PROJECT REPORT

DATE	19 NOVEMBER 2022
PROJECT ID	PNT2022TMID16776
PROJECT NAME	SMART SOLUTION FOR RAILWAYS

1. INTRODUCTION
1.1 Project Overview
1.2 Purpose
2. LITERATURE SURVEY
2.1 Existing problem
2.2 References
2.3 Problem Statement Definition
3. IDEATION & PROPOSED SOLUTION
3.1 Empathy Map Canvas
3.2 Ideation & Brainstorming
3.3 Proposed Solution
3.4 Problem Solution fit
4. REQUIREMENT ANALYSIS
4.1 Functional requirement
4.2 Non-Functional requirements
5. PROJECT DESIGN
5.1 Data Flow Diagrams
5.2 Solution & Technical Architecture
5.3 User Stories
6. PROJECT PLANNING & SCHEDULING
6.1 Sprint Planning & Estimation
6.2 Sprint Delivery Schedule
6.3 Reports from JIRA
7. CODING & SOLUTIONING (Explain the features added in the project along with code)
7.1 Feature 1
7.2 Feature 2
7.3 Database Schema (if Applicable)
8. TESTING
8.1 Test Cases
8.2 User Acceptance Testing
9. RESULTS
9.1 Performance Metrics
10. ADVANTAGES & DISADVANTAGES
11. CONCLUSION
12. FUTURE SCOPE
13. APPENDIX
Source Code
GitHub & Project Demo Link

1.INTRODUCTION

1.1 PROJECT OVERVIEW:

The future of the railway industry is expected to rely upon smart transportation systems that leverage technologies over a large rail network infrastructure to reduce its life-cycle cost. New services, such as integrated security, asset management, and predictive maintenance, are expected to improve timely decision-making for issues like safety, scheduling, and system capacity. Smart railways represent a combination of interconnected technological solutions and components, as well as modern transportation infrastructure like automatic ticketing systems, digital displays, and smart meters. Smart sensors and analytics across the train engine, coaches, and tracks allow rail systems to be remotely checked and repaired before a small issue magnifies into huge trouble. Asset health monitoring through IoT insights implies less of maintenance delays and helps in extending the life of rail infrastructure.

1.2 PURPOSE:

The purpose of the Smart solutions for Railways is It combines software products to make more intelligent use of all rail assets, from tracks to trains, so companies can meet the increasing consumer demand for more efficient and safer services. More improvements are being introduced in almost all fields to reduce human effort and save time. Thinking of the same is trying to introduce automation in the field of track testing. Railroad track is an integral part of any company's asset base, since it provides them with the necessary business functionality. Problems that occur due to problems in railroads need to be overcome. The latest method used by the Indian railroad is the tracking of the train track which requires a lot of manpower and is time-consuming

2. LITERATURE SURVEY

2.1 EXISTING PROBLEM:

SURVEY ON RAILWAY SAFETY:

In the current railway systems, it is becoming more necessary to have safety elements in order to avoid accidents. The problem of landslides can also happen at entrances and exits of tunnels. In these critical areas, systems are usually located to detect the presence of obstacles, so they can inform about it to the control system. There are a lot of dedicated sensorial systems installed in the level crossing area to avoid collision between trains and automobiles, captured on the railway when the crossing gates have started down.

SURVEY ON TICKET BOOKING SYSTEM:

The railway ticketing system is implemented on AWS public cloud using services, such as AWS IoT, Lambda and Dynamo DB. Even supervised machine learning is done in lambda to get some useful insights. Paper tickets also generates a lot of paper waste on a daily basis from millions of commuters travelling from various source to destination. The introduction of technologies has only paved way for quick & fast computing and also information gathering which are directly or indirectly beneficial for the authorities. The Automatic Ticket Vending Machine was introduced to scrap CVM coupons and avoid long queues. But ATVM machines still does not scrap the idea of paper ticketing. Mobile application was further introduced to make it more feasible and convenient for the consumers to book tickets online. But this type of technology requires Internet provisions and Smart phones to avail these

services.

SURVEY ON GPS TRACKING SYSTEM:

Railway is the most commonly used transportation vehicle. Most of the people choose this transportation mainly for low cost and it gives comfort ability. To increase this comfort zone and to reduce the number of accidents, control over the railway level crossing gate is done through smart phones by the engine driver. The existing system aimed to avoid the accidents in the railway tracks. It also planned to reduce the manpower in railway gates. It omits the entire man power in the railway gates. There are many applications to track the location of the train which makes the passengers to arrive the junction at the right time. Also, there is a remainder which alerts the passenger before 5 or 10 minutes the junction arrives.

2.2 REFERENCES:

1. S. Srivastava, R. P. Chourasia, P. Sharma, S. I. Abbas, N. K. Singh, “Railway Track Crack detection vehicle”, IARJSET, Vol. 4, pp. 145-148, Issued in 2, Feb 2017.

2.U. Mishra, V. Gupta, S. M. Ahzam and S. M. Tripathi, “Google Map Based Railway Track Fault Detection Over the Internet”, International Journal of Applied Engineering Research, Vol. 14, pp. 20-23, Number 2, 2019.

3.R. A. Raza, K. P. Rauf, A. Shafeeq, “Crack detection in Railway track using Image processing”, IJARIIT, Vol. 3, pp. 489-496, Issue 4, 2017.

4.N. Bhargav, A. Gupta, M. Khirwar, S. Yadav, and V. Sahu, “Automatic Fault Detection of Railway Track System Based on PLC (ADOR TAST)”, International Journal of Recent Research Aspects, Vol. 3, pp. 91-94, 2016

2.3 PROBLEM STATEMENT DEFINITION:

Among the various modes of transport, railways is one of the biggest modes of transport in the world. Though there are competitive threats from airlines, luxury buses, public transports, and personalized

transports the problem statement is to answer the question “What are the problems faced by the passengers while travelling by train at station and on board” The following are the problem statements which are been discussed in this project.

- A Web page is designed for the public where they can book tickets by seeing the available seats.
- After booking the train, the person will get a QR code which has to be shown to the Ticket Collector while boarding the train.
- The ticket collectors can scan the QR code to identify the personal details.
- A GPS module is present in the train to track it. The live status of the journey is updated in the Web app continuously
- All the booking details of the customers will be stored in the database with a unique ID and they can be retrieved back when the Ticket Collector scans the QR Code.

3. IDEATION & PROPOSED SOLUTION

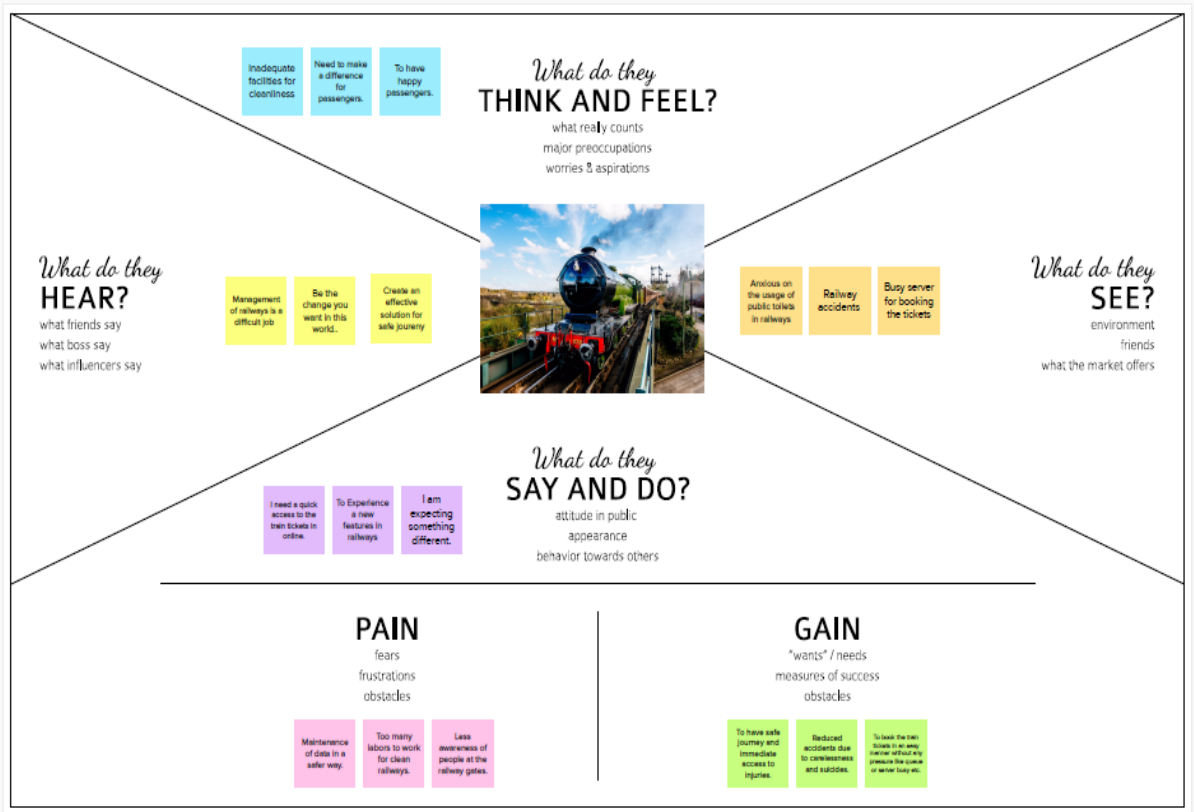
3.1 Empathy Map Canvas:

Empathy Map Canvas

Gain insight and understanding on solving customer problems.

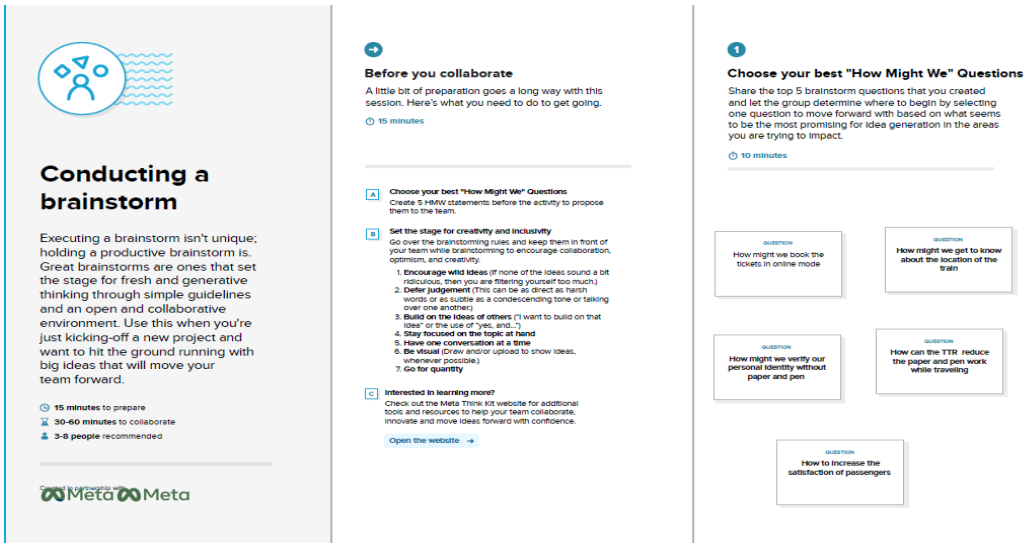
1

Build empathy and keep your focus on the user by putting yourself in their shoes.

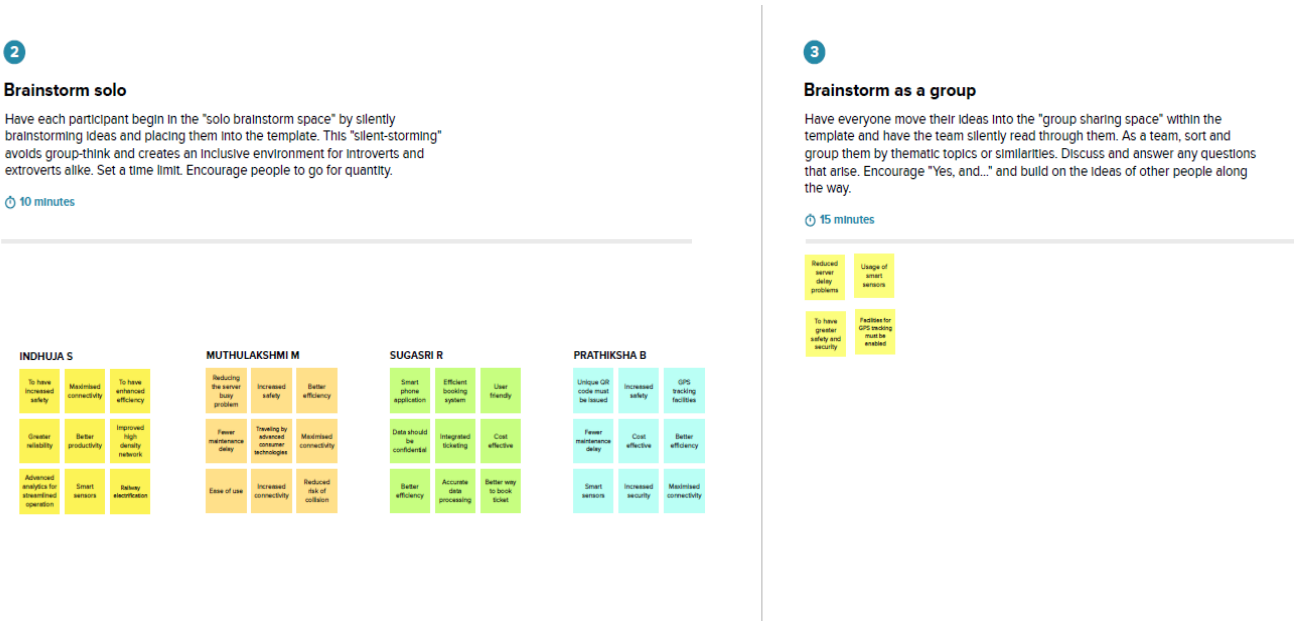


3.2 Ideation & Brainstorming :

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



Step-2: Brainstorm, Idea Listing and Grouping



Step-3: Idea Prioritization

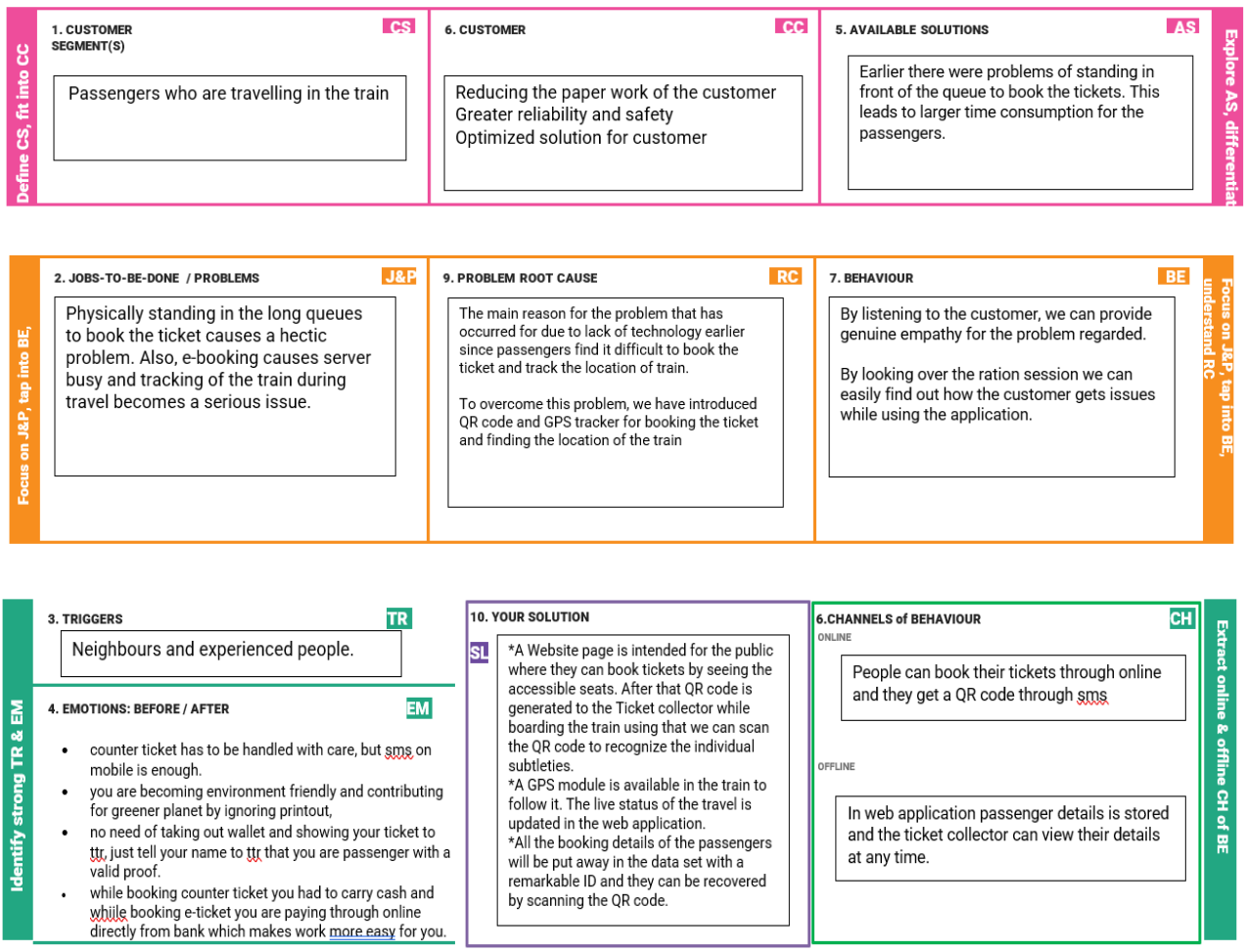
3.3 Proposed Solution:

S. No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	On-site ticket booking may take lot of time and there is a issue of loosing their manual tickets. Even in online booking we should have a copy of ticket as softcopy, in case if that ticket gets erased or lost it will be sometimes difficult to retrieve it. Here we need to show the printed copy or soft copy of tickets and ID card proofs to Ticket checker.
2.	Idea / Solution description	Book tickets using QR Code in railway ticket booking system. We get the details of the passengers. We track the current location of the particular train. We provide unique ID for passengers to secure their information and we will have

		chatbot for customer queries.
3.	Novelty / Uniqueness	• Efficient booking system, verifying validity of the ticket and only register user can book the tickets. • Each passenger will be provided by giving a unique ID to them during first login so that their data will be stored and processed securely. • GPS tracking facility will be provided to track the current location • We provide chatbot for customers queries and that will be solved as soon as possible.
4.	Social Impact / Customer Satisfaction	• Passenger data will be more securely maintained • Prefect way to reserve tickets • User friendly environment • Query section for customer
5.	Business Model (Revenue Model)	Using chat bot, we can contact user's ticket booking. The chat bot can give instructions to the users based on their location. It will store the customer's details and ticket orders in the database. The chat bot will send a notification to customers if the booking is

		confirmed. Chat bots can also help in collecting customer feedback.
6.	Scalability of the Solution	This model can be easily adopted among online users and it can be easily deployed. It can be used and accessed by everyone and it can handle the requests from the customers

3.4 Problem Solution fit:



4. REQUIREMENT ANALYSIS

4.1 Functional requirement:

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Form
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	User QR code generation	QR code is generated
FR-4	GPS tracker	Location is tracked

4.2 Non-functional Requirements:

Following are the non-functional requirements of the proposed solution.

FR No.	Non-Functional Requirement	Description
NFR-1	Usability	Users can navigate easily
NFR-2	Security	The details are secured in the database
NFR-3	Reliability	Reliable to the users without any failure as it is not fixed to limited number of users
NFR-4	Performance	User-friendly
NFR-5	Availability	Available any time at the time of ease
NFR-6	Scalability	Support the users with their needs in reserving ticket and tracking the location.

5.PROJECT DESIGN

5.1 DATA FLOW DIAGRAM:

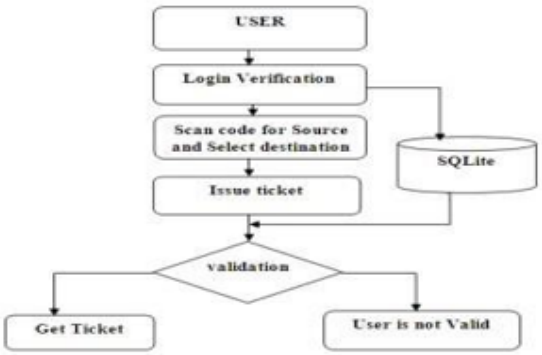
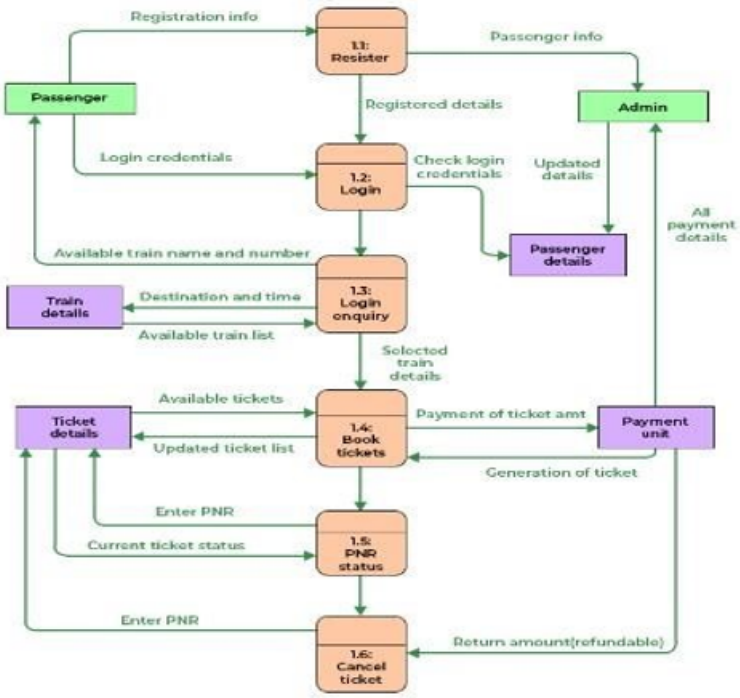


Fig 2. Flow Process of Ticket Booking.



5.2 SOLUTION AND TECHNICAL ARCHITECTURE:

Solution Architecture:

1.If a ticket is need to be booked the passenger can use this qr code method to book their ticket. The python code is written regarding the passengers details to be collected from them. The details are safely stored in a database.

2.Once the details are given the passenger will receive a qr code which is been generated. The qr code is shown to the TTR and he will verify the details of the passenger.

3. After booking the ticket the passenger will get the sms to their email for the successful confirmation of the ticket and the qr code is generated.

4.Also not only qr code is generated using ibm Cloud we are fixing the latitude and longitude of the map where the location of the train is tracked easily by using the GPS tracker.

5.The Passenger can be able to track the location of the train and can be cautious which avoids missing of train because the trains location is easily tracked by using the GPS tracker and the latitute and longitute are used for locating the place.

6.The Iot devices are used from iot platform which connects to the cloud services that provide the storage of details given by the passenger and there will be a unique id issued to each and every passenger in order to avoid collision.

Example - Solution Architecture Diagram:

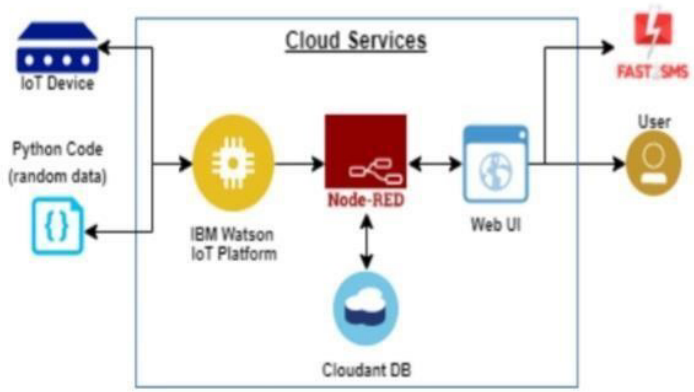


Figure 1: Architecture and data flow of t*he smart solutions for railways*

5.3 USER STORIES:

Sprint	Functional Requirement(Epic)	User Story Numb er	User Story/Task	Story Points	Priority
Sprint-1	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password	15	High

Sprint-1	Reservation & confirmation	USN-2	As a passenger, I can book my tickets and confirm the reservation through receiving mail or sms	5	Medium
Sprint-2	Payment	USN-3	As a passenger, I want to pay my ticket cost in online payment	15	High
Sprint-3	Service Provider	USN-4	The user can clear their doubts by connecting to the service provider	5	Medium
Sprint-3	Service Provider (Admin)	USN-5	Timings and status of the train, which will be updated in the database	10	Medium
Sprint-4	Verifying Tickets	USN-6	As a TC, I want to check the users whether he/she have tickets or not with scanning the QR Code	15	High
Sprint-4	Raise compliant	USN-7	As a user, I should able to raise a ticket if something is wrong	10	Medium

6. PROJECT PLANNING AND SCHEDULING

6.1 SPRINT PLANNING AND ESTIMATION:

Sprint	Functional Requirement(Epic)	User Story Number	User Story/Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password	15	High	Indhuja S Sugasri R Muthulakshmi M Prathiksha B
Sprint-1	Reservation & confirmation	USN-2	As a passenger, I can book my tickets and confirm the reservation through receiving mail or sms	5	Medium	Indhuja S Sugasri R Muthulakshmi M Prathiksha B
Sprint-2	Payment	USN-3	As a passenger, I want to pay my ticket cost in online payment	15	High	Indhuja S Sugasri R Muthulakshmi M Prathiksha B
Sprint-3	Service Provider	USN-4	The user can clear their doubts by connecting to the service provider	5	Medium	Indhuja S Sugasri R Muthulakshmi M Prathik

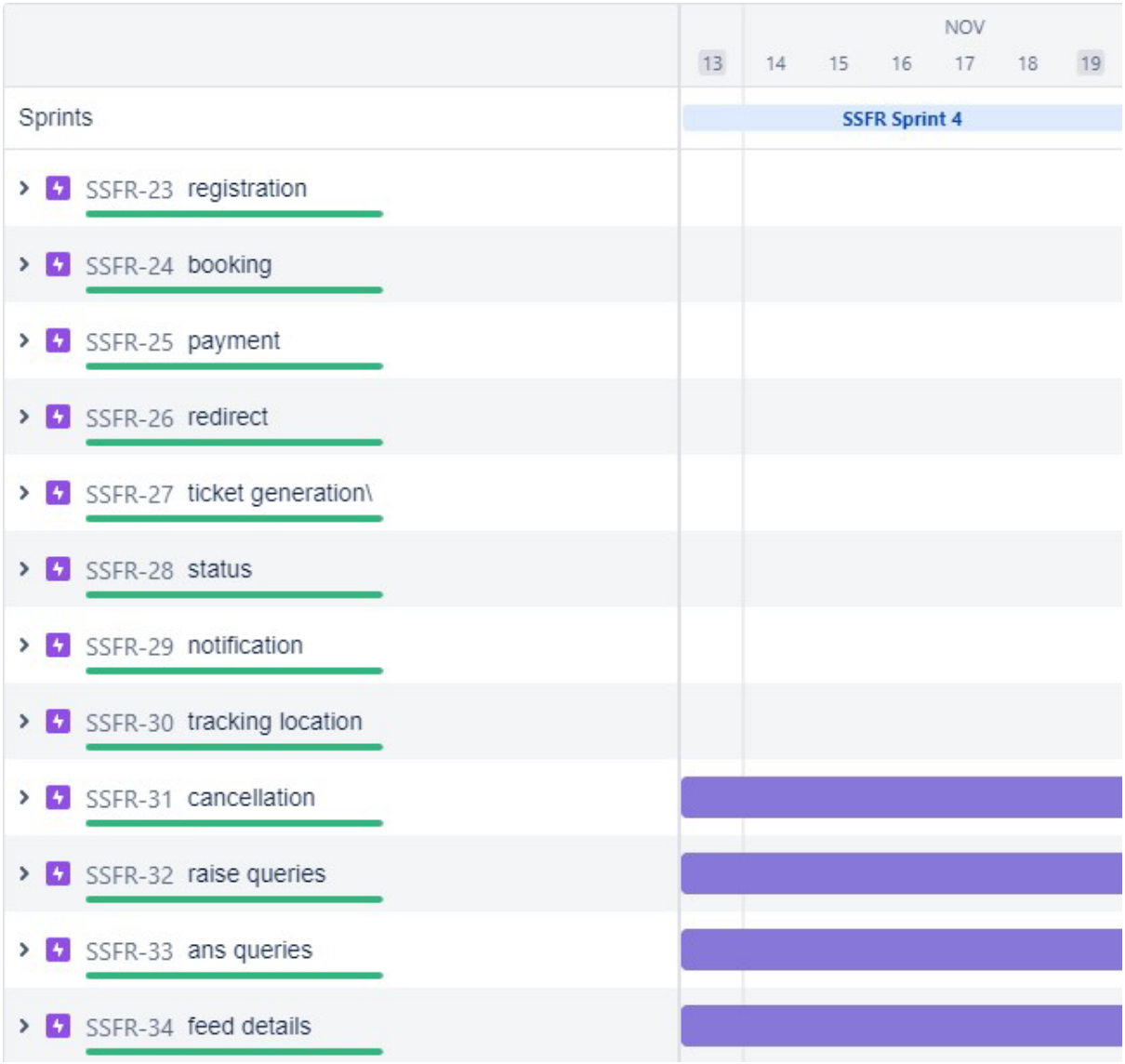
						sha B
Sprint-3	Service Provider (Admi n)	USN-5	Timings and status of the train, which will be updated in the database	10	Medium	Indhuja S Sugasri R Muthulaks hmi M Prathiksha B

Sprint-4	Verifying Tickets	USN-6	As a TC, I want to check the users whether he/she have tickets or not with scanning the QR Code	15	High	Indhuja S Sugasri R Muthulak shmi M Prathi ksha B
Sprint-4	Raise compliant	USN-7	As a user, I should able to raise a ticket if something is wrong	10	Medium	Indhuja S Sugasri R Muthulak shmi

6.2 SPRINT DELIVERY SCHEDULE:

Sprint	Total Story Points	Duration	Sprint StartDate	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	5Days	24Oct2022	1 Nov2022	20	15 Nov 2022
Sprint-2	20	5Days	31Oct2022	05Nov2022	20	17 Nov2022
Sprint-3	20	5Days	07Nov2022	12Nov2022	20	19Nov2022
Sprint-4	20	5Days	14Nov2022	19Nov2022	20	19Nov2022

6.3 REPORTS FROM JIRA:



7. CODING AND SOLUTIONING

7.1 FEATURE 1:

1. IOT device
2. IBM Watson platform
3. Node red
4. Cloudant DB
5. Web UI
6. Geofence MIT App
7. Python code

7.2. FEATURE 2

a. Registration
b. Login
c. Verification
d. Ticket Booking
e. Payment
f. Ticket Cancellation
g. Adding Queries

8. TESTING

WEBPAGE FOR TRAIN BOOKING:

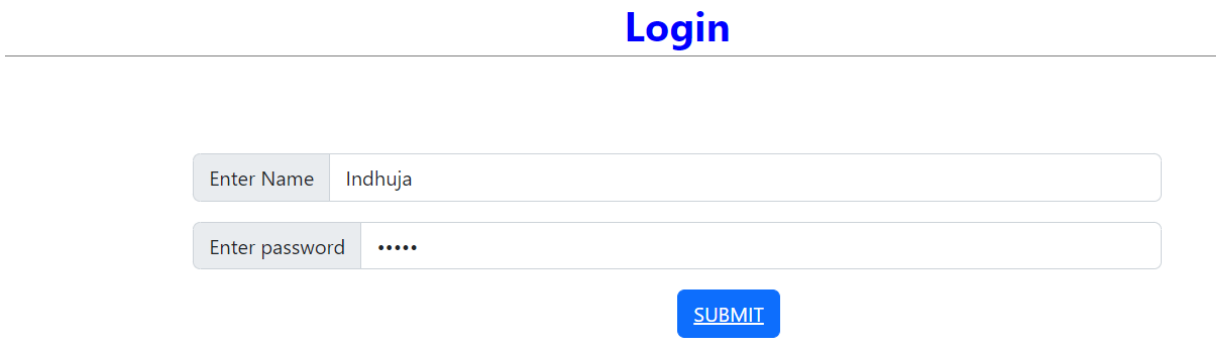
STEP – 1:

Welcome to Railway Ticket Booking !!!

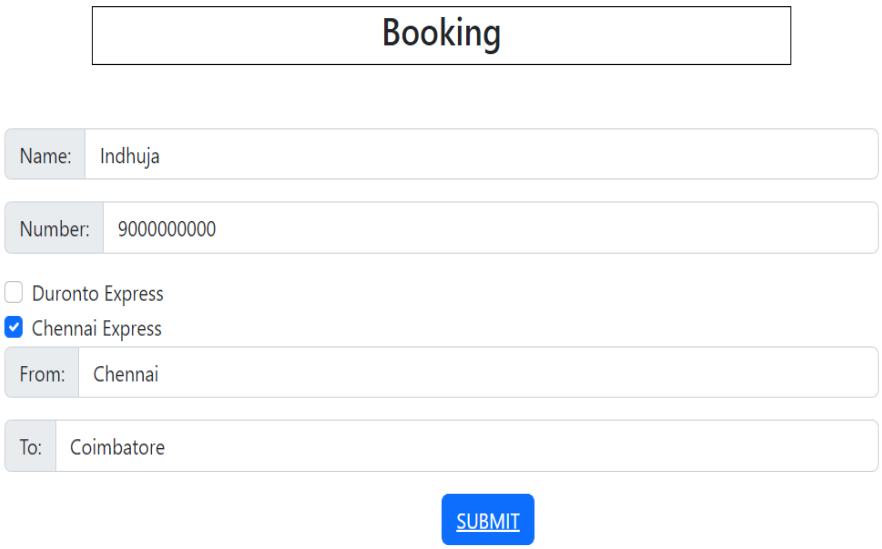
Ticket Booking

Report Issues

STEP – 2:


Login
Enter Name
Indhuja
Enter password
SUBMIT

STEP -3:


Booking
Name:
Indhuja
Number:
9000000000
Duronto Express
Chennai Express
From:
Chennai
To:
Coimbatore
SUBMIT

STEP – 4: (AFTER DOING PAYMETS FOLLOWING WEBPAGE WILL BE DISPLAYED)

Ticket Booked Successfully!!!

Your Booked QR Code

HOME PAGE

9. RESULTS

9.1 PERFORMANCE METRICS:

10. ADVANTAGES AND DISADVANTAGES

10.1. ADVANTAGES

- Openness – compatibility between different system modules, potentially from different vendors;
- Orchestration – ability to manage large numbers of devices, with full visibility over them; o Dynamic scaling – ability to scale the system according to the application needs, through resource virtualization and cloud operation;
- Automation – ability to automate parts of the system monitoring application, leading to better performance and lower operation costs.

10.2. DISADVANTAGES

- Approaches to flexible, effective, efficient, and low-cost data collection for both railway vehicles and infrastructure monitoring, using regular trains;
- Data processing, reduction, and analysis in local controllers, and subsequent sending of that data to the cloud, for further processing;
- Online data processing systems, for real-time monitoring, using emerging communication technologies;
- Integrated, interoperable, and scalable solutions for railway systems preventive maintenance.

11. CONCLUSION

Accidents occurring in Railway transportation system cost a large number of lives. So this system helps us to prevent accidents and giving information about faults or cracks in advance to railway authorities. So that they can fix them and accidents cases becomes less. This project is cost effective. By using more techniques they can be modified and developed according to their applications. By this system many lives can be saved by avoiding accidents. The idea can be implemented in large scale in the long run to facilitate better safety standards for rail tracks and provide effective testing infrastructure for achieving better results in the future.

12. FUTURE SCOPE

In future CCTV systems with IP based camera can be used for monitoring the visual videos captured from the track. It will also increase security for both passengers and railways. GPS can also be used to detect exact location of track fault area, IP cameras can also be used to show fault with the help of video. Locations on Google maps with the help of sensors can be used to detect in which area track is broken.

13. APPENDIX

SOURCE CODE:

FUNCTION NODE COMMAND TO INDICATE THE AVAILABLE SEATS:

```
var a=global.get('a') var s= []

for(let

i=0;i<a.length==0;i++){ s.p

ush(a[i]) } if(s.length==0){

 msg.options=[{"No seats available":0}]

}

else{ msg.

options=

s

}

msg.paylo

ad= s

return msg;
```

FUNCTION NODE COMMAND TO CHOOSE THE AVAILABLE SEATS:

```
var s=global.get('s') var

a=global.get('a') function

reg(x){ for(let

i=0;i<a.length;i++){ if(a[i]==x)

{

 a.splice(i,1)

 }

 } }

if(s==1){ gl

obal.set('s

1',s) reg(s) }

else
```

```
if(s==2){ gl
obal.set('s
2',s) reg(s) }
else
if(s==3){ gl
obal.set('s
3',s) reg(s) }
else
if(s==4){ gl
obal.set('s
4',s) reg(s)
}
else
if(s==4){ gl
obal.set('s
4',s) reg(s) }
return msg;
```

FUNCTION NODE COMMAND TO STORE DATA IN DATABASE:

```
var m=global.get('m')
var d=new Date();
var utc=d.getTime()+(d.getTimezoneOffset()*60000); var
offset=5.5;
newDate=new
Date(utc+(3600000*offset)); var
n=newDate.toISOString() var
date=n.slice(0,10) var time=n.slice(11,19)
var d1=date+','+time msg.payload={
```

```
 "_id":d1,
 "Name":m.Name,
"Age":m.Age,
 "Mobile":m.Num,
 "boarding":global.get('b'),
 "destination":global.get('d'),
 "Seat":global.get('s')
}
return
msg;
```

PYTHON SCRIPT TO SCAN QR CODE:

```
import cv2 import numpy as np import time import pyzbar.pyzbar
as pyzbar from pyzbar.pyzbar import decode from
ibmcloudant.cloudant_v1 import CloudantV1 from ibmcloudant
import CouchDbSessionAuthenticator from
ibm_cloud_sdk_core.authenticators  import BasicAuthenticator
authenticator = BasicAuthenticator('apikey-v2-
125rwcp4ifi6zz2ly1cq0kakyjn98du2ysgc72h53lzi',
'af693938842290ec2c254461754447b5') service =
CloudantV1(authenticator=authenticator)
service.set_service_url('https://apikey-v2-
125rwcp4ifi6zz2ly1cq0kakyjn98du2ysgc72h53lzi:af693938842290ec2c254461754447b5@82d874994395-
4f46-a190-6a186bee5051-bluemix.cloudantnosqldb.appdomain.cloud')
cap= cv2.VideoCapture(0) font =
cv2.FONT_HERSHEY_PLAIN while
True:
 _, frame = cap.read() decodedObjects = pyzbar.decode(frame)
for obj in decodedObjects: #print ("Data", obj.data)
a=obj.data.decode('UTF-8') cv2.putText(frame, "Ticket", (50,
```

```
50), font, 2, (255, 0, 0), 3)
 #print (a)
 try:
 response = service.get_document(db='booking',doc_id = a).get_result()
print(response) time.sleep(5) except Exception as e:
 print("NOT A VALID TICKER")
time.sleep(5)

 cv2.imshow("Frame",frame) if
cv2.waitKey(1) & 0xFF ==ord('q'):
 break
cap.release() cv2.destroyAllWindows()
client.disconnect()
```

PYTHON CODE FOR GPS:

```
import wiotp.sdk.device
import time import
random myConfig = {
"identity": {
"orgId": "12345",
"typeId": "123",
"deviceId":"123"
},
"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()

def pub (data):

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

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

while True:

myData={'name': 'Train1', 'lat':13.08363 , 'lon': 80.27080}

pub (myData)

time.sleep (2)

myData={'name': 'Train2', 'lat': 12.40797, 'lon': 79.81410}

pub (myData)

time.sleep (2)

myData={'name': 'Train1', 'lat': 11.83331, 'lon': 79.37465}

pub(myData)

time.sleep(6)

myData={'name': 'Train1', 'lat': 11.59664, 'lon': 78.69899}

pub (myData)

time.sleep (6)

myData={'name': 'Train1', 'lat': 11.63431, 'lon': 78.11122}

pub (myData)

time.sleep (6)

myData={'name': 'Train1', 'lat': 11.32207, 'lon': 77.61684}

pub (myData)

time.sleep (6)

myData={'name': 'Train1', 'lat': 11.03107, 'lon': 76.96864}

pub (myData) time.sleep (6)

client.commandCallback = myCommandCallback

client.disconnect ()
```

GITHUB LINK:

https://github.com/IBM-EPBL/IBM-Project-19902-1659708735/tree/main

DEMO LINK

https://drive.google.com/file/d/1teCIAFXZDsEqv_sa3DQGXl35UCo2CZar/view?usp=drivesdk