

# **SMART SOLUTIONS FOR RAILWAYS**



## **PET ENGINEERING COLLEGE**

**| Accredited by NAAC with 'B' Grade  
Affiliated to Anna University |  
Approved by AICTE  
Vallioor,  
Tamil Nadu-627117**

*In partial fulfillment for the award of the degree*

*of*

**BACHELOR OF ENGINEERING**

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING  
NOVEMBER 2022**

**A PROJECT REPORT**

*Submitted by*

<b>SNEHA A(Team leader)</b>	<b>(RollNo:963219104034)</b>
<b>LEBILA M</b>	<b>(RollNo:963219104012)</b>
<b>MELDA M</b>	<b>(RollNo:963219104014)</b>
<b>MOHAMMED LAMEES UMAMA S S</b>	<b>(Roll No:963219104019)</b>

# **PROJECT REPORT**

## **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**

7.1 Feature1

7.2 Feature2

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**

# **1. INTRODUCTION**

## **1.1 PROJECTOVERVIEW**

Smart Solutions For Railways is to manage Indian Railways is the largest railway network in Asia and additionally world's second largest network operated underneath a single management. Due to its large size it is difficult to monitor the cracks in tracks manually. This paper deals with this problem and detects cracks in tracks with the help of ultrasonic sensor attached to moving assembly with help of stepper motor. Ultrasonic sensor allows the device to moves back and forth across the track and if there is any fault, it gives information to the cloud server through which railway department is informed on time about cracks and many lives can be saved. This is the application of IoT, due to this it is cost effective system. This effective methodology of continuous observation and assessment of rail tracks might facilitate to stop accidents. This methodology endlessly monitors the rail stress, evaluate the results and provide the rail break alerts such as potential buckling conditions, bending of rails and wheel impact load detection to the concerned authorities.

## **1.2. PURPOSE**

Internet is basically system of interconnected computers through network. But now its use is changing with changing world and it is not just confined to emails or web browsing. Today's internet also deals with embedded sensors and has led to development of smart homes, smart rural area, e-health care's etc. and this introduced the concept of IoT . Internet of Things refers to interconnection or communication between two or more devices without humanto-human and human-to-computer interaction. Connected devices are equipped with sensors or actuators perceive their surroundings. IOT has four major components which include sensing the device, accessing the device, processing the information of the device, and provides application and services. In addition to this it also provides security and privacy of data . Automation has affected every aspect of our daily lives. 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**

### **2.1EXISTINGSYSTEM**

In the Existing train tracks are manually researched. LED (Light Emitting Diode) and LDR (Light Dependent Resister) sensors cannot be implemented on the block of the tracks ]. The input image processing is a clamorous system with high cost and does not give the exact result. The Automated Visual Test Method is a complicated method as the video color inspection is implemented to examine the cracks in rail track which does not give accurate result in bad weather. This traditional system delays transfer of information. Srivastava et al., (2017) proposed a moving gadget to detect the cracks with the help of an array of IR sensors to identify the actual position of the cracks as well as notify to nearest railway station . Mishra et al., (2019) developed a system to track the cracks with the help of Arduino mega power using solar energy and laser. A GSM along with a GPS module was implemented to get the actual location of the faulty tracks to inform the authorities using SMS via a link to find actual location on Google Maps. Rizvi Aliza Raza presented a prototype in that is capable of capturing photos of the track and compare it with the old database and sends a message to the authorities regarding the crack detected. The detailed analysis of traditional railway track fault detection techniques is explained in table.

## 2.2 References

- [1] S.Karthick and A. Velmurugan, “Android sub urban railway ticketing with GPS as ticket checker,” *2012 IEEE International Conference on Advanced Communication Control and Computing Technologies(ICACCCT)*, pp.63-66,2012.
- [2]C.Ulianov,P.HydeandR.Shaltout,“RailwayApplicationsforMonitoringandTrackingSystems,”  
*Marinov,M.(eds)SustainableRailTransport.LectureNotesinMobility.springer,Cham.,2018.*
- [3]B. Mallikarjuna, A. K. R. Doddi and G.Sailaja,“Enhanced Railway Reservation System using Internet of Things,” *2018 IADS International Conference on Computing, Communications & Data Engineering(CCODE)*,2018.
- [4]G. Shelar, V. Rathod and S. Patil, “Railway Ticket Booking System with Restricted Wi-Fi Zone,” *International Journal of Trend in Scientific Research and Development (ijtsrd)*, vol. 2, no. 4, pp.611-615,2018.
- [5]Swarup, M. Mohan, A. Dwivedi, C. Sonkar, R. Prasad, M. Bag and V. Singh, “A QR code based processing for dynamic and transparent seat allocation in Indian railway,” *International Journal of Computer Science Issues (IJCSI)* 9, no.3(2012),p.338, 2012.

## 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 onboard

**Ideation Phase**  
Define the Problem Statements

**Customer Problem Statement Template:**  
Create a problem statement to understand your customer's point of view. The Customer Problem Statement template helps you focus on what matters to create experiences people will love.  
A well-articulated customer problem statement allows you and your team to find the ideal solution for the challenges your customers face. Throughout the process, you'll also be able to empathize with your customers, which helps you better understand how they perceive your product or service.

**Diagram:**

```
graph LR
    A[A Traveller] --> B[Book train tickets on my phone]
    B --> C[It takes long time to book]
    C --> D[The server has to be down]
    D --> E[disappointed]
```

**Table:**

Problem Statement (PS)	I am (Customer)	I'm trying to	But	Because	Which makes me feel
PS-1	A traveller	book train tickets on my phone	It takes long time to book	The server has to be down	disappointed
PS-2	A traveller	Go to Mumbai	The train is too late, the punctuality of the train is too poor	I couldn't reach my destination on time	disappointed

Words: 148

12:50 PM  
11/14/2022



<b>Problem Statement (PS)</b>	<b>I am (Customer )</b>	<b>I'm trying to</b>	<b>But</b>	<b>Because</b>	<b>Which makes me feel</b>
PS-1	A traveller	book train tickets on my phone.	It takes long time to booked.	The server has to be down.	disappointed
PS-2	A traveller	Go to Mumbai .	The train is too late,the punctuality of the train is too poor .	I couldn't reach my destination on time.	disappointed

### **3. IDEATION AND PROPOSED SOLUTION**

#### **3.1 EMPATHY MAP CANVAS**

An empathy map is a simple, easy-to-digest visual that captures knowledge about a user's behaviors and attitudes. It is a useful tool to help teams better understand their users. Empathy mapping is a simple workshop activity that can be done with stakeholders, marketing and sales, product development, or creative teams to build empathy for end users.



## Empathy map canvas

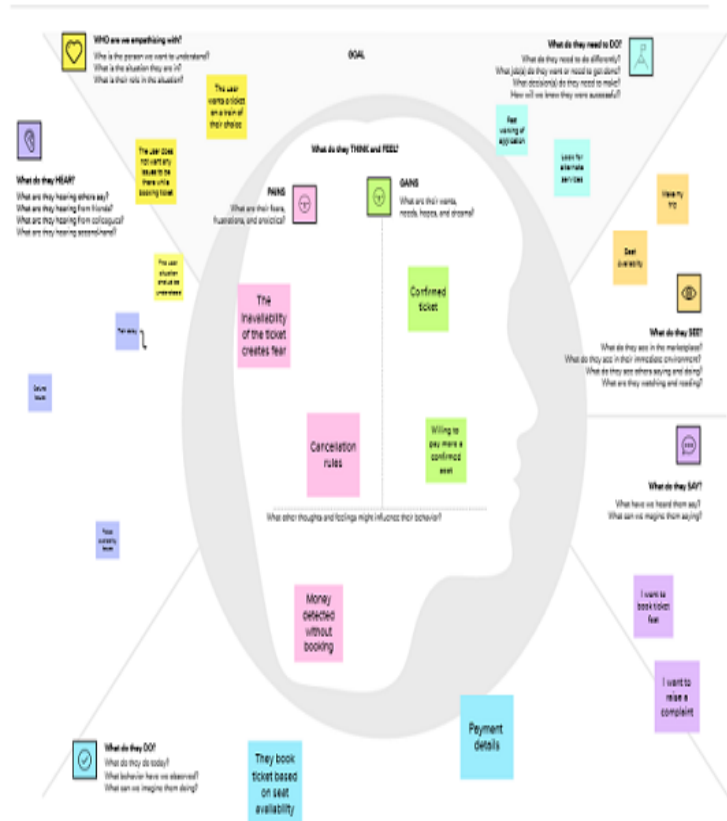
Use this framework to empathize with a customer, user, or any person who is affected by a team's work. Document and discuss your observations and note your assumptions to gain more empathy for the people you serve.

Originally created by Dave Gray at

[Share template feedback](#)

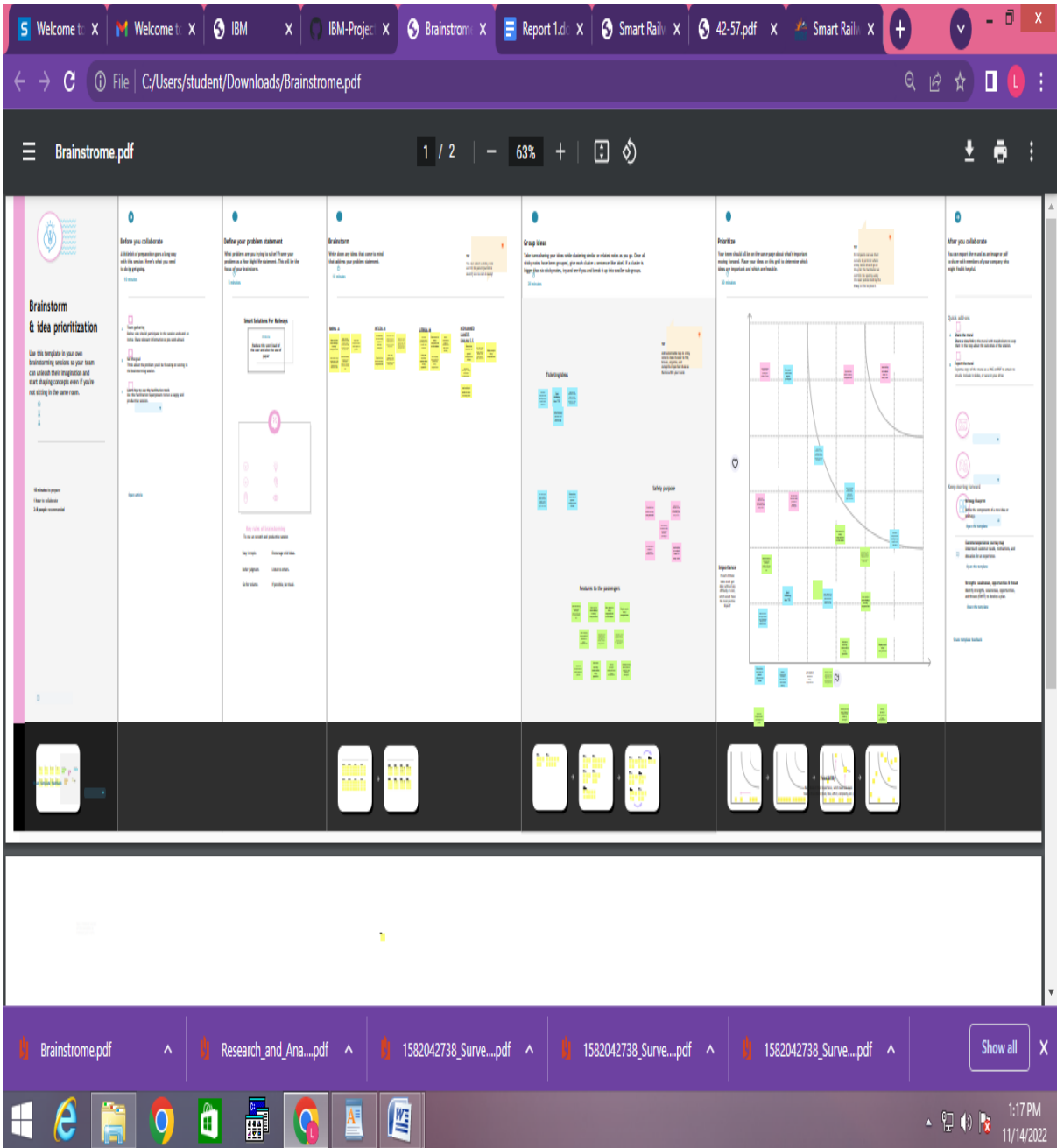
**Develop shared understanding and empathy**

Summarize the data you have gathered related to the people that are impacted by your work. It will help you generate ideas, prioritize features, or discuss decisions.



### **3.2 IDEATION & BRAINSTORMING**

Brainstorming combines a relaxed, informal approach to problem solving with lateral thinking. It encourages people to come up with thoughts and ideas that can, at first, seem a bit crazy. Some of these ideas can be crafted into original, creative solutions to a problem, while others can spark even more ideas. This helps to get people unstuck by "jolting" them out of their normal ways of thinking.



### 3.3ProposedSolution

S.NO	Parameter	Description
1.	Problem Statement (Problem to be solved)	While booking a train ticket the user takes long time to book and sometimes the ticket gets lost. As a solution for this an IOT based web application is introduced which reduces the work load and paper work.
2.	Idea / Solution description	Using this web app the user can check the seat availability while booking a ticket and instead of the ticket paper the QR code is developed for individual user.
3.	Novelty / Uniqueness	This web app enables the user to track the status as the GPS module is present and the status of the train is updated. By using this application user can know the current status like departures, arrivals, delays of the train and in this model the ticket paper is not needed.
4.	Social Impact / Customer Satisfaction	The loss of ticket paper at the last moment makes the user feel stress and the user has to face the loss of pay.

		The poor punctuality of the train makes the customer feel disappointed.
5.	Business Model (Revenue Model)	<p><b>Increased efficiency:</b> Congestion and over crowding create operational inefficiencies. Using deep learning and AI through computer vision, operators can monitor passenger flow and gather data for advanced analytics to help enable more-informed decision-making around staffing and security.</p> <p><b>Reduced downtime:</b> Sensors, cameras and in-vehicle computers empower rail operators to monitor their fleets diagnostic data to minimize breakdowns, predict maintenance repairs and optimize servicing schedules to keep trains in working order and moving.</p>

### **3.3. Problem Solution Fit**

#### **1.customer segment**

All Indian Railways passengers.

#### **2.Problems**

Smart solution for railways are designed to reduce the work load of the user,also the use of paper and to improve the usability of ticket maintenace.

#### **3.Triggers**

User may trigger while booking ticket,get a QRcode ,tracking train location.

#### **4.Available solution**

Currently tickets are verified manually by cross checking ticket number,name etc.Train location can found manually



## Problem Solution Fit:

### 1. CUSTOMER SEGMENT(S)

All Indian Railways  
Passengers

### 6. CUSTOMER CONSTRAINTS

Only one QR can be generated for one ticket. Customers are not allowed to recreate QR code.

### 5. AVAILABLE SOLUTIONS

Currently tickets are verified manually by cross checking ticker no, name etc...  
Train location can find manually

### 2. PROBLEMS

Smart Solutions for railways is designed to reduce the work load of the user, also the use of paper and to improve the usability of ticket maintenance.

### 9. PROBLEM ROOT CAUSE

The main root cause is to find originality of the ticket by verifying manually and also finding train current location.

### 7. BEHAVIOUR

This method will perform actively in ticket counters and it also used to find location of the train.

Verification of the ticket will become easier.

### 3. TRIGGERS

Users may trigger while booking a ticket, get a QR code, tracking train location.

### 10. YOUR SOLUTION

Our project is to develop user friendly webpage and to generate QR code for each ticket and also find the location of the train by using that QR code.

### 8. CHANNELS OF BEHAVIOUR

**Online:** By booking tickets online by entering all the data's of the passengers will be stored in database. Code is to find the location of the train.

**Offline:** Verification of the tickets will be easy to find the originality.

### 4. EMOTIONS : BEFORE / AFTER

**Before:** Customers may feel difficult in finding train details  
**After:** They can easily find train details, locations etc.

## 4. REQUIREMENT ANALYSIS

### 4.1. Functional requirement

FR No.	Functional Requirement(Epic)	Sub Requirement(Story/Sub-Task)
FR-1	User Registration	Registration through Gmail Registration through Facebook Registration through Mobile number
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP Confirmation via call
FR-3	Journey details	Provides From and To information and date of travel.
FR-4	Select Trains	Select the appropriate trains among the list and Also based on these at availability.
FR-5	Book and add passenger	Fill the essential details such as name,contact details age,sex.
FR-6	Proceed to pay	Select an appropriate payment options among UPI ,Internet Banking ,credit card, debit card.
FR-7	Ticket confirmation and Invoices	Ticket confirmation status is send to the irregistered email id
FR-8	GPS	Tracking the live location and the status will be updated to the passengers.
FR-9	GSM	To get a wake up call alarm prior before the destination is reached.
FR-10	Data base management	Entire Journey details will be stored in the server.
FR-11	E-catering	Foods are available for the registered passengers in an effective manner.

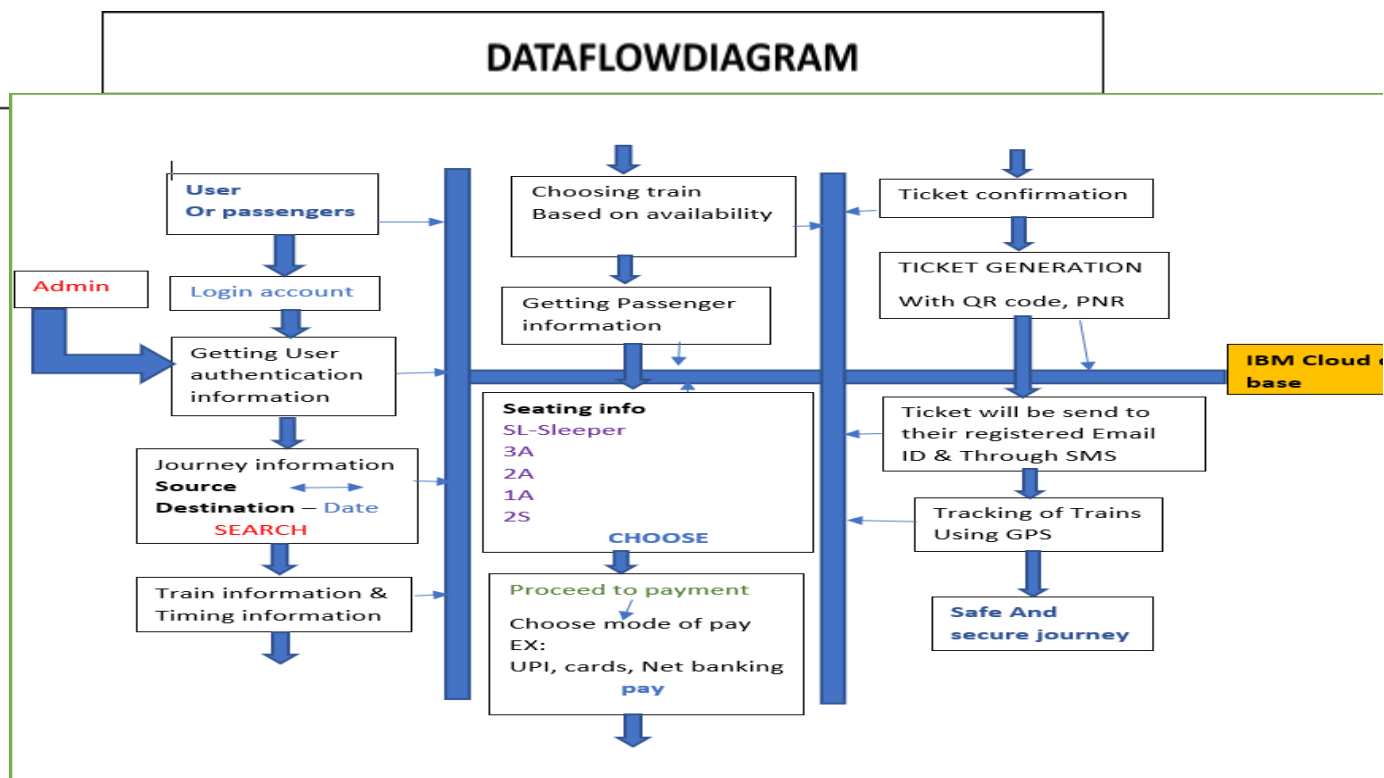
## 4.2. Non-Functional requirements

<b>FR No.</b>	<b>Non-Functional Requirement</b>	<b>Description</b>
NFR-1	<b>Usability</b>	Availability of e-tickets with QR generation instead of physical one.
NFR-2	<b>Security</b>	It protects the details of a passenger against Eaves dropping and denial of service attacks.
NFR-3	<b>Reliability</b>	It enables the user to securely use the app which provides maximum trust to the user.
NFR-4	<b>Performance</b>	No server down problems , many user can access at same the same time. Better performance is provided.
NFR-5	<b>Availability</b>	Accessibility through website or application anytime and from anywhere.
NFR-6	<b>Scalability</b>	No of users concurrently interacting with our web application with higher reliability.

## 5. PROJECT DESIGN

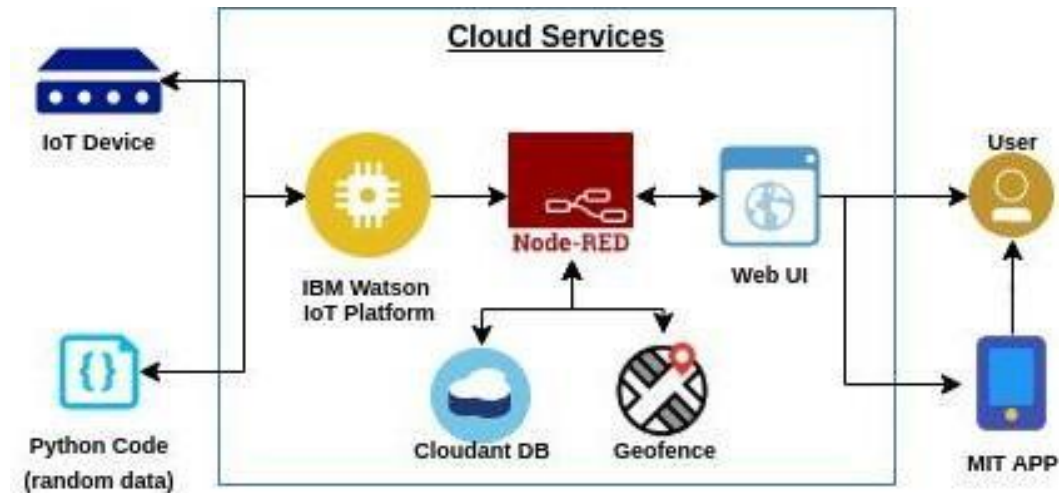
### 5.1. Data Flow Diagrams

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.



## 5.2. Solution & Technical architecture

Technical Architecture is the name of the total concept that is applied to the IT Infrastructure of an organization. IT Infrastructure is a coherent set of interconnected hardware and software, like networks, clouds, servers, clients, printers, tablet PC, smart phones.



### 5.3. User Stories

User Type	Functional Requirement(Epic)	User Story Number	User Story/Task	Acceptance criteria
Passenger	Registration	USN-1	As a passenger, I want to create a login credentials so I can securely access myself service online account.	Input data fieldsto enter 2.Password 3.Re-enter password4. question5.Securityans
		USN-2	As a user, I will receive confirmation email once I have registered for creating an account.	I can receive confirmat confirm.
		USN-3	As a user, I can also create an account using Google.	I can register & acces using Google Login d
		USN-4	As a user, I can also create an account using Face book.	I can register & acces using Face book login detai
	Login	USN-5	As a user ,I can login to the account by entering my email and password.  As a user, I can login to the account through Face book if I previously registered with it.  As a user, I can reset my password if I have forgotten my password.	I can login to the syst information can only
	MyAccount	USN-6	As a user, I can view my personal account.	I can use my personal Account for booking ]

			As a user, I can edit my Profile.	
Customer Care Executive		CCE-1	As a customer care executive, I can take complaints, answer calls from the customers regarding all the queries.	Pays attention to customer feedback to understand what service improvements. Customer care executive is responsible to assist the users by e

## 6.PROJECT PLANNING AND DESIGNING

### 6.1 Sprint planning and estimation

<b>Sprint</b>	<b>Functional Requirement (Epic)</b>	<b>User Story Number</b>	<b>User Story Task</b>	<b>Story Point</b>	<b>Priority</b>	<b>Team Members</b>
Sprint-1	Registration	USN-1	As a user, I can register the application for a convenient use	2	High	2
Sprint-1		USN-2	As a user, I will receive confirmation email once I have registered for the device.	1	High	1
Sprint-2		USN-3	As a user, I can register for the taking care of child tracking location.	2	Low	2
Sprint-1		USN-4	As a device, we can track them and share the notification the user.	2	Medium	2
Sprint-1	Login By user	USN-5	As a user, I can log into the application . And they can track the location of the train by entering the train number	1	High	1
	Dashboard		The user can get lots of notification options, GPS tracker ,alarm in case of emergency.	3	High	3



## 6.2SPRINT DELIVERYSCCHEDULE

<b>Sprint</b>	<b>Total Story Points</b>	<b>Duration</b>	<b>Sprint Start Date</b>	<b>Sprint End Date (Planned)</b>	<b>Story Points Completed (ason Planned End Date)</b>	<b>Sprint Release Date (Actual)</b>
Sprint-1	20	6 Days	24 Oct 2022	29Oct 22	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	19	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

## **7. FEATURES**

### **7.1. FEATURE 1**

- IOT device
- IBM Watson platform
- Node red
- Cloudant DB
- Web UI
- Geo fence
- MIT App
- Python code

### **7.2. FEATURE2**

- Registration
- Login
- Verification
- Ticket Booking
- Payment
- Ticket Cancellation
- Adding Queries

# 8.1. TEST CASES

## SPRINT -1

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)	BUG ID	Executed By
2	UI	verifying OTP	verifying one up or number process				numbers, gmail, Facebook or other social sites and to get otp number	Working as expected	pass				NAVEEN TR
3	Functional	OTP verification	Verify user otp using mail		1.Enter gmail id and enter password 2.click submit	Username: abc@gmail.com password: Testing123	OTP verified is to be displayed	Working as expected	pass				KAVI S
4	Functional	Login page	Verify user is able to log into application with invalid credentials		1.Enter into log in page 2.Click on My Account dropdown button 3.Enter invalid username/email in Email text box 4.Enter valid password in password text box 5.Click on login button	Username: abc@gmail.com password: Testing123	Application should show 'Incorrect email or password' validation message.	Working as expected	pass				NITHINRAJ R
5	Functional	Display Train details	The user can view about the available train details		1.As a user, I can enter the start and destination to get the list of trains available connecting the above	Username: abc@gmail.com password: Testing12367888788878878	A user can view about the available trains to enter start and destination details	Working as expected	fail				NITHINRAJ J

## SPRINT -2

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)	BUG ID	Executed By
1	Functional	Booking	User can provide the basic details such as a name, age, gender etc		1.Enter Method of reservation 2.Enter name,age,gender 3.Enter how many tickets wants to be booked 4.Also enter the number member's details like name,age,gender		Tickets booked to be displayed	Working as expected	pass				NAVEEN TR
2	UI	Booking seats	User can choose the class, seat/berth. If a preferred seat/berth isn't available it can be allocated based on the		1.known to which the seats are available		known to which the seats are available	Working as expected	pass				NITHINRAJ J
3	Functional	Payment	User, I can choose to pay through credit Card/debit card/UPI.		1.user can choose payment method 2.pay using this method		payment for the booked tickets to be done using payment method through either the following methods: credit Card/debit	Working as expected	pass				KAVI S
4	Functional	Redirection	User can be redirected to the selected		1.After payment the user will be redirected to the previous page		After payment the user will be redirected to the previous page	Working as expected	pass				NITHINRAJ R

## SPRINT -3

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)	BUG ID	Executed By
			during my journey.		4.Also enter the number member's details like name,age,gender								
2	UI	Ticket status	a user can see the status of my ticket Whether it's confirmed/waiting/RAC		1.known to the status of the tickets booked		known to the status of the tickets booked	Working as expected	pass				NAVEEN TR
3	Functional	Reminder notification	a user, I get reminders about my journey A day before my actual journey		1.user can get reminder notification		User can get reminder notification	Working as expected	pass				NITHINRAJ R
4	Functional	GPS tracking	User can track the train using GPS and can get information such as ETA, Current stop and		1.tracking train for getting information		tracking process through GPS	Working as expected	pass				NITHINRAJ J

# SPRINT -4

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)	BUG ID	Executed By
1	Functional	Ticket cancellation	user can cancel my tickets there's any Change of plan		1.tickets to be cancelled		Tickets booked to be cancelled	Working as expected	Pass				NITHINRAJ R
2	UI	Raise queries	user can raise queries through the query box or via mail.		1.raise the queries		raise the queries	Working as expected	pass				NITHINRAJ J
3	Functional	Answer the queries	user will answer the questions/doubts Raised by the customers.		1.answer the queries		answer the queries	Working as expected	pass				KAVI S
4	Functional	Feed details	a user will feed information about the trains delays and add extra seats if a new compartment is added.		1.information feeding on trains		information feeding on trains	Working as expected	pass				NAVEEN TR

9. RESULTS

9.1. PERFORMANCE METRICS



## **10. ADVANTAGES & 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;
- 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

### GPS CODE

```
import wiotp.sdk.device

import time

import random

myConfig={

    "identity": ( "orgId":

    "gagtey", "typeId":

    "GPS",

    "deviceId":"12345"},

    "auth": { "token":

    "12345678" }}

def myCommandCallback

(cmd): print ("Message

received from IBM IoT

Platform: %s" %

cmd.data['command'])

mcmd.data['command']

client=

wiotp.sdk.device.DeviceC

lient (config=myConfig,
```

```
logHandlers=None)

client.connect() def pub

(data): client.publishEvent

(eventId="status",

msgFormat="json",

data=myData, qos=0,

print("Published data

Successfully: %s",

myData) while True:

myData={'name': 'Train1',

'lat': 17.6387448, 'lon':

78.4754336) pub

(myData) time.sleep (3)

#myData('name': 'Train2',

'lat': 17.6387448, 'lon':

78.4754336) #pub

(myData) #time.sleep (3)

myData={'name': 'Train1',

'lat': 17.6341908, 'lon':

78.4744722) pub
```

(myData) time.sleep(3)

myData={'name': 'Train1',

'lat': 17.6340889, 'lon':

78.4745052) pub

(myData) time.sleep(3)

myData={'name': 'Train1',

'lat': 17.6248626, 'lon':

78.4720259) pub

(myData) time.sleep (3)

myData={'name': 'Train1',

'lat': 17.6188577, 'lon':

78.4698726) pub

(myData) time.sleep (3)

myData={'name': 'Train1',

'lat': 17.6132382, 'lon':

78.4707318) pub

(myData) time.sleep (3)

client.commandCallback

= myCommandCallback

client.disconnect()

## QR SCANNER CODE:

```
Import cv2 import numpy
as np import time Import
pyzbar.pyzbar as pyzbar
from
ibmcloudant.cloudant_v1
import CloudantV1 from
ibmcloudant import
CouchDbSessionAuthenti
cator from ibm_cloud_
sdk_core.authenticators
import BasicAuthenticator
authenticator=
BasicAuthenticator
('apikey-v2-
16u3crmdpkghhxefdikvps
soh5fwezrmuup5fv5g3ubz
',
'b0ab119f45d3e6255eabb9
78 service Cloudant V1
```

(authenticator-authenticato  
r)

service.set\_service\_url('htt  
ps://apikey-v2-

16u3crmdpkghhxefdikvps

soh5fwezrmuup5fv5g3ubz

:b0ab119f45d3e6255eabb

978e7e2f0 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
Ticket") time.sleep (5)
cv2.imshow("Frame",
frame) if cv2.waitKey(1)
& 0xFF==ord('q'): break
cap.release()
cv2.destroyAllWindows ()
client.disconnect()
```

## **GPRS Location**

```
import time
```

```
import sys
```

```
import ibmiotf.application
```

```
import ibmiotf.device
```

```
import random
```

```
import requests
```

```
import json
```

```
#Provide your IBM Watson
```

```
    Device Credentials
```

```
organization =
```

```
    &quot;0z828r&quot;;
```

```
deviceType =
```

```
    &quot;iotdevice&quot;;
```

```
#Credentials of Watson
```

```
IoT sensor simulator
```

```
deviceId =
```

```
    &quot;1001&quot;;
```



```
authMethod =
```

```
    &quot;token&quot;;
```

```
authToken =
```

```
    &quot;prathyusha&quot;;
```

```
# Initialize the device client.
```

```
L=0
```

```
try:
```

```
deviceOptions =
```

```
    {&quot;org&quot;:
```

```
        organization,
```

```
        &quot;type&quot;:
```

```
        deviceType,
```

```
        &quot;id&quot;: deviceId,
```

```
        &quot;auth-method&quot;;
```

```
    :
```

```
    authMethod,
```

```
    &quot;auth-token&quot;;
```

```
    authToken}
```

```
deviceCli =
```

```
ibmiotf.device.Client(deviceOptions)

#.....

except Exception as e:

    print("&quot;Caught

        exception connecting

        device: %s&quot; %

        str(e))

sys.exit()

# Connect and send a

    datapoint

    &quot;hello&quot; with

    value &quot;world&quot;

    into the cloud as an event

    of type

    &quot;greeting&quot; 10

times

deviceCli.connect()

while True:

    overpass_url =
```

```
&quot;http://overpass-api.  
de/api/interpreter&quot;  
overpass_query =  
    &quot;&quot;&quot;  
[out:json];area[name=&quot;I  
    ndia&quot;];(node[place=  
    &quot;village&quot;])(area  
    ););out;  
&quot;&quot;&quot;
```

```
response = requests.get(  
    overpass_url,  
    params={&#39;data&#39;:  
        overpass_query}  
)
```

```
coords = []
```

```
if response.status_code ==
```

```
    200:
```

```
    data = response.json()
```

```
    places =
```

```

        data.get('#elements&

        #', [])

for place in places:

coords.append((place['lat&

        at#'],

        place['lon#']))

print ('Got %s village

        coordinates!" %

        len(coords))

print (coords[0])

else:

print('Error')

i = random.randint(1,100)

L = coords[i]

#Send random gprs data to

        node-red to IBM Watson

data = {'d': {

        '#Latitude#':

        L[0],

        '#Longitude#':

```

```
L[1]}}
```

```
#print data
```

```
def myOnPublishCallback():
```

```
print("&quot;Published gprs
```

```
location = &quot;, L,
```

```
&quot;to IBM
```

```
Watson&quot;)
```

```
success =
```

```
deviceCli.publishEvent(&
```

```
quot;Data&quot;,
```

```
&quot;;json&quot;, data,
```

```
qos=0,
```

```
on_publish=myOnPublish
```

```
Callback)
```

```
time.sleep(12)
```

```
if not success:
```

```
print("&quot;Not connected to
```

```
IoTF&quot;)
```

```
time.sleep(1)
```

```
deviceCli.disconnect()
```

**DEMO LINK :**

[https://drive.google.com/  
drive/folders/1ZWLz1k  
mt3BEsdY3jUrPmBQT  
nElnw6ics?usp=sharing](https://drive.google.com/drive/folders/1ZWLz1kmt3BEsdY3jUrPmBQTnElnw6ics?usp=sharing)

**GitHub :**

[https://github.com/IBM-  
EPBL/IBM-Project-4670  
7-1660753719](https://github.com/IBM-EPBL/IBM-Project-46707-1660753719)