IOT BASED SMART CROP PROTECTION SYSTEM FOR AGRICULTURE

PROJECT REPORT

CHAPTER 1

INTRODUCTION

1.1PROJECT OVERVIEW

The Internet of Things (IoT) is a popular method of linking objects and collecting data. When it comes to India's farmers, they are currently in dire straits due to factors such as farm size, technology, trade, government regulations, and climate conditions. IoT frameworks are used to handle and interact with data and information via the Internet of Things. Users can register their sensors, generate data streams, and process data in the system. IoT may be used in a variety of agricultural approaches. Soil and plant monitoring, environmental monitoring (such as moisture and temperature), transportation, supply chain management, infrastructure management, control system management, animal monitoring, and pest control, to name a few, are all required for agricultural production. There are a number of additional elements that have a higher impact on productivity. Farmers' farm fields are located miles distant from his residence, allowing them to see precise changes in crop yield. With an estimated population of 1500 million people in India by 2050 with agriculture remains rural people's major source of income, the focus should be on increasing agricultural production. Farmers will be able to collect live data on temperature, soil moisture, and other aspects using an IoT based Agriculture Stick, which will allow them to undertake smart farming and improve their total production and product quality. To increase crop yield, soil characteristics must be improved because they are an important critical aspect in agriculture. Fertilizers are being used to provide more crops. Fertilizers are primarily used to increase crop yields, and they must be utilized to a particular extent. Agriculture is vital to the economies of many countries throughout the

world. Despite economic development, agriculture remains the economy's backbone. However, there will be significant crop loss due to animal interference and fires in agricultural lands. The crop will be damaged altogether. A large number of farmers will lose their livelihoods. To avoid financial losses, it is necessary to protect agricultural fields and farms against animal and fire damage. The project's main goal is to create an intruder alert system for the farm in order to prevent losses due to animals and fire. This Arduino Uno-based device makes use of a microcontroller. A motion sensor detects wild animals approaching the area. As a result, a proper detection system could assist save their lives as well as the crops. This technology assists us in keeping wild animals out of rural areas while also providing surveillance capabilities.

1.2PURPOSE

The purpose of this project is to grant monitoring device for crop safety to animal outbreaks and environment circumstances . This supports to preserve stretch and cash by dipping the physical exertion, else obligatory if the cultivators themselves have to afford guard for their crops with their endless physical administration . Wildlife regularly wreck eminence crops, because of which annual manufacturing of vegetation reduces inflicting monetary victims to cultivators . Agriculturalist suicide is huge bother due to less harvest. This low harvest is duet the circumstance of two most significant purposes i.e. Crop wrecked via untamed animals and Crop wrecked by meteorological conditions The ranchers will treasure these SMS containing location .The prime thing of this task is to furnish a great reply to this distress. Each time either the wild animal or species are identified through PIR sensor which stimulates the web camera and gives rise to alert the buzzer in the locality, associates to the farmer direct to the cloud . When the moisture content is inferior to a terrifying level the sensor planted makes the water pumps to turn on .This ensures the complete safety of crops from animals also as from the weather conditions thus prevent the farmers .

CHAPTER 2

LITERATURE SURVEY

2.1 EXISTING PROBLEM

The biggest challenges faced by IoT in the agricultural sector are lack of information, high adoption costs, and security concerns, etc. Most of the farmers are not aware of the implementation of IoT in agriculture. Boundary walls and solar fences around the sensitive areas are built to prevent the wild animal attacks. But this system doesn't allow the animals to have a large living range and in dependence of movement. Overhead or underground structures as in are built to divert the wild animals into a different path not interfering with vehicle traffic. But this system takes longer duration, labor and moreover not economical and satisfactory some devices of information technology, viz., radio collars with very high frequency, global positioning system and satellite uplink facilities, are being used by the research institutions to monitor the movement of lions, tigers, elephants, olive riley turtles, and other wild animals to understand their movements and their use pattern of the habitat. But installation of the system becomes difficult and is not always possible.

2.2 REFERENCES

1. Development of IoT based smart security and monitoring devices for agriculture T . Baranwal, PK. Pateriya ,2016 6th International ConferenceWireless Sensor System

2. According to the Concept of IoT - Internet of Things .Juan Felipe Corso Arias , Yeison Julian Camargo Barajas , Juan Leonardo Ramirez Lopez ,2018 7th International Conference on wireless sensor system.

3. IoT Based Smart Sensors Agriculture Stick for Live Temprature and Moisture Monitoring using Arduino, Cloud Computing & Solar Technology. Anand Nayyar, Vikram Puri, 2016 The International Conference On Communication And Computing Systems.

4.https://smartinternz.com/assets/docs/Smart%20Home%20Automation%20using%20IBM%20cloud%20Service s%20(1).pdf

5.https://smartinternz.com/assets/docs/Smart%20Home%20Automation%20using%20IBM%20cloud%20Service s%20(1).pdf

6. https://openweathermap.org/

7.https://smartinternz.com/assets/docs/Sending%20Http%20request%20to%20Open%20weather%20map%20website%20to%20get%20the%20weather%20forecast.pdf

8.https://www.youtube.com/watch?v=cicTw4SEdxk

9.https://smartinternz.com/assets/docs/Smart%20Home%20Automation%20using%20IBM%20cloud%20Service s%20(1).pdf

2.3 PROBLEM STATEMENT DEFINITION

To provide efficient decision support system using wireless sensor network which handle different activities of farm and gives useful information related to farm. Information related to Soil moisture, Temperature and Humidity content. Due to the weather condition, water level increasing Farmers get lot of distractions which is not good for Agriculture. Water level is managed by farmers in both Automatic/Manual using that mobile application. It will make more comfortable to farmers. Performing agriculture is very much time consuming .It should utilize minimum resources in terms of hardware and cost. This overcomes the manual operations required to monitor and maintain the agricultural farms in both automatic and manual modes. It should be able to measure the increase or decrease in level of water as well as moisture in the soil. Specifically, small-scale smart irrigation systems are utilized to provide the solution for dissimilar variety of plants in spite of getting the solution for moisture related issues Weather conditions like temperature, humidity and moisture are difficult to check manually frequently . Farmer suicide is turning into big problem due to low productiveness amongst farms . This low productiveness is due to the fact of two main reasons, Crop ruined by means of untamed weather conditions untamed animal attacks, small types of species, insects, some hazardous snakes and weather circumstances. Low productivity is because of the fact of two most important motives i.e. Crop destroyed via untamed animals and Crop damaged by using nature object . The main objective of this assignment is to furnish a fantastic answer to this trouble, as a result with the purpose of the economic losses incurred through the support of our farmers are minimized to get truthful crop yield.

CHAPTER 3

IDEATION & PROPOSED SOLUTION

3.1 EMPATHY MAP CANVAS

An empathy map is a collaborative tool teams can use to gain a deeper insight into their customers. Much like a user persona, an empathy map can represent a group of users, such as a customer segment.

Mural Link:

https://app.mural.co/invitation/mural/arunachalacollegeofengineeri3543/1663867624803?sender=u83fa0371bd3960ac8b936599&key=3f89ed91-3626-4598-857a-ea1269694beb

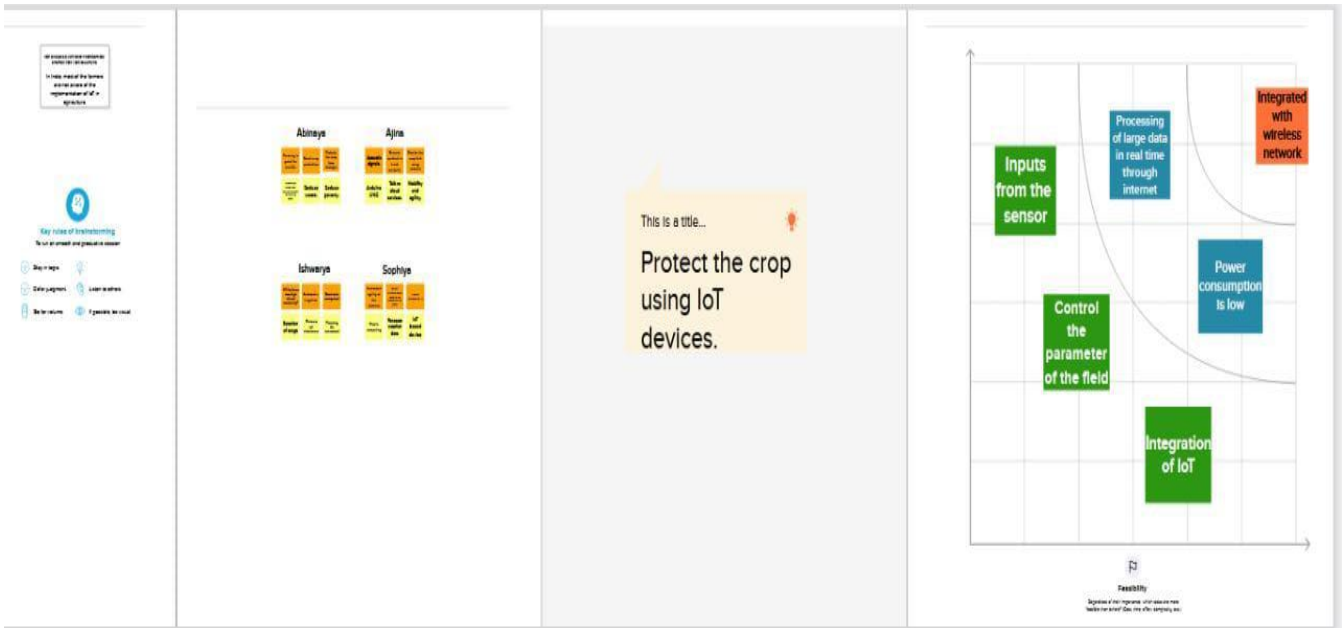
3.2 IDEATION & BRAINSTORMING

Brainstorming is a group problem-solving method that involves the spontaneous contribution of creative ideas and solutions.

Customer Problem Statement :

To provide efficient decision support system using wireless sensor network which handle different activities of farm and gives useful information related to farm. Information related to Soil moisture, Temperature and Humidity content.Water level is managed by farmers in both Automatic/Manual using that mobile application. It will make more comfortable to farmers. Performing agriculture is very much time consuming .It should utilize minimum resources in terms of hardware and cost. This overcomes the manual operations required to monitor and maintain the agricultural farms in both automatic and manual modes. It should be able to measure the increase or decrease in level of water as well as moisture in the soil.

Brainstorming :



3.3 PROPOSED SOLUTION

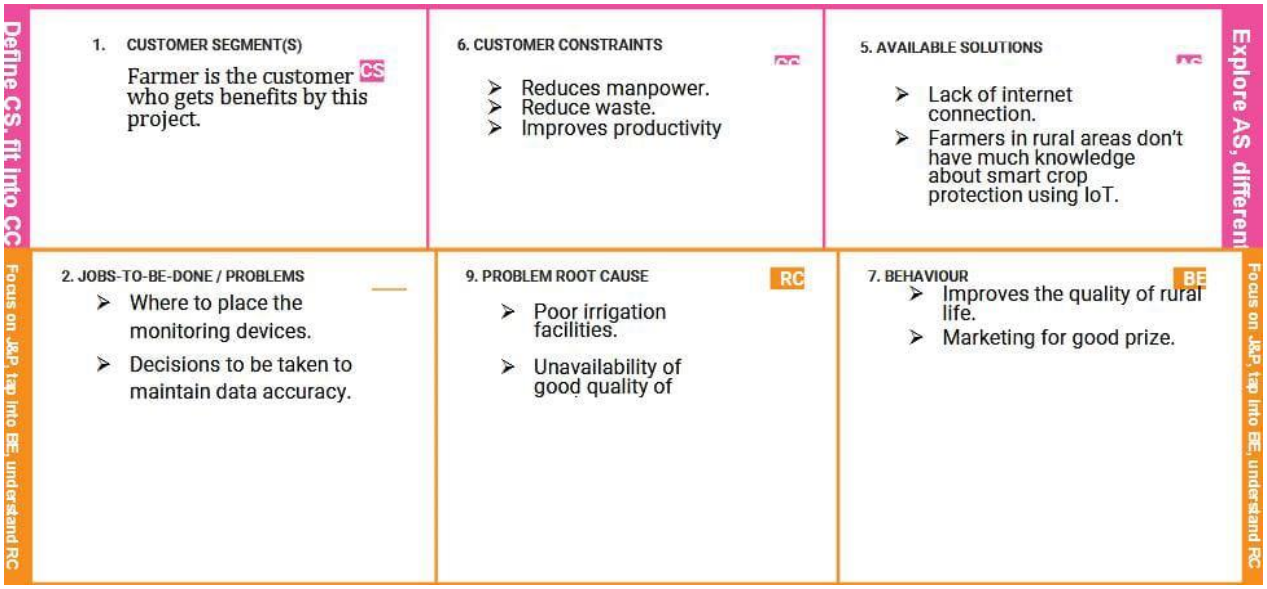
Proposed Solution means the technical solution to be provided by the Implementation agency in response to the requirements and the objectives of the Project.

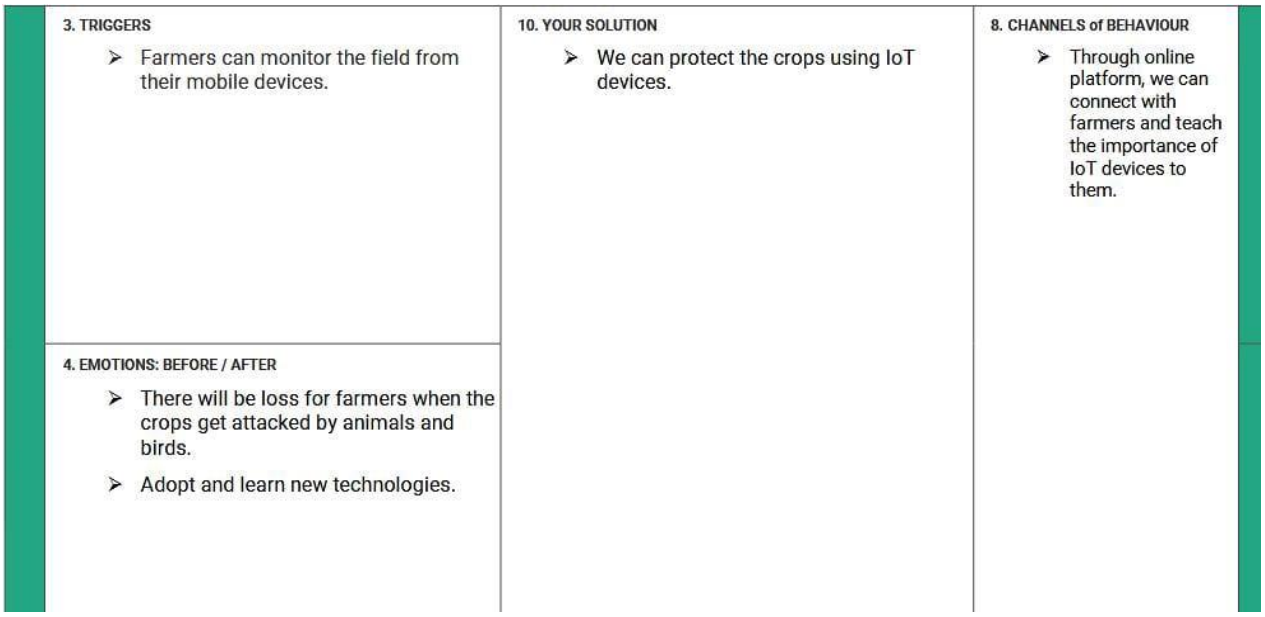
S.NO	Parameter	Description
1.	Problem Statement	❖ To protect the crops using IoT. ❖ Protects the crops from animal and birds attacks. ❖ Also determine the climatic changes and protect the crops.
2.	Solution Description	❖ IoT frameworks are used to handle and interact with information about crops and fields.
3.	Novelty	❖ Specialized sensors are used for monitoring crops. ❖ A PIR sensor is used to detect the animals movement and sends signal to the controller.
4.	Social Impact	❖ There will be no loss for the farmer when the crops are protected using IoT devices ❖ Profit will be high.
5.	Scalability of the Solution	❖ Reduce wastage ❖ Improve productivity

		❖ Process automation ❖ Intelligent data collection
6.	Business Model	❖ This product will be generated and delivered to farmers whoever need smart crop protection for their fields

3.4 PROBLEM SOLUTION FIT

Problem-Solution Fit - this occurs when you have evidence that customers care about certain jobs, pains, and gains.





CHAPTER 4

REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENTS

A functional requirement defines a function of a system or its component, where a function is described as a specification of behavior between inputs and outputs.

FR NO	Functional Requirement	Sub Requirement(story/subtask)
FR -1	Knowledge of seeds	Seeds protect and nourish the embryo or young plant. They usually give a seedling a faster start than a sporeling from a spore, because of the larger food reserves in the seed and the multicellularity of the enclosed embryo.
FR-2	Use of Farmer	Smart farming can make agriculture more profitable for the farmer. Decreasing resource inputs will save the farmer money and labor, and increased reliability of spatially explicit data will reduce risks.

FR-3	Use of Resource	resource inputs in crop production include fuel for tractors and other equipment, water, machinery, fertilizer, pesticides, and packaging materials such as plastic and cardboard.
FR-4	Harvesting	The goal of good harvesting is to maximize crop yield and minimize any crop losses and quality deterioration.
		Harvesting can be done manually, using hands or knifes and it can be done mechanically with the use of rippers, combine harvesters or other machines.

4.2 NON-FUNCTIONAL REQUIREMENTS

Nonfunctional Requirements (NFRs) define system attributes such as security, reliability, performance, maintainability, scalability, and usability. They serve as constraints or restrictions on the design of the system across the different backlogs.

FR NO	Non Functional Requirements	Description
NR-1	Usability	The developed ICT agricultural tools focus on very important agricultural services such as crop disease detection, crop yield predictor will help them to estimate the crop yield which will help them to make decisions in future , recommendation of best crop will help farmers to grow crops that will benefit in their respective region, help famers to locate the pesticide vendors, weather services, discussion forum to communicate.
NR-2	Reliability	Reliability Means Consistency and Accuracy Reliability ensure that a farmer can get a field task accomplished in a timely manner ,it also reduces lost time and productivity

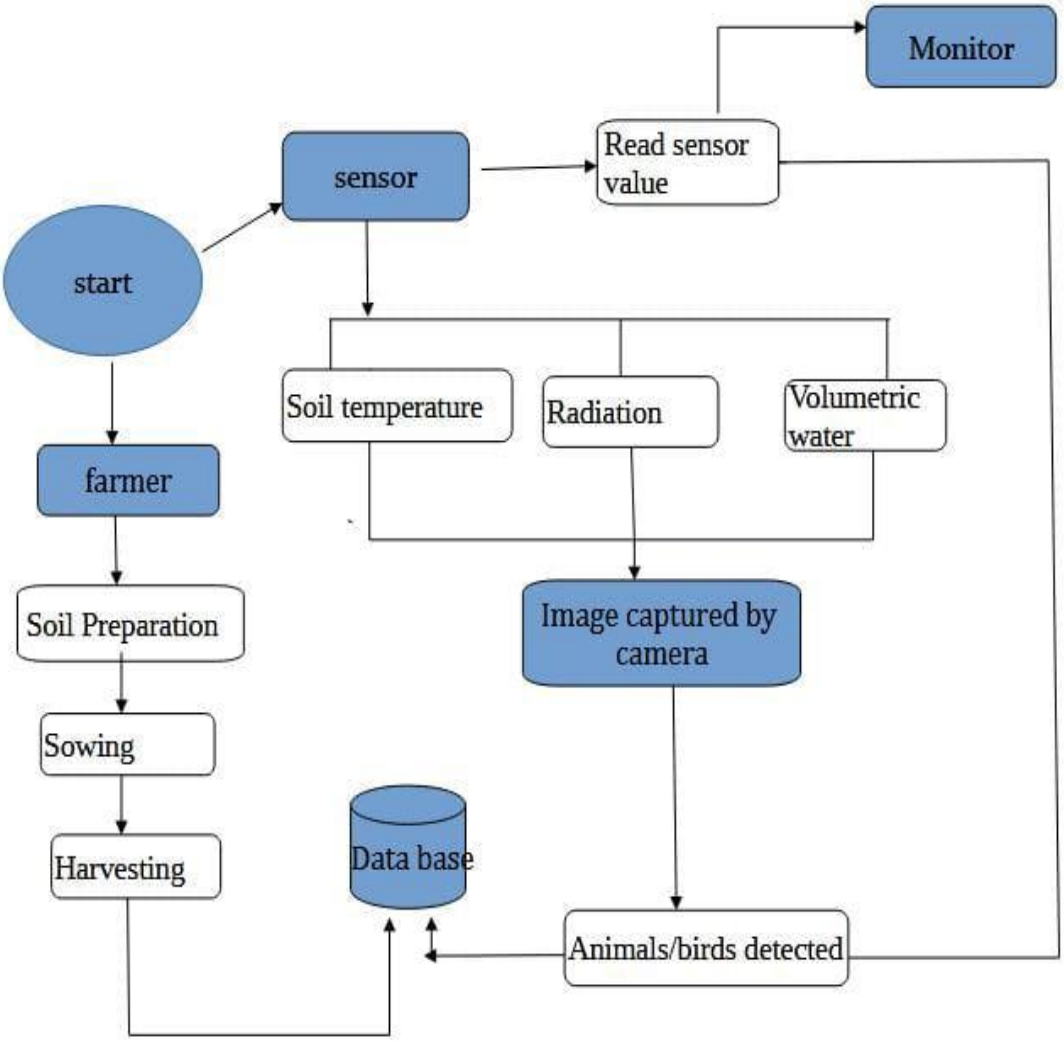
NR-3	Performance	The economic performance of the agricultural industry can be measured in terms of net value added at factor cost, which is gross value added adjusted for the consumption of fixed
		capital, and subsidies and taxes on production.
NR-4	Availability	The basic resources for agriculture are sunlight, soil and water, besides the seeds and animal breeds, and human Endeavour. Another important input is the agro technique. Agricultural production is adversely affected if any of these factors is limited, or disturbed.
NR-5	Scalabilty	Scaling means maximising the impact of agricultural interventions through horizontal or vertical approaches. Horizontal strategies often reach more project beneficiaries by, for example, increasing the size of farms or implementing a service or technological innovation over a wider geographical area.

CHAPTER 5

PROJECT DESIGN

5.1 DATA FLOW DIAGRAMS

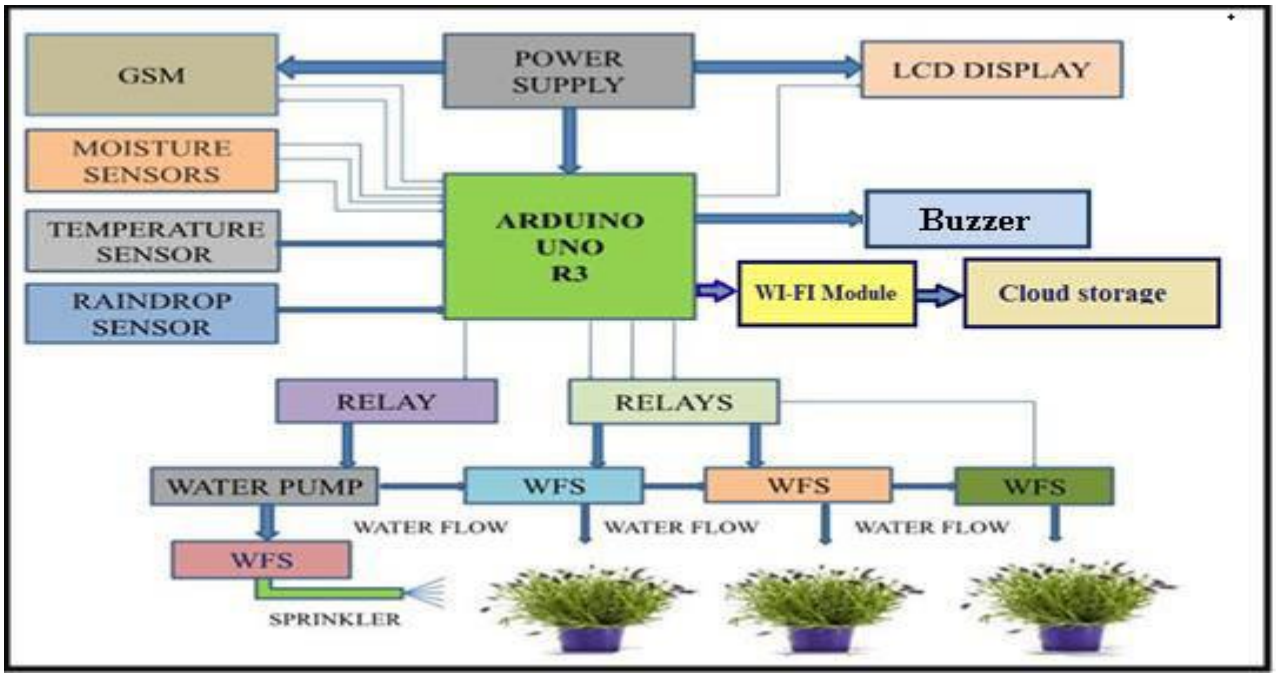
A data flow diagram (DFD) is a graphical or visual representation using a standardized set of symbols and notations to describe a business's operations through data movement.



5.2 SOLUTION & TECHNICAL ARCHITECTURE

Solution Architecture :

Solution architecture is a practice to provide ground for software development projects by tailoring IT solutions to specific business needs and defining their functional requirements and stages of implementation.



Technical Architecture:

Technical Architecture (TA) is a form of IT architecture that is used to design computer systems. It involves the development of a technical blueprint with regard to the arrangement, interaction, and interdependence of all elements so that system-relevant requirements are met.

Guidelines:

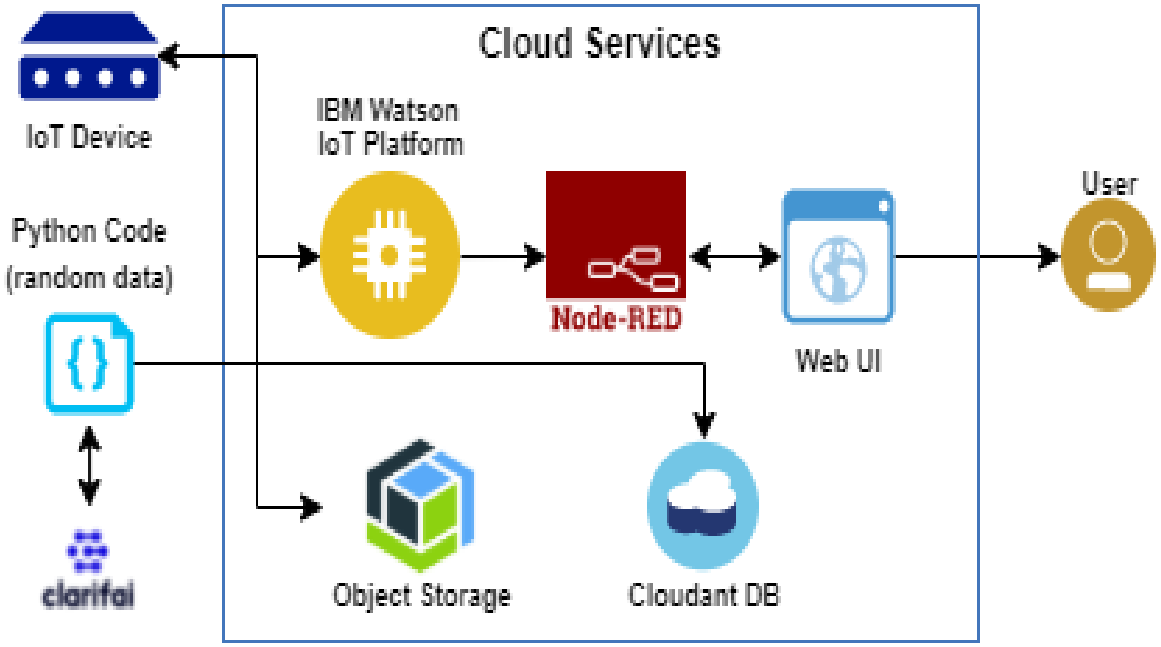
1.To protect the crop from the animals and birds which destroy the crop. This system also helps farmers to monitor the soil moisture levels in the field and also the temperature and humidity values near the field. The motors and sprinklers in the field can be controlled using the mobile application.

2.The temperature and humidity in the field are tracked from a web app using weather API and update automatically.

3.Based on weather change the motors and sprinklers in the field can be controlled using the mobile application.

4.The sensors detect the temperature and humidity values near the field.

5.Attack of birds and animals can be detected by sensors through mobile applications.



5.3 USER STORIES

A user story is an informal, general explanation of a software feature written from the perspective of the end user or customer. The purpose of a user story is to articulate how a piece of work will deliver a particular value back to the customer.

User Type	**Functional Requirement (Epic)**	**User Story Number**	**User Story/Task**	**Acceptance Criteria**	Priority	**Release**
Customer	Registration	USN-1	Login to the application	I can access dashboard	High	Sprint
		USN-2	As a user, I will receive confirmation email once I have registered	I can receive confirmation email & click confirm	High	Sprint
	Login	USN-3	Through the sensor monitor the crop	I can access sensor	High	Sprint
		USN-4	As a user I can Protect crop	Reduce wastage of food	Medium	Sprint

		USN-5	I can get new methods to protect crop	I can handle the situation	High	Sprint
Customer(web user)		USN-6	Use of sensor	Monitor the effects	High	Sprint
Administrator		USN-7	Future updation & monitoring	I can monitor using new technologies	Medium	Sprint

CHAPTER 6

PROJECT PLANNING & SCHEDULING

6.1 SPRINT PLANNING & ESTIMATION

The purpose of sprint planning is to define what can be delivered in the sprint and how that work will be achieved.

TITLE	DESCRIPTION	DATE
Literature survey and Information Gathering	Literature survey on the selected project & gathering information by referring to the technical papers, research publications etc..	24 SEPTEMBER 2022
Prepare Empathy Map	Prepare empathy map canvas to capture the user pains & gains, Prepare list of problem statements.	23 SEPTEMBER 2022
Ideation	List the ideas by organizing the brainstorming session and prioritize the top 3 ideas based on feasibility & importance.	8 OCTOBER 2022
Proposed Solution	Prepare the proposed solution document which includes the novelty, feasibility of idea, business model, social impact, scalability of solution etc..	8 OCTOBER 2022
Problem Solution Fit	Prepare problem - solution fit document.	9 OCTOBER 2022

Solution Architecture	Prepare a solution architecture document.	11 OCTOBER 2022
Functional Requirements	Prepare the functional requirement document.	19 OCTOBER 2022
Data Flow Diagrams	Draw data flow diagram for the project and submit for review.	20 OCTOBER 2022
Customer Journey	Prepare the customer journey maps to understand the user interactions & experiences with the application (entry to exit).	22 OCTOBER 2022
Technology Architecture	Prepare the technology architecture diagram.	19 OCTOBER 2022
Prepare Milestone and Activity List	Prepare the milestone & activity list of the project.	31 OCTOBER 2022
Project development - Delivery of Sprint - 1,2,3 & 4	Develop & submit the developed code by testing it.	19 NOVEMBER 2022

6.2 SPRINT DELIVERY SCHEDULE

A sprint schedule is a document that outlines sprint planning from end to end.

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1		US-1	Create the IBM Cloud services which are being used in this project.	6	High	Abinaya JS, Ajina PA, Ishwarya MT, Sophiya Y
Sprint-1		US-2	Configure the IBM Cloud services which are being used in completing this project.	4	Medium	Abinaya JS, Ajina PA, Ishwarya MT, Sophiya Y
Sprint-2		US-3	IBM Watson IoT platform acts as the mediator to connect the web application to IoT devices, so create the	5	Medium	Abinaya JS, Ajina PA,

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
			IBM Watson IoT platform.			Ishwarya MT, Sophiya Y
Sprint-2		US-4	In order to connect the IoT device to the IBM cloud, create a device in the IBM Watson IoT platform and get the device credentials.	5	High	Abinaya JS, Ajina PA, Ishwarya MT, Sophiya Y
Sprint-3		US-1	Configure the connection security and create API keys that are used in the Node-RED service for accessing the IBM IoT Platform.	10	High	Abinaya JS, Ajina PA, Ishwarya MT, Sophiya Y
Sprint-3		US-2	Create a Node-RED service.	10	High	Abinaya JS, Ajina PA,

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
						Ishwarya MT, Sophiya Y
Sprint-3		US-1	Develop a python script to publish random sensor data such as temperature, moisture, soil and humidity to the IBM IoT platform	7	High	Abinaya JS, Ajina PA, Ishwarya MT, Sophiya Y
Sprint-3		US-2	After developing python code, commands are received just print the statements which represent the control of the devices.	5	Medium	Abinaya JS, Ajina PA, Ishwarya MT, Sophiya Y
Sprint-4		US-3	Publish Data to The IBM Cloud	8	High	Abinaya JS, Ajina PA,

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
						Ishwarya MT, Sophiya Y
Sprint-4		US-1	Create Web UI in Node- Red	10	High	Abinaya JS, Ajina PA, Ishwarya MT, Sophiya Y
Sprint-4		US-2	Configure the Node-RED flow to receive data from the IBM IoT platform and also use Cloudant DB nodes to store the received sensor data in the cloudant DB	10	High	Abinaya JS, Ajina PA, Ishwarya MT, Sophiya Y

Project Tracker, Velocity & Burndown Chart: (4 Marks)

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	20
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20

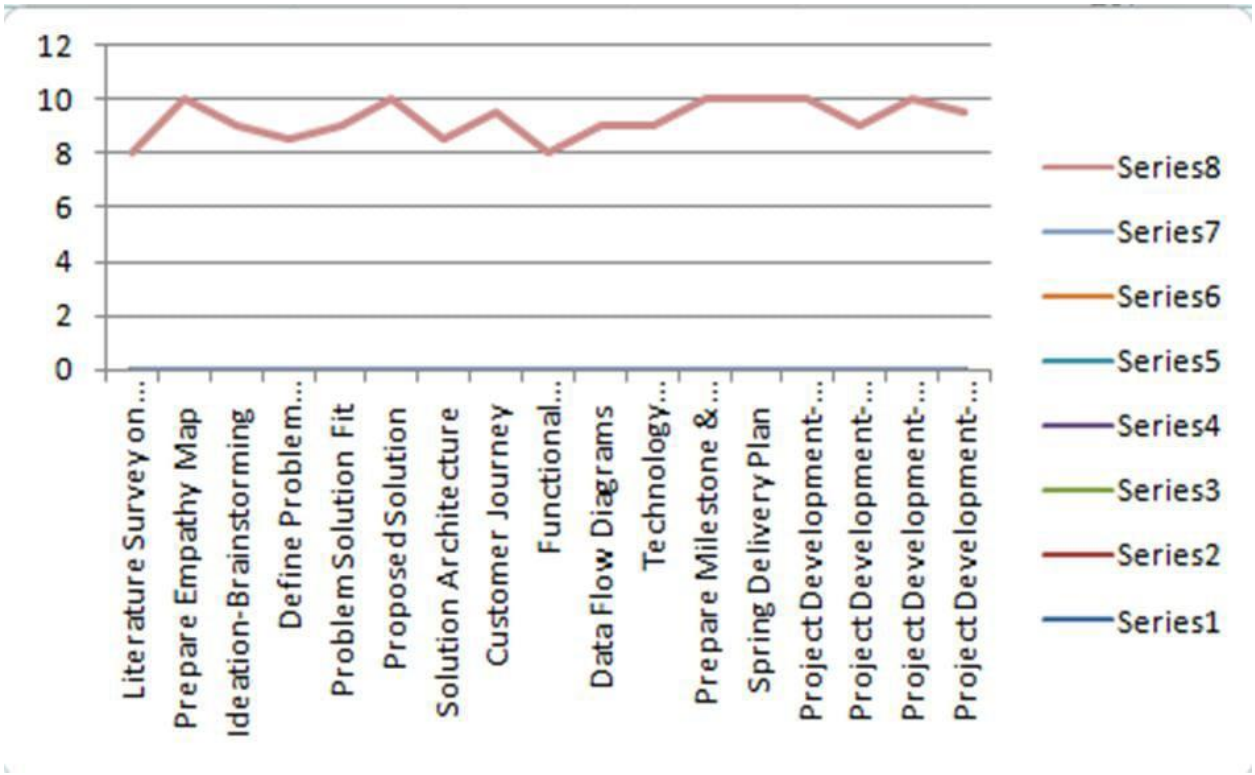
Velocity:

Imagine we have a 10-day sprint duration, and the velocity of the team is 20 (points per sprint). Let's calculate the team's average velocity (AV) per iteration unit (story points per day).

$$AV = \frac{sprint\ duration}{velocity} = \frac{20}{10} = 2$$

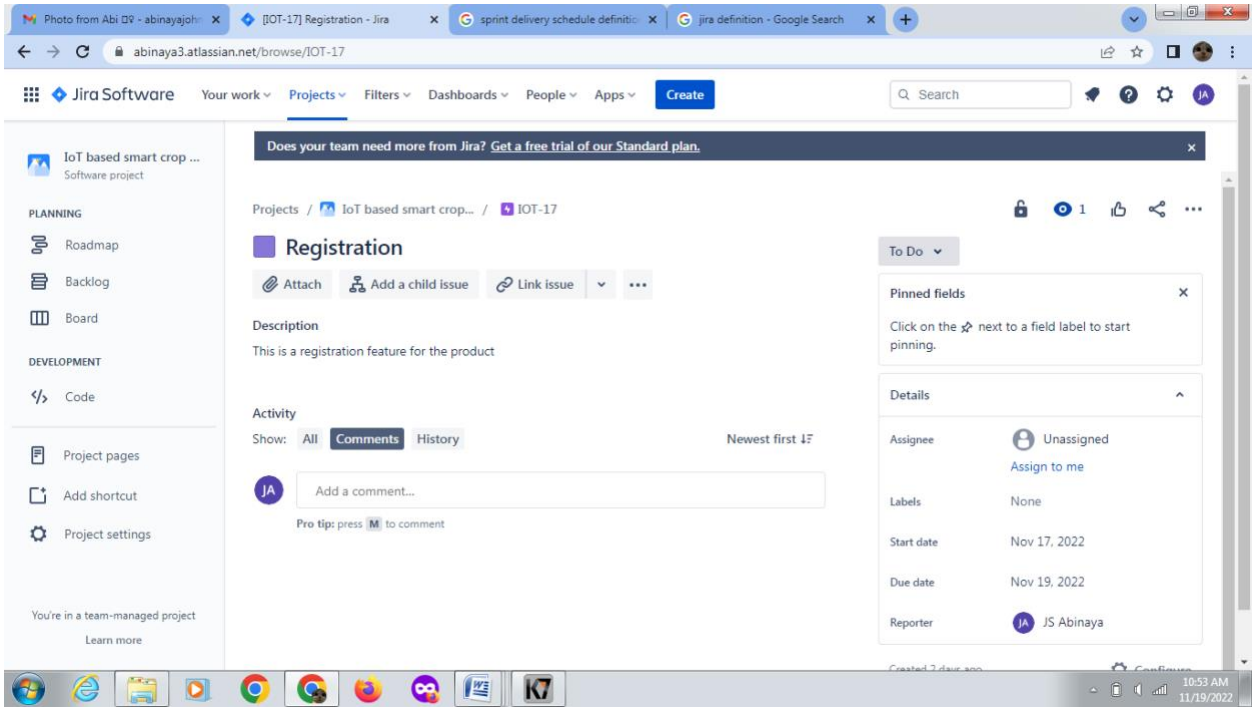
Burn down Chart:

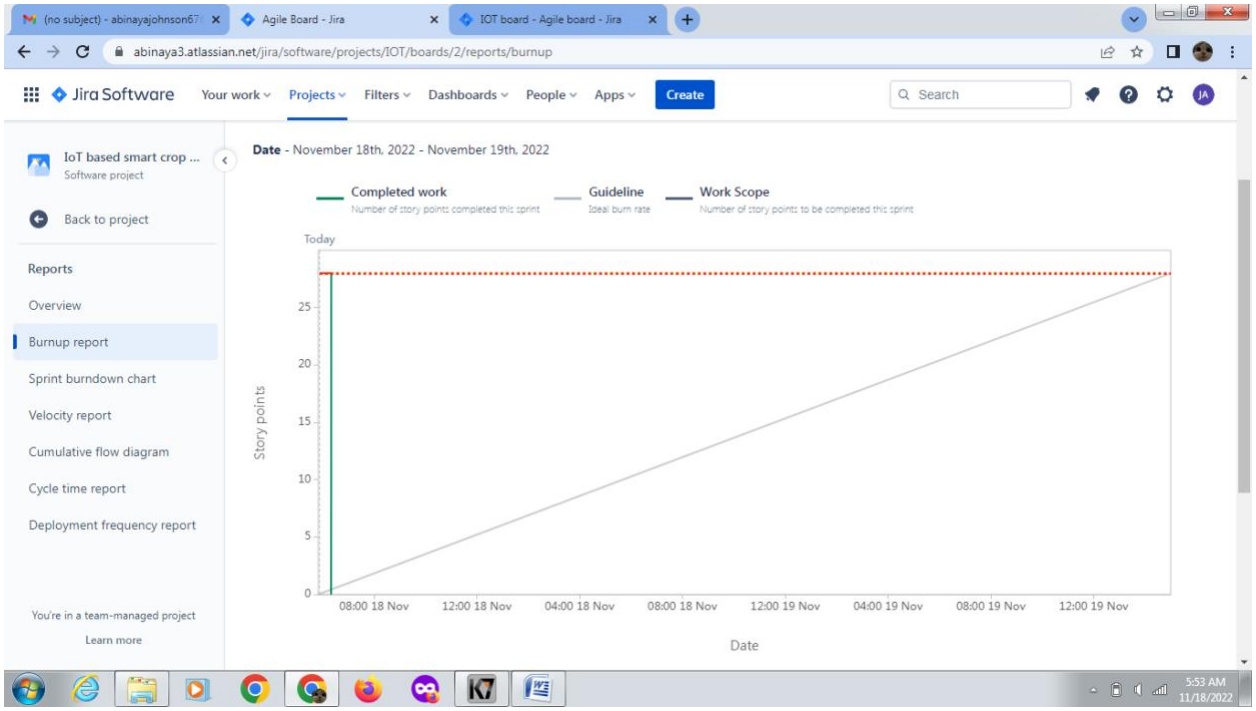
A burn down chart is a graphical representation of work left to do versus time. It is often used in agile software development methodologies such as scrum .However, burn down charts can be applied to any project containing measurable progress overtime.

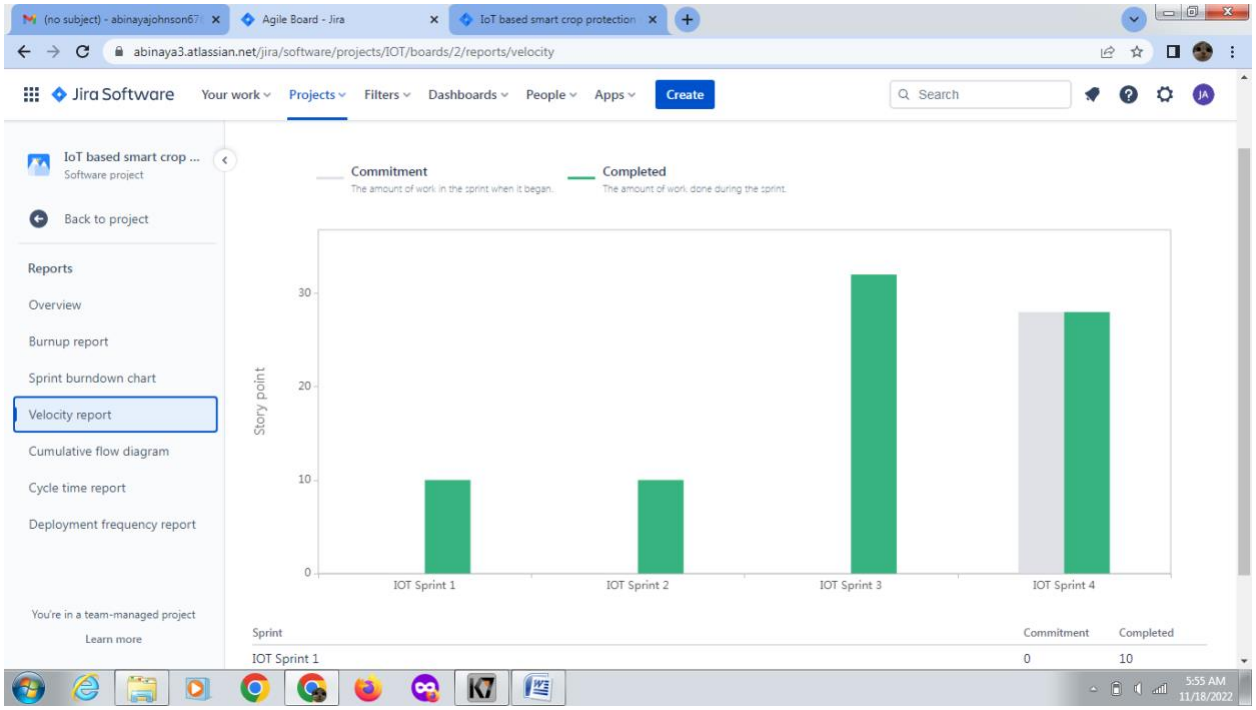

12
10
8
6
4
2
0
Literature Survey on...
Prepare Empathy Map
Ideation-Brainstorming
Define Problem...
Problem Solution Fit
Proposed Solution
Solution Architecture
Customer Journey
Functional...
Data Flow Diagrams
Technology...
Prepare Milestone &...
Spring Delivery Plan
Project Development-...
Project Development-...
Project Development-...
Project Development-...
Series8
Series7
Series6
Series5
Series4
Series3
Series2
Series1

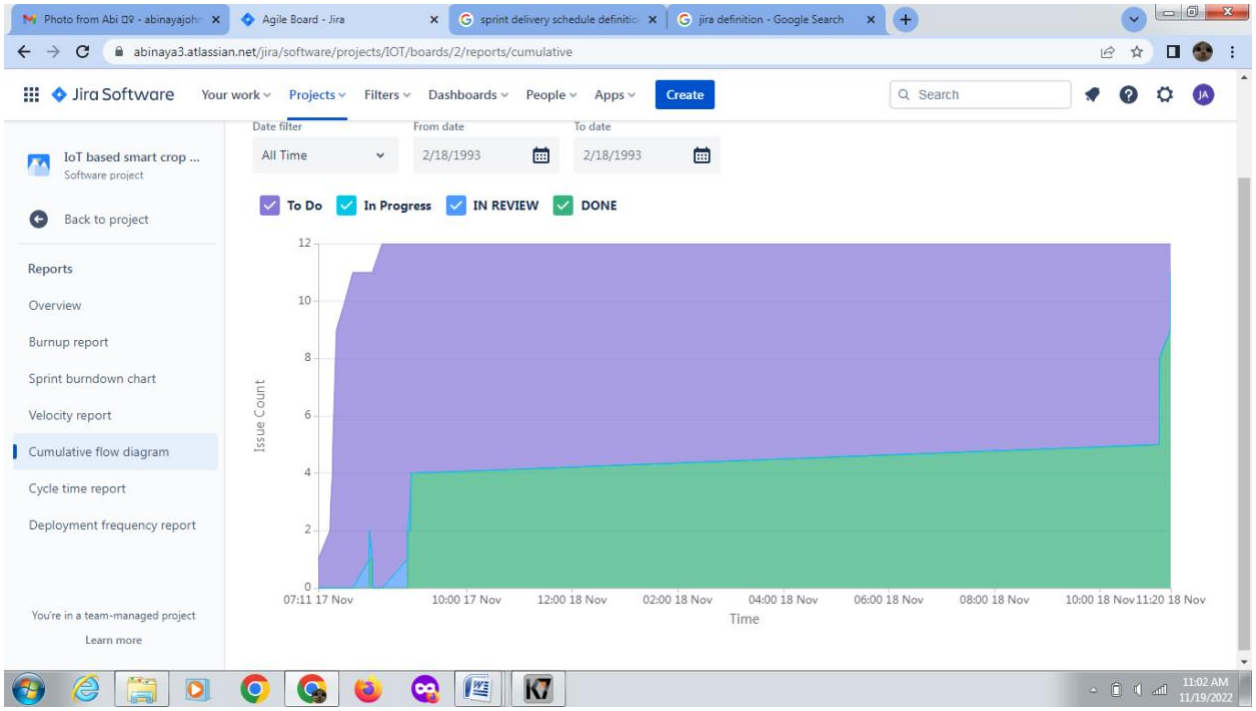
6.3 REPORTS FROM JIRA

Jira is a software application used for issue tracking and project management.




(no subject) - abinayajohnson67
Agile Board - Jira
IOT board - Agile board - Jira
abinaya3.atlassian.net/jira/software/projects/IOT/boards/2/reports/burnup
Jira Software
Your work
Projects
Filters
Dashboards
People
Apps
Create
Search
IoT based smart crop ...
Software project
Back to project
Reports
Overview
Burnup report
Sprint burndown chart
Velocity report
Cumulative flow diagram
Cycle time report
Deployment frequency report
You're in a team-managed project
Learn more
Date - November 18th, 2022 - November 19th, 2022
Completed work
Number of story points completed this sprint
Guideline
Ideal burn rate
Work Scope
Number of story points to be completed this sprint
Today
Story points
0
5
10
15
20
25
08:00 18 Nov
12:00 18 Nov
04:00 18 Nov
08:00 18 Nov
12:00 19 Nov
04:00 19 Nov
08:00 19 Nov
12:00 19 Nov
Date
5:53 AM
11/18/2022


(no subject) - abinayajohnson67
Agile Board - Jira
IoT based smart crop protection
abinaya3.atlassian.net/jira/software/projects/IOT/boards/2/reports/velocity
Jira Software
Your work
Projects
Filters
Dashboards
People
Apps
Create
Search
IoT based smart crop ...
Software project
Back to project
Reports
Overview
Burnup report
Sprint burndown chart
Velocity report
Cumulative flow diagram
Cycle time report
Deployment frequency report
You're in a team-managed project
Learn more
Commitment
The amount of work in the sprint when it began.
Completed
The amount of work done during the sprint.
Story point
0
10
20
30
IOT Sprint 1
IOT Sprint 2
IOT Sprint 3
IOT Sprint 4
Sprint
Commitment
Completed
IOT Sprint 1
0
10
5:55 AM
11/18/2022



CHAPTER 7

CODING & SOLUTIONING

7.1 FEATURE 1

```
import random
import ibmiotf.application
import ibmiotf.device
from time import sleep
import sys
#IBM Watson Device Credentials.
```

```
organization = "kd5lkd"
deviceType = "ibm"
deviceId = "12345678"
authMethod = "use-token-auth"
authToken = "87654321"
def myCommandCallback(cmd):
print("Command received: %s" % cmd.data['command'])
status=cmd.data['command']
if status=="sprinkler_on":
print ("sprinkler is ON")
else :
print ("sprinkler is OFF")
#print(cmd)
try:
deviceOptions = {"org": organization, "type": deviceType, "id": deviceId, "auth-
method": authMethod, "auth-token": authToken}
deviceCli = ibmiotf.device.Client(deviceOptions)
except Exception as e: print("Caught exception connecting device: %s" % str(e))
sys.exit()
#Connecting to IBM watson.
deviceCli.connect()
while True:
#Getting values from sensors
. temp_sensor = round( random.uniform(0,80),2)
PH_sensor = round(random.uniform(1,14),3)
camera = ["Detected","Not Detected","Not Detected","Not Detected","Not
Detected","Not  Detected",]
```

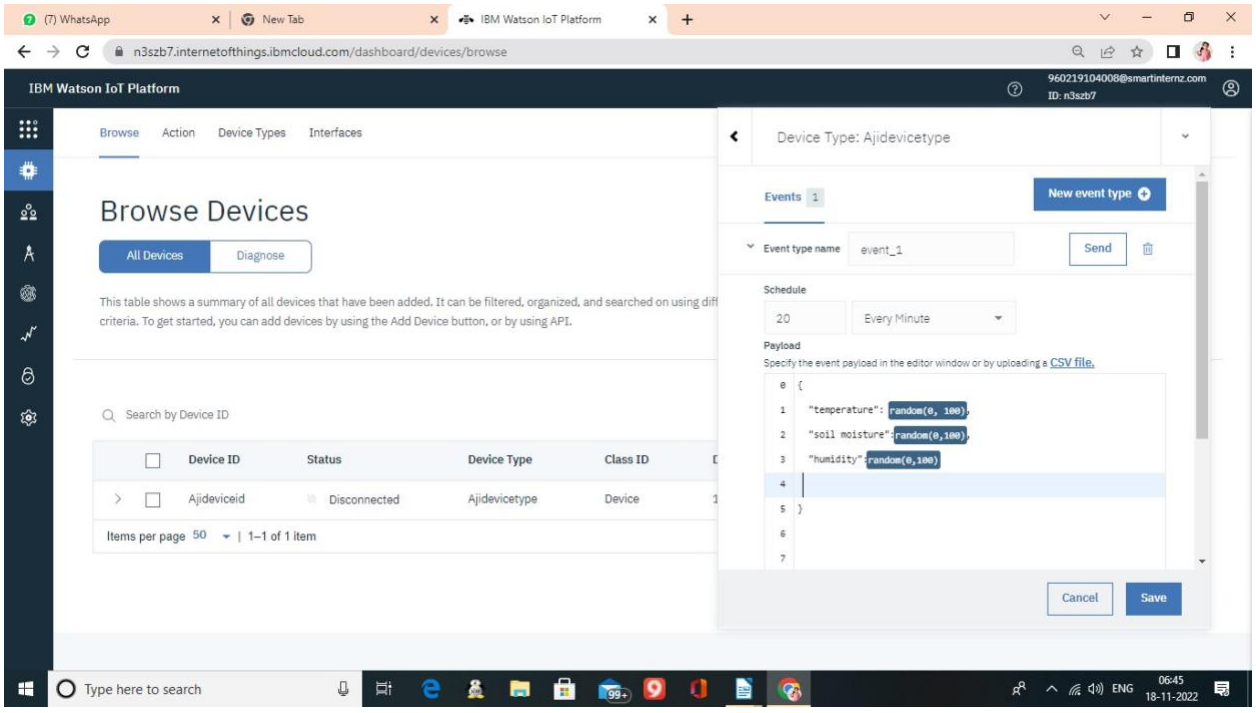
```
camera_reading = random.choice(camera)
flame = ["Detected","Not Detected","Not Detected","Not Detected","Not
Detected","Not  Detected",]
flame_reading = random.choice(flame)
moist_level = round(random.uniform(0,100),2)
water_level = round(random.uniform(0,30),2)
#storing the sensor data to send in json format to cloud.
temp_data = { 'Temperature' : temp_sensor }
PH_data = { 'PH Level' : PH_sensor }
camera_data = { 'Animal attack' : camera_reading}
flame_data = { 'Flame' : flame_reading }
moist_data = { 'Moisture Level' : moist_level}
water_data = { 'Water Level' : water_level}
# publishing Sensor data to IBM Watson for every 5-10 seconds.
success = deviceCli.publishEvent("Temperature sensor", "json", temp_data, qos=0)
sleep(1)
if success:
print (" .....publish ok..... ")
print ("Published Temperature = %s C" % temp_sensor, "to IBM Watson")
success = deviceCli.publishEvent("PH sensor", "json", PH_data, qos=0) sleep(1)
if success:
print ("Published PH Level = %s" % PH_sensor, "to IBM Watson")
success = deviceCli.publishEvent("camera", "json", camera_data, qos=0)
sleep(1)
if success:
print ("Published Animal attack %s " % camera_reading, "to IBM Watson")
success = deviceCli.publishEvent("Flame sensor", "json", flame_data, qos=0)
```

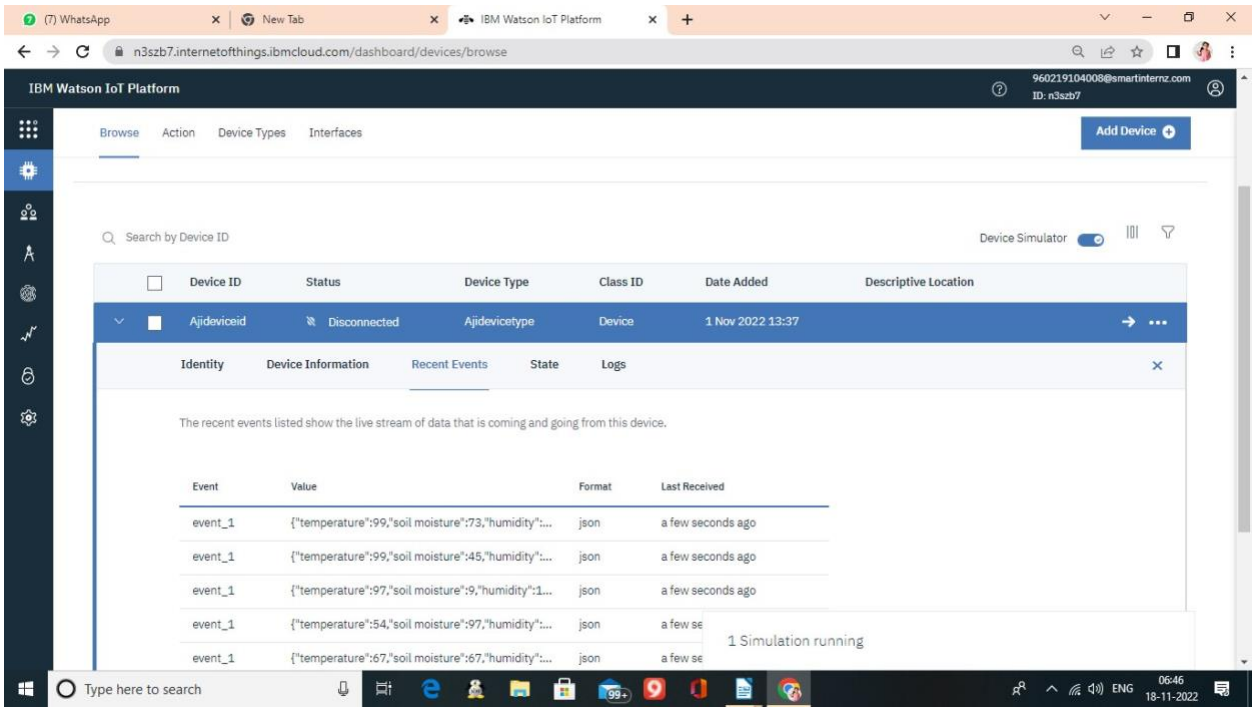
```
sleep(1)
if success:
print ("Published Flame %s " % flame_reading, "to IBM Watson")
success = deviceCli.publishEvent("Moisture sensor", "json", moist_data, qos=0)
sleep(1)
if success:
print ("Published Moisture Level = %s " % moist_level, "to IBM Watson")
success = deviceCli.publishEvent("Water sensor", "json", water_data, qos=0)
sleep(1) i
f success:
print ("Published Water Level = %s cm" % water_level, "to IBM Watson")
print ("")
#Automation to control sprinklers by present temperature an to send alert message
to IBM Watson.
if (temp_sensor > 35):
print("sprinkler-1 is ON")
success = deviceCli.publishEvent("Alert1", "json",{ 'alert1' : "Temperature(%s) is
high, sprinklers are turned ON" %temp_sensor } , qos=0)
sleep(1)
if success:print( 'Published alert1 : ', "Temperature(%s) is high, sprinklers are
turned ON" %temp_sensor,"to IBM Watson")print("")
else:
print("sprinkler-1 is OFF")
print("")
#To send alert message if farmer uses the unsafe fertilizer to crops
. if (PH_sensor > 7.5 or PH_sensor < 5.5):
```

```
success = deviceCli.publishEvent("Alert2", "json",{ 'alert2' : "Fertilizer PH
level(%s) is not safe,use other fertilizer" %PH_sensor } , qos=0)
sleep(1)
if success:
print('Published alert2 : ' , "Fertilizer PH level(%s) is not safe,use other fertilizer"
%PH_sensor,"to IBM Watson")
print("")
#To send alert message to farmer that animal attack on crops.
if (camera_reading == "Detected"):
success = deviceCli.publishEvent("Alert3", "json", { 'alert3' : "Animal attack on
crops detected" },
qos=0)
sleep(1)
if success:
print('Published alert3 : ' , "Animal attack on crops detected","to IBM Watson","to
IBM Watson")
print("")
#To send alert message if flame detected on crop land and turn ON the splinkers to
take immediate action.
if (flame_reading == "Detected"):
print("sprinkler-2 is ON")
success = deviceCli.publishEvent("Alert4", "json", { 'alert4' : "Flame is detected
crops are in danger,sprinklers turned ON" }, qos=0)
sleep(1)
if success: print( 'Published alert4 : ' , "Flame is detected crops are in
danger,sprinklers turned ON","to IBM Watson")
```

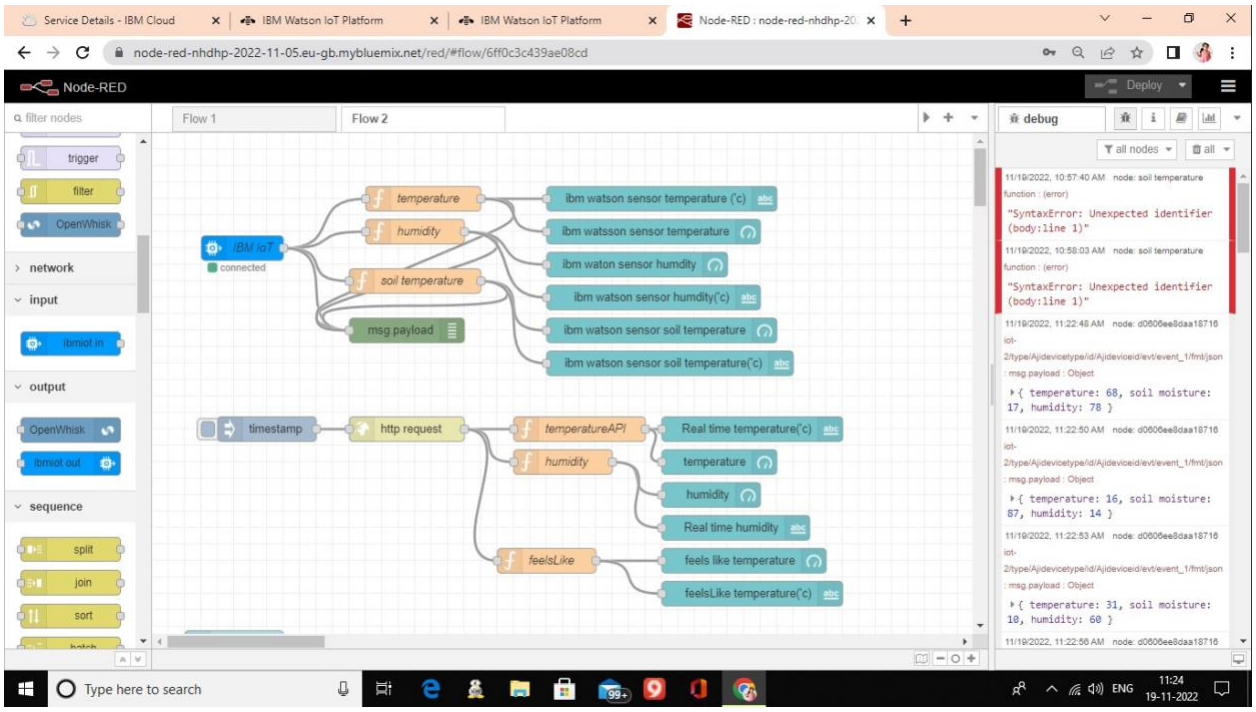
```
#To send alert message if Moisture level is LOW and to Turn ON Motor-1 for irrigation.
if (moist_level < 20):
print("Motor-1 is ON")
success = deviceCli.publishEvent("Alert5", "json", { 'alert5' : "Moisture level(%s) is low, Irrigation started" %moist_level }, qos=0)
sleep(1)
if success:
print('Published alert5 : ' , "Moisture level(%s) is low, Irrigation started" %moist_level,"to IBM Watson"
)
print("")
#To send alert message if Water level is HIGH and to Turn ON Motor-2 to take water out.`
if (water_level > 20):
print("Motor-2 is ON")
success = deviceCli.publishEvent("Alert6", "json", { 'alert6' : "Water level(%s) is high, so motor is ON to take water out " %water_level }, qos=0)
sleep(1)
if success:
print('Published alert6 : ' , "water level(%s) is high, so motor is ON to take water out "
%water_level,"to IBM Watson" )
print("")
#command received by farmer
deviceCli.commandCallback = myCommandCallback
# Disconnect the device and application from the cloud
```

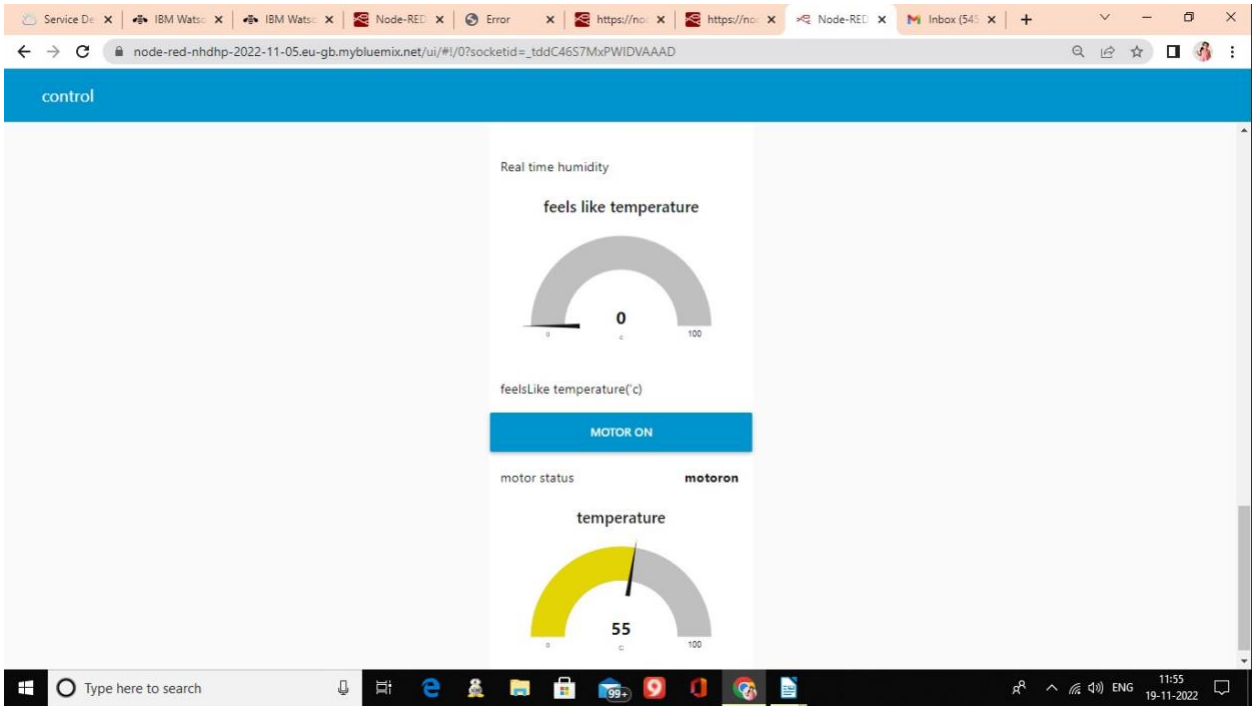
deviceCli.disconnect()

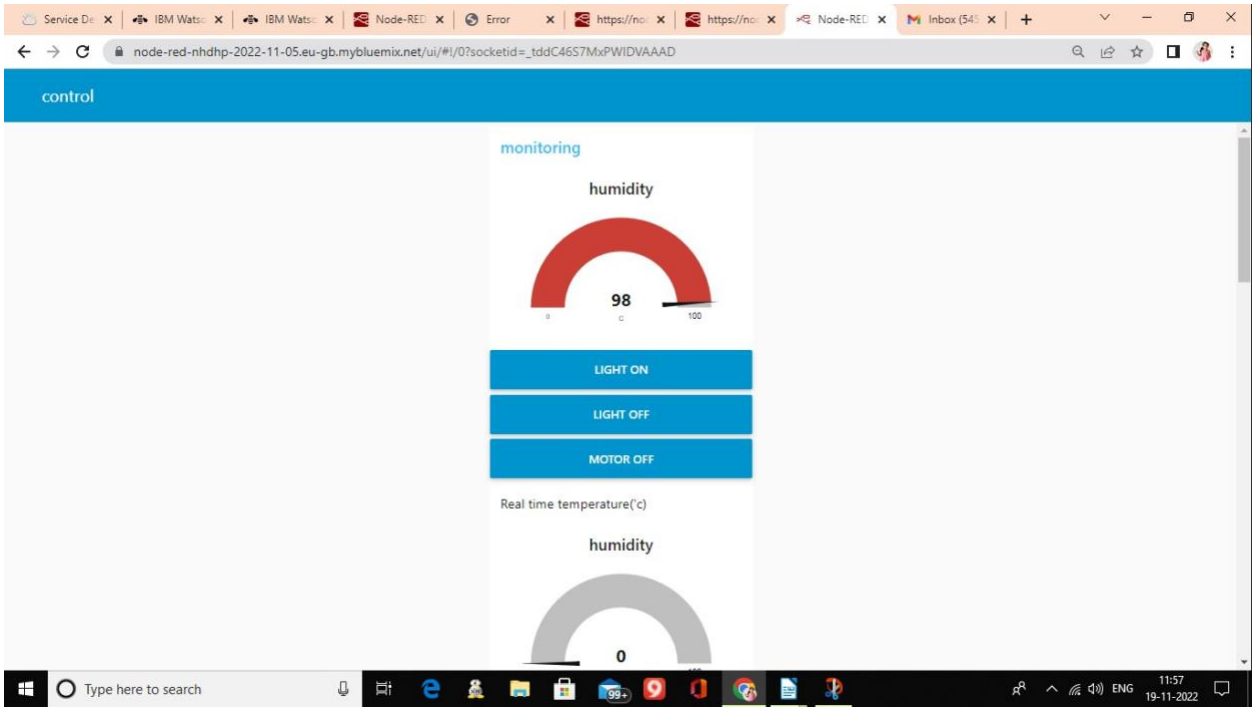


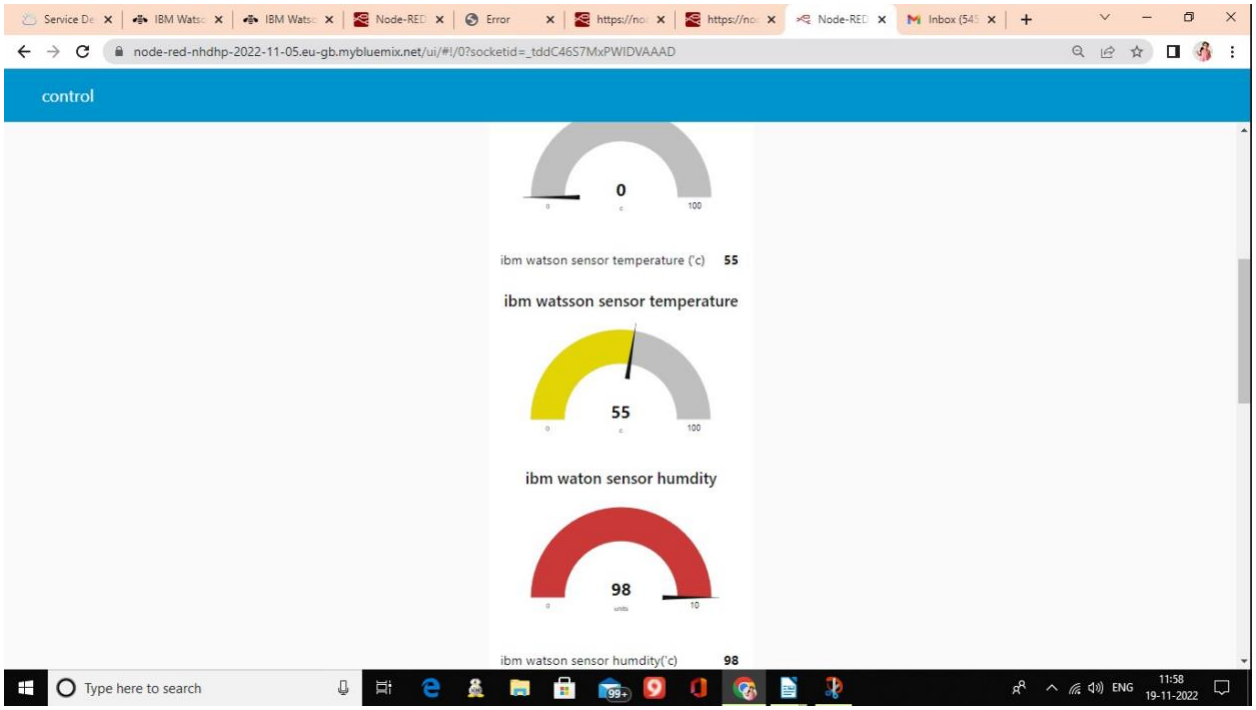


FEATURES :

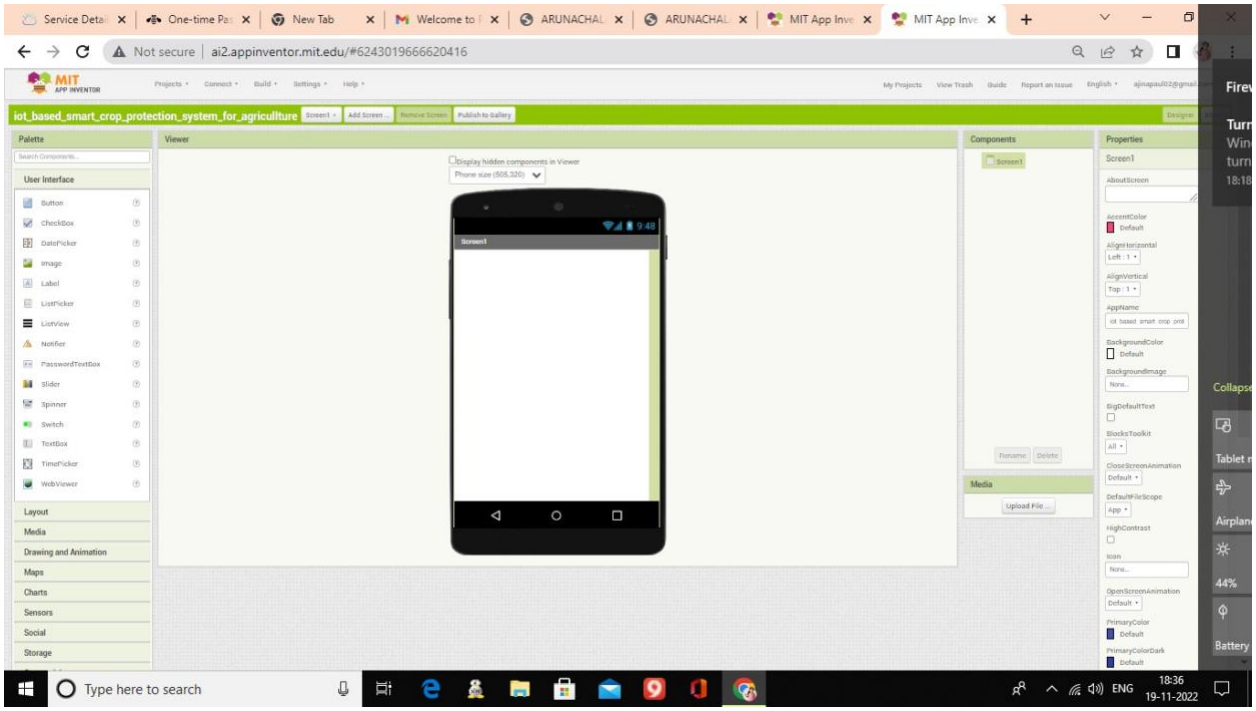

Node-RED
Flow 1
Flow 2
debug
temperature
humidity
soil temperature
msg payload
IBM IoT
connected
ibm watson sensor temperature (°c)
ibm watsson sensor temperature
ibm waton sensor humdity
ibm watson sensor humdity(°c)
ibm watson sensor soil temperature
ibm watson sensor soil temperature(°c)
timestamp
http request
temperatureAPI
humidity
feelsLike
Real time temperature(°c)
temperature
humidity
Real time humidity
feels like temperature
feelsLike temperature(°c)
Type here to search
ENG
11:24
19-11-2022

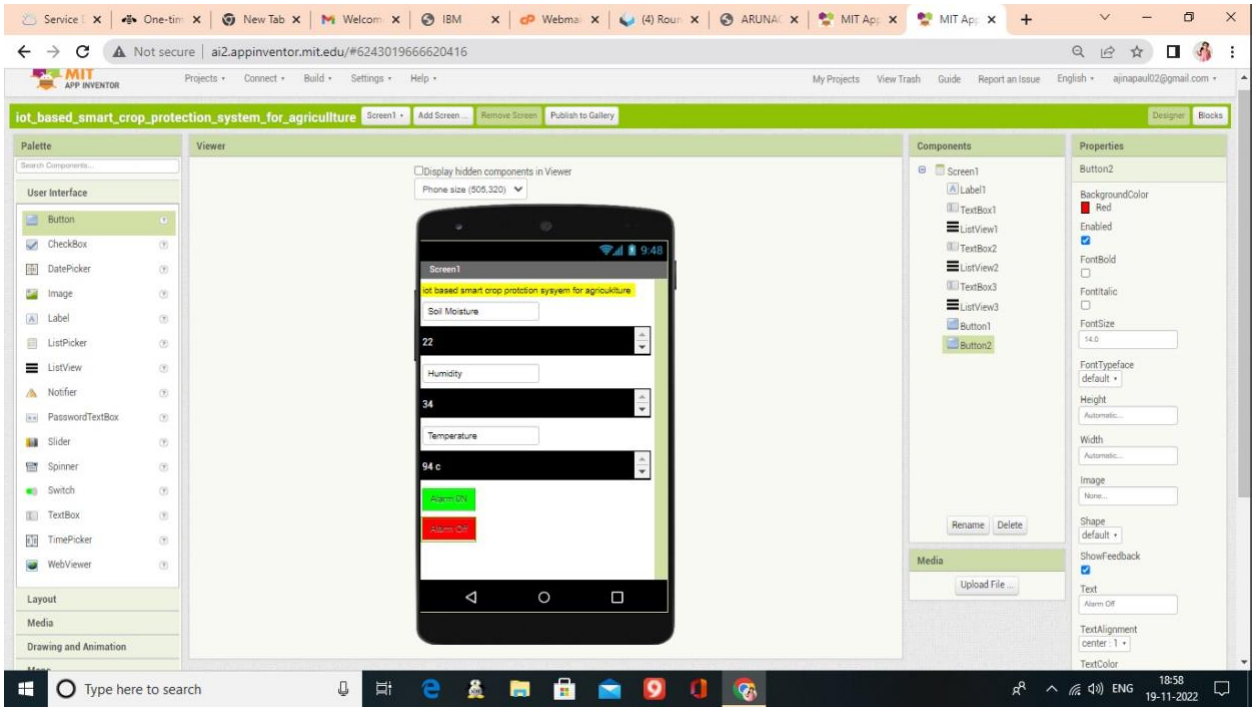

control
Real time humidity
feels like temperature
0
feelsLike temperature(°c)
MOTOR ON
motor status
motoron
temperature
55
Type here to search
ENG
11:55
19-11-2022


node-red-nhdhp-2022-11-05.eu-gb.mybluemix.net/ui/#!/0?socketid=_tddC46S7MxPWIDVAAAD
control
monitoring
humidity
98
LIGHT ON
LIGHT OFF
MOTOR OFF
Real time temperature('c)
humidity
0
Type here to search
ENG
11:57
19-11-2022


node-red-nhdhp-2022-11-05.eu-gb.mybluemix.net/ui/#!/0?socketid=_tddC46S7MxPWIDVAAAD
control
0
ibm watson sensor temperature ('c) 55
ibm watsson sensor temperature
55
ibm waton sensor humdity
98
ibm watson sensor humdity('c) 98
Type here to search
ENG
11:58
19-11-2022

7.2 FEATURE 2





CHAPTER 8

TESTING

8.1 TEST CASES

Test c	Test Scenario	Test Data	Status	Comments
TC_OO1	Create the IBM Cloud services which are being used in this project.	https://cloud.ibm.com/login	Pass	Results verified
TC_OO2	Configure the IBM Cloud services which are being used in completing this project.	https://cloud.ibm.com/login	Pass	Results verified
TC_OO3	IBM Watson IoT platform acts as the mediator to connect the web application to IoT devices, so create the IBM Watson IoT platform.	https://uytbf0.internetofthings.ibmcloud.com/dashboard/devices/browse	Pass	Results verified
TC_OO4	In order to connect the IoT device to the IBM cloud, create a device in the IBM Watson IoT platform and get the device credentials.	Temperature, Humidity , Soil moisture sensor values are generated randomly in simulation	Pass	Results verified
TC_OO5	Configure the connection security and create API keys that are used in the Node-RED service for accessing the IBM IoT Platform.	https://cloud.ibm.com/developer/appservice/create-app?starterKit=59c9d5bd-4d31-3611-897a-f94eea80dc9f&defaultLanguage=undefined	Pass	Results verified
TC_	Create a Node-RED service.	Values of sensors and button for light ON/OFF is displayed	Pass	Results verified
TC_OO7	Develop a python script to publish random sensor data such as temperature, humidity level, soil moisture to the IBM IoT platform	https://www.python.org/downloads/release/python-370/	Pass	Results verified

TC_OO8	After developing python code, commands are received just print the statements which represent the control of the devices.	Get the output from the code	Pass	Results verified
TC_OO9	Publish Data to The IBM Cloud	Publishment of python code	Pass	Results verified
TC_O10	Create Web UI in Node-Red	Sensors values and command values can be seen in the mobile application	Pass	Results verified
TC_O11	Configure the Node-RED flow to receive data from the IBM IoT platform and also use Cloudant DB nodes to store the received sensor data in the cloudant DB	Cloudant is connected by NODE RED	Pass	Results verified

8.2 USER ACCEPTANCE TESTING

1. Purpose of Document

The purpose of this document is to briefly explain the test coverage and open

issues of the IoT based smart crop protection system for agriculture project at the time of the release to User Acceptance Testing (UAT).

2.Defect Analysis

This report shows the number of resolved or closed bugs at each severity level, and how they were resolved.

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	7	2	2	3	14
Duplicate	1	0	3	0	4
External	2	4	0	1	7
Fixed	15	2	5	18	40
Not Reproduced	0	0	1	0	1
Skipped	0	0	1	3	4
Won't Fix	0	7	2	1	10
Totals	25	15	14	26	80

3. Test Case Analysis

This report shows the number of test cases that have passed, failed, and untested.

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	7	0	0	7
Client Application	31	0	0	31
Security	16	0	0	16
Outsource Shipping	9	0	0	9
Exception Reporting	10	0	0	10
Final Report Output	7	0	0	7
Version Control	5	0	0	5

CHAPTER 9

RESULTS

9.1 PERFORMANCE METRICS

1. Requirement Identification
    - Functional Requirements
    - Non-Functional Requirements
2. Implementation result
    - System Implementation results
    - Results of web application Implementation
3. Resource utilization results
    - Foreground activities results
    - Memory usage
    - Energy usage
4. Background activities results

CHAPTER 10

ADVANTAGES & DISADVANTAGES

ADVANTAGES :

- **Efficient resource utilization:**

If we know the functionality and the way that how each device work we definitely increase the efficient resource utilization as well as monitor natural resources.

- **Minimize human effort:**

As the devices of IoT interact and communicate with each other and do lot of task for us, then they minimize the human effort.

- **Save time:**

As it reduces the human effort then it definitely saves out time. Time is the primary factor which can save through IoT platform.

- **Improve security:**

Now, if we have a system that all these things are interconnected then we can make the system more secure and efficient **.**

DISADVANTAGES :

- **Security:**

As the IoT systems are interconnected and over networks. The system offers little control despite any security measures , and it can be lead the various

kinds of network attacks.

- **Privacy:**

Even without the active participation on the user, the IoT system provides substantial personal data in maximum detail.

- **Complexity:**

The designing, developing, and maintaining and enabling the large technology to IoT system is quite complicated.

CHAPTER 11

CONCLUSION

With the incorporation of the IOT, we can upgrade the agriculture farm. These systems enable to check the quality of the soil and the growth of the crop in soil and with these system farmers are able to solve irrigation problems, temperature problems, humidity problems, etc. The availability of sensors for the agricultural parameters and microcontrollers can be easily interfaced with each other and with the help of Internet of Things, wireless sensor networks communication the challenges encountered by the farmers can also be reduced and a better communication path for the transfer of useful data can be achieved between various nodes. So, farmers are able to control various equipment's related to agricultural and monitor their crop on Smartphone or on computers. These systems offer a high application area to the users to improve their skill and output of the crops in better way. Use these systems help to increase the Rice, wheat and maize and other agricultural production in India in the near future. IOT capable to control the condition of the yield and growth, it can also able to check soil, temperature, humidity, etc. with help of IoT.

CHAPTER 12
FUTURE SCOPE

Later on, there will be exceptionally huge extension, this venture can be made dependent on Image processing in which wild animal and fire can be detected by cameras and in the event that it comes towards field, framework will be Straight forwardly initiated through remote organizations. Wild creatures can likewise be recognized by utilizing remote organizations like laser remote sensors and by detecting this, laser or sensor's security framework will be actuated. In such a case the sensor sends signals to the microcontroller to make a move. The micro controller now sounds an alert to shoo away the creatures from the field just as well as sends an SMS to the owner warning about the situation in the field.

CHAPTER 13

APPENDIX

SOURCE CODE:

```
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random
#Provide your IBM Watson Device Credentials
organization = "bxobbs"
deviceType = "b5ibm"
deviceId = "b5device"
authMethod = "token"
authToken = "b55m1eibm"


# Initialize GPIO
def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['command'])
    status=cmd.data['command']
    if status=="lighton":
```

```
        print ("led is on")
    else :
        print ("led is off")
    #print(cmd)
    try:
        deviceOptions = {"org": organization, "type": deviceType, "id": deviceId,
"auth-method": authMethod, "auth-token": authToken}
        deviceCli = ibmiotf.device.Client(deviceOptions)
        #.....................................


except Exception as e:
        print("Caught exception connecting device: %s" % str(e))
        sys.exit()
# Connect and send a datapoint "hello" with value "world" into the cloud as an
event of type "greeting" 10 times
deviceCli.connect()
while True:
        #Get Sensor Data from DHT11
        temp=random.randint(0,100)
        Humid=random.randint(0,100)
        data = { 'temp' : temp, 'Humid': Humid }
        #print data
```

```
        def myOnPublishCallback():
            print ("Published Temperature = %s C" % temp, "Humidity = %s %%"
  % Humid, "to IBM Watson")
  success = deviceCli.publishEvent("IoTSensor", "json", data, qos=0,
on_publish=myOnPublishCallback)
        if not success:
            print("Not connected to IoTF")
        time.sleep(1)
         deviceCli.commandCallback = myCommandCallback
# Disconnect the device and application from the cloud
  deviceCli.disconnect()
```

LINKS :

Source code link :

https://github.com/IBM-EPBL/IBM-Project-40769-1660634508/blob/main/Final%20deliverables/Source%20code.pdf

Node-Red credentials :

https://github.com/IBM-EPBL/IBM-Project-40769-1660634508/blob/main/Final%20deliverables/NODE-RED%20CREDENTIALS.pdf

Github link :

https://github.com/IBM-EPBL/IBM-Project-40769-1660634508

Demo link:

https://drive.google.com/file/d/1nRXRrmneO6NvWoJyGW66ucZ2ktPGl1A6/view?usp=drivesdk