PROJECT REPORT

ELECTRONICS AND COMMUNICATION ENGINEERING

K.L.N. COLLEGE OF ENGINEERING

POTTAPALAYM

INDUSTRY – SPECIFIC INTELLIGENT FIRE MANAGEMENT SYSTEM

TEAM ID :PNT2022TMID11542

Submitted By

VENKATESH B G (910619106066)

SANTHOSE M (910619106048)

SASI KUMAR K (910619106052)

SHARVESH BARATH(910619106055)

TEAM MENTOR:

-DR.N.JANAKIRAMAN

PROF/ECE

INDUSTRY MENTOR:

-SANTHOSI

EVALUATOR:

-DR.P.KARPAGAVALLI

ASP/ECE

BONAFIDE CERTIFICATE

Certified that this project report **“……….TITLE OF THE PROJECT… ……………………….. ”** is the Bonafide work of “**NAME OF THE CANDIDATE(reg number), NAME OF THE CANDIDATE (reg number), NAME OF THE CANDIDATE (reg number)”,** who carried out the project work under our supervision.

SIGNATURE

FACULTY MENTOR

Name

Designation

ELECTRONICS AND COMMUNICATION ENGINEERING

K.L.N COLLEGE OF ENGINEERING

SIGNATURE

FACULTY EVALUATOR

Name

Designation

ELECTRONICS AND COMMUNICATION ENGINEERING

K.L.N COLLEGE OF ENGINEERING

SIGNATURE

DR.V.KEJALAKSHMI

HEAD OF THE DEPARTMENT

ELECTRONICS AND COMMUNICATION ENGINEERING K.L.N COLLEGE OF ENGINEERING

LIST OF FIGURES

LIST OF TABLES

CONTENTS

1.INTRODUCTION

1.1 ProjectOverview

- ThesmartfiremanagementsystemincludesaGassensor,Flame sensorandtemperaturesensorstodetectanychangesintheenvironment.
- BasedonthetemperaturereadingsandifanyGasesarepresenttheexhaustfansarepoweredON.
- Ifanyflameisdetectedthesprinklerswillbeswitchedonautomatically Emergencyalertsare notifiedto theauthorities and Firestation.

1.2 Purpose

Thepurposeofthesystemis:

- Topreventlifelosses,assestsdamageanduncontrollablespreadoffire.
- Toensurethesafetyofworkersandalertthemanagerandfiredepartment.
- Tonottorecklesslyendangerthelifeofthefireworkers.Thiscanbedonebytakingthecontrolmeasures automatically.

2. LITERATURESURVEY

2.1 Existingproblem

The existing problems of the system are:

- **Cost of ownership:** The fire management system should be cost effective. In average,the fire management is expected to last 10 years. The biggest problem is when the system cannot be maintained any longer due to component non-availabililty or due to being unsupported by the manufacturer.
- **Structuralchanges:**Thestructureofthehospitalchangesovertime.Thefire Management system should be easily able to upgrade and adaptable to the changing structure.
- **Evacuationandfirestrategy**:The alert and the control measures are taken immediately ,so that the building can be completely evacuated.
- **Systemperformancechangeswithinspecificenvironments:**The industry will have Unique or specified conditionat sometime. The major problem caused is the false fire alarm.

2.2 References

[1] Gaziweldesyase,BahtaG/meskel,MekonenAbreha,SolomonBaynes,“GSMBasedFireandSmokeDetectionandPreventionSystem”,on08/10/2010,Adigrat,Tigray,Ethiopia.

[2]MayZawTun,HtayMyint,“ArduinobasedFireDetectionandAlarmSystemUsingSmokeSensor”,Volume6, Issue4,on April–2020, Myanmar.

[3]NitinGalugade,MaheshJakka,DevikaNair,MadhurGawas,“Fire MonitoringandControllingSystembased onIot”, 2020,Mumbai, India.

2.3 ProblemStatementDefinition

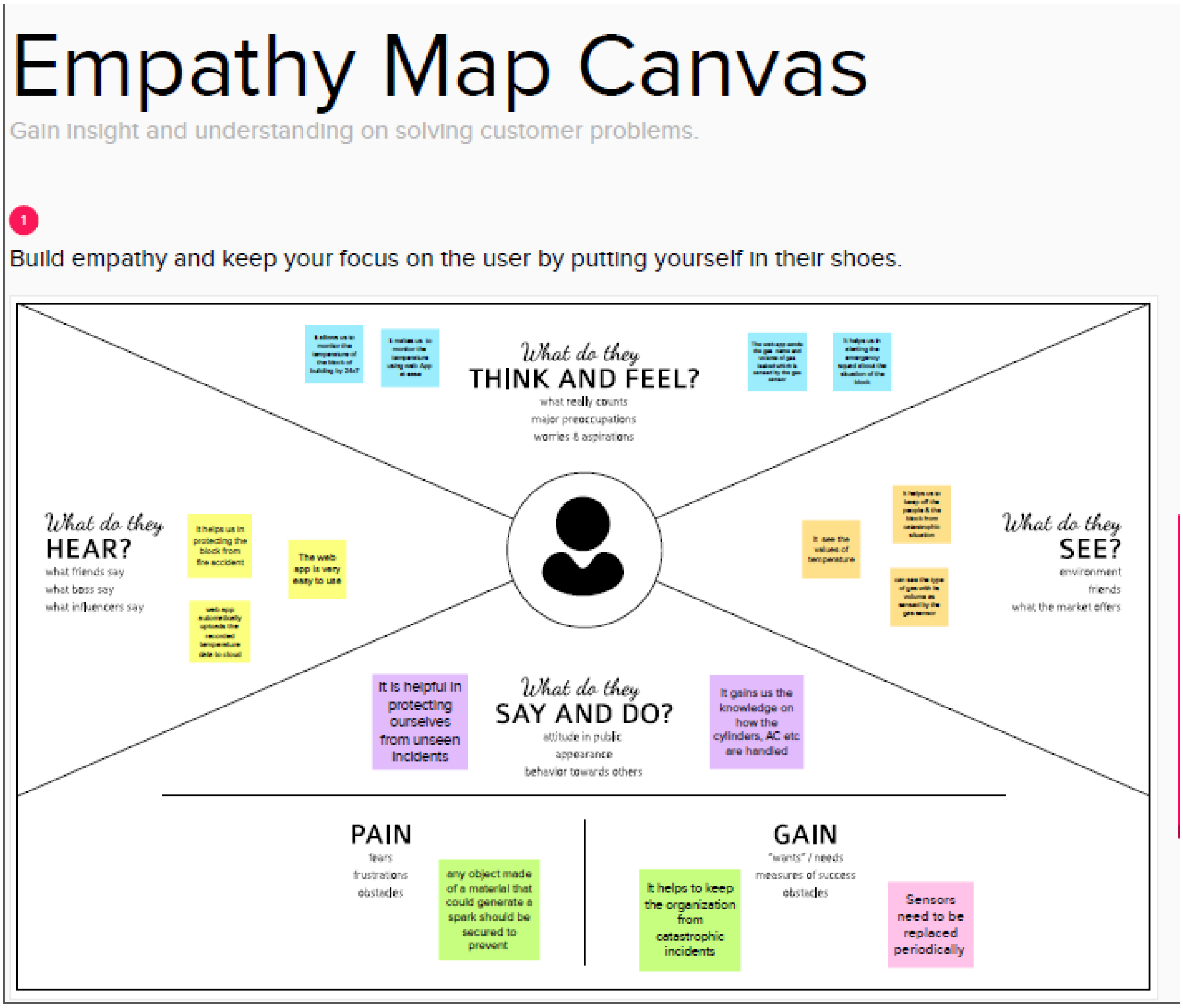
Background: Fire is the rapid oxidation of a material in the exothermic chemical process ofcombustion, releasing heat, light and various reaction products. Although it’s a natural process,it can lead to great destruction. On average, everyday 35 people killed due to Fire-relatedaccidents in the five years between 2016 and 2020, according to a report by Accidental DeathsandSuicidesinIndia(ADSI),maintained by the National Crime Records Bureau. Fire is one ofthe major concerns when analyzing the potential risks on the building. Industrial Fires andExplosions cost companies and governments billions of Rupees every year apart from the loss oflife,whichcan’tbedescribedinmonetaryterms.TheseFiresnotonlyresultsonlyinhugelossof Lives and Property but also disrupt production in the Industry. The Nilflisk says that the fivemajor causes of industrial fires and explosions are Combustible dust, hot works, Flammableliquidsand gasses, equipmentand machinery andElectrical hazards.

Objective:

Based on the temperature readingsand if any Gasses are present the exhaust fans should be powered ON automatically to replacecontaminated and stale air with fresh, healthy air. If any flame is detected the sprinklers will beswitched on automatically. Emergency alerts are notified to the authorities and Fire station. Sothatthe authoritiesandFire Fighterscan controlthe situation.

3 .IDEATION AND PROPOSED SOLUTION

3.1 Empathy Map Canvas



Empathy Map Canvas

Gain insight and understanding on solving customer problems.

1

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

It allows us to monitor the temperature of the block of building by 24x7

It makes us to monitor the temperature using web App at ease

The web app sends the gas name and volume of gas leaked which is sensed by the gas sensor

It helps us in alerting the emergency squad about the situation of the block

What do they HEAR?
what friends say
what boss say
what influencers say

It helps us in protecting the block from fire accident

The web app is very easy to use

web app automatically uploads the recorded temperature data to cloud

What do they SEE?
environment
friends
what the market offers

It see the values of temperature

It helps us to keep all the people & the block from catastrophic situation

can see the type of gas with its volume as sensed by the gas sensor

What do they SAY AND DO?
attitude in public
appearance
behavior towards others

It is helpful in protecting ourselves from unseen incidents

It gains us the knowledge on how the cylinders, AC etc are handled

PAIN
fears
frustrations
obstacles

any object made of a material that could generate a spark should be secured to prevent

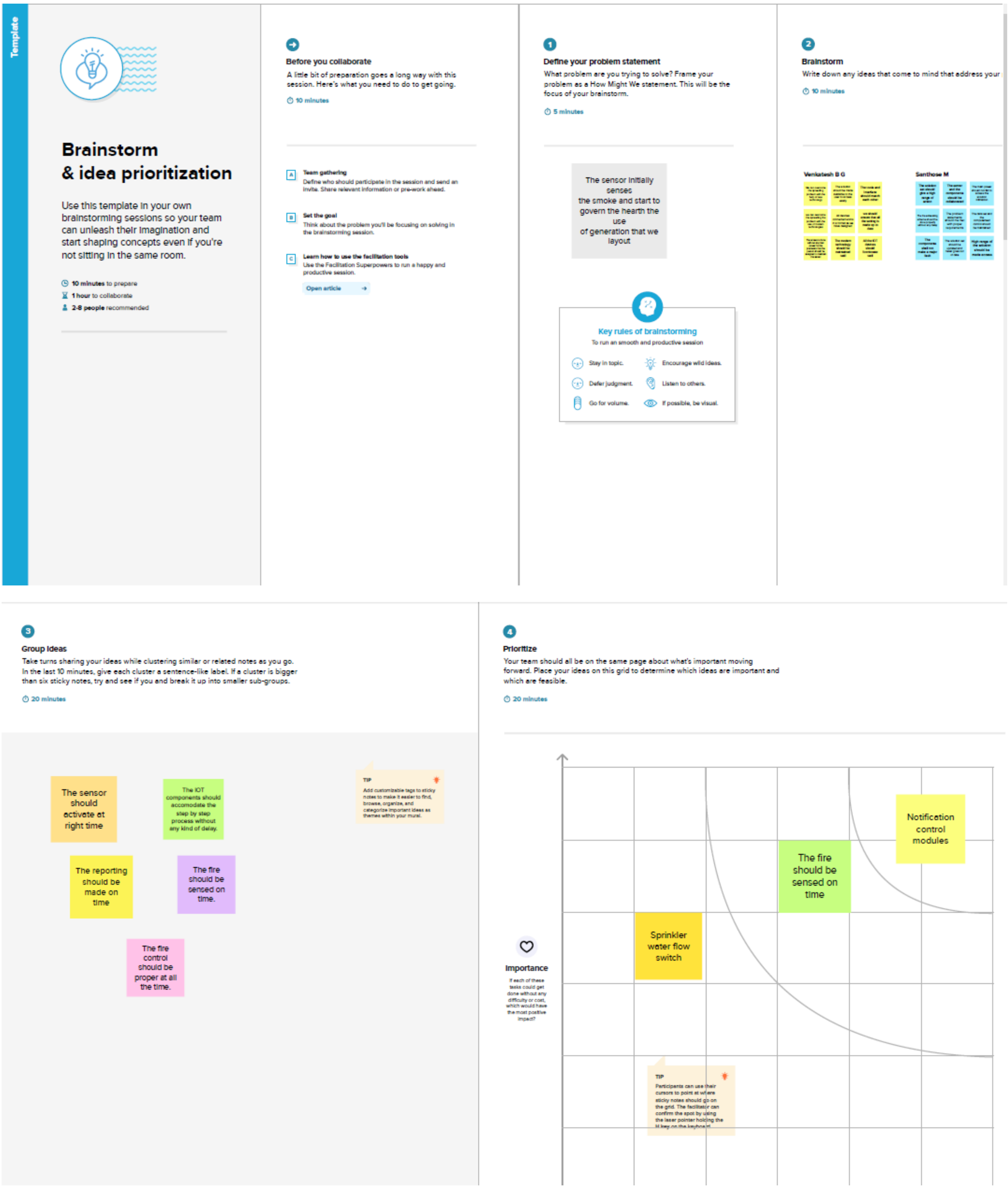
GAIN
"wants" / needs
measures of success
obstacles

It helps to keep the organization from catastrophic incidents

Sensors need to be replaced periodically

Figure 3.1Empathy Map Canvas

3.2 Ideation & 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

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

The sensor initially senses
the smoke and start to govern the hearth the use
of generation that we layout

Key rules of brainstorming
To run an smooth and productive session

- Stay in topic.
- Encourage wild ideas.
- Defer judgment.
- Listen to others.
- Go for volume.
- If possible, be visual.

2 **Brainstorm**

Write down any ideas that come to mind that address your

10 minutes

Venkatesh B G

Santhose M

3 **Group ideas**

Take turns sharing your ideas while clustering similar or related notes as you go. In the last 10 minutes, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you and break it up into smaller sub-groups.

20 minutes

The sensor should activate at right time

The IOT components should accomodate the step by step process without any kind of delay.

The reporting should be made on time

The fire should be sensed on time.

The fire control should be proper at all the time.

TIP
Add customizable tags to sticky notes to make it easier to find, browse, organize, and categorize important ideas as themes within your mural.

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

Importance
If each of these tasks could get done without any difficulty or cost, which would have the most positive impact?

Notification control modules

The fire should be sensed on time

Sprinkler water flow switch

TIP
Participants can use their cursors to point at where sticky notes should go on the grid. The facilitator can confirm the spot by using the laser pointer holding the H key on the keyboard.

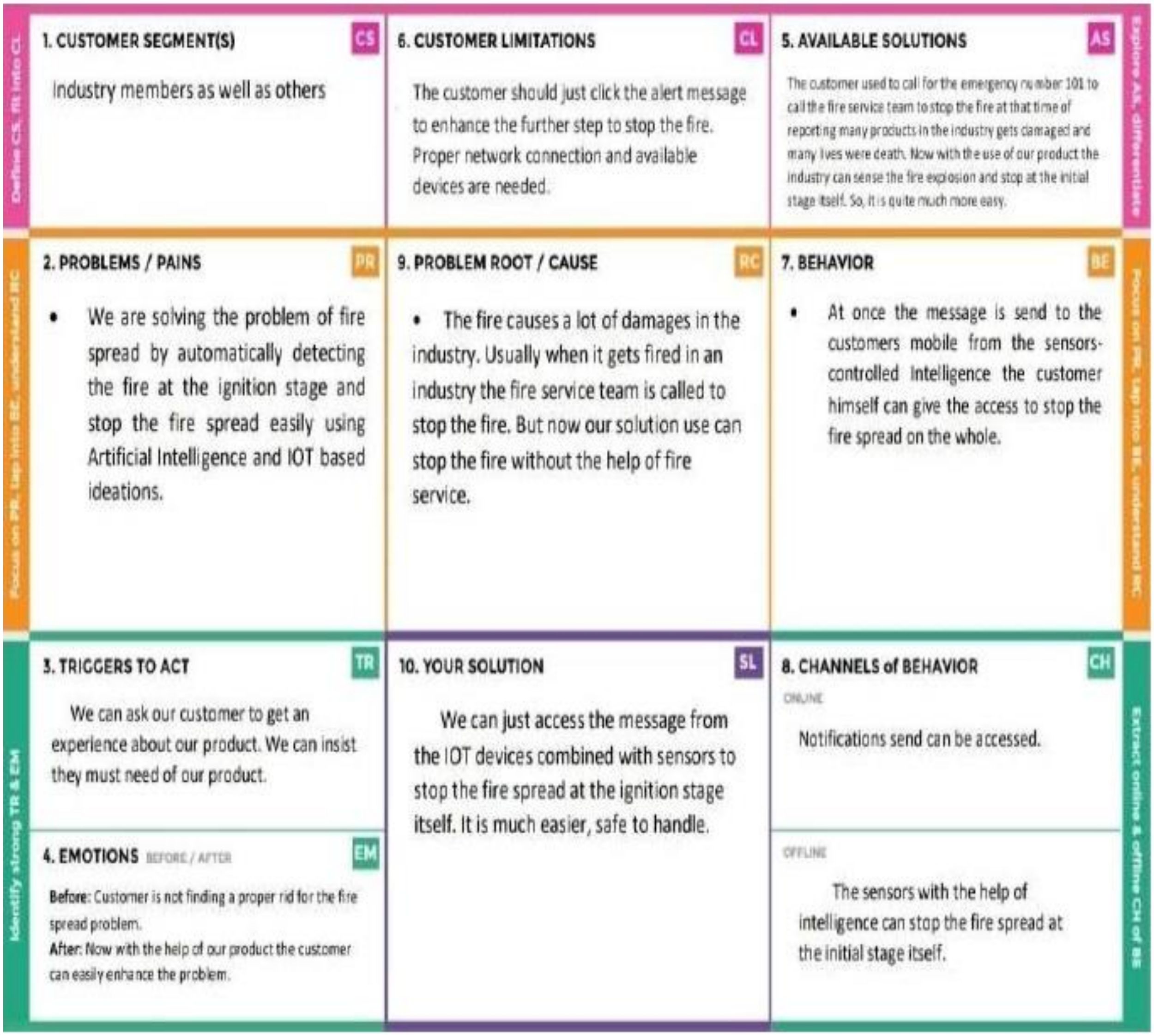
Figure 3.2 Ideation & Brainstorming

TABLE 3.3 Proposed Solution

S.No	Parameter	Description
1.	Problem Statement(Problem tobesolved)	Themaingoalofafirealarmsystemistogivepeopleadvancenoticeofafiresotheycanescapeandtakeswiftactiontoreduceorcompletely extinguishthefire'seffectsassoonasfeasible.
2.	Idea /Solutiondescription	• The exhaust fans are turned on based onthe temperature readings and if any gasesarepresent. • Sprinklerswillbeactivatedautomaticallyifaflame isdetected. • Theauthoritiesandthefirestationarenotifiedof anyemergencyalerts.
3.	Novelty/Uniqueness	• Whenthefirebeginstospread,thetemperature rises, and if any gases arepresent,theexhaust fansareactivated. • If a flame is detected, the sprinklers willactivate automatically and send a messagetohigherauthoritiesandthefire station. • Our proposed system provides a solutionfor secure transmission of the real timedataobtainedfromthesensorstotheIBMcloudratherthanusingthenetworking devices like ZigBee, LORA,GSMmoduleswhichcausestheinterferenceofdataobtainedfrommultipleusers. • Ourproductiscosteffective,sinceforcommunication to higher officials we haveawebdashboardratherusinghardwaredevices. • Design and implementation of highlyscalableproduct.

		• AlltheIOTenddevicesarecontrolledusingstandalonerechargeable batteriesso that the product would last for a longspan.
4.	SocialImpact/CustomerSatisfaction	• Customer experience can be identifiedthroughclientfeedbackprovidedbycustomers who use our kit and providefeedback. • PreventsPollution.
5.	Business Model(RevenueModel)	Thisversionisusedtocalculatetheopportunity of the ignition and spread acrossthelandscape.
6.	ScalabilityoftheSolution	Withthehelpofoursoftware,automatedreal-timedecision-makingispossibleinasettingwherehundredsofthousandsofsensorsareconstantly providing data through a webdashboardwithoutinterferingwith eachother.

3.4 Problem Solution Fit


Define CS, fit into CL
Focus on PR, tap into BE, understand RC
Identify strong TR & EM
Explore AS, differentiate
Focus on PR, tap into BE, understand RC
Extract online & offline CH of BE
CS
CL
AS
PR
RC
BE
TR
EM
SL
CH

1. CUSTOMER SEGMENT(S)

Industry members as well as others

6. CUSTOMER LIMITATIONS

The customer should just click the alert message to enhance the further step to stop the fire. Proper network connection and available devices are needed.

5. AVAILABLE SOLUTIONS

The customer used to call for the emergency number 101 to call the fire service team to stop the fire at that time of reporting many products in the industry gets damaged and many lives were death. Now with the use of our product the industry can sense the fire explosion and stop at the initial stage itself. So, it is quite much more easy.

2. PROBLEMS / PAINS

- We are solving the problem of fire spread by automatically detecting the fire at the ignition stage and stop the fire spread easily using Artificial Intelligence and IOT based ideations.

9. PROBLEM ROOT / CAUSE

- The fire causes a lot of damages in the industry. Usually when it gets fired in an industry the fire service team is called to stop the fire. But now our solution use can stop the fire without the help of fire service.

7. BEHAVIOR

- At once the message is send to the customers mobile from the sensors-controlled Intelligence the customer himself can give the access to stop the fire spread on the whole.

3. TRIGGERS TO ACT

We can ask our customer to get an experience about our product. We can insist they must need of our product.

4. EMOTIONS BEFORE / AFTER

Before: Customer is not finding a proper rid for the fire spread problem.

After: Now with the help of our product the customer can easily enhance the problem.

10. YOUR SOLUTION

We can just access the message from the IOT devices combined with sensors to stop the fire spread at the ignition stage itself. It is much easier, safe to handle.

8. CHANNELS of BEHAVIOR

ONLINE

Notifications send can be accessed.

OFFLINE

The sensors with the help of intelligence can stop the fire spread at the initial stage itself.

Figure 3.4 Problem Solution Fit

4.REQUIREMENT ANALYSIS

TABLE 4.1 Functional requirement

FunctionalRequirements:

Followingarethefunctionalrequirementsoftheproposedsolution.

FRNo.	FunctionalRequirement(Epic)	SubRequirement(Story/Sub-Task)
FR-1	Deviceconfiguration	NewIoTdeviceiscreatedinthecloud Thedeviceisconfiguredwiththenewclouddevice
FR-2	Admindashboard/adminpanel	DatafromsensorsshowninpictorialformControlsaregiveninthebuttonformat
FR-3	Internetconnectivity	Makesurefully-fledgedinternetconnectivityisrequiredforsmoothcommunicationbetweendeviceandcloud
FR-4	SMSAPI	AexternalSMSAPIisrequired

TABLE 4.2 Non-functionalRequirements:

Followingarethenon-functionalrequirementsoftheproposedsolution.

FRNo.	**Non-FunctionalRequirement**	**Description**
NFR-1	**Usability**	The dashboard can be used via a web browserItgivesanabstractviewinaneasy-to-useform.
NFR-2	**Security**	AsthedataissentthroughHTTPSthe dataisencrypted,soitissafe.
NFR-3	**Reliability**	Thesystemiscompletelyreliableaslongasthe internetandpowerisreliable

5.PROJECT DESIGN

5.1 Data Flow Diagrams

Project design is an early phase of a project where the project's keyfeatures, structure,criteria for success, and major deliverables are planned out. The aim is to develop one or moredesigns thatcan be usedto achieve the desired projectgoals.Stakeholders can thenchoosethebest design for the execution of the project. The project design steps might generate variousoutputs,suchassketches,flowcharts,sitetrees,HTMLscreendesigns,prototypes ,photoimpressions,andmore.Theprojectdesign includeseverythingfromwhoisresponsibleforcompletingthe project toa descriptionofthe project,itsgoals,outcomes and objectives.Itdescribes when these goals, outcomes and objectives will be reached, and the major deliverables,products orfeaturesthatwillbecompleted

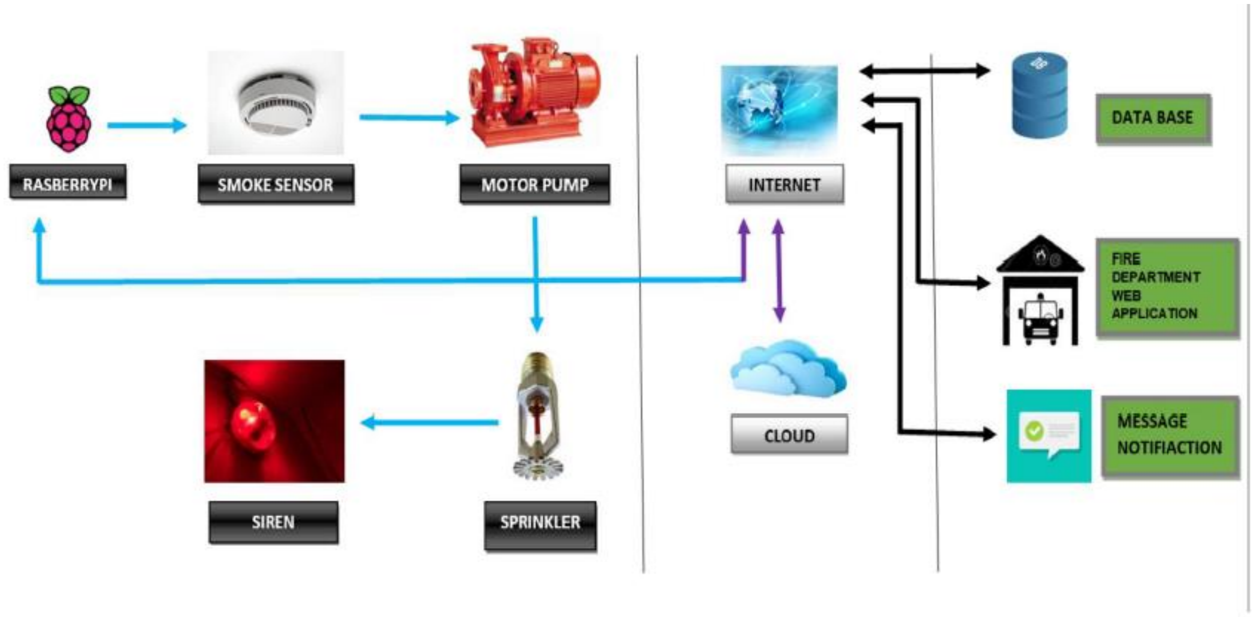


Figure 5.1 Data Flow Diagrams

5.2 Solution and Technical Architecture

SolutionArchitecture:

Solutionarchitectureisacomplexprocess–withmanysub-processes–thatbridgesthegapbetweenbusiness problemsand technologysolutions. Its goalsare to:

- Findthebesttechsolutiontosolveexistingbusinessproblems.

- Describethestructure,characteristics, behavior, and other aspects of the software to projectstakeholders.

- Define features,development phases, andsolution requirements.

- Providespecifications according towhichthe solution is defined,managed, anddelivered

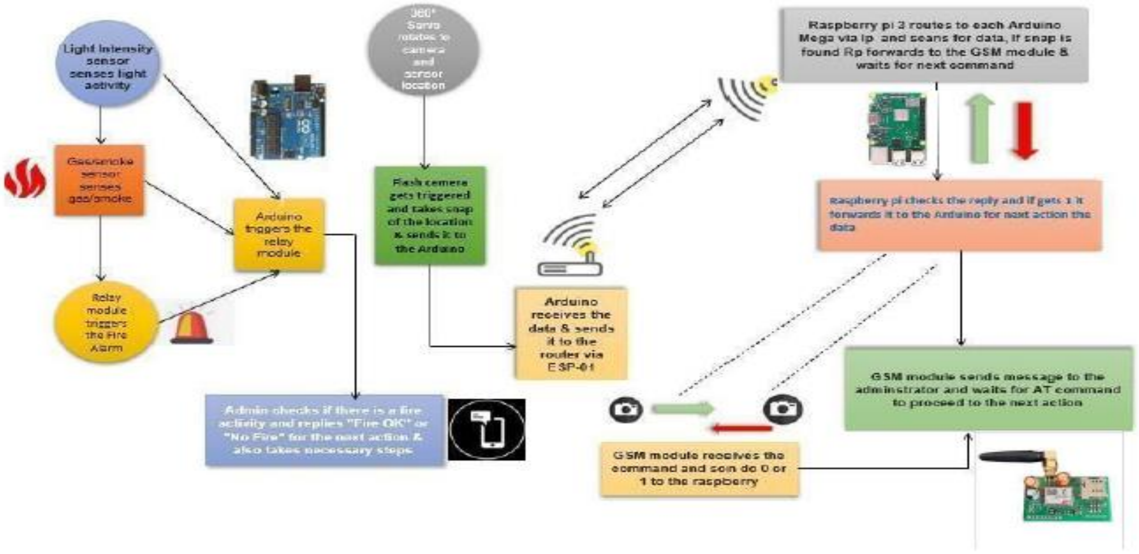

Light Intensity sensor senses light activity
Gas/smoke sensor senses gas/smoke
Relay module triggers the Fire Alarm
Arduino triggers the relay module
360° Servo rotates to camera and sensor location
Flash camera gets triggered and takes snap of the location & sends it to the Arduino
Arduino receives the data & sends it to the router via ESP-01
Admin checks if there is a fire activity and replies "Fire OK" or "No Fire" for the next action & also takes necessary steps
Raspberry pi 3 routes to each Arduino Mega via Ip and scans for data, If snap is found Rp forwards to the GSM module & waits for next command
Raspberry pi checks the reply and if gets 1 it forwards it to the Arduino for next action the data
GSM module sends message to the administrator and waits for AT command to proceed to the next action
GSM module receives the command and sen do 0 or 1 to the raspberry

Figure 5.2 Solution Architecture

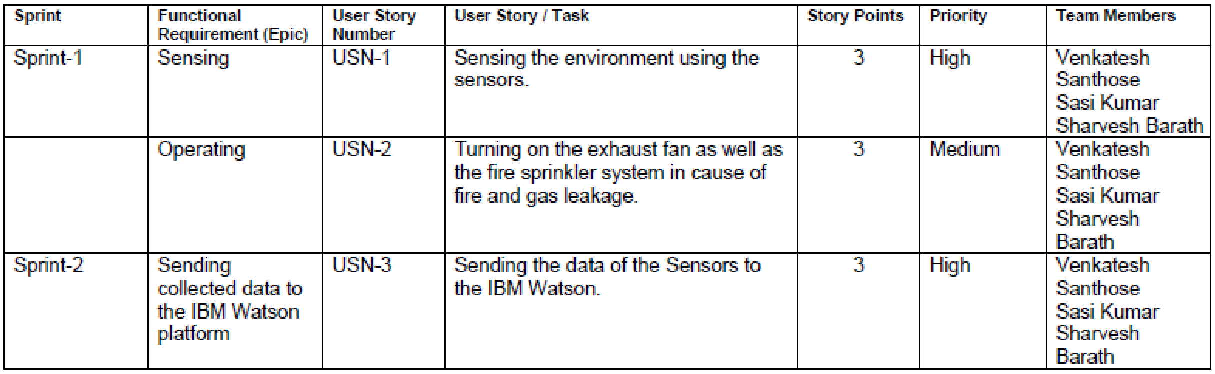
TABLE 5.3 User Stories

UserType	Functionalrequirement	Userstorynumber	Userstory/task	Acceptance criteria	Priority	Release
Customer (Mobileuser, Webuser, Care executive, Administrator)	Registration	USN-1	As auser,I canregister forthe application byentering mymail, password, and confirming my password	Ican access myaccount/ dashboard	High	Sprint-1
		USN-2	As auser,I willreceive confirmation emailonceI have registered forthe application	I canreceive confirmation email&click confirm	High	Sprint-1
	Dashboard	USN-3	As auser,I canregister forthe application through internet	Icanregister &accessthe dashboard withInternet login	Low	Sprint-2

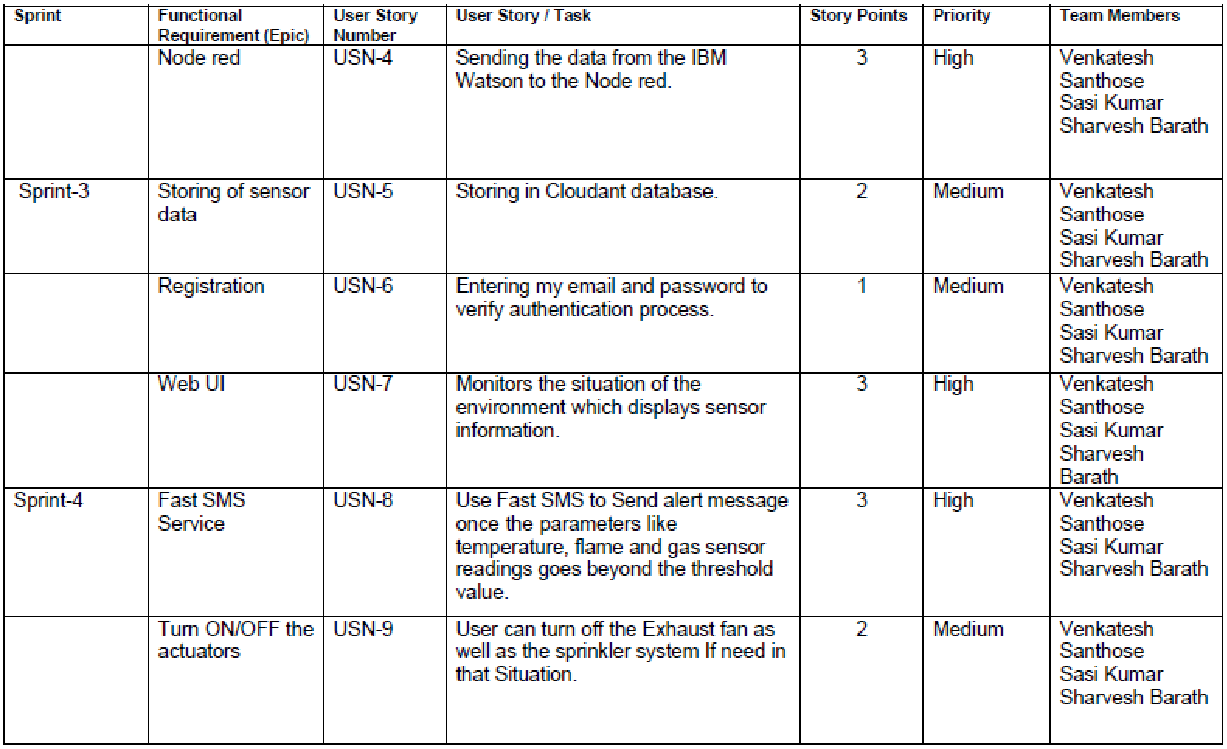
		USN-4	As auser,I canregister forthe application through Gmail	Icanconfirm the registrationin Gmail	Medium	Sprint-3
	Login	USN-5	As auser,I canloginto the application byentering email& password	Icanlogin withmyid andpassword	High	Sprint-4

6. PROJECT PLANNING AND SCHEDULING

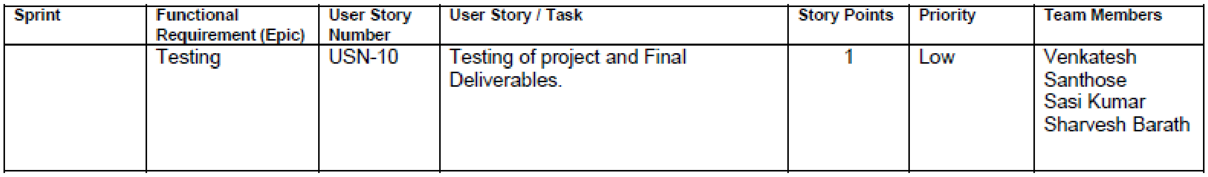
TABLE 6.1 Sprint Planning & Estimation



Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Sensing	USN-1	Sensing the environment using the sensors.	3	High	Venkatesh Santhose Sasi Kumar Sharvesh Barath
	Operating	USN-2	Turning on the exhaust fan as well as the fire sprinkler system in cause of fire and gas leakage.	3	Medium	Venkatesh Santhose Sasi Kumar Sharvesh Barath
Sprint-2	Sending collected data to the IBM Watson platform	USN-3	Sending the data of the Sensors to the IBM Watson.	3	High	Venkatesh Santhose Sasi Kumar Sharvesh Barath

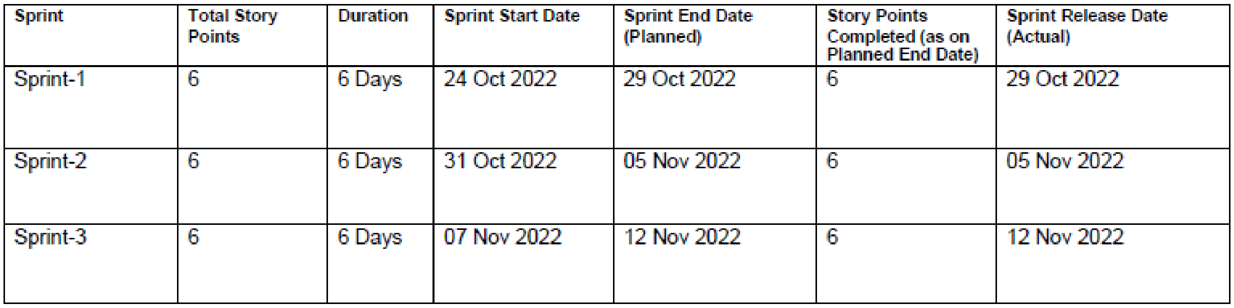


Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
	Node red	USN-4	Sending the data from the IBM Watson to the Node red.	3	High	Venkatesh Santhose Sasi Kumar Sharvesh Barath
Sprint-3	Storing of sensor data	USN-5	Storing in Cloudant database.	2	Medium	Venkatesh Santhose Sasi Kumar Sharvesh Barath
	Registration	USN-6	Entering my email and password to verify authentication process.	1	Medium	Venkatesh Santhose Sasi Kumar Sharvesh Barath
	Web UI	USN-7	Monitors the situation of the environment which displays sensor information.	3	High	Venkatesh Santhose Sasi Kumar Sharvesh Barath
Sprint-4	Fast SMS Service	USN-8	Use Fast SMS to Send alert message once the parameters like temperature, flame and gas sensor readings goes beyond the threshold value.	3	High	Venkatesh Santhose Sasi Kumar Sharvesh Barath
	Turn ON/OFF the actuators	USN-9	User can turn off the Exhaust fan as well as the sprinkler system If need in that Situation.	2	Medium	Venkatesh Santhose Sasi Kumar Sharvesh Barath

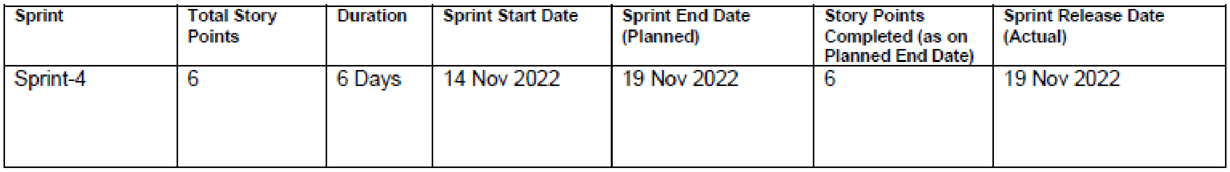


Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
	Testing	USN-10	Testing of project and Final Deliverables.	1	Low	Venkatesh Santhose Sasi Kumar Sharvesh Barath

TABLE 6.2 Sprint Delivery Schedule



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



Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-4	6	6 Days	14 Nov 2022	19 Nov 2022	6	19 Nov 2022

Velocity:

Imaginewehavea10-daysprintduration,andthevelocityoftheteamis20(pointspersprint).Let’scalculate theteam’saveragevelocity (AV)per iteration unit (storypointsper day)

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

AV = 6/6=1

BurndownChart:

Aburndownchartisagraphicalrepresentationofworkleft todoversustime.Itisoftenusedinagilesoftwaredevelopment methodologies such as Scrum. However, burn down charts can be applied to any project containing measurable progressover time

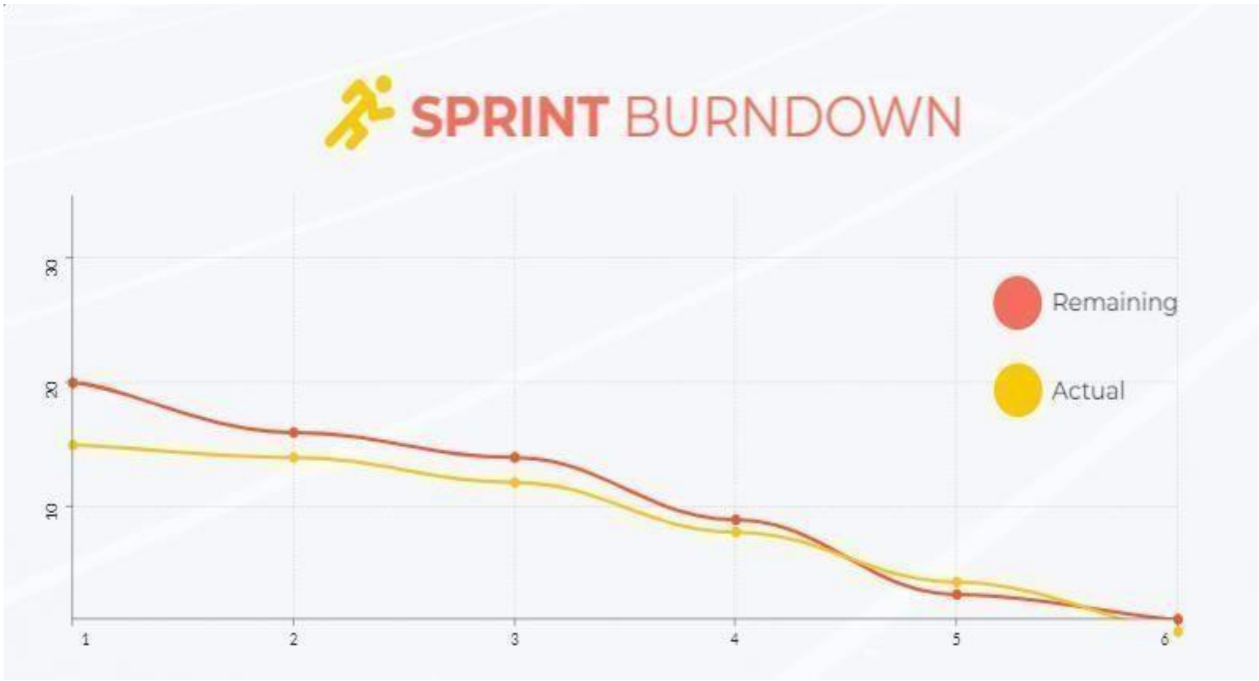
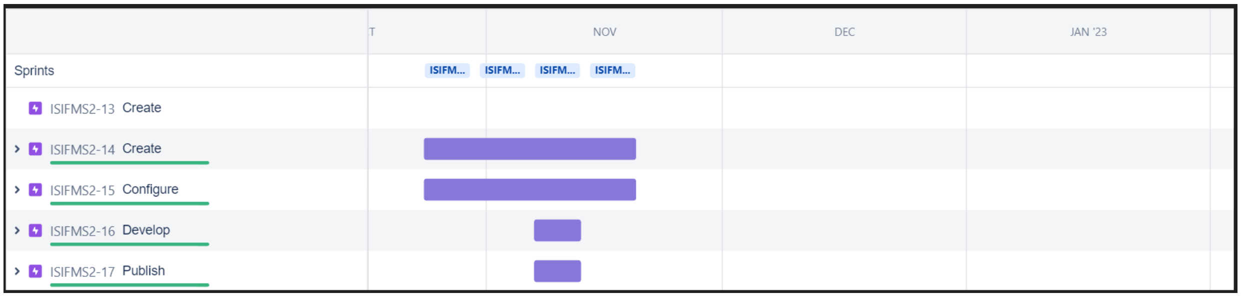


Figure 6.2 Burndown Chart

6.3 Reports from JIRA



Sprints

ISIFMS2-13 Create

ISIFMS2-14 Create

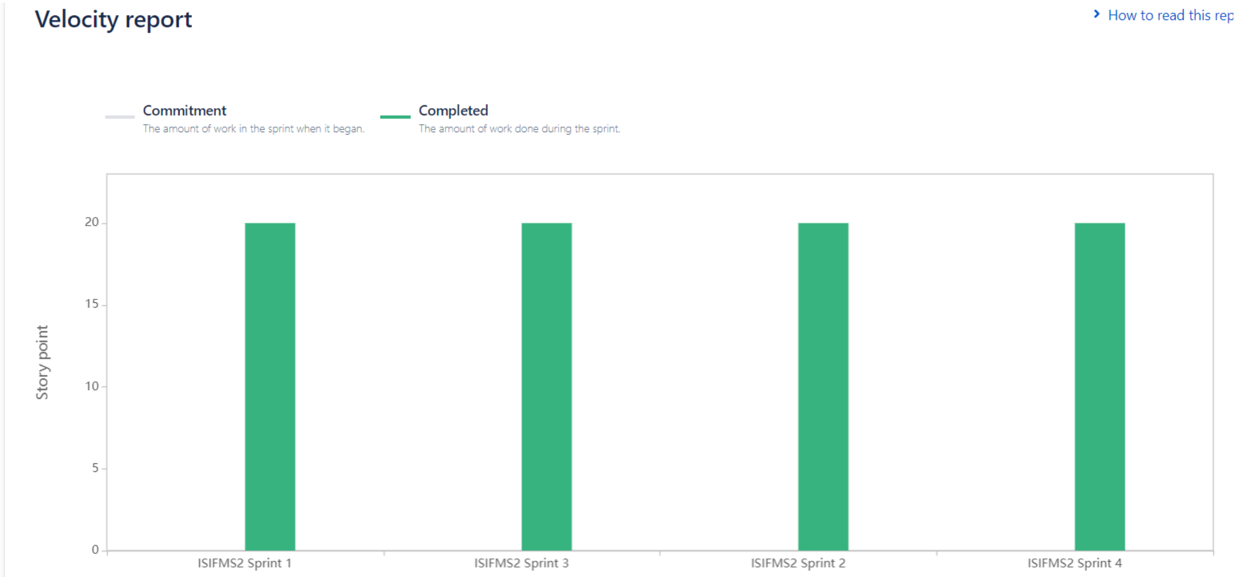
ISIFMS2-15 Configure

ISIFMS2-16 Develop

ISIFMS2-17 Publish

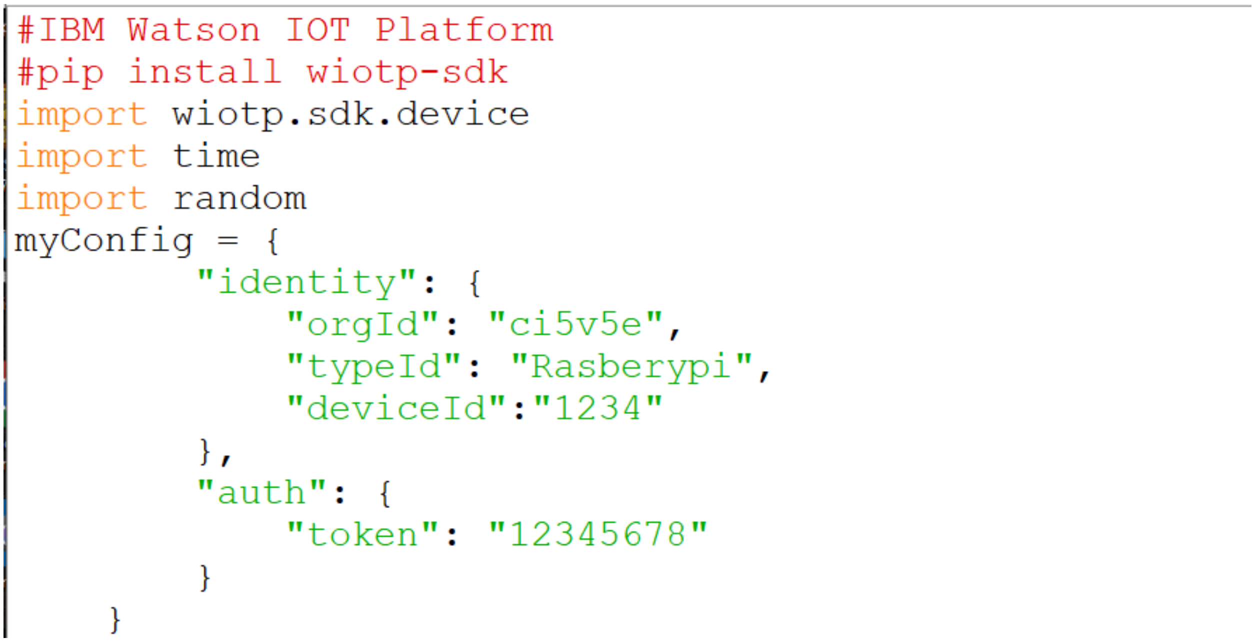
https://pnt2022tmid47460.atlassian.net/jira/software/projects/ISIFMS2/boards/2/roadmap

Figure 6.3 Report from JIRA

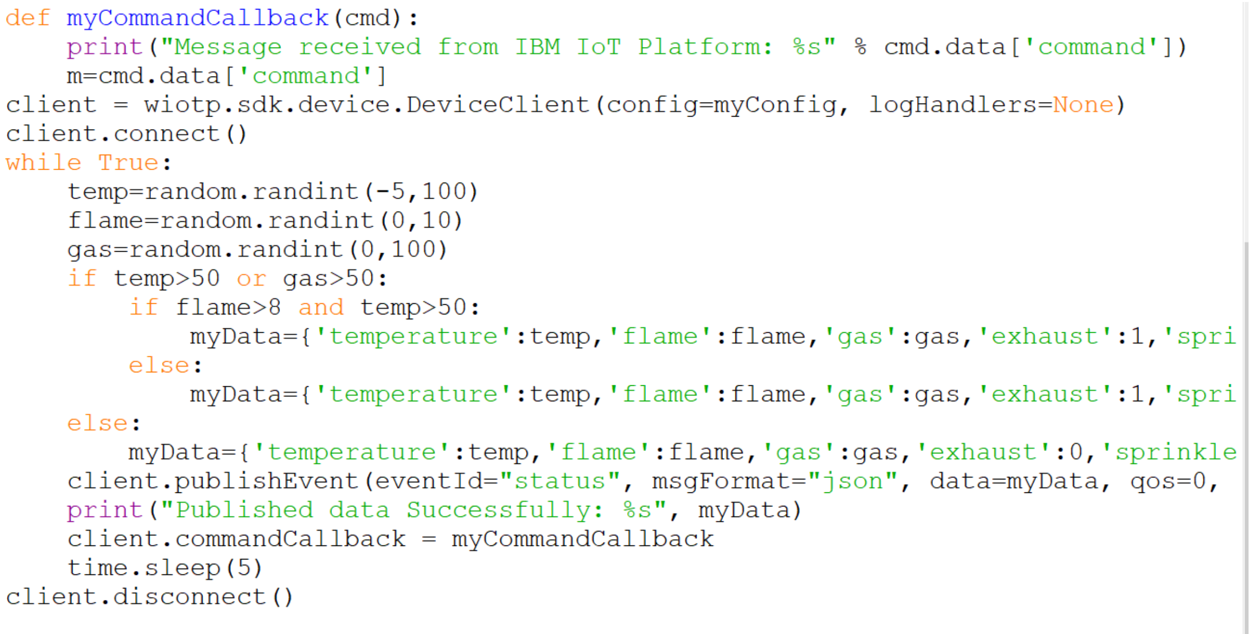


7 CODING AND SOLUTIONING

7.1 Feature 1



```
#IBM Watson IOT Platform
#pip install wiotp-sdk
import wiotp.sdk.device
import time
import random
myConfig = {
        "identity": {
            "orgId": "ci5v5e",
            "typeId": "Rasberypi",
            "deviceId":"1234"
        },
        "auth": {
            "token": "12345678"
        }
    }
```



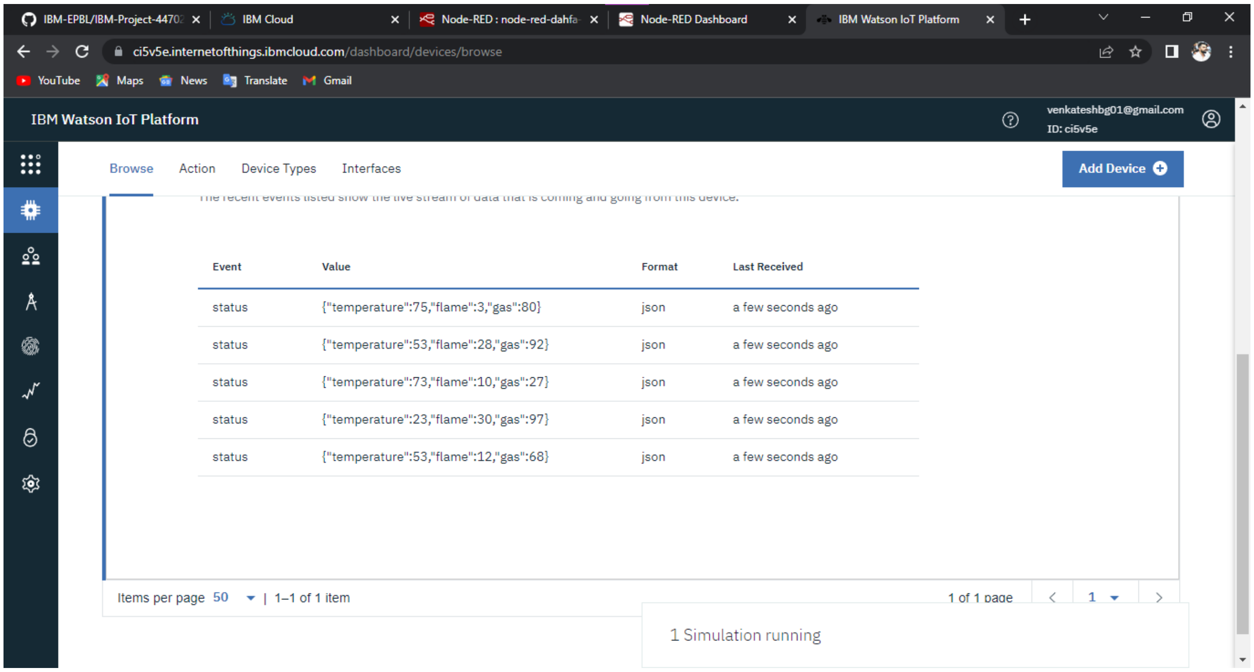
```
def myCommandCallback(cmd):
    print("Message received from IBM IoT Platform: %s" % cmd.data['command'])
    m=cmd.data['command']
client = wiotp.sdk.device.DeviceClient(config=myConfig, logHandlers=None)
client.connect()
while True:
    temp=random.randint(-5,100)
    flame=random.randint(0,10)
    gas=random.randint(0,100)
    if temp>50 or gas>50:
        if flame>8 and temp>50:
            myData={'temperature':temp,'flame':flame,'gas':gas,'exhaust':1,'spri
        else:
            myData={'temperature':temp,'flame':flame,'gas':gas,'exhaust':1,'spri
    else:
        myData={'temperature':temp,'flame':flame,'gas':gas,'exhaust':0,'sprinkle
    client.publishEvent(eventId="status", msgFormat="json", data=myData, qos=0,
    print("Published data Successfully: %s", myData)
    client.commandCallback = myCommandCallback
    time.sleep(5)
client.disconnect()
```

Output

```
Published data Successfully: %s {'temperature': 99, 'flame': 8, 'gas': 41, 'exhaust': 1,
'sprinklers': 0}

Published data Successfully: %s {'temperature': 43, 'flame': 4, 'gas': 22, 'exhaust': 0,
'sprinklers': 0}

Published data Successfully: %s {'temperature': 64, 'flame': 10, 'gas': 73, 'exhaust':
1, 'sprinklers': 1}

Published data Successfully: %s {'temperature': 43, 'flame': 2, 'gas': 87, 'exhaust': 1,
'sprinklers': 0}

Published data Successfully: %s {'temperature': 39, 'flame': 8, 'gas': 19, 'exhaust': 0,
'sprinklers': 0}

Published data Successfully: %s {'temperature': 52, 'flame': 8, 'gas': 6, 'exhaust': 1,
'sprinklers': 0}

Published data Successfully: %s {'temperature': -2, 'flame': 3, 'gas': 98, 'exhaust': 1,
'sprinklers': 0}

Published data Successfully: %s {'temperature': 95, 'flame': 10, 'gas': 75, 'exhaust':
1, 'sprinklers': 1}

Published data Successfully: %s {'temperature': 90, 'flame': 4, 'gas': 25, 'exhaust': 1,
'sprinklers': 0}
```

```
Published data Successfully: %s {'temperature': -2, 'flame': 5, 'gas': 18, 'exhaust': 0,
'sprinklers': 0}

Published data Successfully: %s {'temperature': 99, 'flame': 0, 'gas': 38, 'exhaust': 1,
'sprinklers': 0}

Published data Successfully: %s {'temperature': 80, 'flame': 6, 'gas': 70, 'exhaust': 1,
'sprinklers': 0}

Published data Successfully: %s {'temperature': 48, 'flame': 9, 'gas': 46, 'exhaust': 0,
'sprinklers': 0}

Published data Successfully: %s {'temperature': 1, 'flame': 10, 'gas': 63, 'exhaust': 1,
'sprinklers': 0}

Published data Successfully: %s {'temperature': 95, 'flame': 6, 'gas': 86, 'exhaust': 1,
'sprinklers': 0}

Published data Successfully: %s {'temperature': 29, 'flame': 6, 'gas': 2, 'exhaust': 0,
'sprinklers': 0}
```

7.2 Feature 2



The recent events listed show the live stream of data that is coming and going from this device.

Event	Value	Format	Last Received
status	{"temperature":75,"flame":3,"gas":80}	json	a few seconds ago
status	{"temperature":53,"flame":28,"gas":92}	json	a few seconds ago
status	{"temperature":73,"flame":10,"gas":27}	json	a few seconds ago
status	{"temperature":23,"flame":30,"gas":97}	json	a few seconds ago
status	{"temperature":53,"flame":12,"gas":68}	json	a few seconds ago

Items per page 50 | 1–1 of 1 item 1 of 1 page 1

1 Simulation running

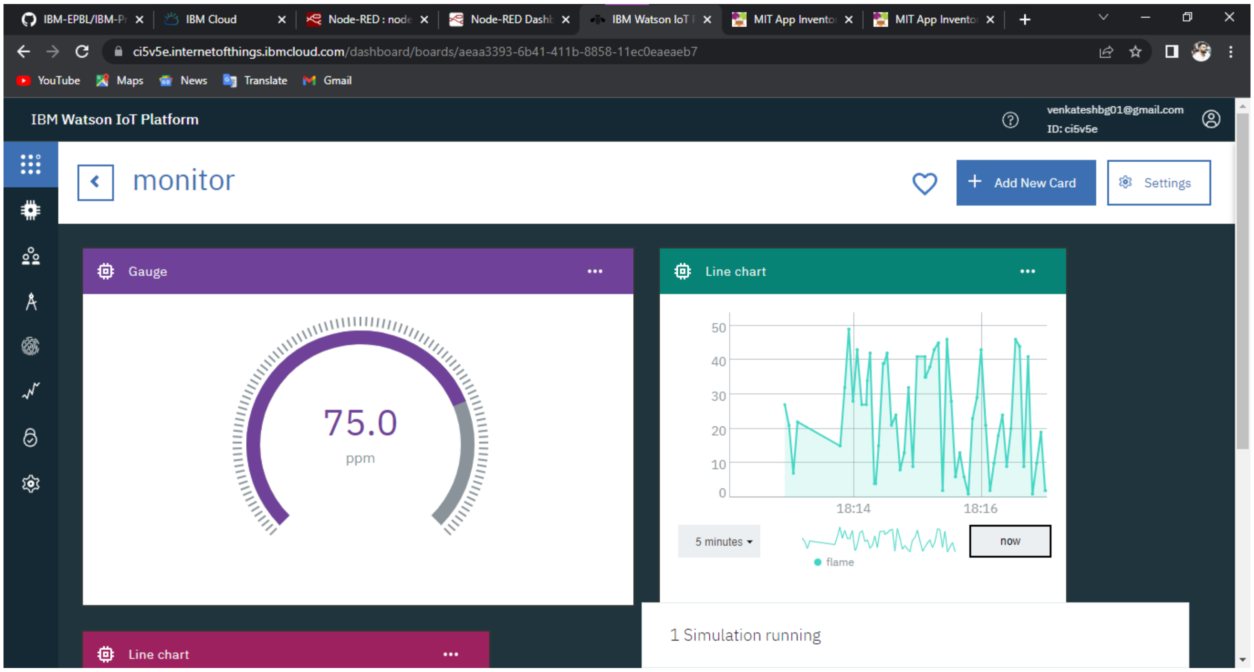
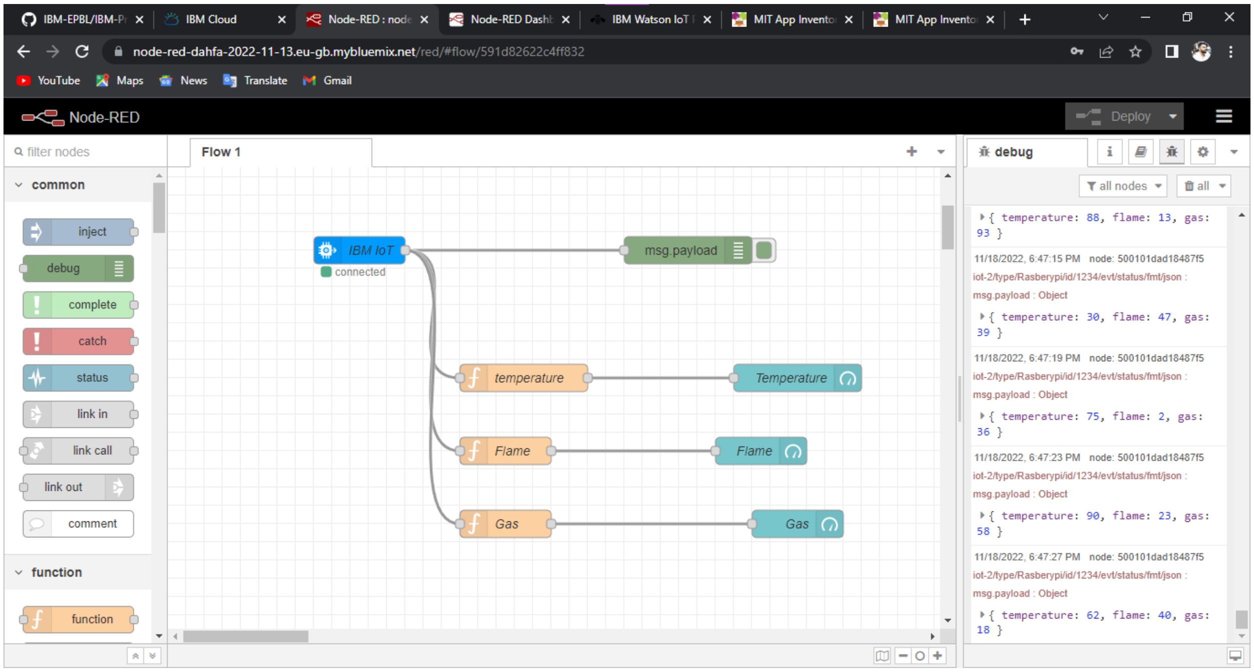


Figure 7.2.1 IBM Watson IoT Platform

Node Red:



```
▸{ temperature: 88, flame: 13, gas:
93 }
11/18/2022, 6:47:15 PM   node: 500101dad18487f5
iot-2/type/Rasberypi/id/1234/evt/status/fmt/json :
msg.payload : Object
▸{ temperature: 30, flame: 47, gas:
39 }
11/18/2022, 6:47:19 PM   node: 500101dad18487f5
iot-2/type/Rasberypi/id/1234/evt/status/fmt/json :
msg.payload : Object
▸{ temperature: 75, flame: 2, gas:
36 }
11/18/2022, 6:47:23 PM   node: 500101dad18487f5
iot-2/type/Rasberypi/id/1234/evt/status/fmt/json :
msg.payload : Object
▸{ temperature: 90, flame: 23, gas:
58 }
11/18/2022, 6:47:27 PM   node: 500101dad18487f5
iot-2/type/Rasberypi/id/1234/evt/status/fmt/json :
msg.payload : Object
▸{ temperature: 62, flame: 40, gas:
18 }
```

Figure 7.2.2 Node Red Platform

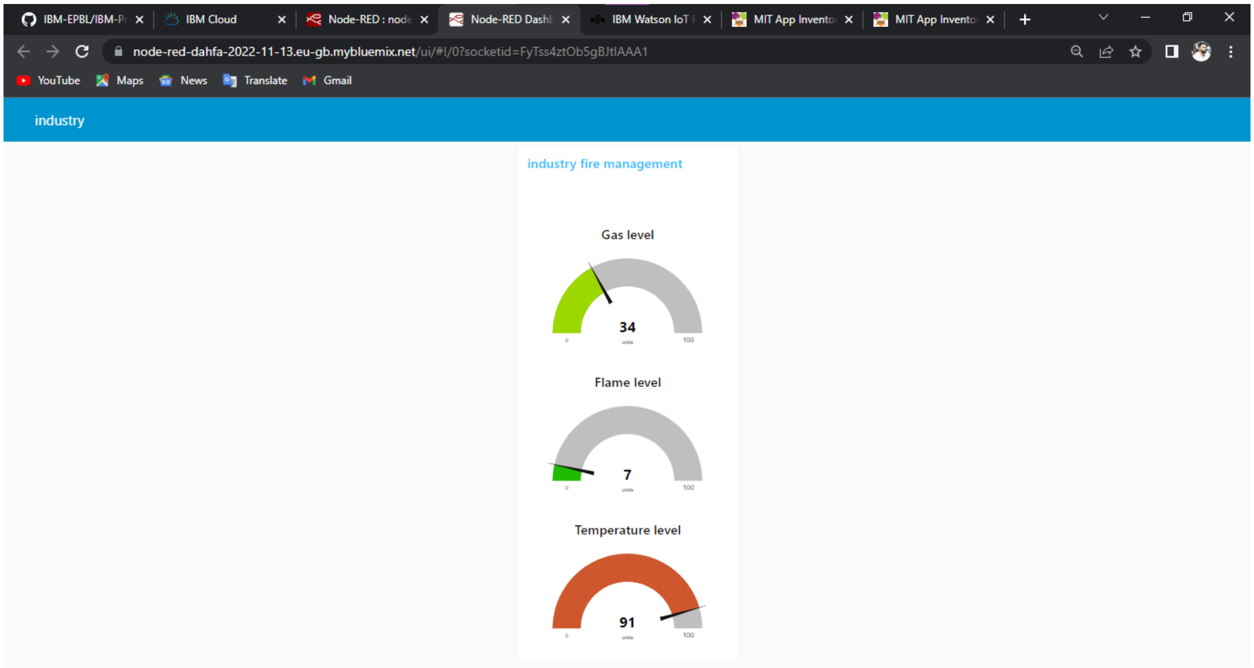
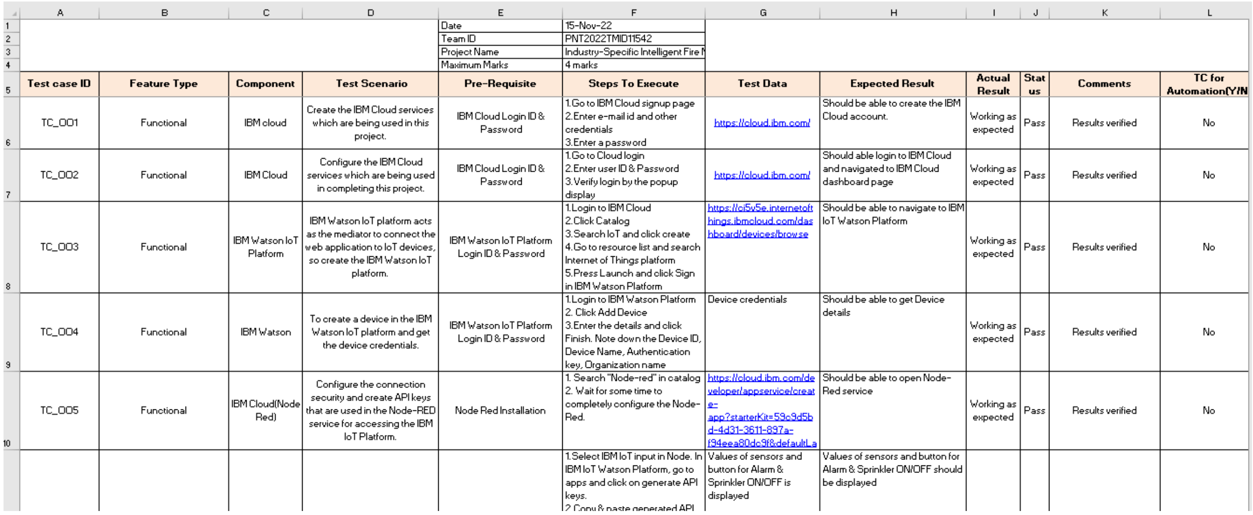


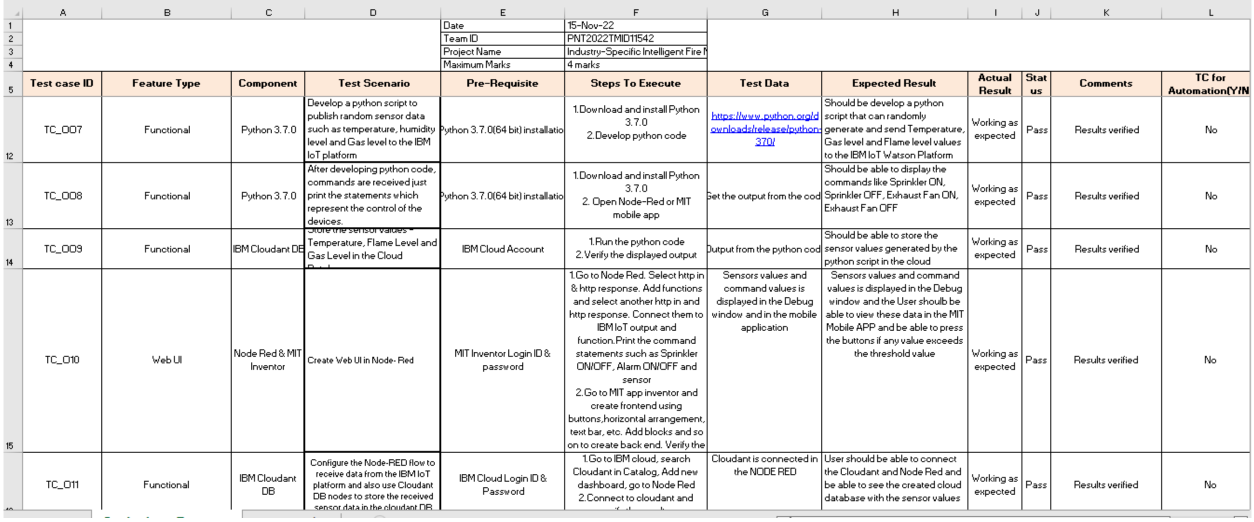
Figure 7.2.3 Web UI

8 TESTING

TABLE 8.1 Test Cases



	A	B	C	D	E	F	G	H	I	J	K	L
1					Date	15-Nov-22						
2					Team ID	PNT2022TMID11542						
3					Project Name	Industry-Specific Intelligent Fire M						
4					Maximum Marks	4 marks						
5	**Test case ID**	**Feature Type**	**Component**	**Test Scenario**	**Pre-Requisite**	**Steps To Execute**	**Test Data**	**Expected Result**	**Actual Result**	**Status**	**Comments**	**TC for Automation(Y/N**
6	TC_001	Functional	IBM cloud	Create the IBM Cloud services which are being used in this project.	IBM Cloud Login ID & Password	1.Go to IBM Cloud signup page 2.Enter e-mail id and other credentials 3.Enter a password	https://cloud.ibm.com/	Should be able to create the IBM Cloud account.	Working as expected	Pass	Results verified	No
7	TC_002	Functional	IBM Cloud	Configure the IBM Cloud services which are being used in completing this project.	IBM Cloud Login ID & Password	1.Go to Cloud login 2.Enter user ID & Password 3.Verify login by the popup display	https://cloud.ibm.com/	Should able login to IBM Cloud and navigated to IBM Cloud dashboard page	Working as expected	Pass	Results verified	No
8	TC_003	Functional	IBM Watson IoT Platform	IBM Watson IoT platform acts as the mediator to connect the web application to IoT devices, so create the IBM Watson IoT platform.	IBM Watson IoT Platform Login ID & Password	1.Login to IBM Cloud 2.Click Catalog 3.Search IoT and click create 4.Go to resource list and search Internet of Things platform 5.Press Launch and click Sign in IBM Watson Platform	https://ci5v5e.internetofthings.ibmcloud.com/dashboard/devices/browse	Should be able to navigate to IBM IoT Watson Platform	Working as expected	Pass	Results verified	No
9	TC_004	Functional	IBM Watson	To create a device in the IBM Watson IoT platform and get the device credentials.	IBM Watson IoT Platform Login ID & Password	1.Login to IBM Watson Platform 2. Click Add Device 3.Enter the details and click Finish. Note down the Device ID, Device Name, Authentication key, Organization name	Device credentials	Should be able to get Device details	Working as expected	Pass	Results verified	No
10	TC_005	Functional	IBM Cloud(Node Red)	Configure the connection security and create API keys that are used in the Node-RED service for accessing the IBM IoT Platform.	Node Red Installation	1. Search "Node-red" in catalog 2. Wait for some time to completely configure the Node-Red.	https://cloud.ibm.com/developer/appservice/create-app?starterKit=59c9d5bd-4d31-3611-897a-f94eea80dc9f&defaultLa	Should be able to open Node-Red service	Working as expected	Pass	Results verified	No
						1.Select IBM IoT input in Node. In IBM IoT Watson Platform, go to apps and click on generate API keys. 2.Copy & paste generated API	Values of sensors and button for Alarm & Sprinkler ON/OFF is displayed	Values of sensors and button for Alarm & Sprinkler ON/OFF should be displayed				



	A	B	C	D	E	F	G	H	I	J	K	L
1					Date	15-Nov-22						
2					Team ID	PNT2022TMID11542						
3					Project Name	Industry-Specific Intelligent Fire M						
4					Maximum Marks	4 marks						
5	**Test case ID**	**Feature Type**	**Component**	**Test Scenario**	**Pre-Requisite**	**Steps To Execute**	**Test Data**	**Expected Result**	**Actual Result**	**Status**	**Comments**	**TC for Automation(Y/N**
12	TC_007	Functional	Python 3.7.0	Develop a python script to publish random sensor data such as temperature, humidity level and Gas level to the IBM IoT platform	Python 3.7.0(64 bit) installatio	1.Download and install Python 3.7.0 2.Develop python code	https://www.python.org/downloads/release/python-370/	Should be develop a python script that can randomly generate and send Temperature, Gas level and Flame level values to the IBM IoT Watson Platform	Working as expected	Pass	Results verified	No
13	TC_008	Functional	Python 3.7.0	After developing python code, commands are received just print the statements which represent the control of the devices.	Python 3.7.0(64 bit) installatio	1.Download and install Python 3.7.0 2. Open Node-Red or MIT mobile app	Get the output from the cod	Should be able to display the commands like Sprinkler ON, Sprinkler OFF, Exhaust Fan ON, Exhaust Fan OFF	Working as expected	Pass	Results verified	No
14	TC_009	Functional	IBM Cloudant DE	Store the sensor values - Temperature, Flame Level and Gas Level in the Cloud	IBM Cloud Account	1.Run the python code 2.Verify the displayed output	Output from the python cod	Should be able to store the sensor values generated by the python script in the cloud	Working as expected	Pass	Results verified	No
15	TC_010	Web UI	Node Red & MIT Inventor	Create Web UI in Node- Red	MIT Inventor Login ID & password	1.Go to Node Red. Select http in & http response. Add functions and select another http in and http response. Connect them to IBM IoT output and function.Print the command statements such as Sprinkler ON/OFF, Alarm ON/OFF and sensor 2.Go to MIT app inventor and create frontend using buttons,horizontal arrangement, text bar, etc. Add blocks and so on to create back end. Verify the	Sensors values and command values is displayed in the Debug window and in the mobile application	Sensors values and command values is displayed in the Debug window and the User shoulb be able to view these data in the MIT Mobile APP and be able to press the buttons if any value exceeds the threshold value	Working as expected	Pass	Results verified	No
	TC_011	Functional	IBM Cloudant DB	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	IBM Cloud Login ID & Password	1.Go to IBM cloud, search Cloudant in Catalog, Add new dashboard, go to Node Red 2.Connect to cloudant and	Cloudant is connected in the NODE RED	User should be able to connect the Cloudant and Node Red and be able to see the created cloud database with the sensor values	Working as expected	Pass	Results verified	No

TABLE 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 Industry-specific intelligent fire management system project at the time of the release to User Acceptance Testing (UAT).

2. Defect Analysis

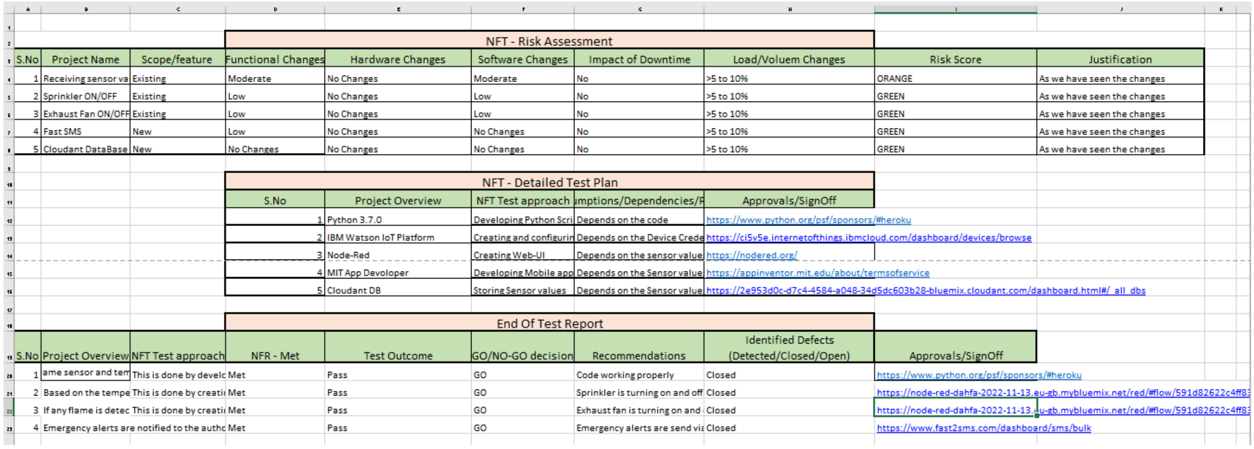
This report shows the number of resolved or closed bugs at each severity level, and how they were resolved **Resolution**	**Severity 1**	**Severity 2**	**Severity 3**	**Severity 4**	**Subtotal**
By Design	10	4	2	3	20
Duplicate	1	0	3	0	4
External	2	3	0	1	6
Fixed	11	2	4	20	37
Not Reproduced	0	0	1	0	1
Skipped	0	0	1	1	2
Won't Fix	0	0	0	1	8
Totals	24	14	13	26	70

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 the Sensor values	7	0	0	7
Client Mobile Application	51	0	0	51
Security	2	0	0	2
Outsource Shipping	3	0	0	3
Exception Reporting	9	0	0	9
Final Report Output	4	0	0	4
Version Control	2	0	0	2

9. RESULTS

TABLE 9.1 Performance Metrics



NFT - Risk Assessment

S.No	Project Name	Scope/feature	Functional Changes	Hardware Changes	Software Changes	Impact of Downtime	Load/Voluem Changes	Risk Score	Justification
1	Receiving sensor va	Existing	Moderate	No Changes	Moderate	No	>5 to 10%	ORANGE	As we have seen the changes
2	Sprinkler ON/OFF	Existing	Low	No Changes	Low	No	>5 to 10%	GREEN	As we have seen the changes
3	Exhaust Fan ON/OFF	Existing	Low	No Changes	Low	No	>5 to 10%	GREEN	As we have seen the changes
4	Fast SMS	New	Low	No Changes	No Changes	No	>5 to 10%	GREEN	As we have seen the changes
5	Cloudant DataBase	New	No Changes	No Changes	No Changes	No	>5 to 10%	GREEN	As we have seen the changes

NFT - Detailed Test Plan

S.No	Project Overview	NFT Test approach	umptions/Dependencies/R	Approvals/SignOff
1	Python 3.7.0	Developing Python Scri	Depends on the code	https://www.python.org/psf/sponsors/#heroku
2	IBM Watson IoT Platform	Creating and configurin	Depends on the Device Crede	https://ci5v5e.internetofthings.ibmcloud.com/dashboard/devices/browse
3	Node-Red	Creating Web-UI	Depends on the sensor value	https://nodered.org/
4	MIT App Devoloper	Developing Mobile app	Depends on the Sensor value	https://appinventor.mit.edu/about/termsofservice
5	Cloudant DB	Storing Sensor values	Depends on the Sensor value	https://2e953d0c-d7c4-4584-a048-34d5dc603b28-bluemix.cloudant.com/dashboard.html#/_all_dbs

End Of Test Report

S.No	Project Overview	NFT Test approach	NFR - Met	Test Outcome	GO/NO-GO decision	Recommendations	Identified Defects (Detected/Closed/Open)	Approvals/SignOff
1	ame sensor and tem	This is done by develc	Met	Pass	GO	Code working properly	Closed	https://www.python.org/psf/sponsors/#heroku
2	Based on the tempe	This is done by creati	Met	Pass	GO	Sprinkler is turning on and off	Closed	https://node-red-dahfa-2022-11-13.eu-gb.mybluemix.net/red/#flow/591d82622c4ff83
3	If any flame is detec	This is done by creati	Met	Pass	GO	Exhaust fan is turning on and	Closed	https://node-red-dahfa-2022-11-13.eu-gb.mybluemix.net/red/#flow/591d82622c4ff83
4	Emergency alerts are notified to the autho		Met	Pass	GO	Emergency alerts are send vi	Closed	https://www.fast2sms.com/dashboard/sms/bulk

10 .ADVANTAGES& DISADVANTAGES

TheAdvantagesofthisIndustry-SpecificIntelligentFireManagementsystemareasfollows

- Theuserneednotrequireexpertiseknowledgetocontrolthissystem.Thissystemissimple.Theusercaneasilyview thesensorvaluesandtake controlactions.
- Thecontrolactionsaretakenautomatically.
- Ifitisimplementedinhardware,thenthecostofimplementationwillbeaffordable.
- Aswearesensingthesensorvaluescontinuously,anyslightchangeintheenvironmentisdetected
- ThissystemisinUser-Friendlyformat.

TheDisadvantageofthisIndustry-SpecificIntelligentFireManagementsystemareasfollows

- Thissystemwillnotbeabletodetecttheoriginoffire.
- Thissystemwillnotprovidetheescaperouteifthereisfireoutbreak.
- Iftheindustryhasspecificchangesintheenvironment,thenthissystemwillgivesfalsealarm.

11.CONCLUSION

An understanding and having Fire Management system in the industry is of utmostimportance .This project is a fire management system that can be user in the industry based onIOT.This system creates a simulation device credentials in IBM WATSON IOT PLATFORM. in node-red,necessary nodes are installed and used.These nodes are installed and used.These nodes aredeployedandthedataiscollected.Intheeventoffire,thissystemcanissuesprinkleron,exhaust fan on.This remote user monitoring system can monitor the system status of eachnodeinrealtime.Thissystem monitors the data continuously so that the any slight change inthe environment can be easily detected.This ensures good control accuracy .This Industry-SpecificIntelligent Fire Management ensures the protection of property, asset and the processes are costeffectiveand theautomatic measuresare incontrol.

12.FUTURE SCOPE

Thefuturescopeofthisprojectistoaddadditionalfeaturesliketriggeringtheextinguisherautomatically, predict the escape route if the fire outbreaks and to implement this system inrealtime using hardware**.**

13.APPENDIX

Source Code

```
#IBM Watson IOT Platform

#pip install wiotp-sdk

import wiotp.sdk.device

import time

import random

myConfig = {

    "identity": {

        "orgId": "ci5v5e",

        "typeId": "Rasberypi",

        "deviceId":"1234"

    },

    "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(-5,100)

    flame=random.randint(0,10)

    gas=random.randint(0,100)

    if temp>50 or gas>50:

        if flame>8 and temp>50:


myData={'temperature':temp,'flame':flame,'gas':gas,'exhaust':1,'sprinklers':
1}

        else:


myData={'temperature':temp,'flame':flame,'gas':gas,'exhaust':1,'sprinklers':
0}
```

```
 else:


myData={'temperature':temp,'flame':flame,'gas':gas,'exhaust':0,'sprinklers':
0}

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

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

client.commandCallback = myCommandCallback

time.sleep(5)

client.disconnect()
```

GitHub link:

https://github.com/IBM-EPBL/IBM-Project-26263-1660022715

Video link:

https://youtu.be/pCsSLY8IoCA