture2 > 800 && level > 800 || moisture1 > 800 && moisture3 > 800
&&      level > 800 || moisture2 > 800 && moisture3 > 800 && level > 800)
{
digitalWrite(vlv, LOW); digitalWrite (pump, LOW);
}

if (moisture1 > 800 && moisture2 > 800 && rain1 == 0 || moisture1 > 800 && moisture3 > 800
&&      rain1 == 0 || moisture2 > 800 && moisture3 > 800 && rain1 == 0)
{
digitalWrite(vlv, LOW); digitalWrite (pump, LOW);
}

```

```

if (moisture1 < 700 && moisture2 < 700 && moisture3 < 700)
{
digitalWrite(vlv, LOW);
//      digitalWrite (pump, LOW);
}
//4,5,6
if (moisture4 > 800 && moisture5 > 800 || moisture4 > 800 && moisture6 > 800 || moisture5 >
800 && moisture6 > 800 && Temp < 20 && light == 1)
{
digitalWrite (pump, HIGH);
digitalWrite(vlv2, HIGH);
}
if (moisture4 > 800 && moisture5 > 800 && level > 800 || moisture4 > 800 && moisture6 > 800
&&      level > 800 || moisture5 > 800 && moisture6 > 800 && level > 800 )
{
digitalWrite (pump, LOW); digitalWrite(vlv2, LOW);
}
if (moisture4 > 800 && moisture5 > 800 && rain1 == 0 || moisture4 > 800 && moisture6 > 800
&&      rain1 == 0 || moisture5 > 800 && moisture6 > 800 && rain1 == 0 )
{
digitalWrite (pump, LOW); digitalWrite(vlv2, LOW);
}
if (moisture4 < 700 && moisture5 < 700 && moisture6 < 700)
{
digitalWrite(vlv2, LOW);
}
// 7,8,9
56
//

```

```

if (moisture7 > 800 && moisture8 > 800 || moisture7 > 800 && moisture9 > 800 || moisture8 >
800 && moisture9 > 800 && Temp < 20 && light == 1)
{
digitalWrite (pump, HIGH);
digitalWrite(vlv3, HIGH);
}
if (moisture7 > 800 && moisture8 > 800 && level > 800 || moisture7 > 800 && moisture9 > 800
&& level > 800 || moisture8 > 800 && moisture9 > 800 && level > 800 )
{
digitalWrite (pump, LOW); digitalWrite(vlv3, LOW);
}
if (moisture7 > 800 && moisture8 > 800 && rain1 == 0 || moisture7 > 800 && moisture9 > 800
&& rain1 == 0 || moisture8 > 800 && moisture9 > 800 && rain1 == 0 )
{
digitalWrite (pump, LOW);
digitalWrite(vlv3, LOW);
}
if (moisture7 < 700 && moisture8 < 700 && moisture9 < 700)
{
digitalWrite(vlv3, LOW);
}
//Tempreture & light & rain
//
//      if (Temp > 20 && light ) || moisture1 > 800 && moisture3 > 800 || moisture2 > 800
&& moisture3 > 800)
//      {
//
//      digitalWrite(vlv, HIGH);
//      digitalWrite (pump, HIGH);
//      }

```

```

float sensorF4 = 0;
float sensorF3 = 0;
float sensorF2 = 0;
float sensorF1 = 0;
sensorF4 = analogRead(Flow4);
sensorF3 = analogRead(Flow3);
sensorF2 = analogRead(Flow2);
sensorF1 = analogRead(Flow1);
lcd.print("Flow2: "); lcd.print(sensorF2 / 1000); lcd.print("L/S"); lcd.println ("   ");
lcd.print("Flow3: "); lcd.print(sensorF3 / 1000); lcd.print("L/S"); lcd.println ("   ");
lcd.print("Flow4: "); lcd.print(sensorF4 / 1000); lcd.print("L/S"); lcd.println ("   ");
lcd.print("Flow1: "); lcd.print(sensorF1 / 1000); lcd.print("L/S"); lcd.println ("   ");
if (Serial.available() > 0)
{
Received = Serial.read();
}
if (Received == '1') {
digitalWrite(vlv, HIGH);
digitalWrite (pump, HIGH);
}
if (Received == '0') {
digitalWrite(vlv, LOW);
digitalWrite (pump, LOW);
}
if (Received == '2') {
digitalWrite(vlv2, HIGH);
digitalWrite (pump, HIGH);
}
}

```

```
if (Received == '3') {  
    digitalWrite(vlv2, LOW);  
    digitalWrite (pump, LOW);  
}  
if (Received == '4') {  
    digitalWrite(vlv3, HIGH);  
    digitalWrite (pump, HIGH);  
}  
if (Received == '5') {  
    digitalWrite(vlv3, LOW);  
    digitalWrite (pump, LOW);  
}  
if (Received == '6') {  
    digitalWrite (pump, HIGH);  
    digitalWrite(vlv, HIGH);  
    digitalWrite(vlv2, HIGH);  
    digitalWrite(vlv3, HIGH);  
}  
if (Received == '7') {  
    digitalWrite (pump, LOW);  
    digitalWrite(vlv, LOW);  
    digitalWrite(vlv2, LOW);  
    digitalWrite(vlv3, LOW);  
}  
}
```

TESTING & DEBUGGING.

Testing and Debugging, this is checking the performance of the system and checking for errors and removal of errors.

Although at each stage of development the system has been tested, after the front end and the backend of the web application has been developed, they are all integrated together for final testing which can be done manually on the company's system before uploading to the internet. The testing is a critical part of the development processes because it helps detect bugs which are being fixed and helps provide a perfect system for user's use.

RELEASE & UPDATE.

The web application is release to the public for use, and there will be an update and development on the application based on the user's review

HARDWARE & SOFTWARE FEATURES

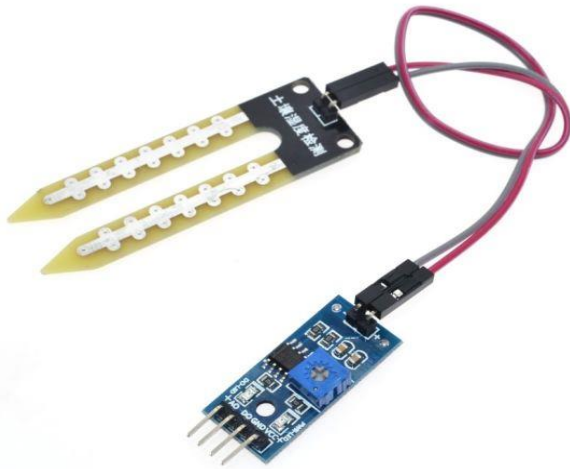
The Hardware and Software are important parts of this application. The application cannot exist without these two modules

Hardware Features,

- Tensiometer (Moisture Sensor), are devices that measure soil moisture tension. They important hardware, because they'll help the application to know how moist the soil is and helps the user to know how frequent the sprinklers would be wetting the crops(soil)



- Soil Temperature Sensor, ST01 is a high quality temperature sensor that is specifically designed for soil temperature measurements in extreme environments. This is also as important as the tensiometer, it helps the user to be able to monitor the system and the crops(soil).

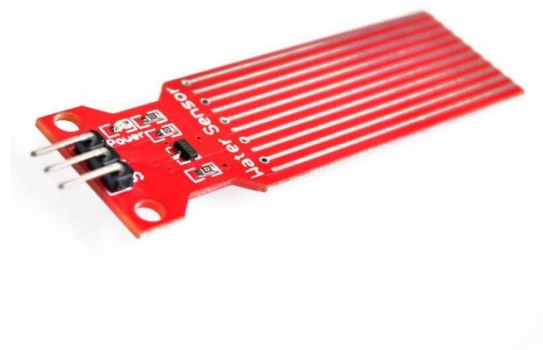


- Sprinklers, is a device that sprays water. Sprinklers are very Important in this system, because they are used to wet the crops, the sprinklers are controlled by a timer which alerts the sprinklers when to spray water.

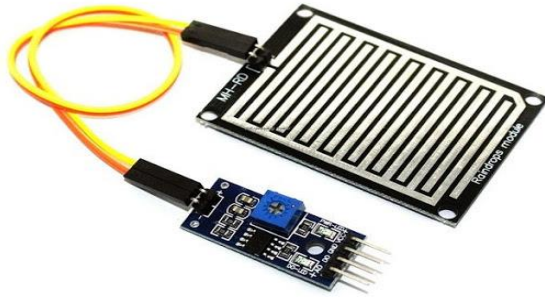


- Pipe System, this will be used to transport water from the Reservoir to the sprinklers. The pipe system plays a very important role in the system.

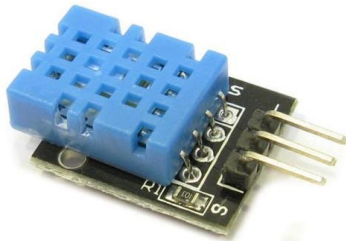
- Tank / Reservoir, this is where water is stored temporarily, and where the pipe system takes water from.
- Source of Water, the source of water could either be from well, borehole, rain, etc. but the most reliable source, is borehole.
- Pump, the pump is used to generate or bring out water from the source.
- Water level Sensor, Level sensors are used to detect the level of substances that can flow. Such substances include liquids, slurries, granular material and powders. Such measurements can be used to determine the amount of materials within a closed container or the flow of water in open channels.



- Rain Sensor, is one kind of switching device which is used to detect the rainfall. It works like a switch and the working principle of this sensor is, whenever there is rain, the switch will be normally closed.



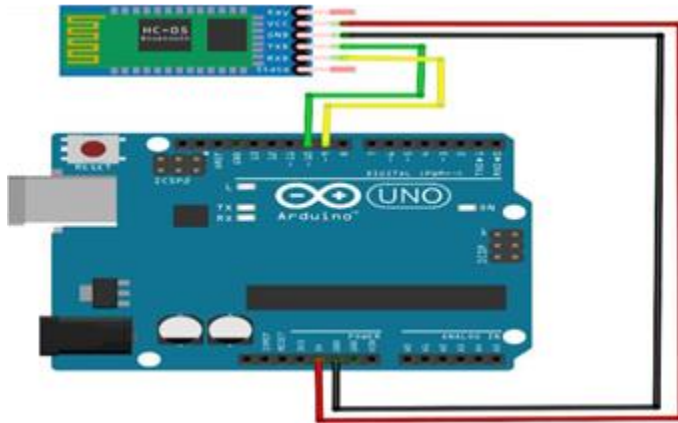
- Relative Humidity Sensor, A humidity sensor (or hygrometer) senses, measures and reports both moisture and air temperature. The ratio of moisture in the air to the highest amount of moisture at a particular air temperature is called relative humidity.



- Light Sensor, it is a sensor that can be used to detect light is an LDR



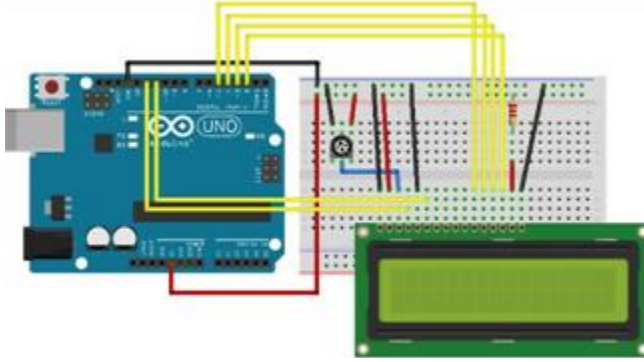
- Bluetooth Module, HC 05/06 this tool works on serial connections and information is sent via Bluetooth when a particular button is pressed. Data is sent to the ARDWINO via TX, the signal is transmitted from the ARDWINO and the RX receives the signal from ARDWINO. The data is verified if gotten information is 1 the Driven turns on turns OFF when gotten information is 0



- Arduino, is "an open-source electronics platform based totally on clean-to-use hardware and software". The Arduino control panel programmed by Arduino c and is based on C and C ++



- Arduino LCD(Liquid Crystal Library), This library permits an Arduino board to control Liquid Crystal Display (LCD) that is on the Hitachi HD44780 chipset, which is on most text-based LCDs



Software Features,

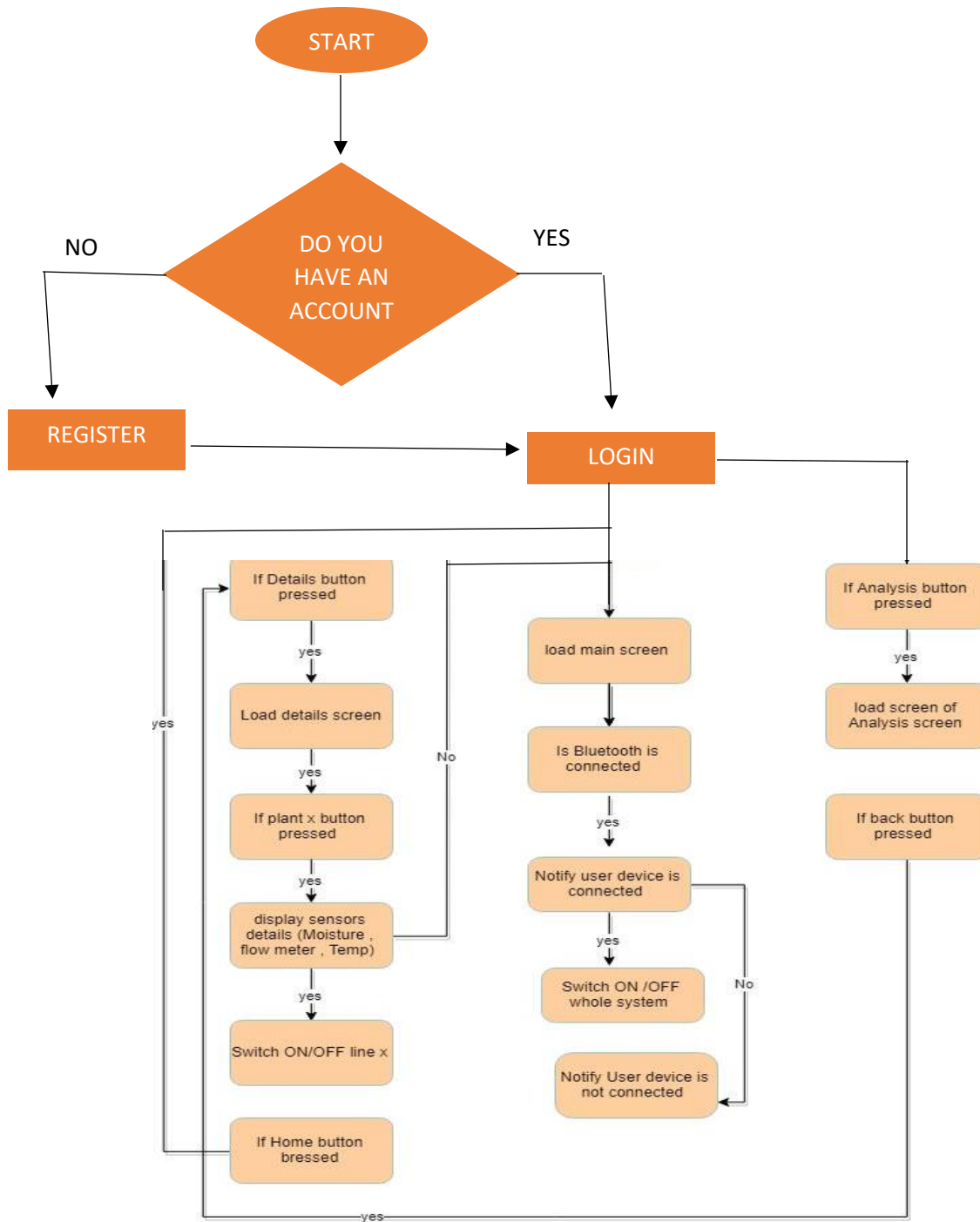
- GUI (Graphical User Interface), The graphical user interface is a form of user interface that allows users to interact with electronic devices through graphical icons and audio indicator such as primary notation, instead of text-based user interfaces, typed command labels or text navigation
- Data Base Management, A database management system (DBMS) is a software package designed to define, manipulate, retrieve and manage data in a database. A DBMS generally manipulates the data itself, the data format, field names, record structure and file structure. It also defines rules to validate and manipulate this data
- Timer, used to indicate how many times someone has done something, but in this case, its to indicate the number of times the system tells the sprinklers to wet the crops.
- GPS, which stands for Global Positioning System, is a radio navigation system that allows land, sea, and airborne users to determine their exact location at any point in time. This is very useful in ABUAD Farm cause of the large size, if the user gets lost.
- Notification System, a notification system is a combination of software and hardware that provides a means of delivering a message to a set of recipients
- Error Detection System, error detection refers to the techniques used to detect noise or other impairments introduced into data

while it is transmitted from source to destination. When a error id detected in the system, both the email of the user and the application will notify the user.

- Weather Forecast, it is used to predict the conditions of the atmosphere for a given location and time.
- Access Control, is a security technique that regulates who or what can view or use resources in the system. It is a fundamental concept in security that minimizes risk to the organization
- MIT App Inventor, it's an app inventor used to control the system from a far.



FLOW CHART



ALGORITHMS.

1. Start
2. Button. If it is clicked, it goes to the user to About Page, which have information about the app.
3. Button. If it is clicked, it goes to Help Page, which have information about the app.
4. List Picker. When it is clicked, it shows a list of all connected Bluetooth devices When a Bluetooth device is clicked.
5. Connection. If the device is connected, the text color change to green for feedback and says 'Connected'.
6. ON /OFF buttons allow the user to switch on or off the system.
7. Details button. When clicked take the user to a details page which shows all the information and let user control system line by line.
8. List Picker. When it is clicked, it shows a list of all connected Bluetooth devices When a Bluetooth device is clicked.
9. These two labels, shows the temperature and amount of water go from tank flow4 is the main one.
10. Plant button. When clicked it shows user list of details about plant such as (Name of the plant, humidity and amount of water).
11. These three labels, it is not visible unless the user clicks on plant button, when user click on plant1 all these labels become visible.
12. Responsive button, when it is clicked, it takes a user to the Home page.
13. ON/OFF button in each line allow a user to switch on or off each line individually
14. End

TOP BOTTOM DIAGRAM.

