

Gas Leakage Monitoring & Alerting system for Industries

Repo ID : IBM-Project-16053-1659606982

Team ID : PNT2022TMID08134

Team Members :

KARAN S

HARIHARAN P

MADHANKUMAR

NANDHAKUMAR

Project Guided by :

Mrs.A.P.NITHIYAPRIYA

Assistant Professor

1. INTRODUCTION	
1. Project Overview	4
2. Purpose	4
2. LITERATURE SURVEY	
1. Existing problem	5
2. References	5
3. Problem Statement Definition	6
3. IDEATION & PROPOSED SOLUTION	
1. Empathy Map Canvas	7
2. Ideation & Brainstorming	8
3. Proposed Solution	11
4. Problem Solution fit	12
4. REQUIREMENT ANALYSIS	
1. Functional requirement	13
2. Non-Functional requirements	13
5. PROJECT DESIGN	
1. Data Flow Diagrams	14
2. Solution & Technical Architecture	15
3. User Stories	16
6. PROJECT PLANNING & SCHEDULING	
1. Sprint Planning & Estimation	17
2. Sprint Delivery Schedule	19
3. Reports from JIRA	23
7. CODING & SOLUTIONING	
1. Feature 1	26
2. Feature 2	53

8. TESTING	
1. Test Cases	53
2. User Acceptance Testing	
9. RESULTS	
1. Performance Metrics	55
10. ADVANTAGES & DISADVANTAGES	56
11. CONCLUSION	56
12. FUTURE SCOPE	57
13. APPENDIX	
Source Code	61
GitHub & Project Demo Link	68

1. INTRODUCTION

1.1 Project Overview

As to develop a system for detecting and alerting industries that utilise hazardous gases in production. The industries benefit from this project's assistance in tracking hazardous gas emissions. The gas sensors will be integrated in a number of locations to track gas leaks. If a gas leak is discovered anywhere, the administrators will be informed along with the location. Administrators can see the sensor parameters on the web application.

1.2 Purpose

The usage of gas detectors includes the detection of combustible, flammable, and hazardous gases as well as oxygen loss and gas leak detection. It generates a warning sound and directs operators to investigate the location where the leak is occurring. and can prevent fire accidents.

2. LITERATURE SURVEY

1.Existing problem

- In the existing method, different gas sensing technology is used. The LPG leakage is detected by the semiconductor sensor.
- In recent times LPG accidents are prevalent. The main reason for these accidents is due to the leakage of LPG.
- This leakage of LPG starts when we forget to close the main regulator valve. This is the basis of these kinds of accidents.
- Already there are some sorts of remedial measures such as when the leakage is detected, an alert message is sent to the fire station and the owner.

2.References

Sensor based Smart Automated Gas Leakage & Detection and Prevention System

<https://drive.google.com/file/d/1HOkIGsl4v75sZzrTznNNe2micKUSOFzs/view?usp=drivesdk>

Gas Leakage Detection and Alert System using IoT

<https://drive.google.com/file/d/1HZOaVb9pqUHQTc0zI6D823sfCLObw4ER/view?usp=drivesdk>

A smart gas leakage monitoring system for use in hospitals

https://drive.google.com/file/d/1-Ahb05x-rfZWKXMOIjLfOXincvVpxsIN/view?usp=share_link

IoT Based LPG Gas Leakage Detector

https://drive.google.com/file/d/1Gzr5TUW5Jb5VOSXWdFZqpFOqMo_b3V3k/view?usp=share_link

Mobile Based Gas Leakage Monitoring Using IOT

https://drive.google.com/file/d/1SroGSQaz9K0m3HOaj1ofkohoFaelSfZI/view?usp=share_link

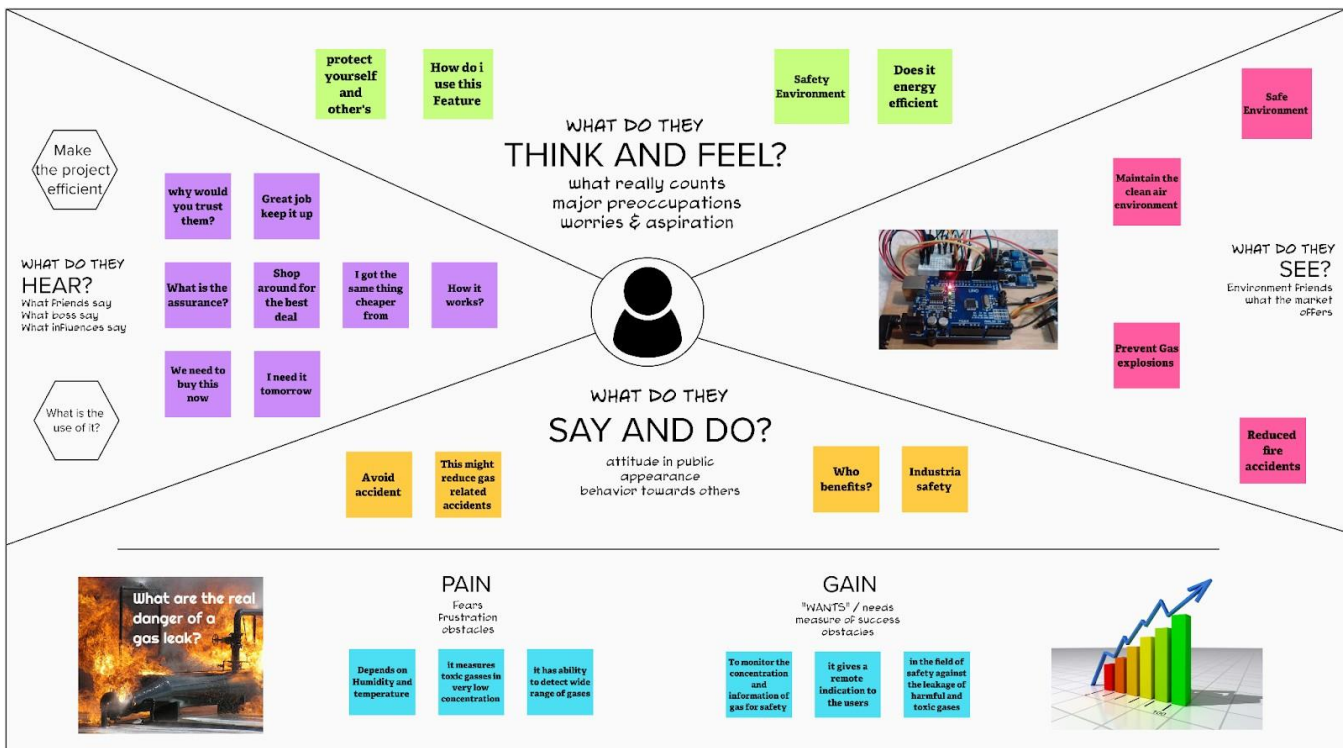
Problem Statement Definition

The presence of dangerous gas leaks in industries, workworkplaces, gas containers. The leakage of gas detection is used to sound the alarm with the help of a buzzer. The buzzer gives an audible signal to the presence of a hydrogen or methane sensor that acts as a switch with different instructions. Each detected gas level is sent to the Arduino which acts as an existing gas level analyzer controller. The controller is also responsible for activating lights, buzzers, GPRS and the Global System Mobility module (GSM). The system has two lights indicating the detected gas level. Green light means “safe level” and “Synchronized”, red means “danger level”. The buzzer is activated when the controller detects a hazard and switches ON red light. The Application shows at which location the gas level is detected using GPRS Module and the GSM module sends Short Message Service (SMS) as a notification to the concerned people.

3. IDEATION & PROPOSED SOLUTION

Empathy Map Canvas

A compassion map is a basic, simple to-process visual that catches information about a client's ways of behaving and perspectives. It is a valuable device to assist teams with better comprehension of their clients. Making a compelling arrangement requires grasping the genuine issue and the individual who is encountering it. The activity of making the guide helps members consider things according to the client's point of view alongside their objectives and challenges.




Step-1: Team Gathering, Collaboration and Select the Problem Statement

In this step team members gather and provide their ideas and collaborate those ideas and select their problem statement. The ideas should be relevant to their problem statement

Brainstorming

Template



Brainstorm & idea prioritization

Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.

🕒 10 minutes to prepare
🕒 1 hour to collaborate
👤 2-8 people recommended

[Share template feedback](#)

➔

Before you collaborate

A little bit of preparation goes a long way with this session. Here's what you need to do to get going.

🕒 10 minutes

A

Team gathering

Define who should participate in the session and send an invite. Share relevant information or pre-work ahead.

B

Set the goal

Think about the problem you'll be focusing on solving in the brainstorming session.

C

Learn how to use the facilitation tools

Use the Facilitation Superpowers to run a happy and productive session.

[Open article](#) ➔

1


Define your problem statement

What problem are you trying to solve? Frame your problem as a How Might We statement. This will be the focus of your brainstorm.

🕒 5 minutes

PROBLEM

How might we implement Hazardous gas sensor?



Key rules of brainstorming

To run a smooth and productive session

🗣️ Stay in topic.

💡 Encourage wild ideas.

⏸️ Defer judgment.

👂 Listen to others.

🗣️ Go for volume.

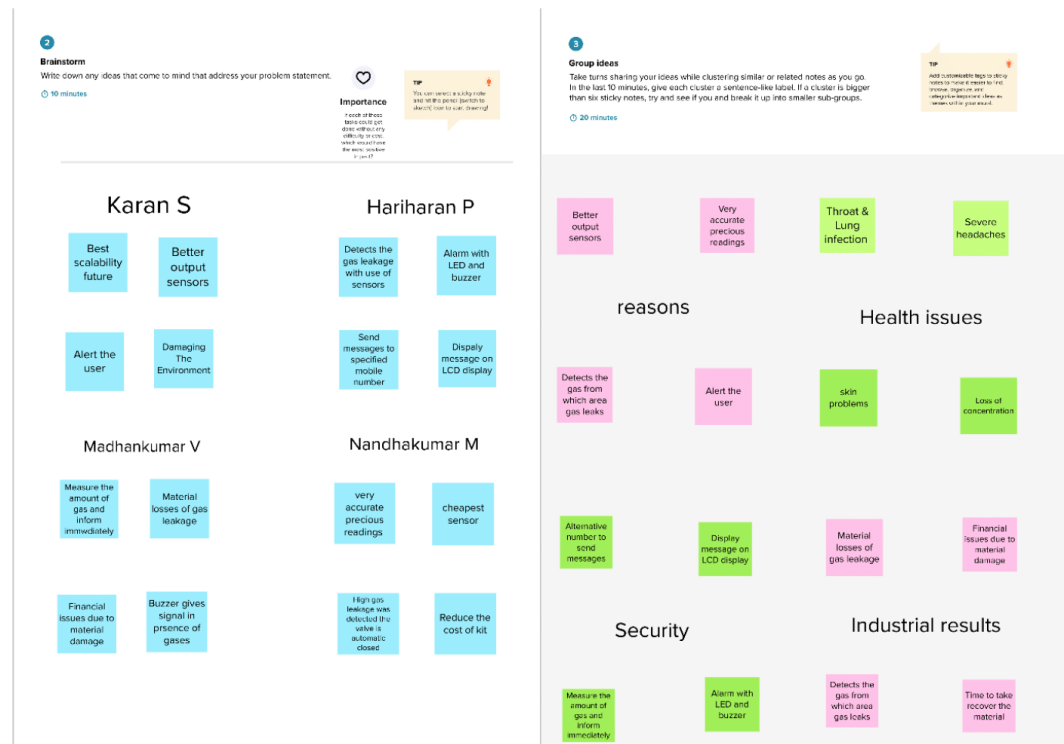
👁️ If possible, be visual.

8

Brainstorm, Idea Listing and Grouping

In this step they put their ideas and views which are prioritised based on their importance and the ideas are grouped. These ideas are categorised according to their relevant classifications.

Step-2: Brainstorm, Idea Listing and Grouping



Step-3: Idea Prioritisation

As mentioned, idea prioritisation is just a part of the idea management process. Having a structured idea management process and a systematic way of gathering, evaluating and prioritising new ideas takes time. To make it work, the entire idea management process should be integrated into everyday ways of working.

4

Prioritize

Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which ideas are important and which are feasible.

⌚ 20 minutes

After you collaborate

You can export the mural as an image or pdf to share with members of your company who might find it helpful.

Quick add-ons

- A Share the mural**
Share a view link to the mural with stakeholders to keep them in the loop about the outcomes of the session.
- B Export the mural**
Export a copy of the mural as a PNG or PDF to attach to emails, include in slides, or save in your drive.

Keep moving forward

- Strategy blueprint**
Define the components of a new idea or strategy.
[Open the template →](#)
- Customer experience journey map**
Understand customer needs, motivations, and obstacles for an experience.
[Open the template →](#)
- Strengths, weaknesses, opportunities & threats**
Identify strengths, weaknesses, opportunities, and threats (SWOT) to develop a plan.
[Open the template →](#)

[Share template feedback](#)

Proposed Solution

S.NO.	Parameter	Description
1	Problem Statement (Problem to be solved)	Monitoring harmful gases leakage sensors for industry.
2	Idea / Solution description	By using Arduino UNO with MQ Gas sensors it detects the Gas leakage and alert the customer
3	Novelty / Uniqueness	We use location tagging and alert services so that the customers and fire department team will be notified of the exact location as soon as the leakage is detected.
4	Social Impact / Customer Satisfaction	saves labourers' lives and resources from fire accidents.
5	Business Model (Revenue Model)	By using : safety from fire accidents, strong financial, By without using Fire accident, loss of lives, loss of resources.
6	Scalability of the Solution	It is well advanced in saving the lives of peoples of the industries.

Problem Solution fit

Project Title : Gas Leakage Monitoring and Alerting System

Team ID : PNT2022TMID08134

Define CS, fit into CC	1. CUSTOMER SEGMENT(S) CS <p>The business owners who employ gases in their manufacturing.</p>	6. CUSTOMER CONSTRAINTS CC <p>Due to expensive installation costs for alternative products, they have moved away from current technology.</p>	5. AVAILABLE SOLUTIONS AS <p>-Automatic shutoff valve. -In past they identify the leakage by smell which causes the delay i taking actions. -Pros & Cons they easily identify the locations and neccessary actions.</p>	Explore AS, differentiate
	2. JOBS-TO-BE-DONE / PROBLEMS J&P <p>People are concerned about gas leaks after the work is done or are unsure if they closed it or not.</p>	9. PROBLEM ROOT CAUSE RC <p>-Industries without enough training frequently use faulty valves. -Industries without enough training frequently use faulty valves without control.without control.</p>	7. BEHAVIOUR BE <p>To determine the gas's characteristics and solve the issue, they will locate the leak and identify the warning.</p>	
Identify strong TR & EM	3. TRIGGERS TR <p>Safe precautions for the workers to work without any fear.</p>	10. YOUR SOLUTION SL <p>Low cost IoT based device that can be easily accessed and fixed by people.</p>	8. CHANNELS of BEHAVIOUR CH <p>Promoting through social media. With the help of social media entrepreneurs.</p>	Extract online & offline CH of BE
	4. EMOTIONS: BEFORE / AFTER EM <p>If they faced a problem or a situtaion, they could confidently handle the situation and ease to access.</p>		8.2 OFFLINE <p>The identification of the area of leakage and take precautionary actions</p>	



Problem-Solution fit canvas is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 license
 Created by Daria Nepriakhina / Amaltama.com



4. REQUIREMENT ANALYSIS

Functional Requirements:

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	Uploading Image	By uploading the images of spirals and waves the further process will take place.
FR-2	Result Generation	With the given data the result can be generated as if the person is healthy or not healthy. It also generates the stage of the disease along with its diagnosis.

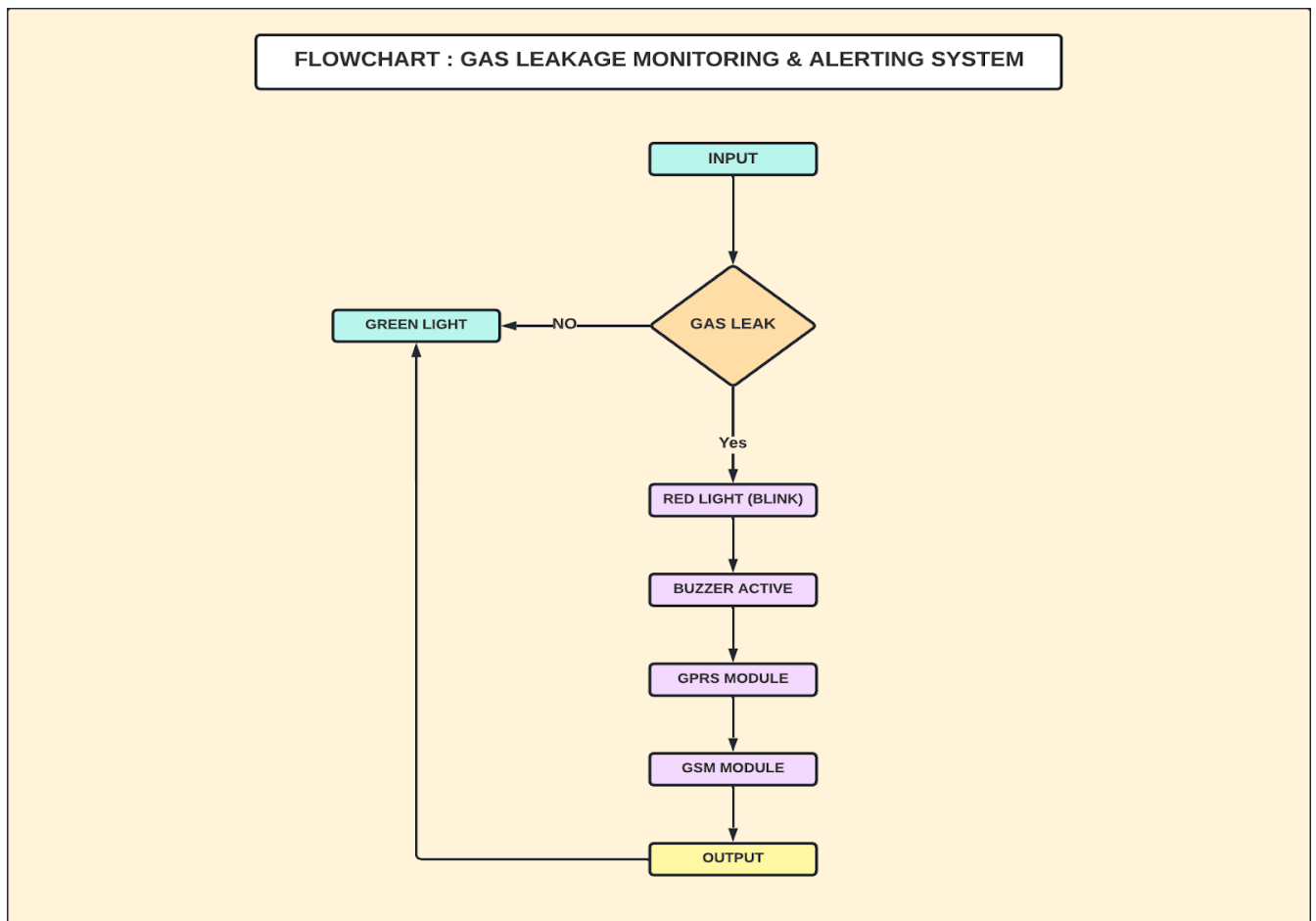
Non-functional Requirements:

FR No.	Non-Functional Requirement	Description
NFR-1	Availability	It will be available to everyone .
NFR-2	Security	The data given by the user will be very secure.
NFR-3	Scalability	System is able to handle many users within the given time.
NFR-4	Performance	Load time for the user interface screen will be not more than 2 seconds.
NFR-5	Usability	The user interface can be used by anyone very easily. It is a user-friendly website.
NFR-6	Reliability	Users can access the website all the time without any issues.

5. PROJECT DESIGN

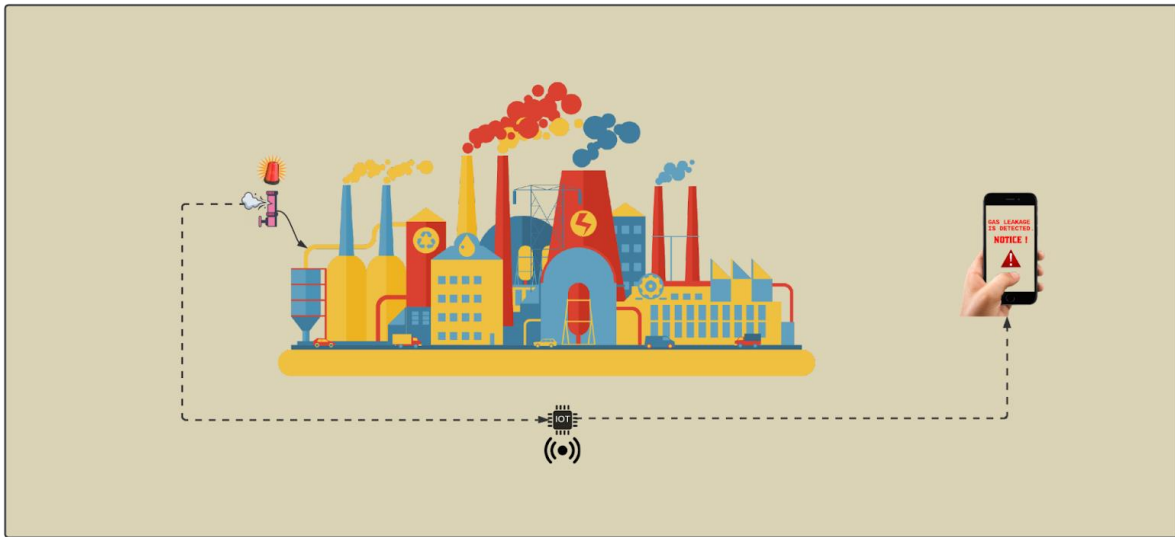
Data Flow Diagram

An information stream graph shows the manner in which data courses through a cycle or framework. It incorporates information data sources and results, information stores, and the different sub processes the information as it travels through them. DFDs are constructed utilising normalised images and documentation to portray different substances and their connections.



Solution Architecture

Arrangement engineering is a training to give ground to programming improvement projects by fitting IT answers for explicit business needs and characterising their utilitarian necessities and phases of execution. It involves numerous subprocesses that draw direction from different endeavour design perspectives.



USER STORIES

The owners of chemical enterprises with a risk of fire accidents are the exclusive target market for this product.

Sprint-1 :

- As a client, I can enrol for the application by entering my email, secret key, and affirming my secret phrase. It has a high need.
- As a client, I will get an affirmation email whenever I have enlisted for the application. It has a high need.
- Enrollment as a client can be affirmed utilising an OTP or check code. It has a low need.

Sprint-2 :

- Users can log into the mobile interface by storing or using the registered login credentials. It has medium priority.
- As a user, I can log into the application by entering email and password. It has medium priority.
- As a user, I should be able to login the profile or status dashboard. It has medium priority.
- Check the Dashboard for routine tests and get the Live updates with Temperature, Gas ppm value, Humid. It has low priority.

Sprint-3 :

- On the occasion of Gas Release the application will spring up notice and caution the Client.
- The Client can extract the Workers and save their lives.

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

Milestone name	Milestone number	Description	Mandatory
Prerequisites	M-01	We will to learn about cloud services and we will install python and required libraries	Yes
Project objectives	M-02	Gain knowledge of Watson Platform, IBM Cloudant DB, web application development and Generating QR codes with the required data	Yes
Create and configure IBM cloud services	M-03	We will create and configure the IBM Cloud services which are being used in completing this project	Yes
Develop a python script To publish And Subscribe To IBM IOT platform	M-04	We will develop the python script to publish the data and subscribe the data from the IBM Watson IOT Platform	Yes
Develop A Web Application Using Node-RED Service	M-05	A Web UI will be created in Node-RED using dashboard nodes available in it	Yes

Ideation phase	M-06	Prepare Literature Survey on the selected Project and Information Gathering, empathy map and ideation	Yes
Project design phase-I	M-07	Prepare Proposed solution, problem-solution fit and Solution Architecture	Yes
Project Design Phase-II	M-08	Prepare Customer journey functional requirements, Data Flow diagram and Technology Architecture	Yes
Project Planning Phase	M-09	Prepare Milestone list, Activity list and Sprint Delivery Plan	Yes
Project Development Phase	M-10	Project Development delivery of Sprint 1, Sprint 2, Sprint 3, Sprint 4	Yes

6.2. Sprint Delivery Schedule

Activity Number	Activity	Sub activity	Assigned member	Status
1	Prerequisites	IBM cloud services	Karan Hariharan Madhankumar Nandhakumar	Completed
		software	Karan Hariharan	Completed
2	Project objectives	Select Activities	Madhankumar nandhakumar	Completed
3	Create and configure IBM cloud services	Create And Configure IBM cloud services	Karan Hariharan	Completed
		Create Node-RED service	Madhan Nandha	
4	Develop a python script To publish And Subscribe To IBM IOT platform	Develop the Python code	Karan hariharan	Completed
		Publish data to the IBM cloud	Karan Hariharan	

5	Develop A Web Application Using Node-RED Service	Develop The Web Application Using Node-RED	Madhan Nandha	Completed
		Use Dashboard Nodes For Creating UI(Web App)	Madhan Nandha	
6	Ideation phase	10.1. Literature Review.	All	Completed
		10.2. Empathy map.	All	Completed
		10.3. Ideation.	All	Completed
7	Project design phase-I	10.1 Proposed Solution	All	Completed
		10.2 Problem solution fit.	All	Completed
		10.3 Solution Architecture.	All	Completed

8	Project Design Phase-II	11.1 Customer journey.	All	Completed
		12.2. Functional requirement.	All	Completed
		12.3. Data flow Diagrams.	All	Completed
		12.4. Technology Architecture.	All	Completed
9	Project Planning Phase	13.1. Prepare milestones and activity lists.	All	Completed
		13.2. Sprint delivery plan	All	Completed

10	Project Development Phase	14.1.Project development- Delivery of Sprint-1.	All	In-progress
		14.2.Project development- Delivery of Sprint-2.	All	In-progress
		14.3. Project development- Delivery of Sprint 3.	All	In-progress
		14.4.Project development- Delivery of Sprint-4.	All	In-progress

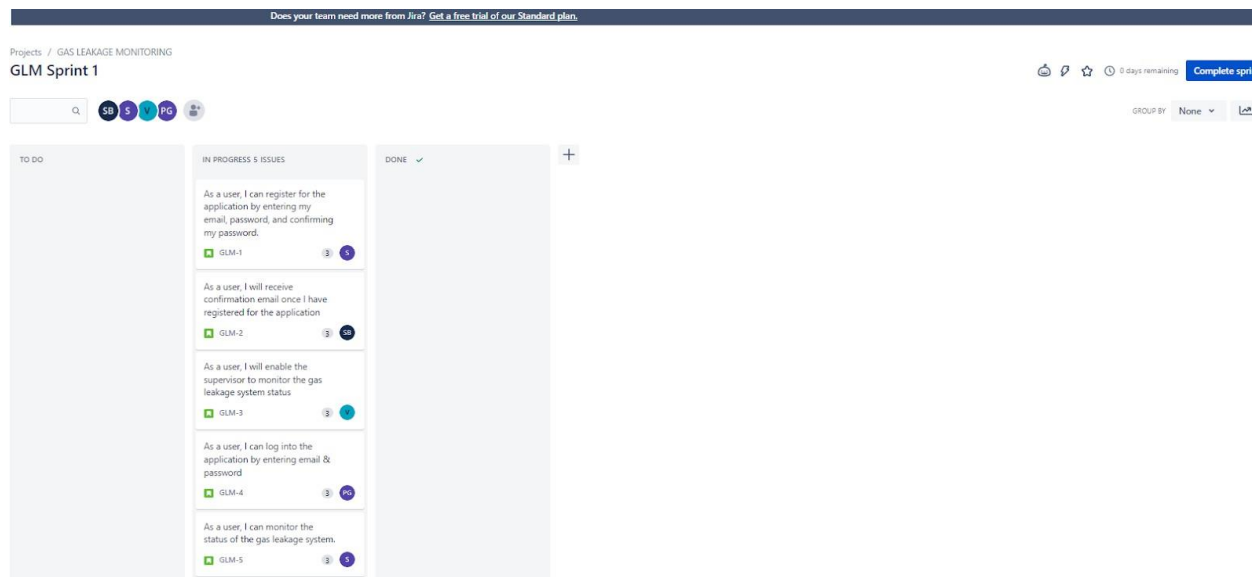
6.3. Reports from JIRA

Burndown chart report :

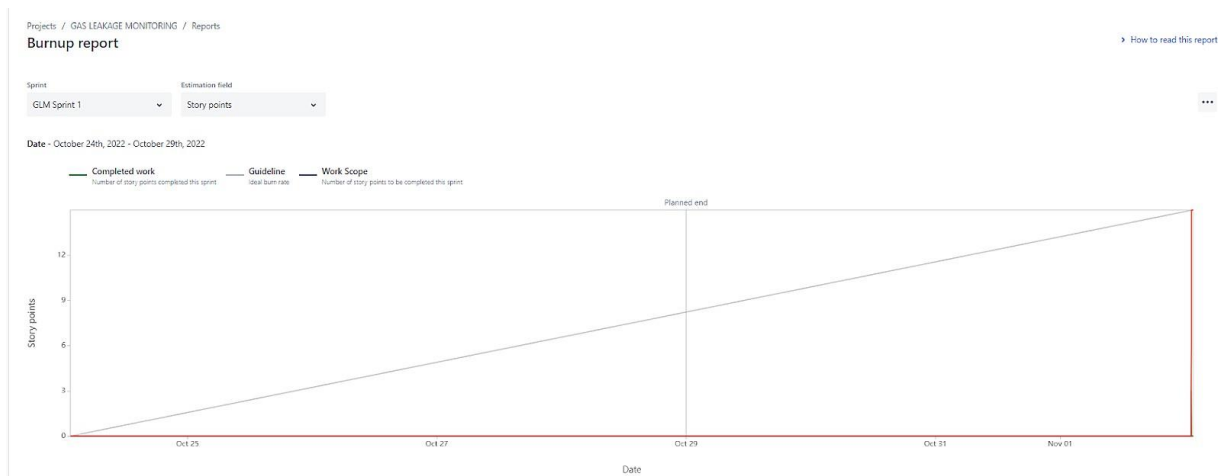
A burndown chart shows the amount of work still to be done about time and work already performed. Agile software development approaches like scrum and jira frequently use it.

Burndown charts, however, can be used for any project with measurable time.

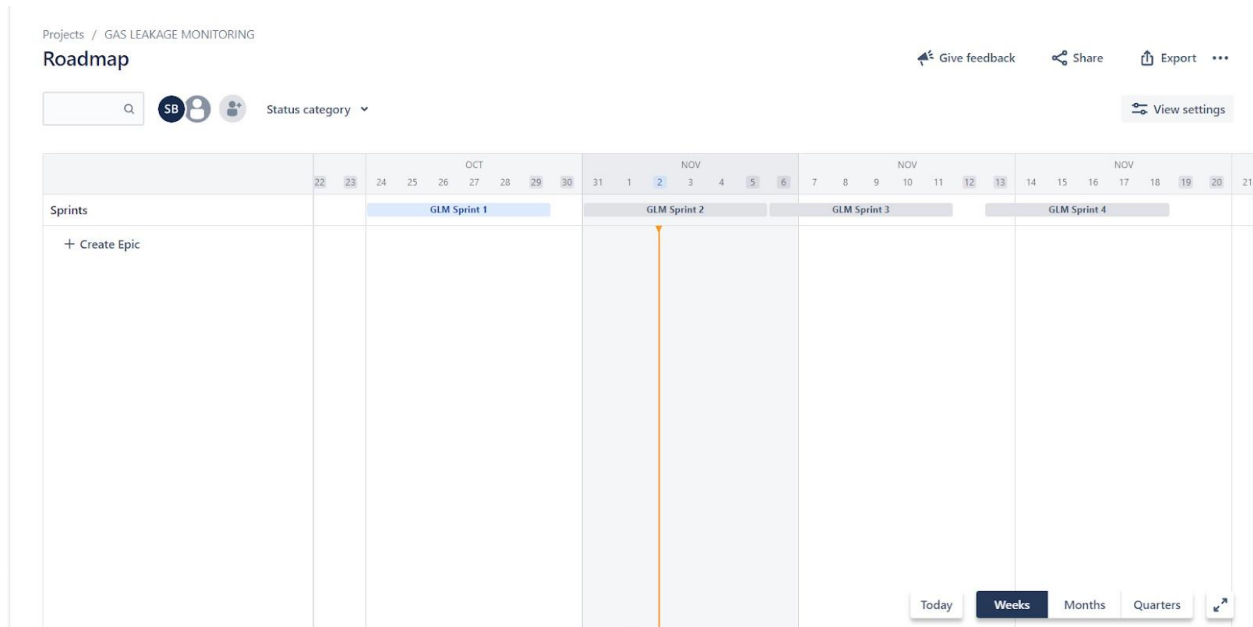
Board



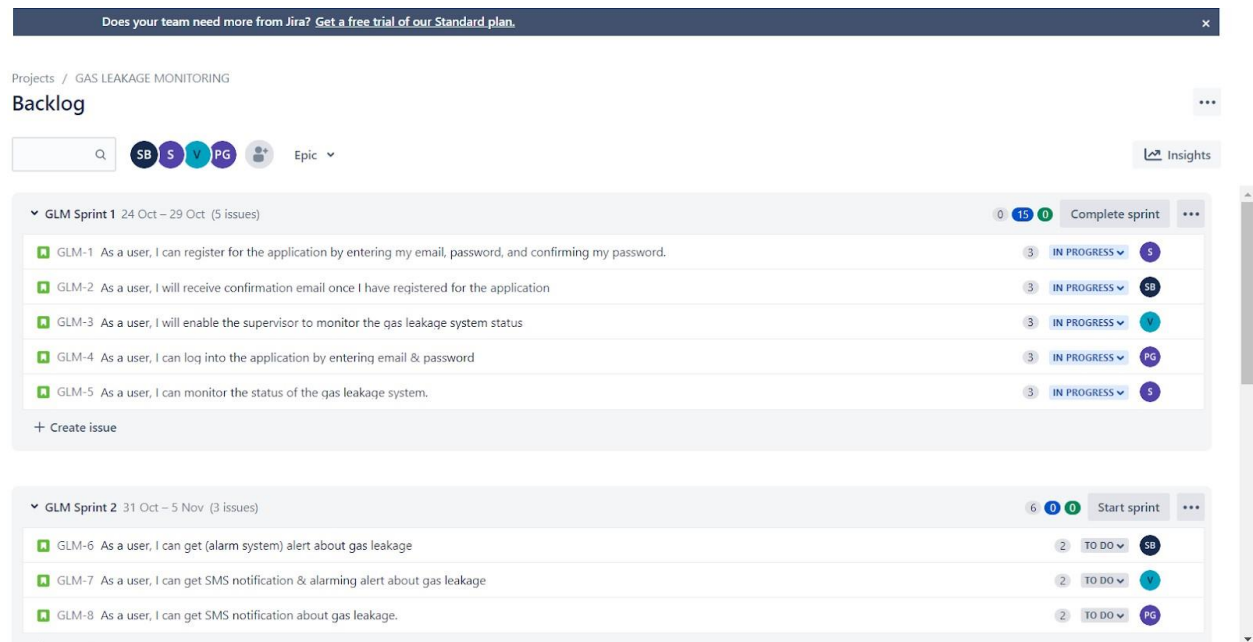
Burnup report



Roadmap



Backlog 1



Backlog 2

Projects / GAS LEAKAGE MONITORING

Backlog

SB S V PG Epic ▾ Insights

GLM Sprint 3 6 Nov – 11 Nov (3 issues)

3

0

0

Start sprint

...

GLM-9 As a user, I can sign-up using Facebook login

1

TO DO ▾

S

GLM-10 As a user, I can request for service in case of any issue with gas leakage monitoring system

1

TO DO ▾

SB

GLM-11 As a user, I can request for scaling up the gas leakage monitoring system

1

TO DO ▾

V

+ Create issue

GLM Sprint 4 13 Nov – 18 Nov (1 issue)

1

0

0

Start sprint

...

GLM-12 As a user, I can sign-up using Facebook login.

1

TO DO ▾

PG

+ Create issue

Backlog (0 issues)

0

0

0

Create sprint

26

7. CODING & SOLUTIONING

Feature 1

User interface feature:

Here the user gets an interface where the user can enter into a details form to check the current status of Gas Leakages. It is done through html, javascript and css languages.

Source Code

```
<!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" />

    <title>Gas Leakage Monitoring and Alerting System</title>

    <link rel="stylesheet" href="style.css" />

  </head>

  <body>

    <main>

      <div class="box">

        <div class="inner-box">

          <div class="forms-wrap">

            <form action="index.html" autocomplete="off" class="sign-in-form">
```

<div class="logo">

<h4>Gas Leakage Monitoring and Alerting System for
Industries
PNT2022TMIDO8134</h4>

</div>

<div class="heading">

<h2>Welcome Back</h2>

<h6>Not registred yet?</h6>

Sign up

<a href="https://ibm-nalaiyathiran.s3.ams03.cloud- object-
storage.appdomain.cloud/gmaps.html" class="toggle">
Skip
SignIn

</div>

<div class="actual-form">

<div class="input-wrap">

<input

type="text" minlength="4"

class="input-field"

autocomplete="off"

required

/>

```
<label>Name</label>
```

```
</div>
```

```
<div class="input-wrap">
```

```
<input
```

```
  type="password"
```

```
  minlength="4"
```

```
  class="input-field"
```

```
  autocomplete="off"
```

```
  required
```

```
<label>Password</label>
```

```
</div>
```

```
<input type="submit" value="Sign In" class="sign-btn" />
```

```
<link rel="stylesheet" type="text/css"
```

```
href="//fonts.googleapis.com/css?family=Open+Sans" />
```

```
<p class="text">
```

Forgotten your password or you login details?

[Get help](#) signing in

</p>

</div>

</form>

<form action="index.html" autocomplete="off" class="sign-up-form">

<div class="logo">

<h4>Gas Leakage Monitoring and Alerting System for Industries

PNT2022TMIDO8020</h4>

</div>

<div class="heading">

<h2>Get Started</h2>

<h6>Already have an account?</h6>

Sign in

</div>

<link rel="stylesheet" type="text/css"

href="//fonts.googleapis.com/css?family=Open+Sans" />

<div class="actual-form">

<div class="input-wrap">

<input

```
        type="text" minlength="4"

        class="input-field"

        autocomplete="off"

        required

    />

    <label>Name</label>

</div>
```

```
<div class="input-wrap">

    <input

        type="email" class="input-

        field" autocomplete="off"

        required

    />

    <label>Email</label>

</div>
```

```
<div class="input-wrap">

    <input

        type="password"
```



```

        minlength="4"

        class="input-field"

        autocomplete="off"

        required

    />

    <label>Password</label>

</div>

<input type="submit" value="Sign Up" class="sign-btn" />

<p class="text">

    By signing up, I agree to the

    <a href="#">Terms of Services</a> and

    <a href="#">Privacy Policy</a>

</p>

</div>

</form>

</div>

<div class="carousel">

    <div class="images-wrapper">

```

```






</div>

<div class="text-slider">

  <div class="text-wrap">

    <div class="text-group">

      <h2>Create your own industrial ID</h2>

      <h2>overview of our application</h2>

      <h2>checking status</h2>

    </div>

  </div>

  <div class="bullets">

    <span class="active" data-value="1"></span>

    <span data-value="2"></span>

    <span data-value="3"></span>

  </div>

</div>

</div>

</div>
```

</div>

```
</main>

<!-- Javascript file -->

<script src="app.js"></script>

</body>

</html>
```

app.js

```
const inputs = document.querySelectorAll(".input-field"); const toggle_btn =
document.querySelectorAll(".toggle"); const main = document.querySelector("main");
const bullets = document.querySelectorAll(".bullets span"); const images =
document.querySelectorAll(".image"); inputs.forEach((inp) => {

  inp.addEventListener("focus", () => {

    inp.classList.add("active");

  });

  inp.addEventListener("blur", () => { if (inp.value !== "")

    return; inp.classList.remove("active");

  });

});

toggle_btn.forEach((btn) => {
```

```

btn.addEventListener("click", () => {

    main.classList.toggle("sign-up-mode");

});

});

function moveSlider() {

    let index = this.dataset.value;

    let currentImage = document.querySelector(`.img-${index}`); images.forEach((img) =>

    img.classList.remove("show")); currentImage.classList.add("show");

    const textSlider = document.querySelector(".text-group"); textSlider.style.transform = `translateY(${-(index -

    1) * 2.2}rem)`;

    bullets.forEach((bull) => bull.classList.remove("active")); this.classList.add("active");

}

bullets.forEach((bullet) => { bullet.addEventListener("click",

    moveSlider);

});

```

style.css

```
@import url("https://fonts.googleapis.com/css2?family=Poppins:wght@200;300;400;500;600;700;800&display=swap");
```

```
*,
```

```
*::before,
```

```
*::after { padding:
```

```
0;
```

```
margin: 0;
```

```
box-sizing: border-box;
```

```
}
```

```
body,
```

```
input {
```

```
font-family: "Poppins", sans-serif;
```

```
}
```

```
main {
```

```
width: 100%;
```

```
min-height: 100vh; overflow: hidden;
```

```
background-color: #ff8c6b;
```

```
padding: 2rem;

display: flex;

align-items: center; justify-content:

center;

}

.box {

position: relative; width:

100%;

max-width: 1020px; height:

640px; background-color: #fff;

border-radius: 3.3rem;

box-shadow: 0 60px 40px -30px rgba(0, 0, 0, 0.27);

}

.inner-box { position: absolute;

width: calc(100% - 4.1rem); height:

calc(100% - 4.1rem); top: 50%;

left: 50%;
```

```
    transform: translate(-50%, -50%);  
  }
```

```
.forms-wrap { position:  
  
    absolute; height: 100%;  
  
    width: 45%;  
  
    top: 0;  
  
    left: 0; display: grid;  
  
    grid-template-columns: 1fr; grid-template-  
  
    rows: 1fr; transition: 0.8s ease-in-out;  
}
```

```
form {  
  
    max-width: 260px;  
  
    width: 100%;  
  
    margin: 0 auto;  
  
    height: 100%;  
  
    display: flex;  
  
    flex-direction: column;
```



```
justify-content: space-evenly; grid-column:

1 / 2;

grid-row: 1 / 2;

transition: opacity 0.02s 0.4s;

}
```

```
form.sign-up-form { opacity:

0;

pointer-events: none;

}
```

```
.logo { display: flex;

align-items: center;

}
```

```
.logo img { width:

27px;

margin-right: 0.3rem;

}
```

```
.logo h4 {
```

```
font-size: 1.1rem; margin-top: -  
  
9px; letter-spacing: -0.5px; color:  
  
#151111;  
  
}
```

```
.heading h2 {  
  
font-size: 2.1rem; font-  
  
weight: 600;  
  
color: #151111;  
  
}
```

```
.heading h6 { color:  
  
#bababa; font-weight:  
  
400;  
  
font-size: 0.75rem; display:  
  
inline;  
  
}
```

```
.toggle {  
  
color: #151111;  
  
  
text-decoration: none;
```

```
font-size: 0.75rem; font-  
  
weight: 500; transition:  
  
0.3s;  
}
```

```
.toggle:hover { color:  
  
#8371fd;  
}
```

```
.input-wrap { position: relative;  
  
height: 37px;  
  
margin-bottom: 2rem;  
}
```

```
.input-field { position:  
  
absolute; width: 100%;  
  
height: 100%;  
  
background: none;  
  
border: none; outline:  
  
none;
```

```
border-bottom: 1px solid #bbb; padding: 0;

font-size: 0.95rem; color:

#151111; transition: 0.4s;

}
```

```
label {

    position: absolute; left: 0;

    top: 50%;

    transform: translateY(-50%); font-size:

    0.95rem;

    color: #bbb;

    pointer-events: none;

    transition: 0.4s;

}
```

```
.input-field.active {

    border-bottom-color: #151111;

}
```

```
.input-field.active + label { font-size:
```

```
0.75rem;
```

```
top: -2px;
```

```
}
```

```
.sign-btn {
```

```
display: inline-block; width:
```

```
100%;
```

```
height: 43px;
```

```
background-color: #151111; color:
```

```
#fff;
```

```
border: none; cursor:
```

```
pointer;
```

```
border-radius: 0.8rem; font-
```

```
size: 0.8rem; margin-bottom:
```

```
2rem; transition: 0.3s;
```

```
}
```

```
.sign-btn:hover {
```

```
background-color: #8371fd;
```

```
}
```

```

.text { color: #bbb;

        font-size: 0.7rem;

}

.text a { color: #bbb;

        transition: 0.3s;

}

.text a:hover { color:

        #8371fd;

}

main.sign-up-mode form.sign-in-form { opacity: 0;

        pointer-events: none;

}

main.sign-up-mode form.sign-up-form { opacity: 1;

        pointer-events: all;

}

main.sign-up-mode .forms-wrap { left: 55%;

```

```
}
```

```
main.sign-up-mode .carousel { left: 0%;
```

```
}
```

```
.carousel { position: absolute;
```

```
height: 100%;
```

```
width: 55%;
```

```
left: 45%;
```

```
top: 0;
```

```
background-color: #ffe0d2; border-
```

```
radius: 2rem; display: grid;
```

```
grid-template-rows: auto 1fr; padding-
```

```
bottom: 2rem; overflow: hidden;
```

```
transition: 0.8s ease-in-out;
```

```
}
```

```
.images-wrapper {
```

```
display: grid;

grid-template-columns: 1fr; grid-

template-rows: 1fr;

}
```

```
.image { width:

100%;

grid-column: 1/2; grid-

row: 1/2; opacity: 0;

transition: opacity 0.3s, transform 0.5s;

}
```

```
.img-1 {

transform: scale(0.4, 0.5);

}
```

```
.img-2 {

transform: scale(0.4, 0.5);

}
```

```
.img-3 {

transform: scale(0.4, 0.5);

}
```



```
.image.show { opacity: 1;

    transform: none;

}

.text-slider { display:

    flex;

    align-items: center; justify-content:

    center; flex-direction: column;

}

.text-wrap {

    max-height: 2.2rem; overflow:

    hidden; margin-bottom: 2.5rem;

}

.text-group { display:

    flex;

    flex-direction: column; text-align:

    center; transform: translateY(0);

    transition: 0.5s;

}
```

```
.text-group h2 {  
  
    line-height: 2.2rem; font-  
  
    weight: 600; font-size:  
  
    1.6rem;  
  
}  
  
.bullets { display: flex;  
  
    align-items: center; justify-content:  
  
    center;  
  
}  
  
.bullets span { display:  
  
    block; width: 0.5rem;  
  
    height: 0.5rem;  
  
    background-color: #aaa; margin:  
  
    0 0.25rem;  
  
    border-radius: 50%; cursor:  
  
    pointer; transition: 0.3s;  
  
}  
  
.bullets span.active {
```

```
width: 1.1rem;

background-color: #151111; border-

radius: 1rem;

}
```

```
@media (max-width: 850px) {

  .box {

    height: auto;

    max-width: 550px;

    overflow: hidden;

  }
```

```
.inner-box { position: static;

  transform: none; width:

  revert; height: revert;

  padding: 2rem;

}
```

```
.forms-wrap { position:

  revert;
```

```
width: 100%;
```

```
height: auto;
```

```
}
```

```
form {
```

```
max-width: revert;
```

```
padding: 1.5rem 2.5rem 2rem;
```

```
transition: transform 0.8s ease-in-out, opacity 0.45s linear;
```

```
}
```

```
.heading { margin: 2rem
```

```
0;
```

```
}
```

```
form.sign-up-form { transform:
```

```
translateX(100%);
```

```
}
```

```
main.sign-up-mode form.sign-in-form { transform:
```

```
translateX(-100%);
```

```
}
```

```
main.sign-up-mode form.sign-up-form { transform:
```

```
translateX(0%);
```

```

}

.carousel { position: revert;

    height: auto; width: 100%;

    padding: 3rem 2rem;

    display: flex;

}

.images-wrapper {

    display: none;

}

.text-slider { width:

    100%;

}

}

@media (max-width: 530px) { main {

    padding: 1rem;

}

```

```
.box {  
  
    border-radius: 2rem;  
  
}  
  
.inner-box { padding:  
  
    1rem;  
  
}  
  
.carousel {  
  
    padding: 1.5rem 1rem; border-  
  
    radius: 1.6rem;  
  
}  
  
.text-wrap {  
  
    margin-bottom: 1rem;  
  
}  
  
.text-group h2 { font-size:  
  
    1.2rem;  
  
}  
  
}
```

```
.box {
```

```
    border-radius: 2rem;
```

```
}
```

```
.inner-box { padding:
```

```
    1rem;
```

```
}
```

```
.carousel {
```

```
    padding: 1.5rem 1rem; border-
```

```
    radius: 1.6rem;
```

```
}
```

```
.text-wrap {
```

```
    margin-bottom: 1rem;
```

```
}
```

```
.text-group h2 { font-size: 1
```

SIGN IN DASHBOARD

Feature 2

24x7 Accessible:

The Dashboard we configured can be accessible by 24*7 and user can access any time and from any where and monitor the status in the Dashboard.

In this Dashboard the cleint can check the live status of

- Temperature
- Humid
- Gas ppm

And lead his team further.

Gas Leakage Monitoring A

Set Gas Level:

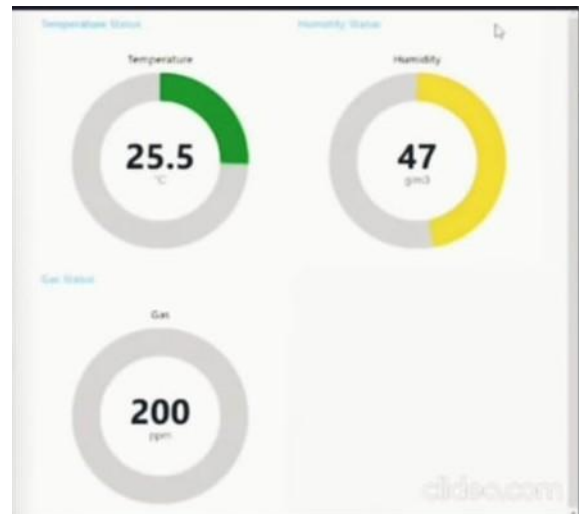
Current Gas Level: 992.00 ppm

Set Temperature:

Current Temperature: 3.00 °C

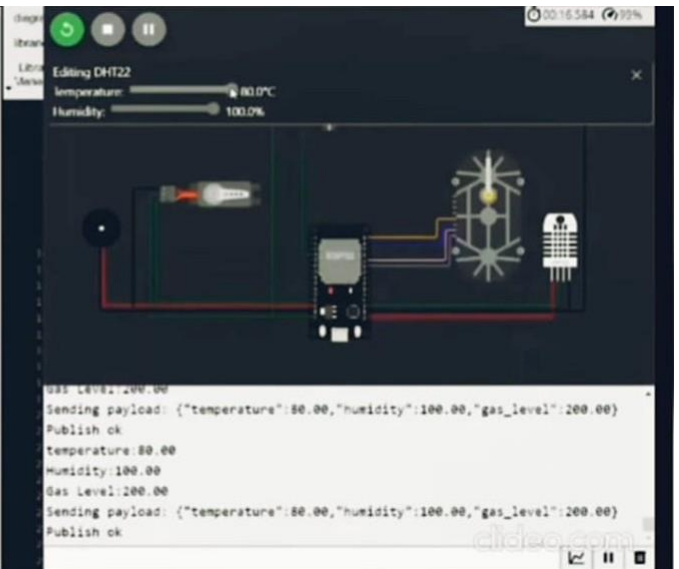
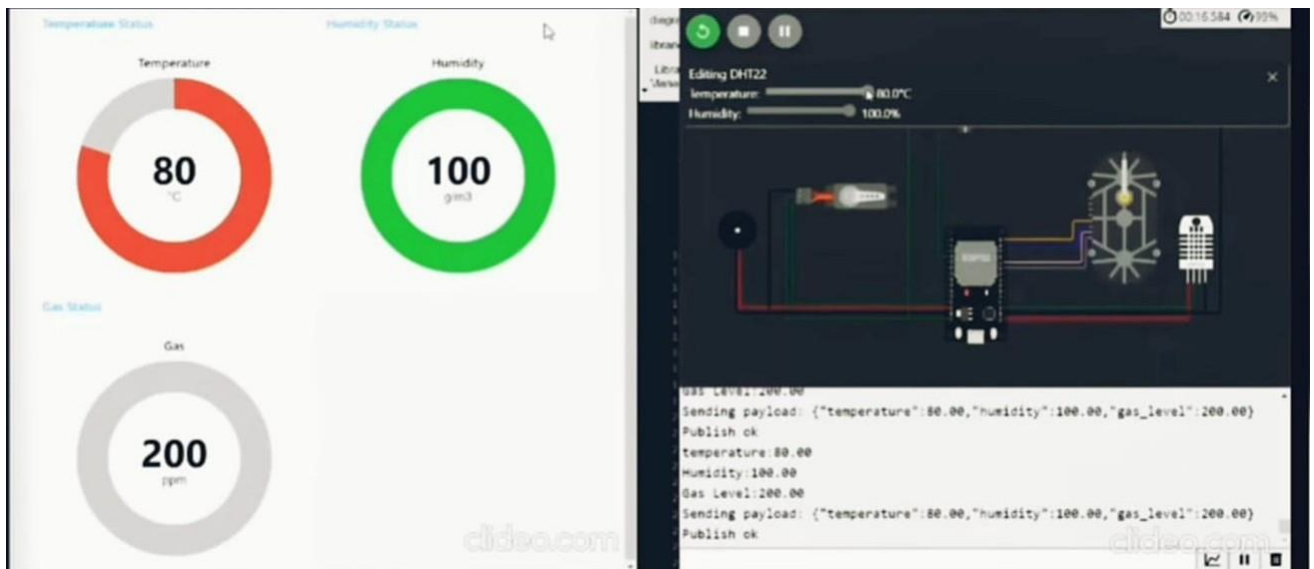
Set Humidity:

Current Humidity: 36.00 %



8. TESTING

We need to check whether the person is eligible or not based on details entered. If user is eligible he/she will be displayed to the Dashboard of Status.



9. RESULTS

Performance Metrics

With this user interface users can easily showcase the status and alert of the gas leakage.



ADVANTAGES & DISADVANTAGES

Advantages

- Comparing genuine alarm to existing
- Reusability
- Rapid response

Disadvantages

- Costly Investment
- Internet access is required.
- Gas mixture cannot be identified.

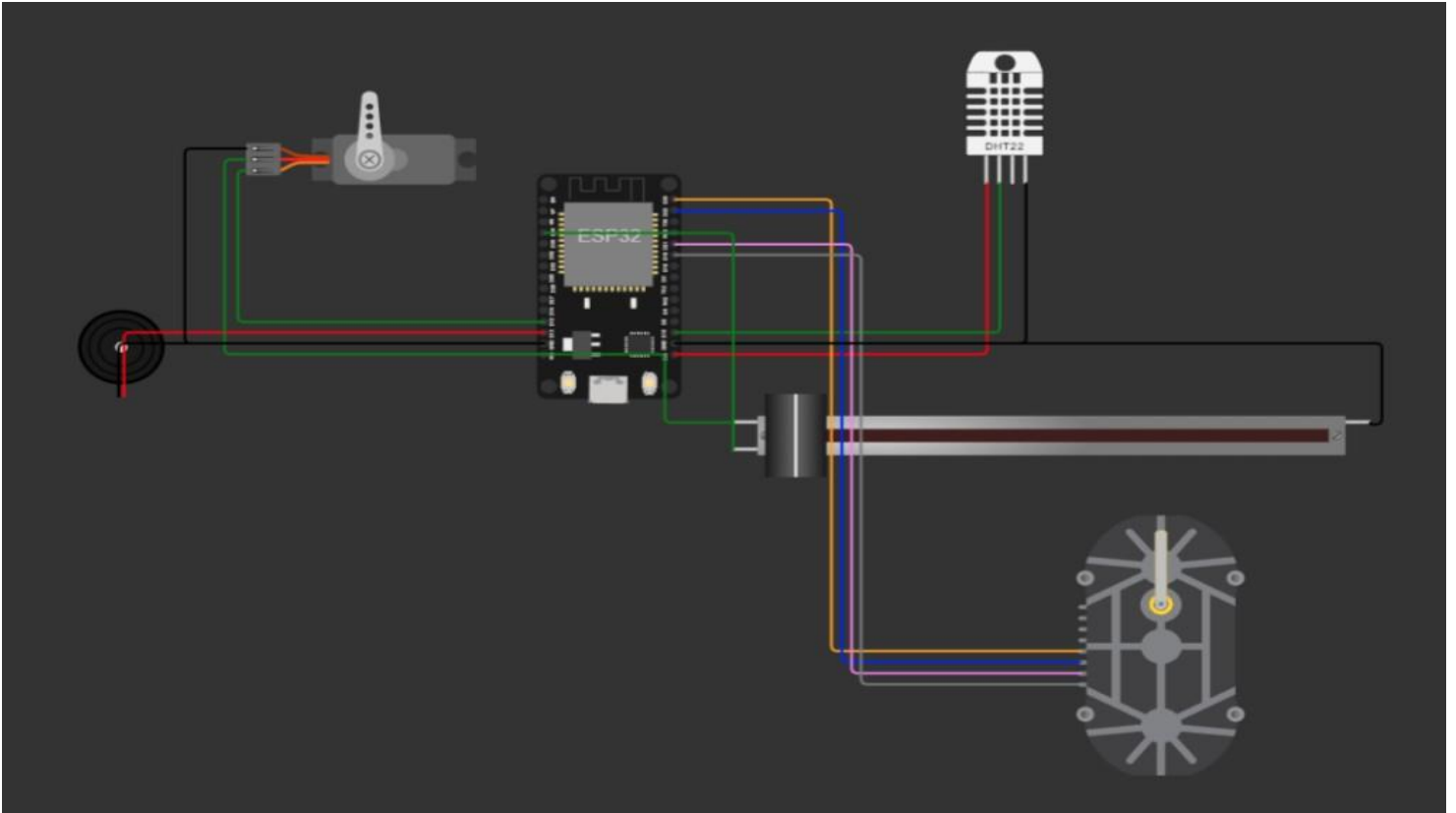
CONCLUSION

It has become more efficient, more applicable to today's applications, and smarter. The work has presented solutions to several problems and issues that have not been addressed in previous work. The principle of operation of an IOT based gas leakage and monitoring system was demonstrated by operating the Arduino UNO attached to an embedded system with required input and output gas levels with the help of gas sensors. This results in a more efficient operation because it is connected to a common web page specially built to notify the responsible authority and reduces the stress of constant monitoring. The choice of using a real time gas leakage monitoring and sensing the output levels of gas has been clearly observed with the help of this system.

FUTURE WORK

Voice feedback systems can be included in GSM based harmful GAS leakage detection and alerting systems. Users will get intimation through pre-recorded voice messages like the weight of the gas Cylinder is ABC kg. In the future, some other wireless technology can be used to sense gases and can be helpful for control of gas leakage. A robot has been utilized in trading humans for taking care of different errands in a risky and perilous working environment where human life may be in danger. A portable gas detecting robot can be built to detect the spillage of gas through pipelines as the robot can proceed onward along a track which is arranged along the length of the pipeline.

NodeMCU (ESP32) Configuration



Mobile Application

Mobile App



Google Maps API



SOURCE CODE

IBM Watson IOT Platform

```
#IBM Watson IOT Platform #pip

install wiotp-sdk import

wiotp.sdk.device import time

import random

myConfig = {

    "identity": {

        "orgId": "hj5fmy",

        "typeId": "NodeMCU",

        "deviceId": "12345"

    },

    "auth": {

        "token": "12345678"

    }

}

def myCommandCallback(cmd):

    print("Message received from IBM IoT Platform: %s" %

cmd.data['command'])
```



```

        m=cmd.data['command']

client = wiotp.sdk.device.DeviceClient(config=myConfig, logHandlers=None)

client.connect() while
True:

    temp=random.randint(-20,125) hum=random.randint(0,100)

    myData={'temperature':temp, 'humidity':hum}

    client.publishEvent(eventId="status", msgFormat="json", data=myData, qos=0,
onPublish=None)

    print("Published data Successfully: %s", myData)

    client.commandCallback = myCommandCallback time.sleep(2)

client.disconnect()

```

Wokwi (NodeMCU .ino code)

```
#NodeMCU ESP32 Code
#TEAM ID : PNT2022TMID08134
#include <ESP32Servo.h> #include
<WiFi.h> #include <Stepper.h>
#include <PubSubClient.h> #include
<DHTesp.h> #define DHTPIN 15
#define GAS_LEVER 34
#define buzzer 13
#define LED 5
const int servoPin = 12; Servo valve;
DHTesp dhtsensor;
Stepper stepper(1000, 19,21,22,23);
void callback(char* subscribetopic, byte* payload, unsigned int payloadLength);

#define ORG "8o05yw"
#define DEVICE_TYPE "esp32" #define
DEVICE_ID"sam123" #define TOKEN
"123456789"

String data3; float
h, t, g; int pos=0;
boolean valve_open=true;

char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
```

```
char publishTopic[] = "iot-2/evt/Data/fmt/json"; char subscribetopic[] = "iot-2/cmd/test/fmt/String"; char authMethod[] = "use-token-auth";  
char token[] = TOKEN;  
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
```

```
WiFiClient wifiClient;  
PubSubClient client(server, 1883, callback ,wifiClient); void setup()  
{  
    Serial.begin(115200);  
    dhtsensor.setup(DHTPIN,DHTesp::DHT22);  
    stepper.setSpeed(100); valve.attach(servoPin);  
    pinMode(GAS_LEVER, INPUT);  
    pinMode(buzzer,OUTPUT);  
    delay(10); Serial.println();  
    wificonnect(); mqttconnect();  
    valve.write(90);  
  
}  
  
void loop()  
{  
    TempAndHumidity data=dhtsensor.getTempAndHumidity();  
    t=data.temperature;  
    h=data.humidity;  
    g=map(int(analogRead(GAS_LEVER)), 0, 4095, 200, 2000);
```

```

Serial.print("temperature:");
Serial.println(t); Serial.print("Humidity:");
Serial.println(h); Serial.print("Gas Level:");
Serial.println(g);

if(g>500){ tone(buzzer, 1000);
    stepper.step(1000);
    valve.write(180);
}
else{
    valve.write(90);
    noTone(buzzer);
}
PublishData(t, h, g);
delay(1000);
if (!client.loop()) {
    mqttconnect();
}
}

void PublishData(float temp, float humid, float gas_level) { mqttconnect();
    String payload = "{\"temperature\":"; payload +=
temp;
    payload += ", \"humidity\":"; payload +=
humid;

```

```

payload += "," "\"gas_level\":"; payload +=
gas_level;
payload += "}";

Serial.print("Sending payload: ");
Serial.println(payload);

if (client.publish(publishTopic, (char*) payload.c_str())) { Serial.println("Publish ok");
} else {
    Serial.println("Publish failed");
}

}

void mqttconnect() {
    if (!client.connected()) { Serial.print("Reconnecting client to ");
        Serial.println(server);
        while (!client.connect(clientId, authMethod, token)) { Serial.print(".");
            delay(500);
        }

        initManagedDevice();
        Serial.println();
    }
}

void wificonnect()
{
    Serial.println();

```

```

Serial.print("Connecting to ");

WiFi.begin("Wokwi-GUEST", "", 6);
while (WiFi.status() != WL_CONNECTED) { delay(500);
    Serial.print(".");
}
Serial.println(""); Serial.println("WiFi
connected"); Serial.println("IP address: ");
Serial.println(WiFi.localIP());
}

void initManagedDevice() {
    if (client.subscribe(subscribetopic)) {
        Serial.println((subscribetopic)); Serial.println("subscribe
to cmd OK");
    } else {
        Serial.println("subscribe to cmd FAILED");
    }-
}

void callback(char* subscribetopic, byte* payload, unsigned int payloadLength)
{

    Serial.print("callback invoked for topic: ");
    Serial.println(subscribetopic);
    for (int i = 0; i < payloadLength; i++) { data3 +=
        (char)payload[i];
    }

    Serial.println("data: "+ data3); data3="";
}

```

GitHub Repository Link : <https://github.com/IBM-EPBL/IBM-Project-16053-1659606982>

Project Demo Link : https://drive.google.com/file/d/1wpT_GIBo-2AAAn9wbkOSN2RNkvS2G_2DT/view?usp=sharing

