SmartFarmer - IoT Enabled Smart Farming Application

Project Name	SmartFarmer - IoT Enabled Smart Farming Application
Team ID	PNT2022TMID51460
Team Members	V.Priya M.Navaniya Shanmugabharathi R.Sandhiya X.Sneha

1. INTRODUCTION

1.1. Project Overview

The objectives of this report is to proposed IoT based Smart Farming System which will enable farmers to have live data of soil moisture environment temperature at very low cost so that live monitoring can be done.

1.2. Purpose

IoT based Smart Farming improves the entire Agriculture system by monitoring the field in real-time. With the help of sensors and interconnectivity, the Internet of Things in Agriculture has not only saved the time of the farmers but has also reduced the extravagant use of resources such as Water and Electricity.

2. LITERATURE SURVEY

2.1. Existing Problem

- Lack of Infrastructure: Even if the farmers adopt IoT technology they won't be able to take benefit of this technology due to poor communication infrastructure. ...
- High Cost: Equipment needed to implement IoT in agriculture is expensive.

2.2. References

[1] Zuraida Muhammad, Muhammad Azri Asyraf Mohd Hafez, Nor Adni Mat ”Smart Agriculture Using Internet of Things with Raspberry Pi.” 2020.

[2] Divya J., Divya M.,Janani V.”IoT based Smart Soil Monitoring System for Agricultural Production” 2017.

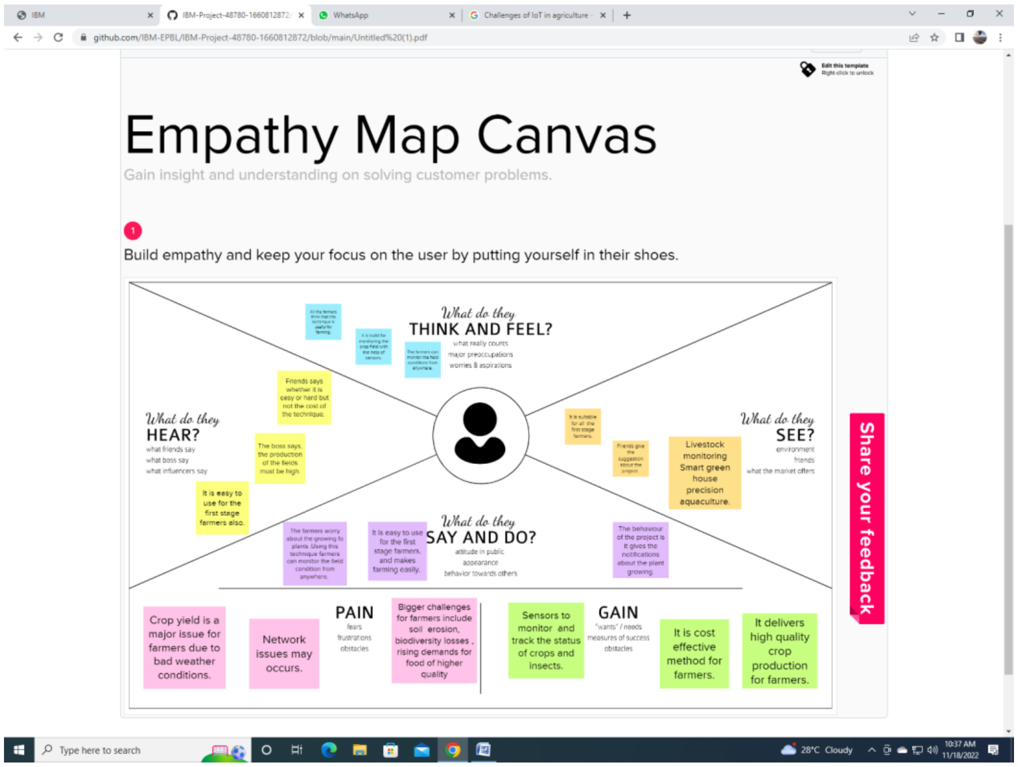
[3] H.G.C.R.Laksiri , H.A.C.Dharmagunawardhana , J.V.Wijayakulasooriya ”Design and Optimization of loT Based Smart Irrigation System in Sri Lanka”2019

2.3. Problem Statements

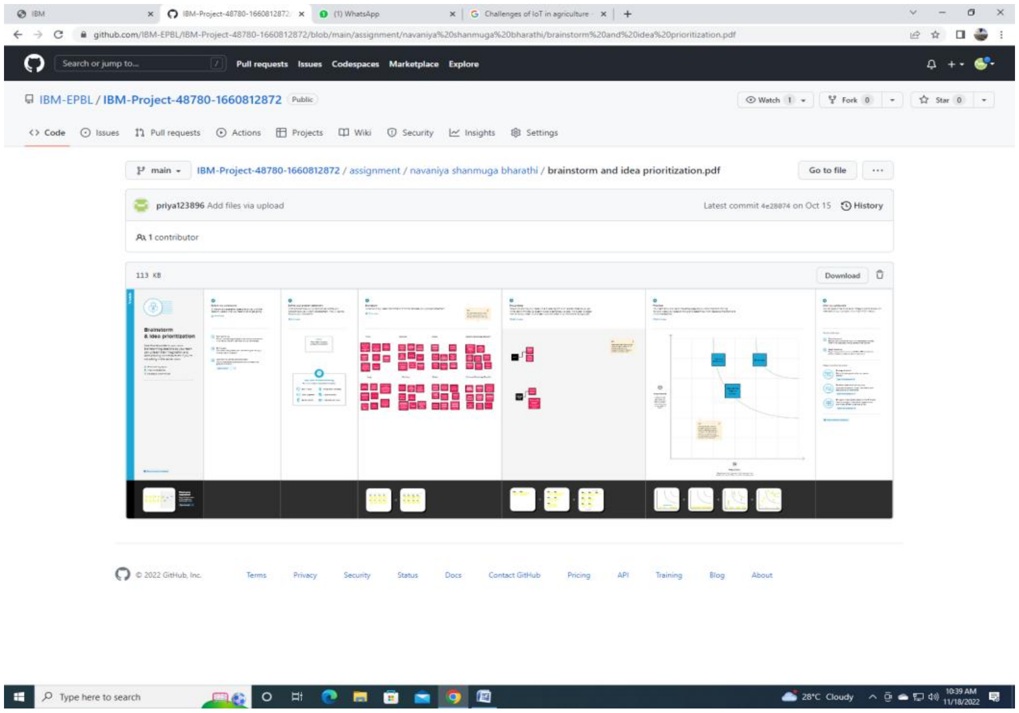
The traditional agriculture and allied sector cannot meet the requirements of modern agriculture which requires high-yield, high quality and efficient output. Thus, it is very important to turn towards modernization of existing methods and using the information technology and data over a certain period to predict the best possible productivity and crop suitable on the very particular land.

3. IDEATION AND PROPOSED SOLUTION

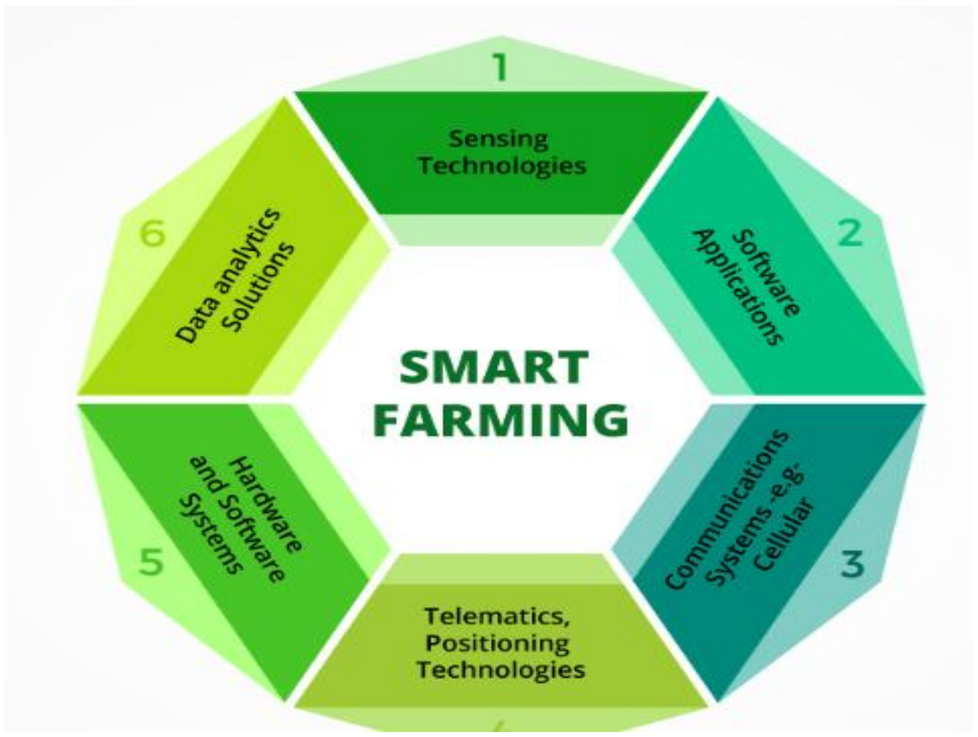
3.1. Empathy Map Canvas


Empathy Map Canvas
Gain insight and understanding on solving customer problems.
Build empathy and keep your focus on the user by putting yourself in their shoes.
What do they THINK AND FEEL?
what really counts
major preoccupations
worries & aspirations
All the farmers think that this technique is useful for farming.
It is build for monitoring the crop field with the help of sensors.
The farmers can monitor the field conditions from anywhere.
What do they HEAR?
what friends say
what boss say
what influencers say
Friends says whether it is easy or herd but not the cost of the technique.
The boss says, the production of the fields must be high.
It is easy to use for the first stage farmers also.
What do they SEE?
environment
friends
what the market offers
It is suitable for all the first stage farmers.
Friends give the suggestion about the project.
Livestock monitoring Smart green house precision aquaculture.
What do they SAY AND DO?
attitude in public
appearance
behavior towards others
The farmers worry about the growing fo plants .Using this technique farmers can monitor the field condition from anywhere.
It is easy to use for the first stage farmers, and makes farming easily.
The behaviour of the project is it gives the notifications about the plant growing.
PAIN
fears
frustrations
obstacles
Crop yield is a major issue for farmers due to bad weather conditions.
Network issues may occurs.
Bigger challenges for farmers include soil erosion, biodiversity losses , rising demands for food of higher quality
GAIN
"wants" / needs
measures of success
obstacles
Sensors to monitor and track the status of crops and insects.
It is cost effective method for farmers.
It delivers high quality crop production for farmers.
Share your feedback
github.com/IBM-EPBL/IBM-Project-48780-1660812872/blob/main/Untitled%20(1).pdf
Type here to search
28°C Cloudy
10:37 AM
11/18/2022

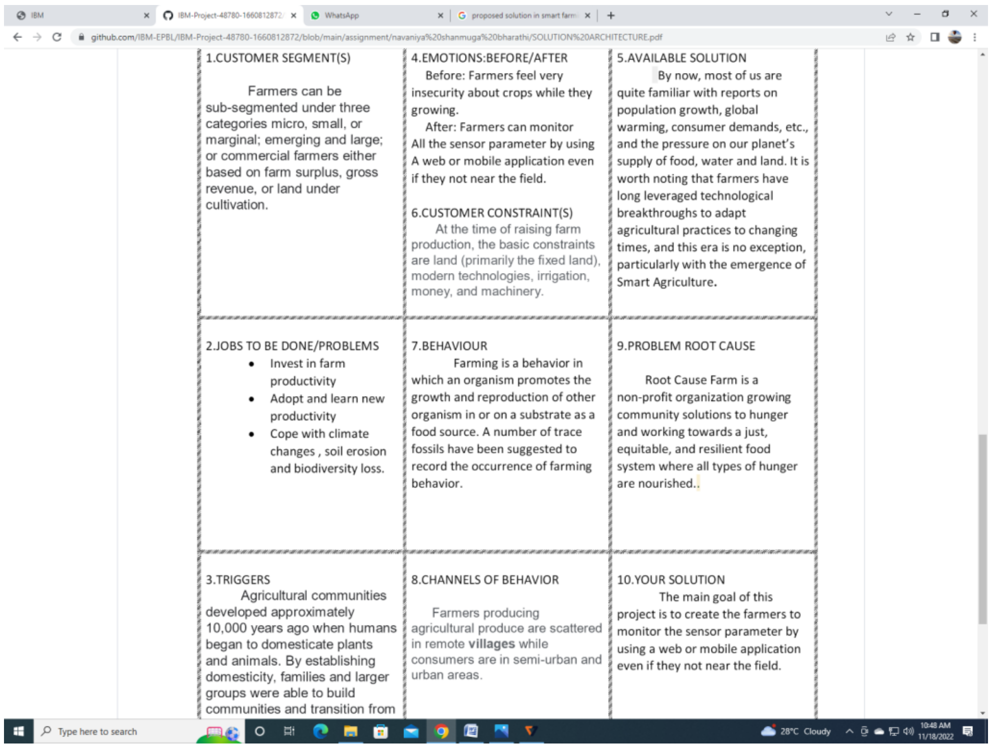
3.2. IDEATION AND BRAINSTORMING


github.com/IBM-EPBL/IBM-Project-48780-1660812872/blob/main/assignment/navaniya%20shanmuga%20bharathi/brainstorm%20and%20idea%20prioritization.pdf
IBM-EPBL / IBM-Project-48780-1660812872 Public
Code Issues Pull requests Actions Projects Wiki Security Insights Settings
main IBM-Project-48780-1660812872 / assignment / navaniya shanmuga bharathi / brainstorm and idea prioritization.pdf
Go to file
priya123896 Add files via upload
Latest commit 4e28074 on Oct 15 History
1 contributor
113 KB
Download
Brainstorm & idea prioritization
© 2022 GitHub, Inc. Terms Privacy Security Status Docs Contact GitHub Pricing API Training Blog About
Type here to search
28°C Cloudy
10:39 AM
11/18/2022

3.3. PROPOSED SOLUTION


1
Sensing Technologies
2
Software Applications
3
Communications Systems -e.g- Cellular
Telematics, Positioning Technologies
5
Hardware and Software Systems
6
Data analytics Solutions
SMART FARMING

3.4. PROBLEM SOLUTION FIT


IBM
IBM-Project-48780-1660812872
WhatsApp
proposed solution in smart farm
github.com/IBM-EPBL/IBM-Project-48780-1660812872/blob/main/assignment/navaniya%20shanmuga%20bharathi/SOLUTION%20ARCHITECTURE.pdf
Type here to search
28°C Cloudy
10:48 AM
11/18/2022

1.CUSTOMER SEGMENT(S) Farmers can be sub-segmented under three categories micro, small, or marginal; emerging and large; or commercial farmers either based on farm surplus, gross revenue, or land under cultivation.	4.EMOTIONS:BEFORE/AFTER Before: Farmers feel very insecurity about crops while they growing. After: Farmers can monitor All the sensor parameter by using A web or mobile application even if they not near the field. 6.CUSTOMER CONSTRAINT(S) At the time of raising farm production, the basic constraints are land (primarily the fixed land), modern technologies, irrigation, money, and machinery.	5.AVAILABLE SOLUTION By now, most of us are quite familiar with reports on population growth, global warming, consumer demands, etc., and the pressure on our planet's supply of food, water and land. It is worth noting that farmers have long leveraged technological breakthroughs to adapt agricultural practices to changing times, and this era is no exception, particularly with the emergence of Smart Agriculture.
2.JOBS TO BE DONE/PROBLEMS • Invest in farm productivity • Adopt and learn new productivity • Cope with climate changes , soil erosion and biodiversity loss.	7.BEHAVIOUR Farming is a behavior in which an organism promotes the growth and reproduction of other organism in or on a substrate as a food source. A number of trace fossils have been suggested to record the occurrence of farming behavior.	9.PROBLEM ROOT CAUSE Root Cause Farm is a non-profit organization growing community solutions to hunger and working towards a just, equitable, and resilient food system where all types of hunger are nourished..
3.TRIGGERS Agricultural communities developed approximately 10,000 years ago when humans began to domesticate plants and animals. By establishing domesticity, families and larger groups were able to build communities and transition from	8.CHANNELS OF BEHAVIOR Farmers producing agricultural produce are scattered in remote **villages** while consumers are in semi-urban and urban areas.	10.YOUR SOLUTION The main goal of this project is to create the farmers to monitor the sensor parameter by using a web or mobile application even if they not near the field.

4. REQUIREMENT ANALYSIS

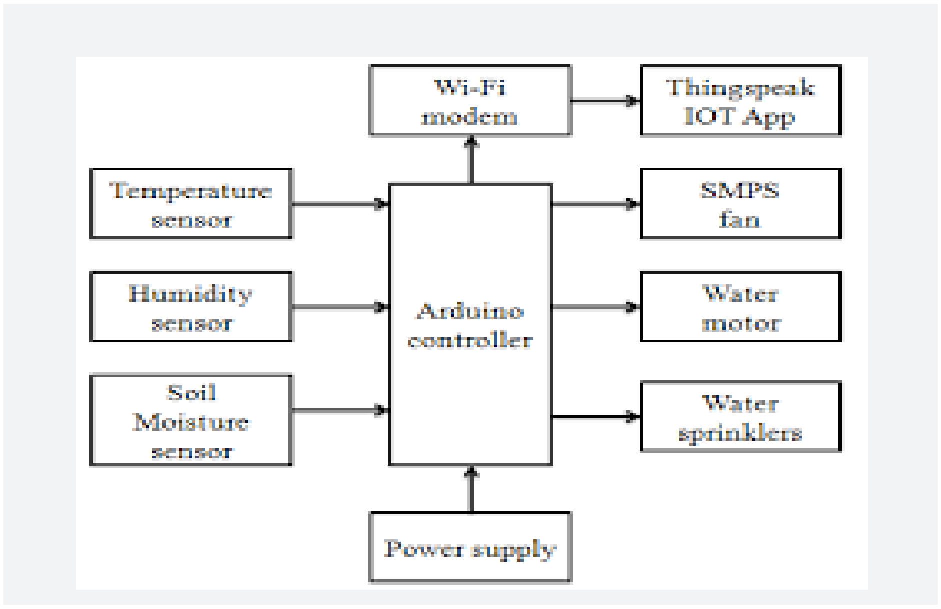
4.1. Funtional Requirements

Functional requirements are product features or functions that developers must implement to enable users to accomplish their tasks. So, it's important to make them clear both for the development team and the stakeholders. Generally, functional requirements describe system behavior under specific conditions.

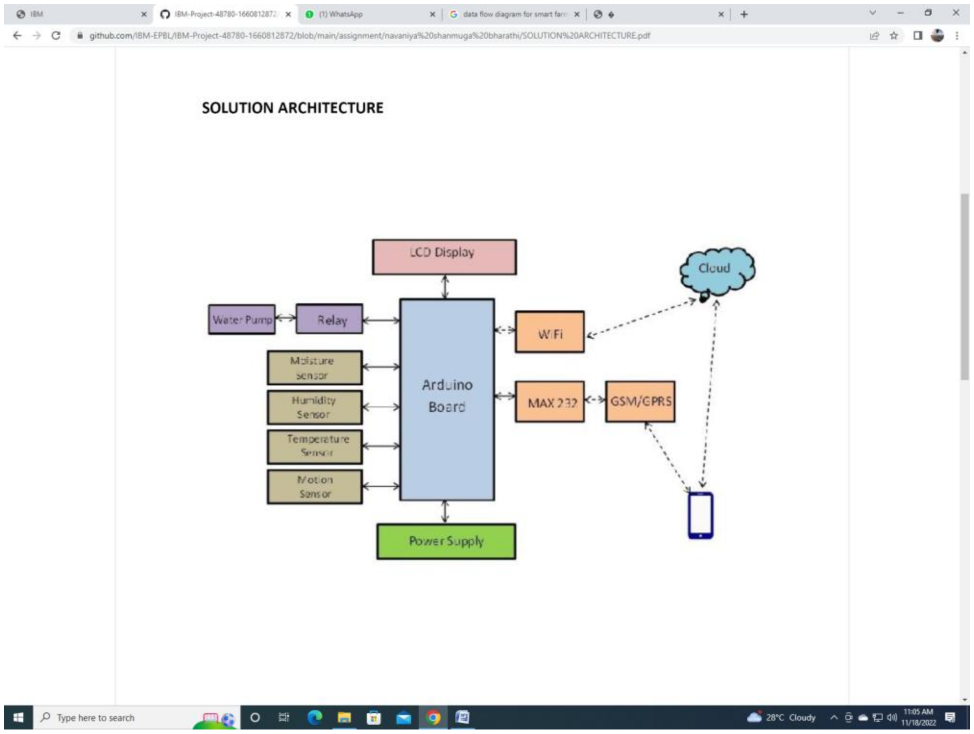
4.1. Non Functional Requirements

Non Functional requirement analysis is done for the Internet of Things (IoT) based traffic management unit which indicates the traffic density. Few quality characteristics of the design are analyzed during the development process. Design model decisions are governed by these Non Functional Requirement (NFR) design parameters. These quality characteristics considered are cost, sensitivity, design complexity, storage capacity, development process, response criteria and environmental impact. Having analyzed the quality attributes the design components are selected to deploy the design unit. Considerable effort needs to be put at the system design level to streamline the Internet of Things based design process.

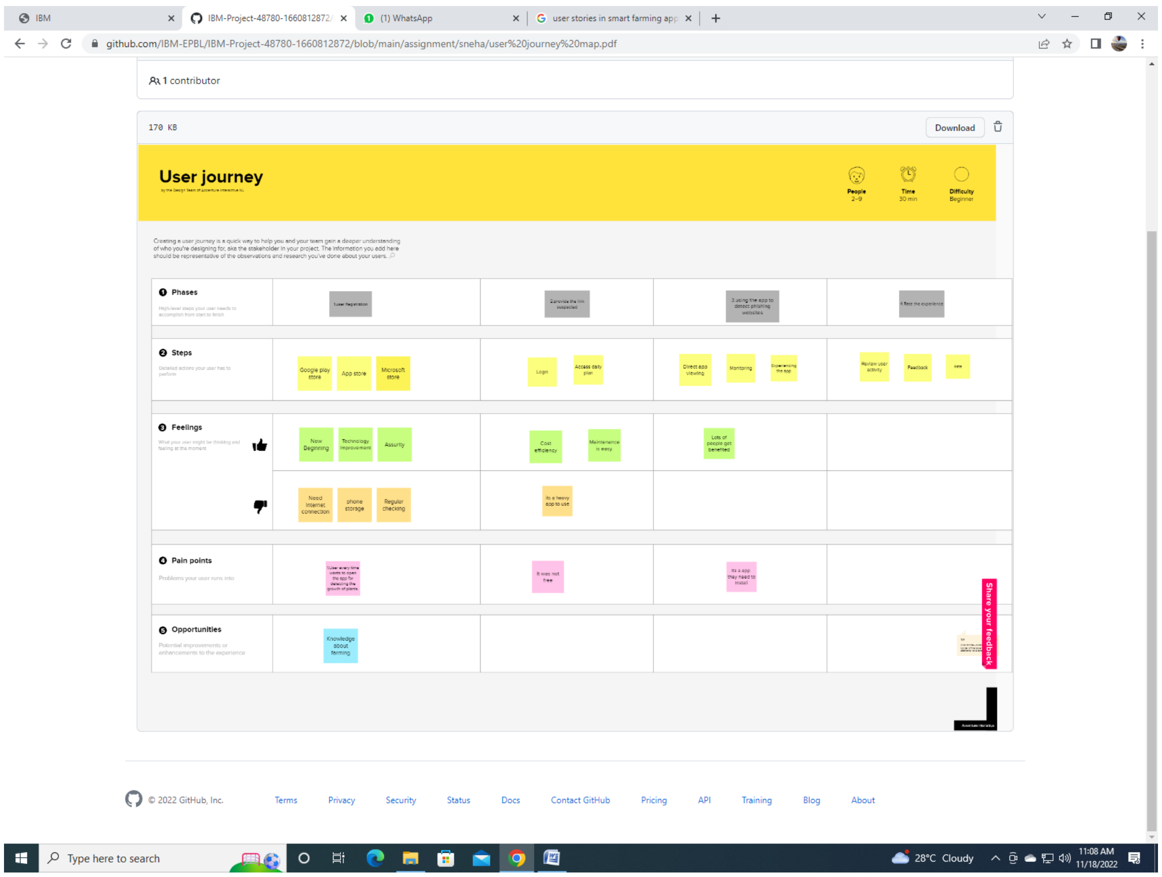
5. PROJECT DESIGN



5.2 Solution and Technical Architecture



5.3. User Stories


github.com/IBM-EPBL/IBM-Project-48780-1660812872/blob/main/assignment/sneha/user%20journey%20map.pdf
1 contributor
170 KB
Download

User journey

by the Design Team of Accenture Interactive NL

People 2–9 | Time 30 min | Difficulty Beginner

Creating a user journey is a quick way to help you and your team gain a deeper understanding of who you're designing for, aka the stakeholder in your project. The information you add here should be representative of the observations and research you've done about your users.

1 Phases High-level steps your user needs to accomplish from start to finish	1.user Registration	2.provide the link suspected	3.using the app to detect phishing websites	4.Rate the experience
2 Steps Detailed actions your user has to perform	Google play store; App store; Microsoft store	Login; Access daily plan	Direct app viewing; Monitoring; Experiencing the app	Review user activity; Feedback; Rate
3 Feelings What your user might be thinking and feeling at the moment 👍	New Beginning; Technology improvement; Assurity	Cost efficiency; Maintenance is easy.	Lots of people get benefited	
👎	Need internet connection; phone storage; Regular checking	its a heavy app to use		
4 Pain points Problems your user runs into	1.User every time wants to open the app for detecting the growth of plants.	It was not free	its a app they need to install	
5 Opportunities Potential improvements or enhancements to the experience	Knowledge about farming			

© 2022 GitHub, Inc. Terms Privacy Security Status Docs Contact GitHub Pricing API Training Blog About

6. PROJECT PLANNING AND SCHEDULING

6.1. Sprint Planning and Estimation

Planning and Estimation are essential in software projects to achieve predictability, reduce the risks involved, and set a basic expectation for all stakeholders. Planning brings a lot of focus on preparation and forecasting whereas Estimation is a process to forecast project-related variables i.e., effort, scope, schedule, etc.

6.2. Sprint Delivery Scheduling

Delivery Scheduling is required irrespective of the project management methodologies that the team follows, whether it is Waterfall or Agile. Planning gives the project team a perspective on how to meet the objective in a systematic way and helps project stakeholders to keep a tab on the project progress and investments done.

6.3. Reports From JIRA

Smart farming is an emerging concept that refers to managing farms using technologies like IoT, robotics, drones and AI to increase the quantity and quality of products while optimizing the human labor required by production.

7. CODING AND SOLUTIONING

7.1. Feature 1

Connecting Sensors with Arduino using C++ code

```
#include "Arduino.h"

 #include "dht.h"

#include "SoilMoisture.h"

#define dht_apin A0

 const int sensor_pin = A1; //soil moisture

 int pin_out = 9; dht DHT;
 int c=0; void setup()
 {
pinMode(2, INPUT); //Pin 2 as INPUT
pinMode(3, OUTPUT); //PIN 3 as OUTPUT
 pinMode(9, OUTPUT);//output for pump
 }
void loop()
 {
 if (digitalRead(2) == HIGH)
{
 digitalWrite(3, HIGH); // turn the LED/Buzz ON
 delay(10000); // wait for 100 msecond
 digitalWrite(3, LOW); // turn the LED/Buzz OFF
delay(100);
 }
Serial.begin(9600);
delay(1000);
 DHT.read11(dht_apin); //temperature
 float h=DHT.humidity;
 float t=DHT.temperature;
delay(5000);

 Serial.begin(9600);

float moisture_percentage;

 int sensor_analog;

sensor_analog = analogRead(sensor_pin);

moisture_percentage = ( 100 - ( (sensor_analog/1023.00) * 100 ) );

 float m=moisture_percentage;

delay(1000);

 if(m=0) { mySerial.begin(9600);

 Delay(15000);

Serial. Begin(9600);

 delay(1000);

 Serial.print("\r");

delay(1000);

 Serialprint((String)"update-
>"+(String)"Temperature="+t+(String)"Humidity="+h+(String
)"Moisture="+m);

delay(1000);

}

}
```

7.2. Feature 2

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-
scale=1.0">
<link rel="stylesheet" href="/css/fireoauth.css">
<link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/nprogress/0.2.0/nprogre
ss.min.css">
<link rel="shortcut icon"
href="https://raw.githubusercontent.com/tharunoptimus-
pd/firepwa/main/favicon.ico?token=GHSAT0AAAAAABR46HVJ5
M5L3QGFRZRQXOISYUJUWAA" type="image/x-icon">
<style>
html, body {
height: 100%;
margin: 0;
font-family: -apple-system, BlinkMacSystemFont, "Segoe UI",
Roboto, Oxygen, Ubuntu, Cantarell, "Open Sans", "Helvetica
Neue", sans-serif; font-weight: 300;
}
a {
text-decoration: none; color: #007bff;
font-weight: 500; font-size: 1.2rem;
}
h3 {
```

```
font-size: 1.4rem;
}
h3, h4 { margin: 0;
padding: 0.3rem 0;
}
.wrapper { display: flex;
flex-direction: column; align-items: center; justify-content: center;
height: 100%;
text-align: center;
}
.oneClickSignin { padding: 0.5rem;
border: 1px solid #44444444; border-radius: 5px;
box-shadow: 0 0 3px 0px #44444444;
opacity: 0.2;
pointer-events: none;
}
.qrcode { opacity: 0.1;
}
.learnAboutFire { padding-top: 1.25em;
}
.qrHolder { display: none;
margin-top: 3rem;
}
.qrContainer {
align-items: center; display: flex;
justify-content: center; padding: 8px;
margin: 2rem auto;
box-shadow: 0 0px 6px 1px rgb(0 0 0 / 16%); border: 1px solid
#44444444;
border-radius: 6px; width: 200px; height: 200px;
}
</style>
<title>Fire OAuth</title>
<script>
```

```
if (window.location.hostname !== "localhost") { if (location.protocol
!== "https:") {
location.replace(
`https:${location.href.substring( location.protocol.length
)}`
)
}
}
</script>
</head>
<body>
<div class="wrapper">
<h3 class="pageTitle">Login with Fire ??</h3>
<div class="qrAuthorize">
<h4 class="subTitle">Scan QR from your Fire OAuth App??</h4>
<div class="qrContainer">
<canvas id="qr-code" class="qrcode"></canvas>
</div>
</div>
<div class="oneClickSignin">
<h4>Have Fire PWA on this device?</h4>
<a target="_blank" id="authorizeOverLink"
href="https://firepwa.netlify.app/authorize?sessionId"
rel="noopener">Click to Authorize ?? </a>
</div>
<div class="learnAboutFire">
<a target="_blank" href="https://fireoauth.netlify.app"
rel="noopener">Learn More about Fire
??</a>
</div>
</div>
<script
src="https://cdnjs.cloudflare.com/ajax/libs/nprogress/0.2.0/nprogres
s.min.js"></script>
```

```
<script
src="https://cdnjs.cloudflare.com/ajax/libs/qrious/4.0.2/qrious.min.j
s"></script>
<script
src="https://cdnjs.cloudflare.com/ajax/libs/socket.io/4.2.0/socket.io.j
s"></script>
<script>
const FIRE_API_KEY = "635b790a3bcc6b59c4b772d0"
const FIRE_ENDPOINT =
"https://fire.adaptable.app/api/apis/generate" const
CHANNEL_NAME = "fireOAuthChannel"
const broadCastingChannel = new
BroadcastChannel(CHANNEL_NAME)
const FIRE_SERVER_SOCKET_ENDPOINT =
"https://fire.adaptable.app" let socket =
io(FIRE_SERVER_SOCKET_ENDPOINT)
let qr
let qrcode = document.querySelector(".qrcode")
let oneClickSignin = document.querySelector(".oneClickSignin") let
pageTitle = document.querySelector(".pageTitle")
let subTitle = document.querySelector(".subTitle")
function setOpacity(opacity) { oneClickSignin.style.opacity =
opacity
oneClickSignin.style.pointerEvents = opacity === "1" ? "auto" :
"none" qrcode.style.opacity = opacity
}
async function getSessionID() { let response
try {
response = await fetch(`${FIRE_ENDPOINT}/${FIRE_API_KEY}`,
{ method: "GET",
headers: {
"Content-Type": "application/json",
}
})
} catch (error) { console.log(error) return null }
let { sessionId, chatRoomId } = data return { sessionId, chatRoomId
}
}
function generateQR(value) { (qr = new QRious({
element: document.getElementById("qr-code"), size: 200,
level: 'M', value: value,
}))
}
function changeHREF ({sessionId, chatRoomId}) {
let firePwaUrlHostname = https://firepwa.netlify.app
let originURL = encodeURIComponent(window.location.origin)
let url =
`${firePwaUrlHostname}/authorize.html?sessionId=${sessionId}&c
hatRoomId=${chatRoomId}&url=${ori ginURL}`
let a = document.getElementById("authorizeOverLink") a.href =
url
}
async function fire() { NProgress.set(0.4)
let { sessionId, chatRoomId } = await getSessionID()
null) {
if(sessionId === undefined || chatRoomId === undefined || sessionId
=== null || chatRoomId ===
pageTitle.innerHTML = "Something went wrong ???"
subTitle.innerHTML = "Please try again later ????" return
}
setOpacity("1")
NProgress.done() let data = {
sessionId,
url: encodeURIComponent(window.location.origin)
}
data = JSON.stringify(data) generateQR(data)
changeHREF({sessionId, chatRoomId}) socket.emit("join room",
sessionId)
}
fire() return null }
```

```
let data = await response.json()
let { sessionId, chatRoomId } = data return { sessionId, chatRoomId
}
}
function generateQR(value) { (qr = new QRious({
element: document.getElementById("qr-code"), size: 200,
level: 'M', value: value,
}))
}
function changeHREF ({sessionId, chatRoomId}) {
let firePwaUrlHostname = "https://firepwa.netlify.app"
let originURL = encodeURIComponent(window.location.origin)
let url =
`${firePwaUrlHostname}/authorize.html?sessionId=${sessionId}&c
hatRoomId=${chatRoomId}&url=${ori ginURL}`
let a = document.getElementById("authorizeOverLink") a.href =
url
}
async function fire() { NProgress.set(0.4)
let { sessionId, chatRoomId } = await getSessionID()
null) {
if(sessionId === undefined || chatRoomId === undefined || sessionId
=== null || chatRoomId ===
pageTitle.innerHTML = "Something went wrong ???"
subTitle.innerHTML = "Please try again later ????" return
}
setOpacity("1")
NProgress.done() let data = {
sessionId,
url: encodeURIComponent(window.location.origin)
}
data = JSON.stringify(data) generateQR(data)
changeHREF({sessionId, chatRoomId}) socket.emit("join room",
sessionId) }
fire()
```

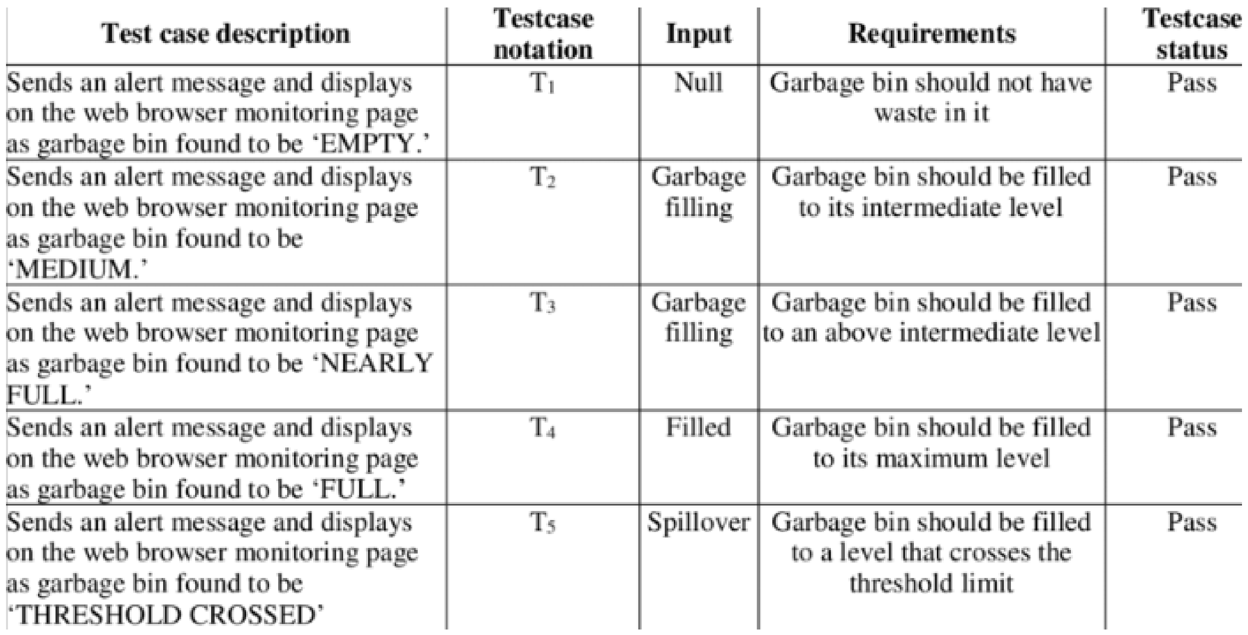
```
socket.on("trusted token", (token) => {
let data = {} data.success = true data.token = token
broadCastingChannel.postMessage(data)
window.close()
})
</script>
</body>
</html>
DASHBOARD:
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-
scale=1.0">
<link rel="stylesheet" href="./css/dashboard.css">
<title>Dashboard</title>
<script src="./localforage.js"></script>
</head>
<body>
<div class="wrapper">
<div class="header">
<span class="heading">Dashboard</span>
<span class="right">
<span class="username">Hello User</span>
<span>
<img class="profilePic"
src="https://avatars.dicebear.com/api/avataaars/asdfasdfds.svg"
alt="User Profile" height="30" width="30">
</span>
</span>
</div>
<div class="actionCenter">
<div class="action">
```

```
<span>Create Child Card</span>
</div>
<div class="action">
<span class="logout">Log out</span>
</div>
</div>
<div class="childCardContainer">
<div class="childCard">
<div class="childCardHeader">
<span>Child Name</span>
<span>Age 12</span>
</div>
<div class="actions">
<span>View</span>
<span>GeoFence</span>
</div>
</div>
</div>
</div>
<script>
async function main() {
let userData = await localforage.getItem('userData') if(userData ==
null) {
window.location.href = "/login"
}
document.querySelector(".username").innerHTML = `Hello
${userData.firstName}` document.querySelector(".profilePic").src =
userData.profilePic
}
main()
document.querySelector(".logout").addEventListener("click",
async () => { await localforage.setItem('userData', null)
window.location.href = "/login"
})
</script>
```

</body>
</html>

8. TESTING

8.1. Test Cases



Test case description	Testcase notation	Input	Requirements	Testcase status
Sends an alert message and displays on the web browser monitoring page as garbage bin found to be 'EMPTY.'	T_1	Null	Garbage bin should not have waste in it	Pass
Sends an alert message and displays on the web browser monitoring page as garbage bin found to be 'MEDIUM.'	T_2	Garbage filling	Garbage bin should be filled to its intermediate level	Pass
Sends an alert message and displays on the web browser monitoring page as garbage bin found to be 'NEARLY FULL.'	T_3	Garbage filling	Garbage bin should be filled to an above intermediate level	Pass
Sends an alert message and displays on the web browser monitoring page as garbage bin found to be 'FULL.'	T_4	Filled	Garbage bin should be filled to its maximum level	Pass
Sends an alert message and displays on the web browser monitoring page as garbage bin found to be 'THRESHOLD CROSSED'	T_5	Spillover	Garbage bin should be filled to a level that crosses the threshold limit	Pass

8.2. User Acceptance Testing

- **Acceptance testing** is formal testing based on user requirements and function processing.
- It determines whether the software is conforming specified requirements.

9. Results

9.1. Performance Metrics

Smart farming helps farmers to better understand the important factors such as water, topography, aspect, vegetation and soil types. This **allows farmers to determine the best uses of scarce resources within their production environment and manage these in an environmentally and economically sustainable manner.**

10. ADVANTAGES

- Increased work efficiency. One of the greatest things about Smart Farming is its potential to save valuable time. ...
- Improved fuel efficiency. Smart Farming allows farmers to be much more precise. ...
- Reduced consumables.
- Increased yields.

DISADVANTAGES

- Poor living conditions and hygiene for livestock.
- Excessive use of agro-chemicals. ...
- Deforestation and alteration of the natural environment. ...
- Risks to human health.
- The use of chemical hormones in food. ...
- Possibility of poor quality food products.

11. CONCLUSION

Through IoT, the farmers and the companies enjoy profits in the form of better productivity of field crops and safety to the farmers and other workers. Application of fertilizer nutrients, water, herbicides, insecticides, fungicides and pesticides are minimized. This process reduces the runoff loss of harmful chemicals to ground and surface water like rivers and lakes. This results in less stress on the ecosystem. IoT has a positive impact on the healthcare systems by reducing the healthcare costs on the government and improving the quality of life, and reduced carbon footprints. IoT may be served as a climate-resilient technology or system in agriculture.

12. FUTURE SCOPE

Smart farming is certainly a leading enabler in producing more food with less for an increasing world population. In particular, smart farming enables increased yield through more efficient use of natural resources and inputs, and improved land and environmental management.

13. APPENDIX

Source code

```
#include <ESP8266WiFi.h>
#include <ESP8266HTTPClient.h>
String thingSpeakAddress= "http://api.thingspeak.com/update?";
String writeAPIKey;
String tsfield1Name;
String request_string,request_string1;
HTTPClient http;
#include <DHT.h> // Including library for dht
// library
// https://github.com/adafruit/DHT-sensor-library
#include <ESP8266WiFi.h>
String apiKey = "77921LPMGM2OAGQE";    // Enter your Write API
```

```
key from ThingSpeak
const char *ssid = "DESKTOP";     // replace with your wifi
ssid and wpa2 key
const char *pass = "asdfghjkl";     // WIFI Password
const char* server = "api.thingspeak.com";
#define DHTPIN 0      //pin D3 where the dht11 is connected
DHT dht(DHTPIN, DHT11);
WiFiClient client;
void setup()
{  dht.begin();
Serial.begin(115200);
delay(3000);
WiFi.disconnect();
Serial.println("START");
WiFi.begin("DESKTOP","asdfghjkl");
while ((!(WiFi.status() == WL_CONNECTED))){
delay(300);
Serial.println("...");
}
Serial.println("I AM CONNECTED");
}
void loop()
{
if (client.connect("api.thingspeak.com",80))
{
request_string = thingSpeakAddress;
request_string += "key=";
request_string += "77921LPMGM2OAGQE";
request_string += "&";
request_string += "field3";
request_string += "=";
request_string += analogRead(A0);
http.begin(request_string);
http.GET();
http.end();
}
delay(10);
float h = dht.readHumidity();
float t = dht.readTemperature();
if (isnan(h) || isnan(t))
{
Serial.println("Failed to read from DHT sensor!");
return;
}
if (client.connect(server,80))  //  "184.106.153.149" or
api.thingspeak.com
{
```

```
String postStr = apiKey;
postStr +="&field1=";
postStr += String(t);
postStr +="&field2=";
postStr += String(h);
postStr += "\r\n\r\n";
client.print("POST /update HTTP/1.1\n");
client.print("Host: api.thingspeak.com\n");
client.print("Connection: close\n");
client.print("X-THINGSPEAKAPIKEY: "+apiKey+"\n");
client.print("Content-Type: application/x-www-form-
urlencoded\n");
client.print("Content-Length: ");
client.print(postStr.length());
client.print("\n\n");
client.print(postStr);
Serial.print("Temperature: ");
Serial.print(t);
Serial.print(" degrees Celcius, Humidity: ");
Serial.print(h);
Serial.print(" Soil Sensor ");
Serial.print(A0);
Serial.println("%. Send to Thingspeak.");
}
client.stop();
Serial.println("Waiting...");
// thingspeak needs minimum 15 sec delay between updates
delay(10);
}
```