SMART SOLUTION FOR RAILWAYS

IBM NALAIYATHIRAN (HX8001)
PROJECT REPORT

HARRISH KUMAR S	**211719106026**
HARIHARASUDHAN A	**211719106022**
IRFAN SHAKIL S	**211719106029**
JOSHIBA M	**211719106032**

in partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

IN

ELECTRONICS AND COMMUNICATION ENGINEERING

RAJALAKSHMI INSTITUTE OF TECHNOLOGY

ANNA UNIVERSITY: CHENNAI 600 025

NOVEMBER 2022

BONAFIDE CERTIFICATE

Certified that this project report **"SMART SOLUTION FOR RAILWAYS"** is the bonafide work of **"HARRISH KUMAR S (211719106026) , HARIHARASUDHAN A (211719106022), IRFAN SHAKIL S (211719106029) , JOSHIBA M (211719106032) ",** who carried out the project work under my supervision.

SIGNATURE:

Dr. S.MANJULA M.E., Ph.D.

HEAD OF THE DEPARTMENT,

Dept. of Electronics and Communication Engg..

Rajalakshmi Institute of Technology,

Kuthambakkam Post,

Chennai - 600 124

SIGNATURE:

Mr. A.BALAJI M.E., Ph.D,

MENTOR,

Dept. of Electronics and Communication Engg..

Rajalakshmi Institute of Technology,

Kuthambakkam Post,

Chennai - 600 124

The viva-voce is held on _____.

INTERNAL EXAMINER

EXTERNAL EXAMINER

Project Report

INTRODUCTION

1.1. PROJECT OVERVIEW

SMART SOLUTIONS FOR RAILWAYS

The Indian Railways (IR) carries about 5.5 lakhs passengers in reserved accommodation every day. The Computerized Passenger Reservation System (PRS) Facilites the booking and cancellation of tickets from any of the 4000 terminals (i.e. PRS booking window all over the countries). These tickets can be booked or cancelled for journeys commencing in any part of India and ending in any other part, with travel time as long as 72hours (about 3 days) and distance upto several thousand kilometers. In the given project we will be developing a website which will help users to find train details, book and cancel tickets and the exact rates of their tickets to the desired destination. With the help of online booking people can book their tickets online through the internet, sitting in their home with a single click of mouse. Using their credit cards people can easily get their tickets done within minutes.

1.2. PURPOSE

Railway passengers frequently need to know about their ticket reservation status, ticket availability on a particular train or for a place, train arrival or departure details, special trains etc. Customer information centers at the railway stations are unable to serve such queries at peak periods. Our website has various kinds of information that helps regarding booking of tickets via railways. Users will be able to search the train availability ,the exact fare ,the arrival and departure time of the train and they can also book the ticket by using the debit ,credit or master card and after booking the ticket The online railway ticket reservation system aims to develop a web application which aims at providing trains details, trains availability, as well as the facility to book ticket in online for customers.

LITERATURE SURVEY

2.1 EXISTING SYSTEM

The Indian railway reservation system the passengers face many difficulties to book the tickets, cancelling tickets, checking Passenger Name Record number (PNR) [6]. The developed countries IOT had a major impact on the smart transportation industry, with the advent of autonomous vehicles and improved cargo management [1]. Shri Suresh Prabhu Indian railway minister on February 26th, 2015, announced in Lok Sabha,the IT industry must implement in IoT for railways, the proposed research on IoT Investment Plan 2015-2022 has been assigned Rs 5,000 crore Indian rupees [6]. Handling the passenger reservation data has been a key point of consideration in most railway services [1]. The smart railways research report also provides an in-depth analysis of proposed and ongoing projects by various countries [2].

2.2. REFERENCES

- D. Hesse, "Rail Inspection Using Ultrasonic Surface Waves"Thesis, ImperialCollege of London, 2007.
- Md. Reya Shad Azim1, Khizir Mahmud2 and C. K. Das. Automatic railway track switching system, International Journal of Advanced Technology, Volume 54, 2014.

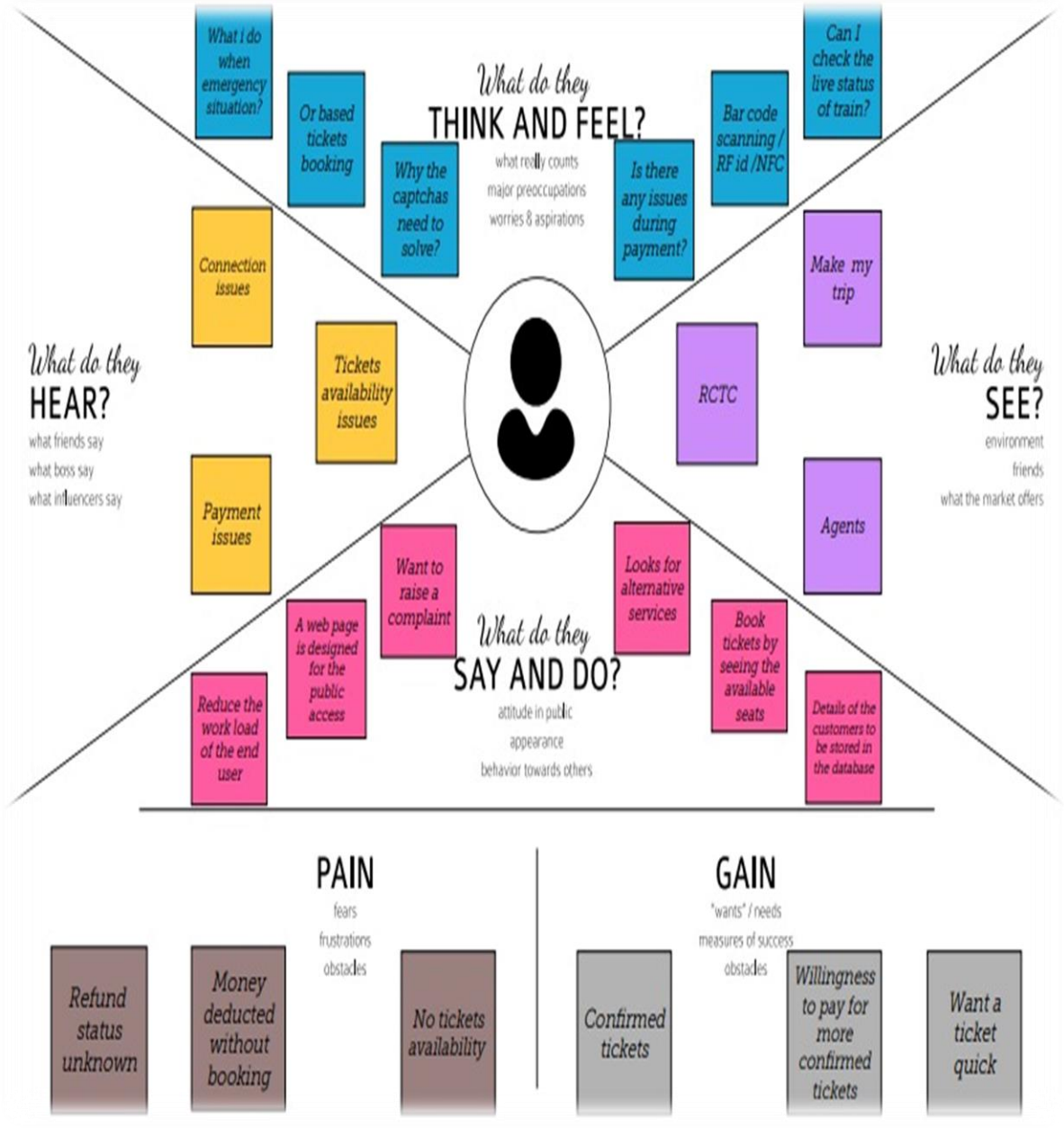
- S. Somal Raju, V. Murali, G. Saha and V. Vaidehi, “Title-robust railway crackdetection scheme using LED (Light Emitting Diode) - LDR (Light Dependent Resistor) assembly IEEE 2012.
- S. Srivastava, R. P. Chourasia, P. Sharma, S. I. Abbas, N. K. Singh, “RailwayTrack Crack detection vehicle”, IARJSET, Vol. 4, pp. 145-148, Issued in 2, Feb 2017.
- U. Mishra, V. Gupta, S. M. Ahsam and S. M. Tripathi, “Google Map Based Railway Track Fault Detection Over the Internet”, International Journal of AppliedEngineering Research, Vol. 14, pp. 20-23, Number 2, 2019.
- 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.
- N. Bhargav, A. Gupta, M. Khiara, S. Yadav, and V. Sahu, “Automatic FaultDetection of Railway Track System Based on PLC (ADOR TAST)”, InternationalJournal 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 modesof 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 travellingby train at station and on board”

IDEATION AND PROPOSED SOLUTON

3.1. EMPATHY MAP CANVAS



3.2. IDEATION & BRAINSTORMING

2

Brainstorm

Write down any ideas that come to mind that address your problem statement.

10 minutes

3

Group ideas

Take turns sharing your ideas while clustering similar or related notes as you go. Once all sticky notes have been grouped, 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

3.3. PROPOSED SOLUTION

S.NO	PARAMETERS	DESCRIPTIONS
1	Problem Statement (Problem to be solved)	In order to satisfy the passengers, the Railways provides various services to its passengers But, the passengers can face some problems.
2	Idea / Solution description	The idea is to minimize the ticket booking problems among the passengers by providing Online mode of booking rather than papers. . In queues in front of the ticket counters in railway stations have been drastically increased over the time.
3	Novelty / Uniqueness	Online mode of booking is most common and so ease of access to everyone that makes more efficient uniqueness of utilizing the technique. People can book their ticket through online and they get a QR code through SMS
4	Social Impact / Customer Satisfaction	Customers for sure get satisfied as they are in thefast-roaming world and this technique makes more easier for travelling passengers. A web page is designed in which the user can book tickets and will be provided with the QR code, which will be shown to the ticket collector and by scanning the QR code the ticket collector will get the passenger details

3.4. PROBLEM SOLUTION FIT

owing

REQUIREMENT ANALYSIS

4.1. FUNCTIONAL REQUIREMENTS

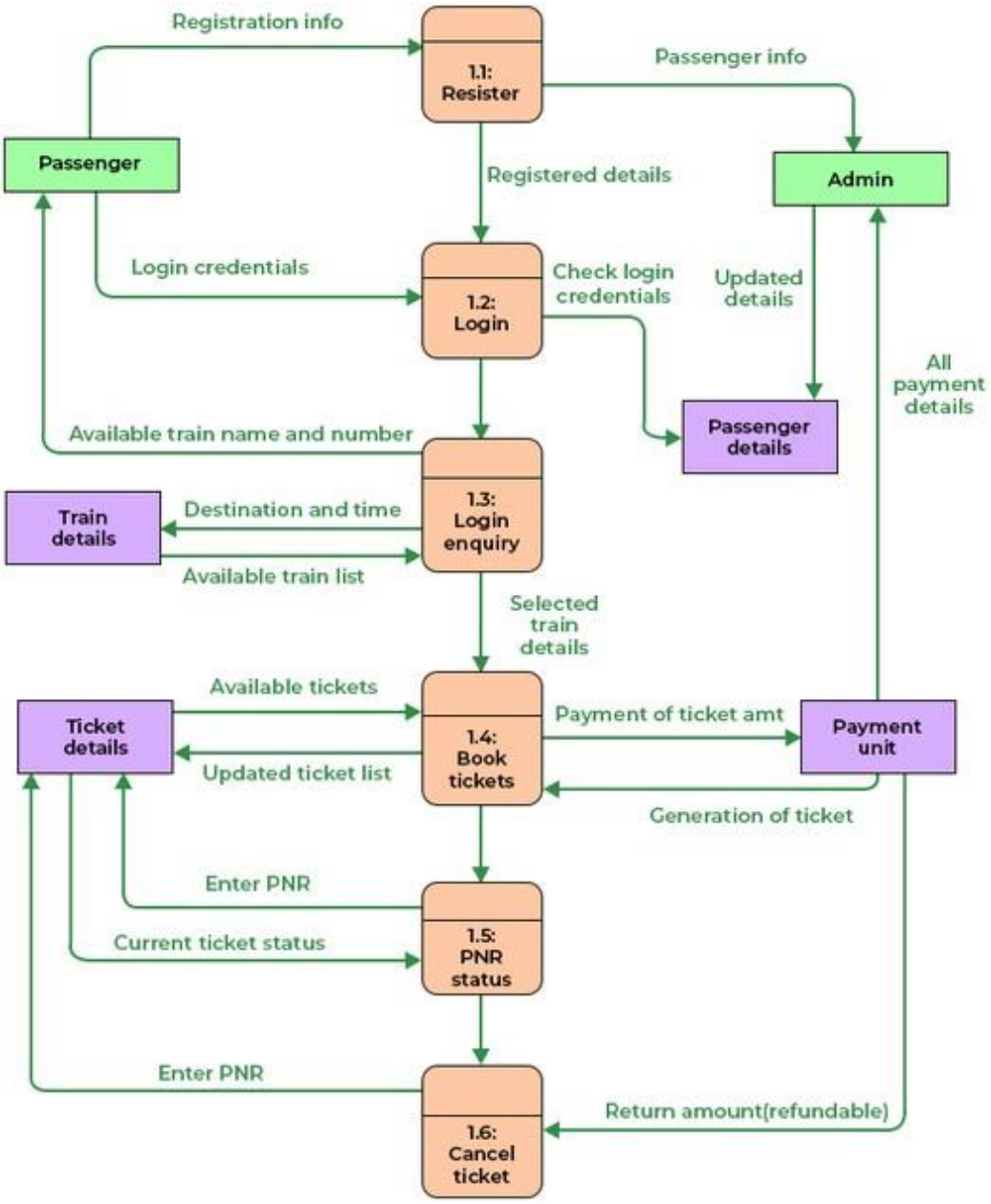
FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	**User Registration**	Registration through Official website, Registration through Form, Registration through App.
FR-2	**User Confirmation**	Confirmation via QR code, Confirmation via . message.
FR-3	**Ticket Verification**	Ticket collectors check the tickets using the QR code which is generated during ticket booking.
FR-4	**TTE workload reduction**	TTE can simply scan the QR code to identify the personal details and also to reduce the use of paper.
FR-5	**Database Storage**	All the booking details of the customer is stored in the database with unique ID which can be retrieved when the ticket collector scans the QR code.

4.2. NON-FUNCTIONAL REQUIREMENTS

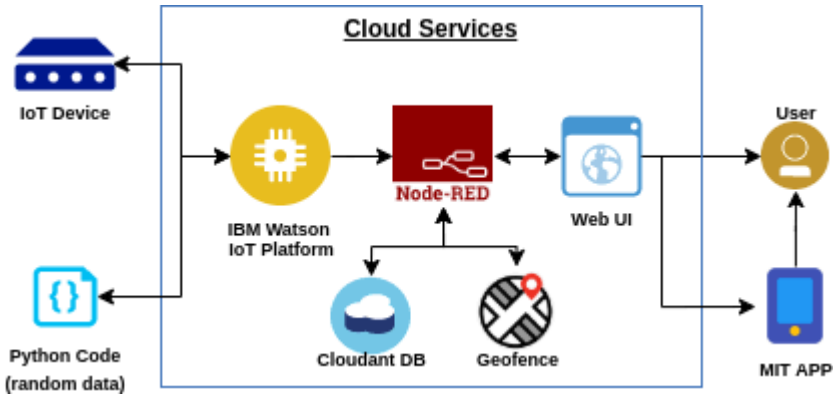
FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	The user can easily book the at their own pace based on the availability of seats without waiting in the ticket counter.
NFR-2	**Security**	User's personal details are safely stored in the Cloud. QR code reduces the duplicate train tickets
NFR -3	**Reliability**	As IBM cloud is used to store the information about the customers this product is highly reliable.
NFR-4	**Performance**	The Web UI provides smooth user experience and improves the performance of this solution.
NFR-5	**Availability**	App is readily available on play store for mobile phone users or user can access the website using web browser.
NFR-6	**Scalability**	Use of captcha and encryption toavoid bots from booking tickets

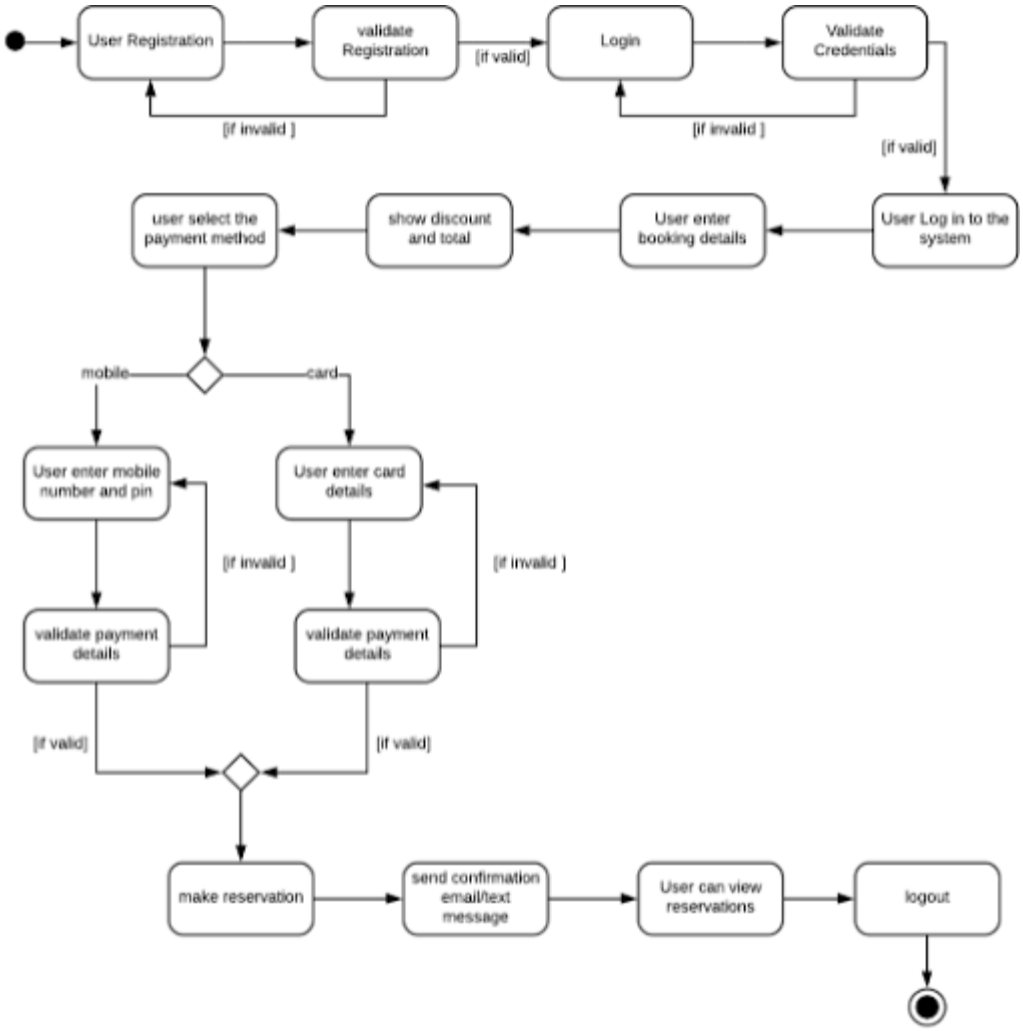
PROJECT DESIGN

5.1.DATA FLOW DIAGRAMS



5.2 SOLUTION & TECHNICAL ARCHITECTURE




User Registration
validate Registration
[if valid]
Login
Validate Credentials
[if invalid]
[if invalid]
[if valid]
User Log in to the system
User enter booking details
show discount and total
user select the payment method
mobile
card
User enter mobile number and pin
User enter card details
[if invalid]
[if invalid]
validate payment details
validate payment details
[if valid]
[if valid]
make reservation
send confirmation email/text message
User can view reservations
logout

5.3 USER STORIES

5.3 USER STORIES

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user, Web user)	Registration	USN-1	As a user, I can register through the form by Filling in my details	I can register and create my account / dashboard	High	Sprint-1
		USN-2	As a user, I can register through phone numbers, Gmail, Facebook or other social sites	I can register & create my dashboard with Facebook login or other social sites	High	Sprint-2
	Conformation	USN-3	As a user, I will receive confirmation through email or OTP once registration is successful	I can receive confirmation email & click confirm.	High	Sprint-1
	Authentication/Login	USN-4	As a user, I can login via login id and password or through OTP received on register phone number	I can login and access my account/dashboard	High	Sprint-1
	Display Train details	USN-5	As a user, I can enter the start and destination to get the list of trains available connecting the above	I can view the train details (name & number), corresponding routes it passes through based on the start and destination entered.	High	Sprint-1
	Booking	USN-6	As a use, I can provide the basic details such as a name, age, gender etc…	I will view, modify or confirm the details enter.	High	Sprint-1
		USN-7	As a user, I can choose the class, seat/berth. If a preferred seat/berth isn't available I can be allocated based on the availability.	I will view, modify or confirm the seat/class berth selected	High	Sprint-1
	Payment	USN-8	As a user, I can choose to pay through credit Card/debit card/UPI.	I can view the payment Options available and select my desirable choice To proceed with the payment	High	Sprint-1

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
			completion of payment I'll be redirected to the booking website.	be done I can move back to the initial payment page		
	Ticket generation	USN-10	As a user, I can download the generated e-ticket for my journey along with the QR code which is used for authentication during my journey.	I can show the generated QR code so that authentication can be done quickly.	High	Sprint-1
	Ticket status	USN-11	As a user, I can see the status of my ticket Whether it's confirmed/waiting/RAC.	I can confidentially get the Information and arrange alternate transport if the ticket isn't Confirmed	High	Sprint-1
	Remainders notification	USN-12	As a user, I get remainders about my journey A day before my actual journey.	I can make sure that I don't miss the journey because of the constant notifications.	Medium	Sprint-2
		USN-13	As a user, I can track the train using GPS and can get information such as ETA, Current stop and delay.	I can track the train and get to know about the delays pian accordingly	Medium	Sprint-2
	Ticket cancellation	USN-14	As a user, I can cancel my tickets if there's any Change of plan	I can cancel the ticket and get a refund based on how close the date is to the journey.	High	Sprint-1
	Raise queries	USN-15	As a user, I can raise queries through the query box or via mail.	I can view my pervious queries.	Low	Sprint-2
Customer care Executive	Answer the queries	USN-16	As a user, I will answer the questions/doubts Raised by the customers.	I can view the queries and make it once resolved	Medium	Sprint-2
Administrator	Feed details	USN-17	As a user, I will feed information about the trains delays and add extra seats if a new compartment is added.	I can view and ensure the corrections of the information fed.	High	Sprint-1

PROJECT PLANNING AND SCHEDULING

6.1 SPRINT PLANNING& ESTIMATION

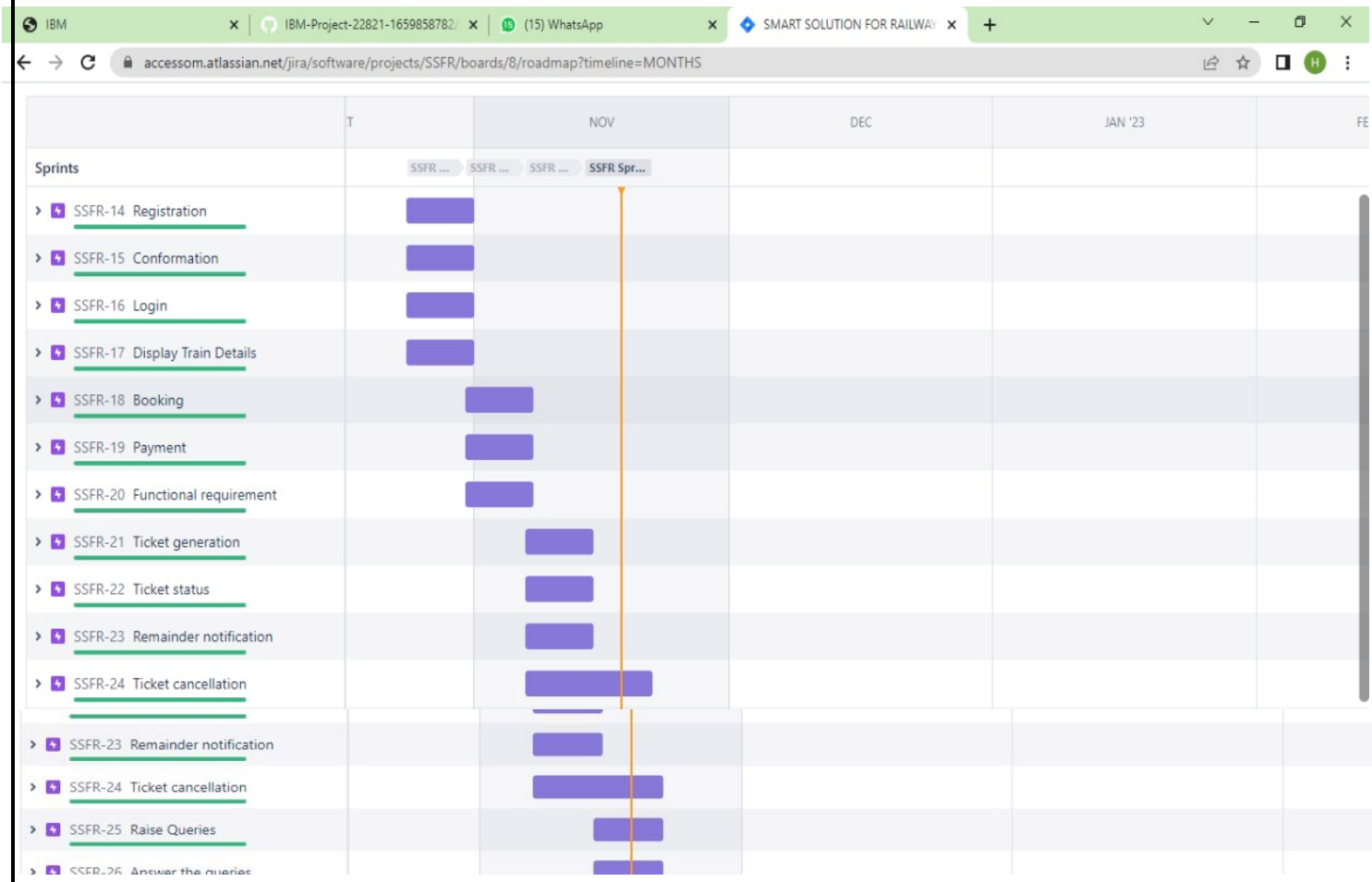
Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	As a user, I can register through the form by Filling in my details	2	High	JOSHIBA
Sprint-1		USN-2	As a user, I can register through phone numbers, Gmail, Facebook or other social sites	1	High	HARRISH KUMAR
Sprint-1	Conformation	USN-3	As a user, I will receive confirmation through email or OTP once registration is successful	2	Low	IRFAN SHAKIL
Sprint-1	login	USN-4	As a user, I can login via login id and password or through OTP received on register phone number	2	Medium	JOSHIBA
Sprint-1	Display Train details	USN-5	As a user, I can enter the start and destination to get the list of trains available connecting the above	1	High	HARIHARA SUDHAN
Sprint-2	Booking	USN-6	As a use, I can provide the basic details such as a name, age, gender etc…	2	High	IRFAN SHAKIL
Sprint-2		USN-7	As a user, I can choose the class, seat/berth. If apreferred seat/berth isn't available I can be allocated based on the availability	1	Low	JOSHIBA
Sprint-2	Payment	USN-8	As a user, I can choose to pay through credit Card/debit card/UPI.	1	High	IRFAN SHAKIL
Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	
Sprint-2		USN-9	As a user, I will be redirected to the selected	2	High	HARRISH KUMAR
Sprint-3	Ticket generation	USN-10	As a user, I can download the generated e- ticket for my journey along with the QR code which is used for authentication during my journey.	1	High	JOSHIBA

			Whether it's confirmed/waiting/RAC.			
Sprint-3	Remainders notification	USN-12	As a user, I get remainders about my journey A day before my actual journey.	1	High	HARIHARA SUDHAN
Sprint-3	Ticket cancellation	USN-13	As a user, I can track the train using GPS and can get information such as ETA, Current stop and delay	2	High	HARRISH KUMAR
Sprint-4		USN-14	As a user, I can cancel my tickets if there's any Change of plan	1	High	IRFAN SHAKIL
Sprint-4	Raise queries	USN-15	As a user, I can raise queries through the query box or via mail.	2	Medium	JOSHIBA
Sprint-4	Answer the queries	USN-16	As a user, I will answer the questions/doubts Raised by the customers.	2	High	HARIHARA SUDHAN
Sprint-4	Feed details	USN-17	As a user, I will feed information about the trains delays and add extra seats if a new compartment is added.	1	High	HARRISH KUMAR

6.2SPRINT 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	20	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	5 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-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov2022

REPORTS FROM JIRA



CODING AND SOLUTIONING

7.1. FEATURE 1

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

7.2. FEATURE 2

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

```
labl_0 = Label(base, text="Registration form",width=20,font=("bold",20))
labl_0.place(x=90,y=53)


lb1= Label(base, text="Enter Name", width=10, font=("arial",12))lb1.place(x=20, y=120)
en1= Entry(base) en1.place(x=200,
y=120)

lb3= Label(base, text="Enter Email", width=10, font=("arial",12))lb3.place(x=19, y=160)
en3= Entry(base) en3.place(x=200,
y=160)

lb4= Label(base, text="Contact Number", width=13,font=("arial",12))lb4.place(x=19, y=200)
en4= Entry(se) en4.place(x=200,
y=200)

lb5= Label(base, text="Select Gender", width=15, font=("arial",12))lb5.place(x=5, y=240)

var = IntVar()

Radiobutton(base, text="Male", padx=5,variable=var,value=1).place(x=180, y=240)

Radiobutton(base, text="Female", padx =10,variable=var,value=2).place(x=240,y=240)
Radiobutton(base, text="others", padx=15, variable=var,value=3).place(x=310,y=240)

list_of_cntry = ("United States", "India", "Nepal", "Germany")cv = StringVar()
drplist= OptionMenu(base, cv, *list_of_cntry)drplist.config(width=15)
cv.set("United States")
lb2= Label(base, text="Select Country", width=13,font=("arial",12))lb2.place(x=14,y=280)
drplist.place(x=200, y=275)

lb6= Label(base, text="Enter Password", width=13,font=("arial",12))lb6.place(x=19, y=320)
en6= Entry(base, show='*')
en6.place(x=200, y=320)

lb7= Label(base, text="Re-Enter Password",
width=15,font=("arial",12))
lb7.place(x=21, y=360)
en7 =Entry(base, show='*')
en7.place(x=200, y=360)

Button(base, text="Register", width=10).place(x=200,y=400)base.mainloop()
```

```
def generateOTP() :

    # Declare a digits variable# which stores
    all digits digits = "0123456789" OTP =
    ""

   # length of password can be changed# by changing value
   in range
    for i in range(4) :
        OTP += digits[math.floor(random.random() * 10)]return OTP

# Driver code
if____name_____== "_____main_____" :


    print("OTP of 4 digits:", generateOTP())digits="0123456789"
OTP=""
for i in range(6): OTP+=digits[math.floor(random.random()*10)]
otp = OTP + " is your OTP"msg= otp
s = smtplib.SMTP('smtp.gmail.com', 587)s.starttls()
s.login("Your Gmail Account", "You app password")emailid = input("Enter your
email: ")

.sendmail('&&&&&&&&&&',emailid,msg)   a = input("Enter Your OTP
                                    >>: ")
if a == OTP: print("Verified")
else:
    print("Please Check your OTP again")roo
```

s

TESTING

8.1. TEST CASES

Test case ID	Feature Type	Component	Test Scenario	Pre-Requisite	Steps To Execute	Test Data	Expected Result	Actual Result	Status	Commnets	TC for Automati	BUG
1	Functional	Registration	Registration through the form by Filling in my details		1.Click on register 2.Fill the registration form 3.click Register		Registration form to be filled is to be displayed	Working as expected	Pass			
2	UI	Generating OTP	Generating the otp for further process		1.Generating of OTP number		user can register through phone numbers, Gmail, Facebook or other social sites and to get oto number	Working as expected	pass			
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 verifed is to be displayed	Working as expected	pass			
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 password: Testing123	Application should show 'Incorrect email or password' validation message.	Working as expected	pass			
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: Testing123678868678687 6876	A user can view about the available trains to enter start and destination details	Working as expected	fail			

Feature Type	Component	Test Scenario	Pre-Requisite	Steps To Execute	Test Data	Expected Result	Actual Result	Status	Commnets	TC for Automation(Y/N	BUG ID
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			
UI	Booking seats	User can choose the class, seat/berth. If a preferred seat/berth isn't available I can be allocated based on the availability		1,.known to which the seats are available		known to which the seats are available	Working as expected	pass			
Functional	Payment	user, I can choose to pay through credit Card/debit card/UPI.		1.user can choose payment method 2.pay using tht method		payment for the booked tickets to be done using payment method through either the following methods credit Card/debit card/UPI.	Working as expected	pass			
Functional	Redirection	user can be redirected to the selected		1.After payment the usre will be redirected to the previous		After payment the usre will be redirected to the previous page	Working as expected	pass			

Test case ID	Feature Type	Component	Test Scenario	Pre-Requisit	Steps To Execute	Test Data	Expected Result	Actual Result	Status	Commnets	TC for Autom	BUG ID
10	Functional	Ticket generation	a user can download the generated e ticket for my journey along with the QR code which is used for authentication during my journey.		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			
11	UI	Ticket status	a usercan see the status of my ticket Whether it's confirmed/waiting/RAC		1.known to the status of the tivkets booked		known to the status of the tivkets booked	Working as expected	pass			
12	Functional	Remainder notification	a user, I get remainders about my journey A day before my actual journey		1.user can get reminder nofication		user can get reminder nofication	Working as expected	pass			
13	Functional	GPS tracking	user can track the train using GPS and can get information such as ETA, Current stop and delay		1.tracking train for getting information		tracking process through GPS	Working as expected	pass			

Test case ID	Feature Type	Component	Test Scenario	Pre-Requisite	Steps To Execute	Test Data	Expected Result	Actual Result	Status	Commnets	TC for Automation(Y	BUG ID
14	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			
15	UI	Raise queries	user can raise queries through the query box or via		1,raise the queries		raise the queries	Working as expected	pass			
16	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			
17	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			

RESULTS

9.1PERFORMANCE METRICS

ADVANTAGES &DISADVANTAGES

10.1ADVANTAGES

- The online reservation system is a very good feature, because standing in Queues all the time and jostling away wasting your time is not the best thing to do, so the IRCTC website does help a great deal to book your tickets.

- PNR status: The status enquiry feature is a lot better than before, when you had to dial your PNR number over and over again to no avail. The website does give your status fairly easily.

- Counter ticket has to be handled with care, but SMS on mobile is more than enough.

- You are becoming environment friendly and contributing for greener planet by ignoring printout.

10.2DISADVANTAGES

Main disadvantage of railway reservation system is that we are not sure of getting a berth of our choice after first day of reservation in 120 day advance reservation period.This makes most of senior citizens women with infants and small children who are badly in need of lower berth at the mercy of other passengers

CONCLUSION

In this paper the survey of railway industry broadband technologies like GSM-R, LTE, 5G, IEEE 802.11 are examined. GSM-Railway is a international wireless mobile communications standard for various railway applications has been described. In Indian railway reservation system to replace the present 4G network to GSM-R, 5G, LTE and IEEE 802.11 brought a new standardized telecommunication network. In present PRS system there are many dilemmas passenger to book the ticket and railway reservation system to issues the ticket. Passenger must wait on IRCTC website more 5 minutes to book ticket. Hence present PRS is updated to smart railway reservation under the vision of India 2022. The passenger vending machine read the passenger details that can be interacted with railway reservation system to UIDAI database. The result proved that passengers obtained the ticket with fraction of mile seconds, the reservation chart contains the passengers name with photo, effective passenger reservations will be maintained.

FUTURE SCOPE

- A major future scope for this project is the use of ECC (Elliptic curve Cryptography) for securing the sensitive fingerprint data. Since ECC methodology is perfect for securing the data generated by low power devices such as Arduino. And since ECC provides the same level of security as that of RSA and DES, we can completely ignore the two-way encryption and provide only a single-level of encryption with ECC.
- Another prospect of this project is to implement the same application in the form of a mobile application, through which the user can book tickets directly from their mobiles and there is no need for a separate kiosk for that purpose.

APPENDIX

13.1. SOURCE PROGRAM

```
Import math,random
Import os

 from bs4 import BeautifulSoup
 from django.contrib.auth.base_user import AbstractBaseUserfrom
 django.db import models
 i
from plyer import notification
import time
import numpy as np
import matplotlib.pyplot as plt
from PIL import Image, ImageDraw
from pickle import load,dump import
smtplib, ssl
from email.mime.text import MIMEText
from email.mime.multipart import MIMEMultipart
import email

from email import encoders
from email.mime.base import MIMEBase


  import attr
  from flask import Blueprint, flash, redirect, request, url_forfrom
  flask.views import MethodView
  from flask_babelplus import gettext as _
  from flask_login import current_user, login_required
  from pluggy import HookimplMarker

  ()
  base.geometry("500x500")
  base.title("registration form")
```

```
labl_0 = Label(base, text="Registration form",width=20,font=("bold",20))
labl_0.place(x=90,y=53)

lb1= Label(base, text="Enter Name", width=10, font=("arial",12))
lb1.place(x=20, y=120)
en
lb3= Label(base, text="Enter Email", width=10, font=("arial",12))
lb3.place(x=19, y=160)
lb4= Label(base, text="Contact Number", width=13,font=("arial",12))
lb4.place(x=19, y=200)

lb5= Label(base, text="Select Gender", width=15, font=("arial",12))
lb5.place(x=5, y=240)
var = IntVar()
Radiobutton(base, text="Male", padx=5,variable=var,
Radiobutton(base, text="Female", padx =10,variable=var,
value=2).place(x=240,y=240)
Radiobutton(base, text="others", padx=15, variable=var,
value=3).place(x=310,y=240)

list_of_cntry = ("United States", "India", "Nepal", "Germany")cv =
StringVar()
drplist= OptionMenu(base, cv, *list_of_cntry)
drplist.config(width=15)
cv.set("United States")
lb2= Label(base, text="Select Country", width=13,font=("arial",12))
lb2.place(x=14,y=280)
drplist.place(x=200, y=275)
```

```
lb6= Label(base, text="Enter Password", width=13,font=("arial",12))
lb6.place(x=19, y=320)
en6= Entry(base, show='*')
en6.place(x=200, y=320)

lb7= Label(base, text="Re-Enter Password",
width=15,font=("arial",12))
lb7.place(x=21, y=360)
)

Button(base, text="Register", width=10).place(x=200,y=400)
base.mainloop()

def generateOTP() :
    # declarea digits variable#
    which stores all digits
    digits = "0123456789"
    OTP = ""

   # length of password can be changed#
   by changing value in range
    for i in range(4) :
        OTP += digits[math.floor(random.random() * 10)]


        return OTP

    Home.title("Python: Simple Login Application")
    width = 600
    height = 500
    screen_width = root.winfo_screenwidth()
```

```
    screen_height = root.winfo_screenheight()x
    = (screen_width/2) - (width/2)
    y = (screen_height/2) - (height/2)
    root.resizable(0, 0)
    Home.geometry("%dx%d+%d+%d" % (width, height, x, y)) lbl_home =
    Label(Home, text="Successfully Login!", font=('times new
roman', 20)).pack()
    btn_back = Button(Home, text='Back',
command=Back).pack(pady=20, fill=X)

def Back():
```

GIT HUB LINK –
https://github.com/IBM-EPBL/IBM-Project-22821-165985878

Project Demo Link-

https://youtu.be/z-Zk8hR9sgw