# PROJECT REPORT
# SMART SOLUTION FOR RAILWAYS

**TEAM ID: PNT2022TMID27503**

**TEAM MEMBERS:** BHUVANESH K
DEEPAK SARAVANAN S
DHAMINI M
MERISHA L V
JUDE ANTO BENHUR A

## 1. **INTRODUCTION**

### 1.1 Project Overview

As trains are one of the most preferred modes of transportation among middle class and impoverished people as it attracts for its amenities. Simultaneously there is an increase at risk from thefts and accidents like chain snatching, derailment, fire accident. In order to avoid or in better words to stop all such brutality we came up with a solution by providing an application which can be accessed by the user after booking their tickets. With a single click this app addresses issues by sending a text message to TC and RPF as an alert. In our project we use Node-Red service, app-development, IBM cloud platform to store passenger data.

### 1.2 Purpose

The purpose of this project is to report and get relived from the issues related to trains.

## 2. LITERATURE SURVEY

### 2.1 Existing problem

- 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.

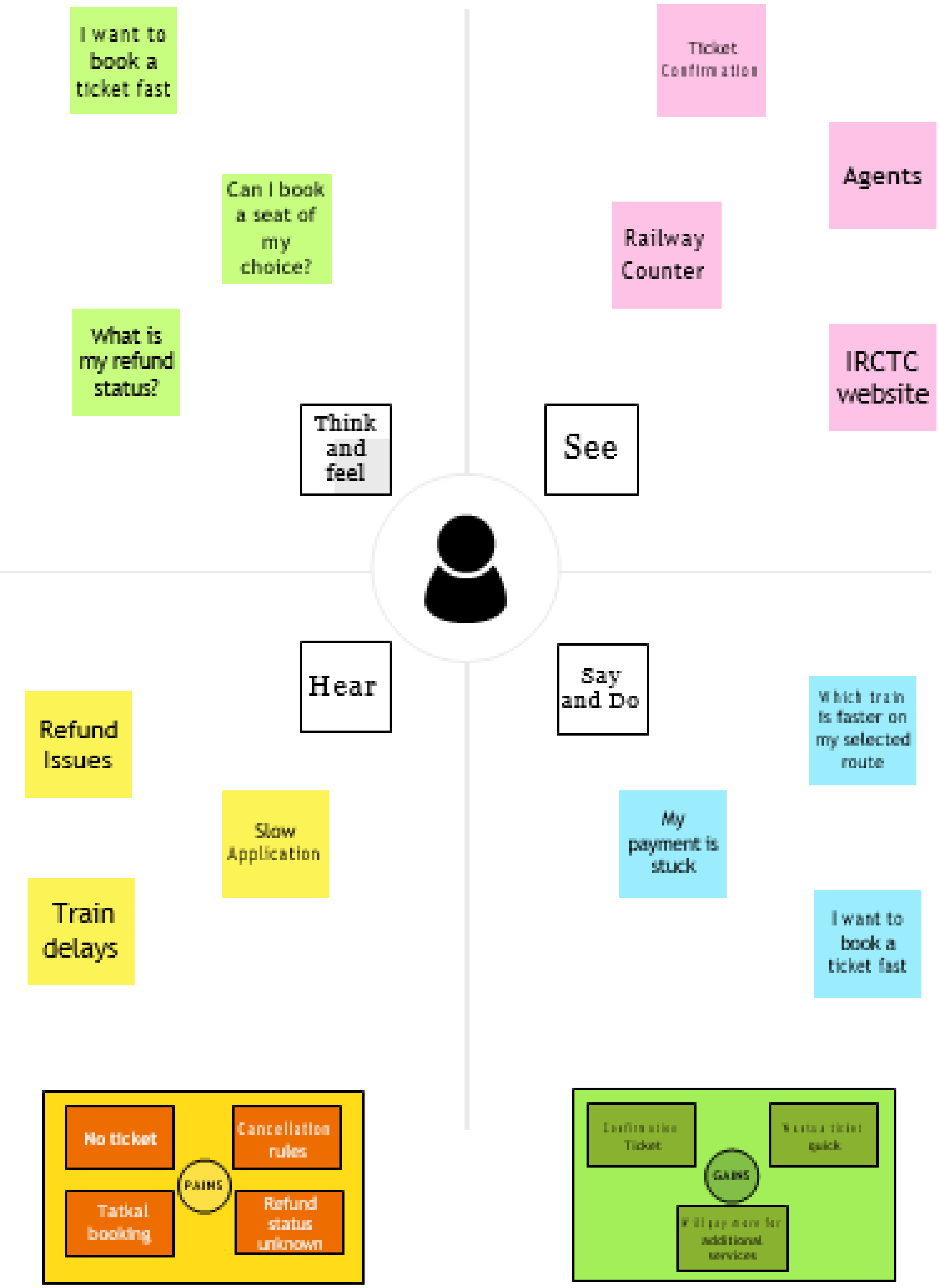
## 2.2 REFERENCES

| | | | |
|---|---|---|---|
| Smart Train Detector using IOT | Payal Srivastava,Rana Majumdar, Bonny Paulose, Sunil Kumar. | January 2019 | https://ieeexplore.ieee.org/document/8776894 |
| Smart Train Accident Detection and Prevention System using IOT Technology | Lakshmi Devi R,Saravanan G, Sangeetha K, Pavithra S, Thiygarajan S. | July 2021 | https://ieeexplore.ieee.org/document/9526413 |
| Railways Components Wear: A smart platform. | Alessandro, Massaro, Emanuele, Cannella | June 2021 | https://ieeexplore.ieee.org/document/9488486 |

## 3. IDEATION & PROPOSED SOLUTION

### 3.1 Empathy Map Canvas

# Smart Solution for Railways



Think and feel

I want to book a ticket fast

Can I book a seat of my choice?

What is my refund status?

See

Ticket Confirmation

Agents

Railway Counter

IRCTC website

Hear

Refund Issues

Slow Application

Train delays

Say and Do

Which train is faster on my selected route

My payment is stuck

I want to book a ticket fast

PAINS

No ticket

Cancellation rules

Tatkal booking

Refund status unknown

GAINS

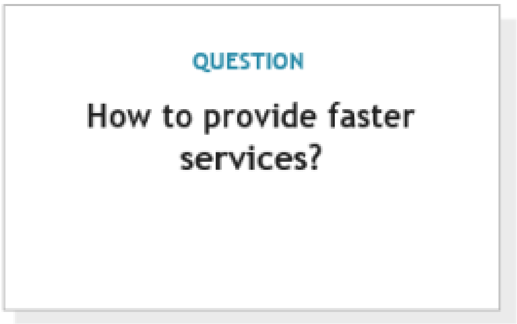
Confirmation Ticket

Status ticket quick

Will pay more for additional services

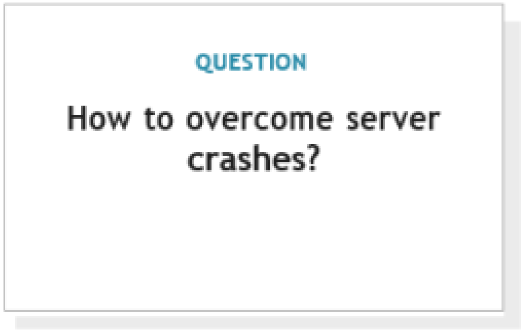
## 3.2 IDEATION AND BRAINSTORMING

### PROBLEM:



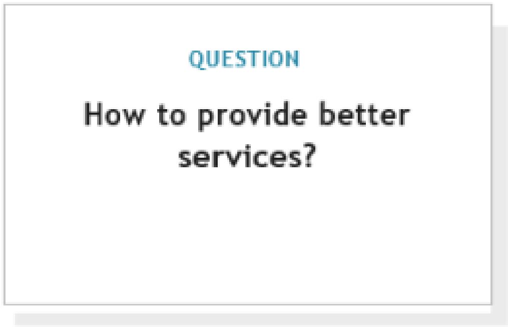
QUESTION

How to provide faster services?



QUESTION

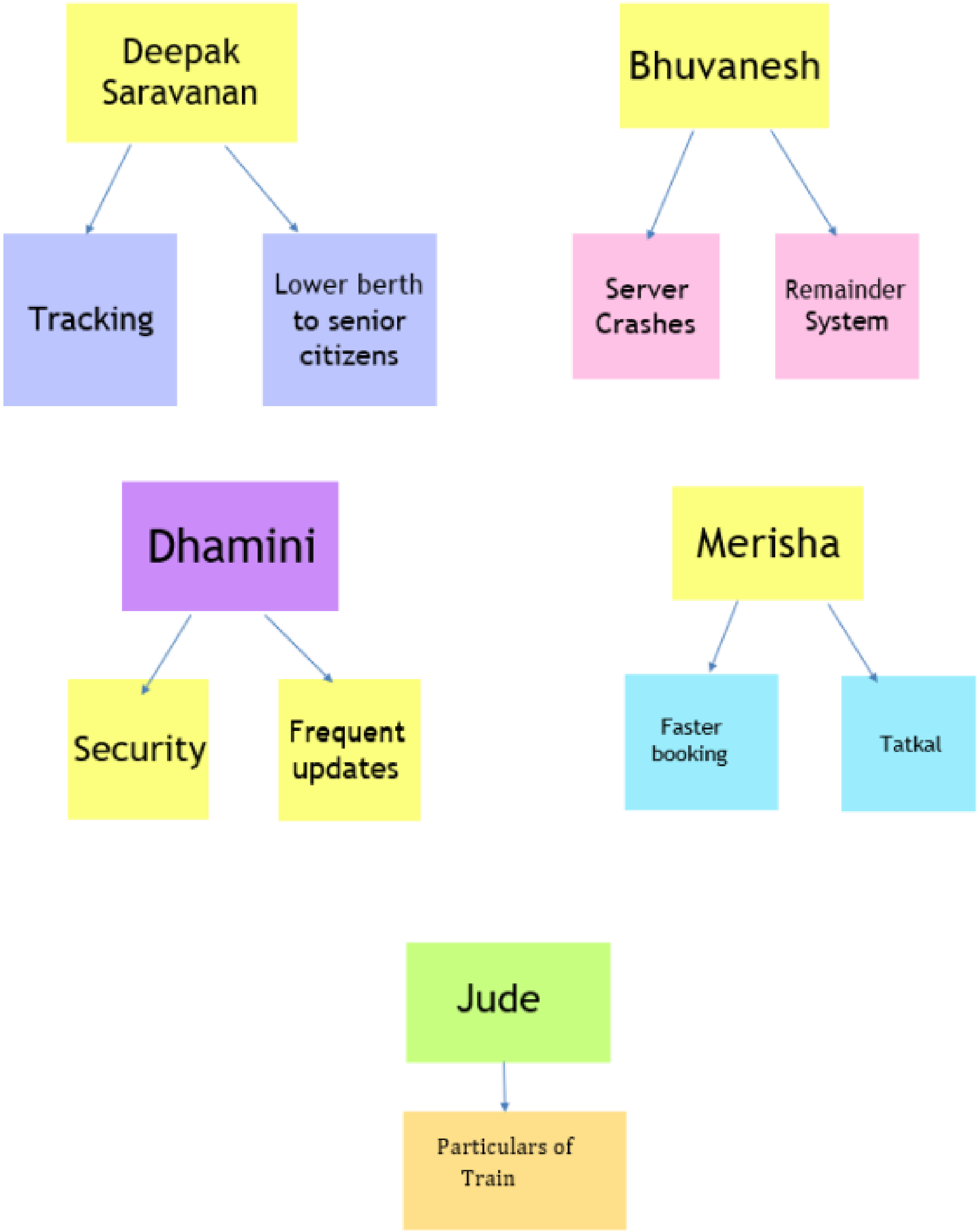
How to overcome server crashes?



QUESTION

How to provide better services?

# INDIVIDUAL IDEAS:



Deepak Saravanan

Tracking

Lower berth to senior citizens

Bhuvanesh

Server Crashes

Remainder System

Dhamini

Security

Frequent updates

Merisha

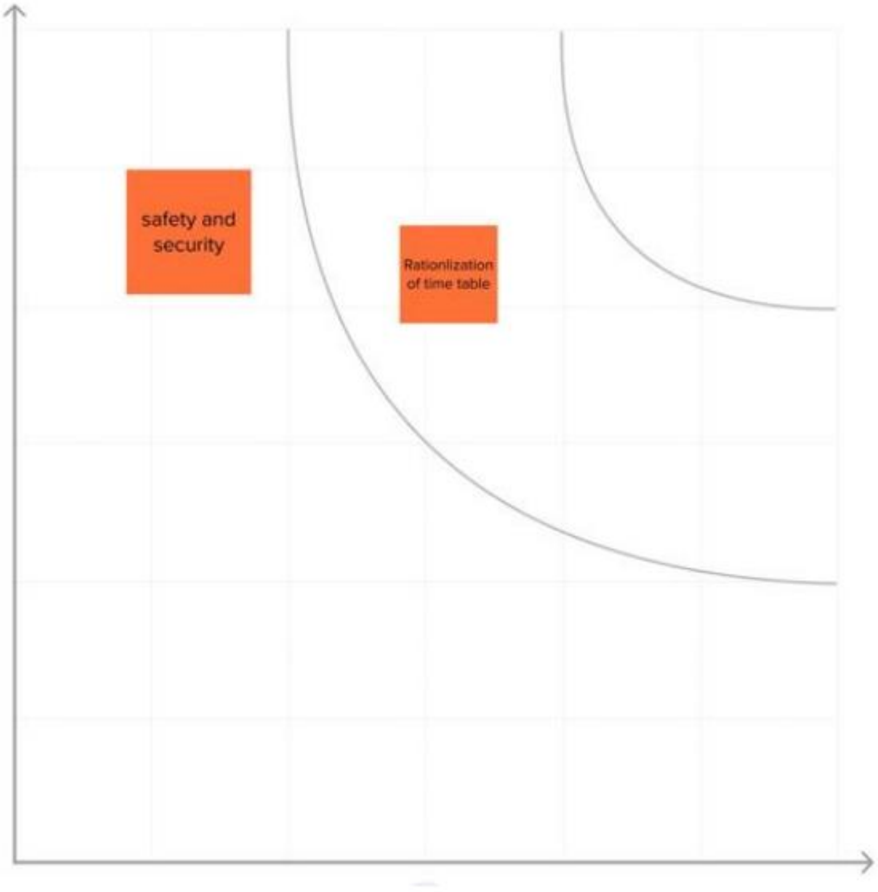
Faster booking

Tatkal

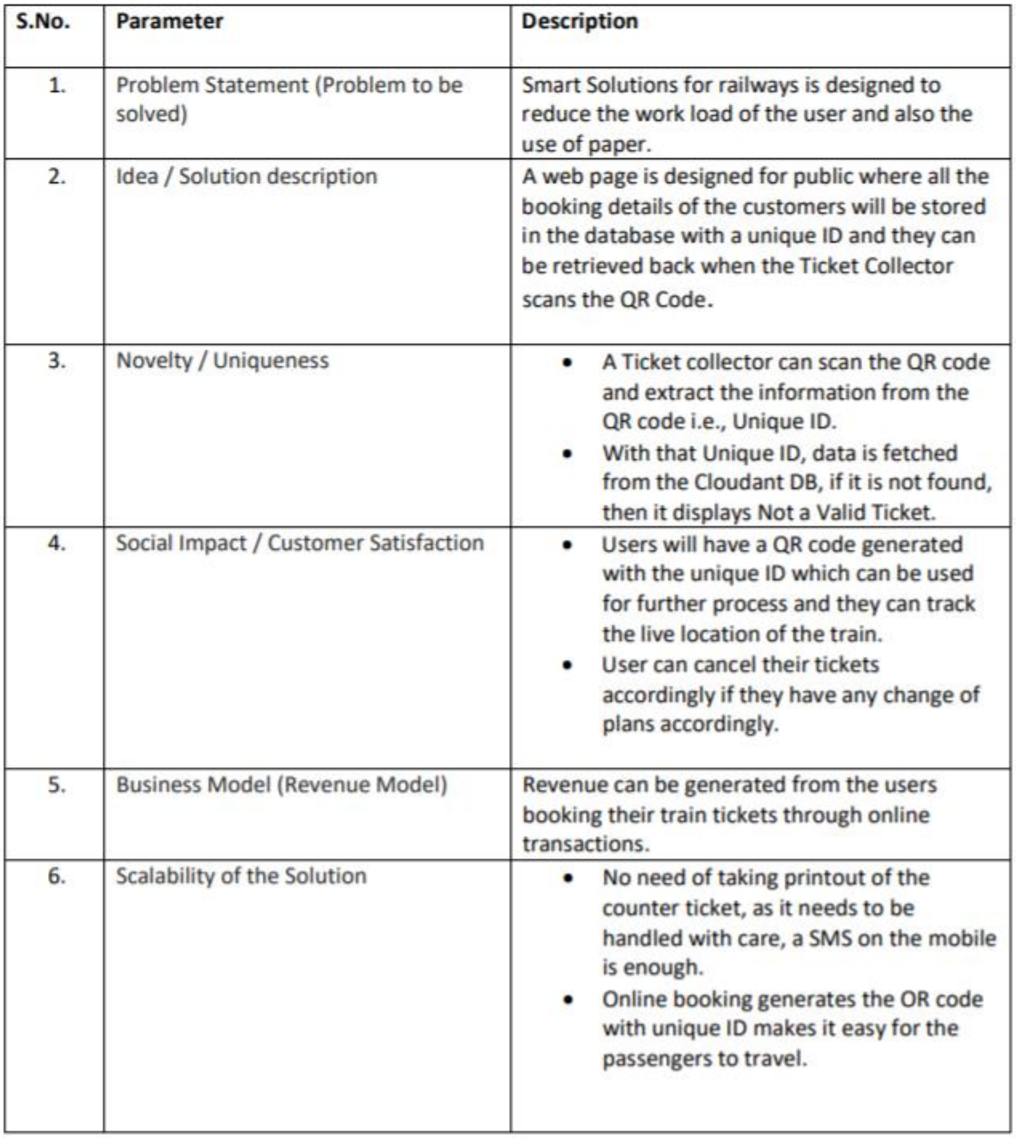
Jude

Particulars of Train

## PRIORITIZE:

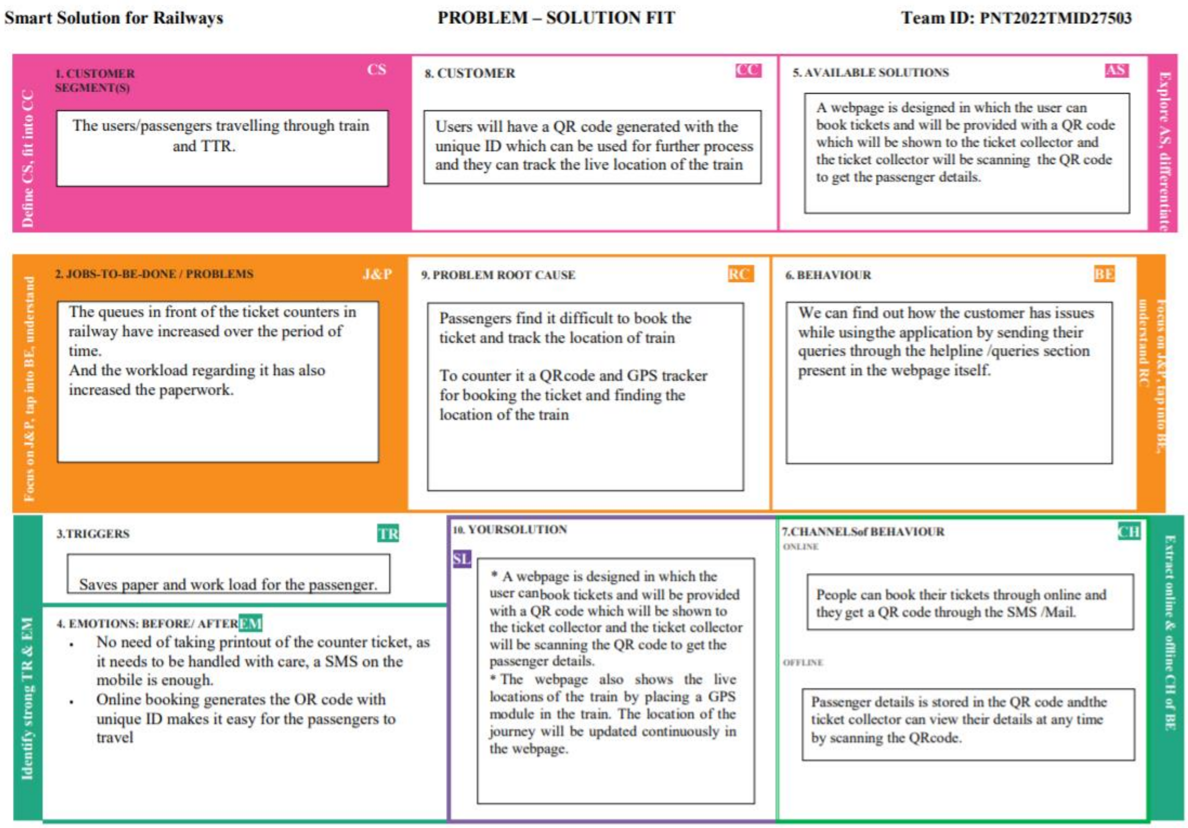


## 3.3 PROPOSED SOLUTION



| S.No. | Parameter | Description |
|---|---|---|
| 1. | Problem Statement (Problem to be solved) | Smart Solutions for railways is designed to reduce the work load of the user and also the use of paper. |
| 2. | Idea / Solution description | A web page is designed for public where 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. | Novelty / Uniqueness | • A Ticket collector can scan the QR code and extract the information from the QR code i.e., Unique ID.<br>• With that Unique ID, data is fetched from the Cloudant DB, if it is not found, then it displays Not a Valid Ticket. |
| 4. | Social Impact / Customer Satisfaction | • Users will have a QR code generated with the unique ID which can be used for further process and they can track the live location of the train.<br>• User can cancel their tickets accordingly if they have any change of plans accordingly. |
| 5. | Business Model (Revenue Model) | Revenue can be generated from the users booking their train tickets through online transactions. |
| 6. | Scalability of the Solution | • No need of taking printout of the counter ticket, as it needs to be handled with care, a SMS on the mobile is enough.<br>• Online booking generates the OR code with unique ID makes it easy for the passengers to travel. |

## 3.4 PROPOSED SOLUTION FIT



Smart Solution for Railways

PROBLEM – SOLUTION FIT

Team ID: PNT2022TMID27503

Define CS, fit into CC

1. CUSTOMER SEGMENT(S) CS

The users/passengers travelling through train and TTR.

8. CUSTOMER CC

Users will have a QR code generated with the unique ID which can be used for further process and they can track the live location of the train

5. AVAILABLE SOLUTIONS AS

A webpage is designed in which the user can book tickets and will be provided with a QR code which will be shown to the ticket collector and the ticket collector will be scanning the QR code to get the passenger details.

Explore AS, differentiate

Focus on J&P, tap into BE, understand

2. JOBS-TO-BE-DONE / PROBLEMS J&P

The queues in front of the ticket counters in railway have increased over the period of time.
And the workload regarding it has also increased the paperwork.

9. PROBLEM ROOT CAUSE RC

Passengers find it difficult to book the ticket and track the location of train

To counter it a QRcode and GPS tracker for booking the ticket and finding the location of the train

6. BEHAVIOUR BE

We can find out how the customer has issues while usingthe application by sending their queries through the helpline /queries section present in the webpage itself.

Focus on J&P, tap into BE, understand RC

Identify strong TR & EM

3.TRIGGERS TR

Saves paper and work load for the passenger.

4. EMOTIONS: BEFORE/ AFTER EM

- No need of taking printout of the counter ticket, as it needs to be handled with care, a SMS on the mobile is enough.
- Online booking generates the OR code with unique ID makes it easy for the passengers to travel

10. YOURSOLUTION SL

* A webpage is designed in which the user canbook tickets and will be provided with a QR code which will be shown to the ticket collector and the ticket collector will be scanning the QR code to get the passenger details.
* The webpage also shows the live locations of the train by placing a GPS module in the train. The location of the journey will be updated continuously in the webpage.

7.CHANNELSof BEHAVIOUR CH

ONLINE

People can book their tickets through online and they get a QR code through the SMS /Mail.

OFFLINE

Passenger details is stored in the QR code andthe ticket collector can view their details at any time by scanning the QRcode.

Extract online & offline CH of BE

# 4.REQUIREMENT ANALYSIS

## 4.1 FUNCTIONAL REQUIREMENT

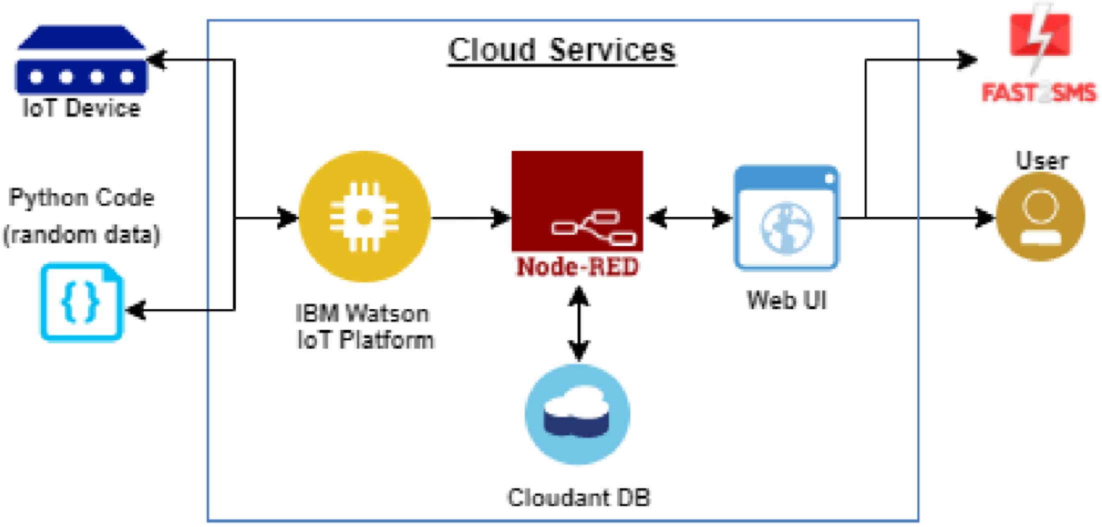
| FR No. | Functional Requirement (Epic) | Sub Requirement (Story / Sub-Task) |
|---|---|---|
| FR-1 | User Registration | Registration through Online Registration through Gmail |
| FR-2 | User Confirmation | Confirmation via Email<br>Confirmation via OTP |
| FR-3 | Application installation | The application is installed through the given link |
| FR-4 | User access | Access the app requirements |

## 4.2 NON FUNCTIONAL REQUIREMENT

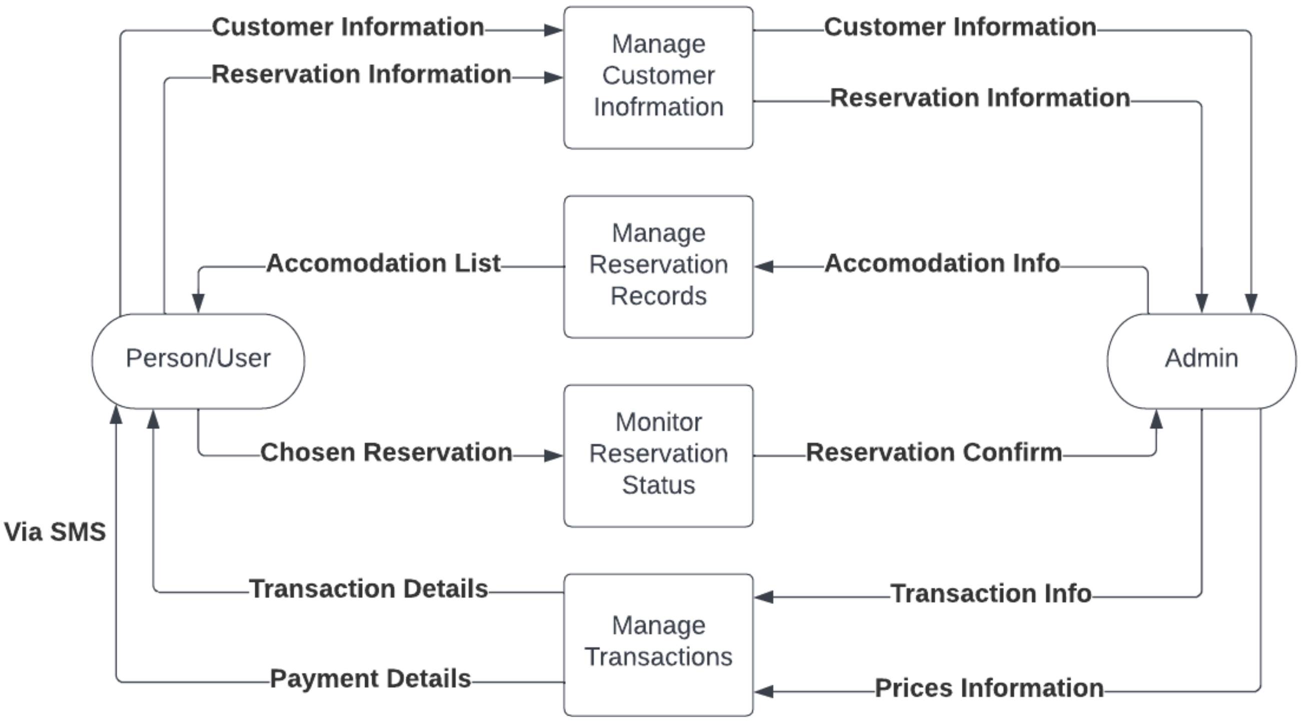
| FR No. | Non-Functional Requirement | Description |
|---|---|---|
| NFR-1 | **Usability** | • The app can be used during the travelling time<br>• Easy and simple<br>• Efficiency is high |
| NFR-2 | **Security** | By clicking on the icon, the alert will be given to the respective officials |
| NFR-3 | **Reliability** | Highly reliable to use |
| NFR-4 | **Performance** | Low error rate |
| NFR-5 | **Availability** | Free source |
| NFR-6 | **Scalability** | It is scalable enough to support many users at the same time |

## 5.PROJECT DESIGN

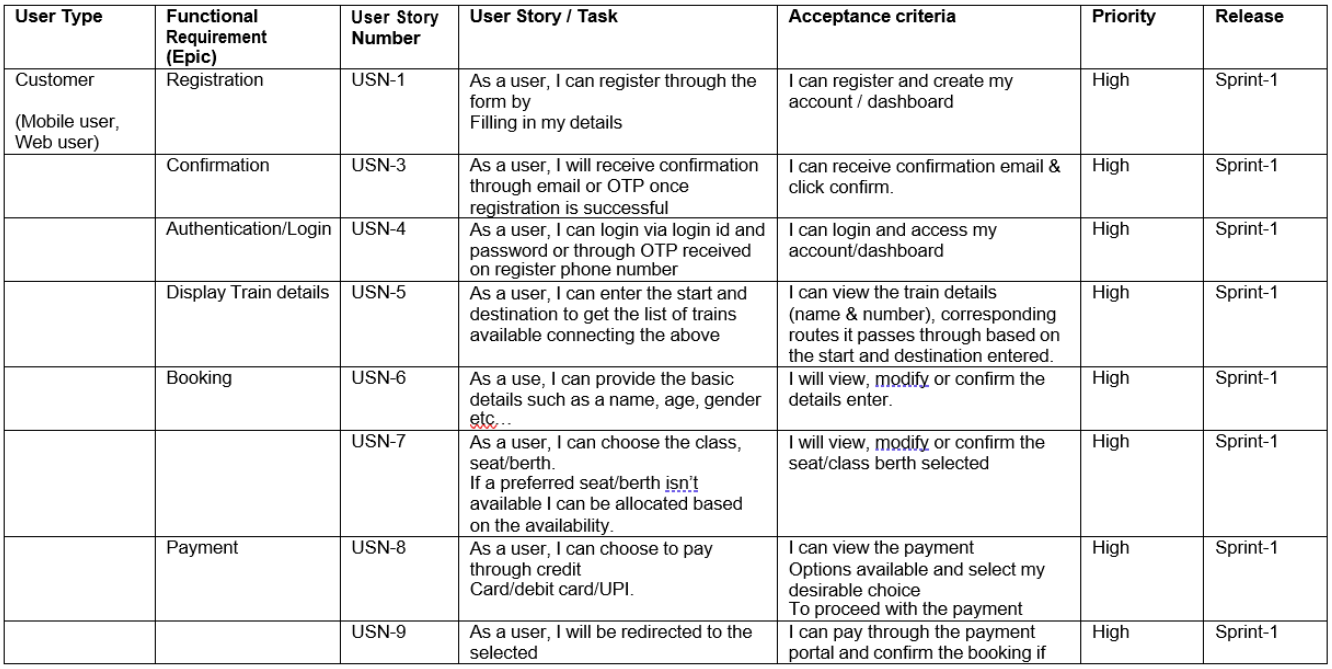
## 5.1 SOLUTION ARCHITECTURE



## 5.2 DATA FLOW DIAGRAM



## 5.3 USER STORIES

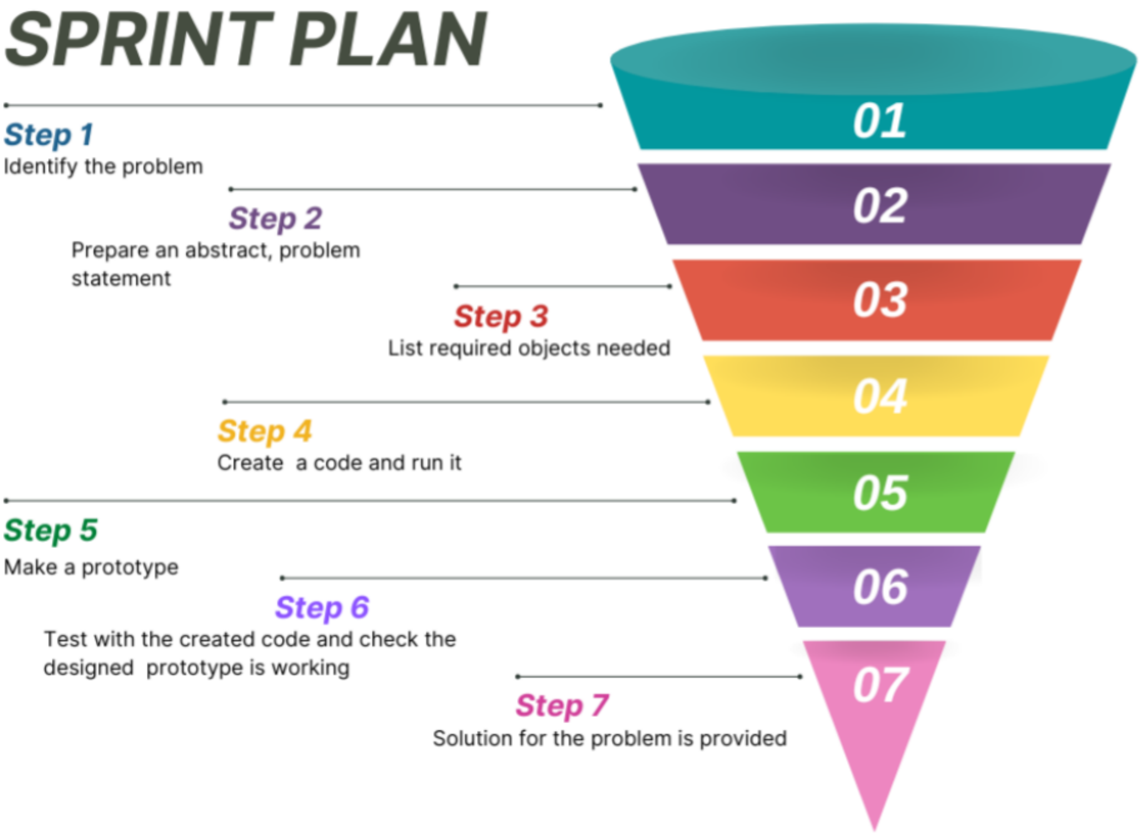


| User Type | Functional Requirement (Epic) | User Story Number | User Story / Task | Acceptance criteria | Priority | Release |
|---|---|---|---|---|---|---|
| Customer<br>(Mobile user, Web user) | Registration | USN-1 | As a user, I can register through the form by Filling in my details | I can register and create my account / dashboard | High | Sprint-1 |
| | Confirmation | 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 |
| | | USN-9 | As a user, I will be redirected to the selected | I can pay through the payment portal and confirm the booking if | High | Sprint-1 |

| User Type | Functional Requirement (Epic) | User Story Number | User Story / Task | Acceptance criteria | Priority | Release |
|---|---|---|---|---|---|---|
| | | | Payment gateway and upon successful completion of payment I'll be redirected to the booking website. | any changes need to 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 transport if the ticket isn't Confirmed | High | Sprint-1 |
| | Remainders notification | USN-12 | I get remainders aboutmy 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 | I can track the train usingGPS 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 | User can cancel tickets ifthere'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 | User can raise queries through the query box or via mail. | I can view my pervious queries. | Low | Sprint-2 |

# 6.PROJECT PLANNING AND SCHEDULING

## 6.1 SPRINT PLANNING AND ESTIMATION


SPRINT PLAN

Step 1
Identify the problem

Step 2
Prepare an abstract, problem statement

Step 3
List required objects needed

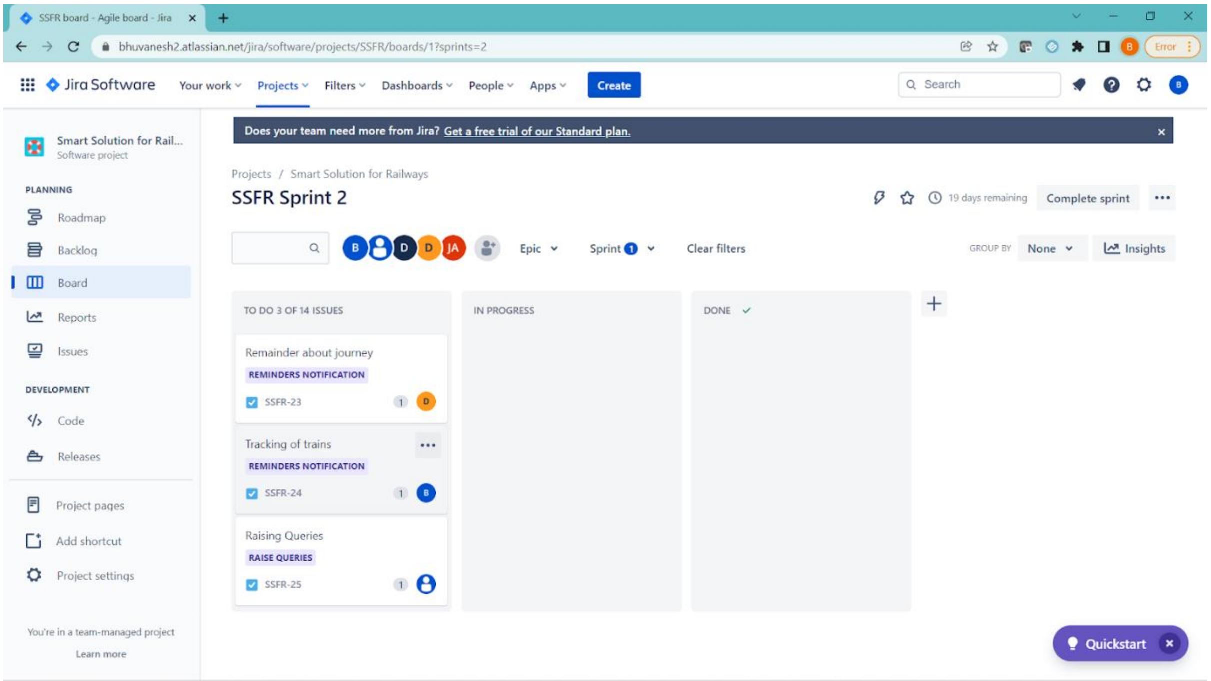
Step 4
Create a code and run it

Step 5
Make a prototype

Step 6
Test with the created code and check the designed prototype is working

Step 7
Solution for the problem is provided

## 6.2 REPORTS FROM JIRA



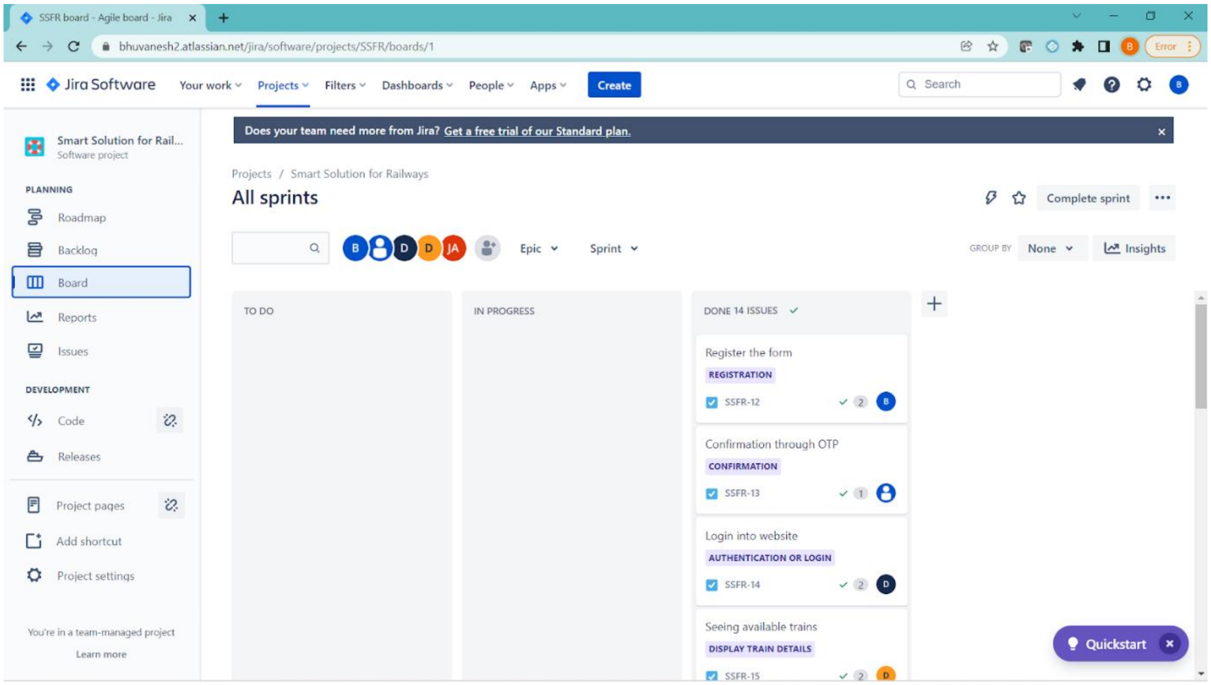
SSFR Sprint 2

TO DO 3 OF 14 ISSUES

- Remainder about journey — REMINDERS NOTIFICATION — SSFR-23 — 1 — D
- Tracking of trains — REMINDERS NOTIFICATION — SSFR-24 — 1 — B
- Raising Queries — RAISE QUERIES — SSFR-25 — 1

IN PROGRESS

DONE ✓


SSFR board - Agile board - Jira
bhuvanesh2.atlassian.net/jira/software/projects/SSFR/boards/1
Jira Software
Your work
Projects
Filters
Dashboards
People
Apps
Create
Search
Does your team need more from Jira? Get a free trial of our Standard plan.
Smart Solution for Rail...
Software project
PLANNING
Roadmap
Backlog
Board
Reports
Issues
DEVELOPMENT
Code
Releases
Project pages
Add shortcut
Project settings
You're in a team-managed project
Learn more
Projects / Smart Solution for Railways
Complete sprint
Epic
Sprint
GROUP BY
None
Insights
Quickstart

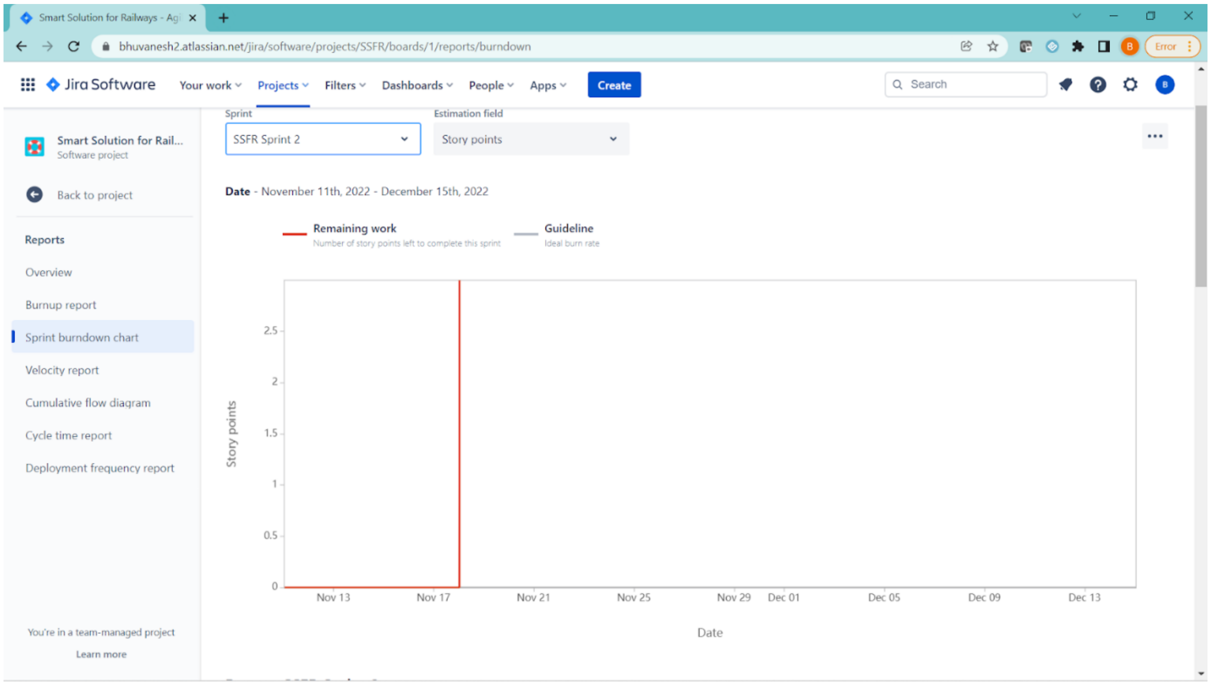
All sprints

TO DO

IN PROGRESS

DONE 14 ISSUES ✓

- Register the form — REGISTRATION — SSFR-12 — ✓ 2 — B
- Confirmation through OTP — CONFIRMATION — SSFR-13 — ✓ 1
- Login into website — AUTHENTICATION OR LOGIN — SSFR-14 — ✓ 2 — D
- Seeing available trains — DISPLAY TRAIN DETAILS — SSFR-15 — ✓ 2 — D



bhuvanesh2.atlassian.net/jira/software/projects/SSFR/boards/1/reports/burndown

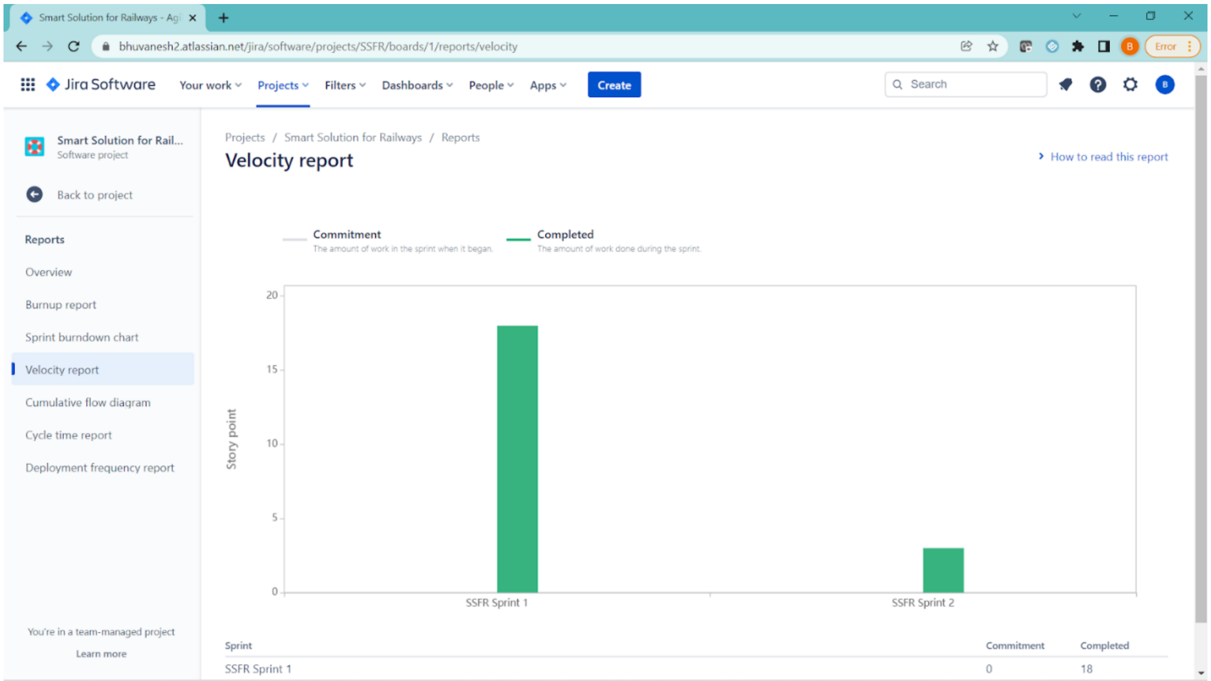
Sprint: SSFR Sprint 2

Estimation field: Story points

**Date** - November 11th, 2022 - December 15th, 2022

🟥 Remaining work
Number of story points left to complete this sprint

Guideline
Ideal burn rate



bhuvanesh2.atlassian.net/jira/software/projects/SSFR/boards/1/reports/velocity

Projects / Smart Solution for Railways / Reports

Velocity report

› How to read this report

Commitment
The amount of work in the sprint when it began.

🟩 Completed
The amount of work done during the sprint.

| Sprint | Commitment | Completed |
|---|---|---|
| SSFR Sprint 1 | 0 | 18 |

# 7.CODING AND SOLUTIONING

## 7.1 Feature 1

- IOT Device
- IBM Watson Platform
- Node Red
- Cloudant DB
- Web UI
- MIT App Inventor
- Python code

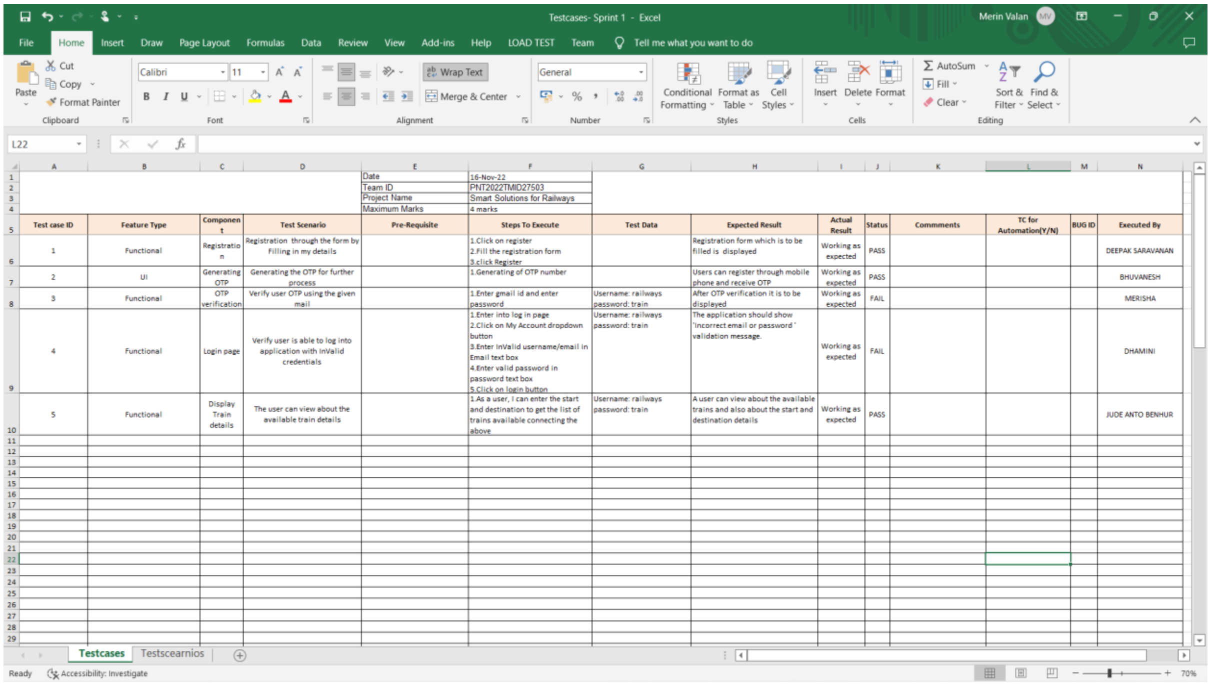
## 7.2 Feature 2

- Login
- Verification
- Ticket Booking
- Adding rating

# 8.TESTING AND RESULTS

## 8.1 Test Cases

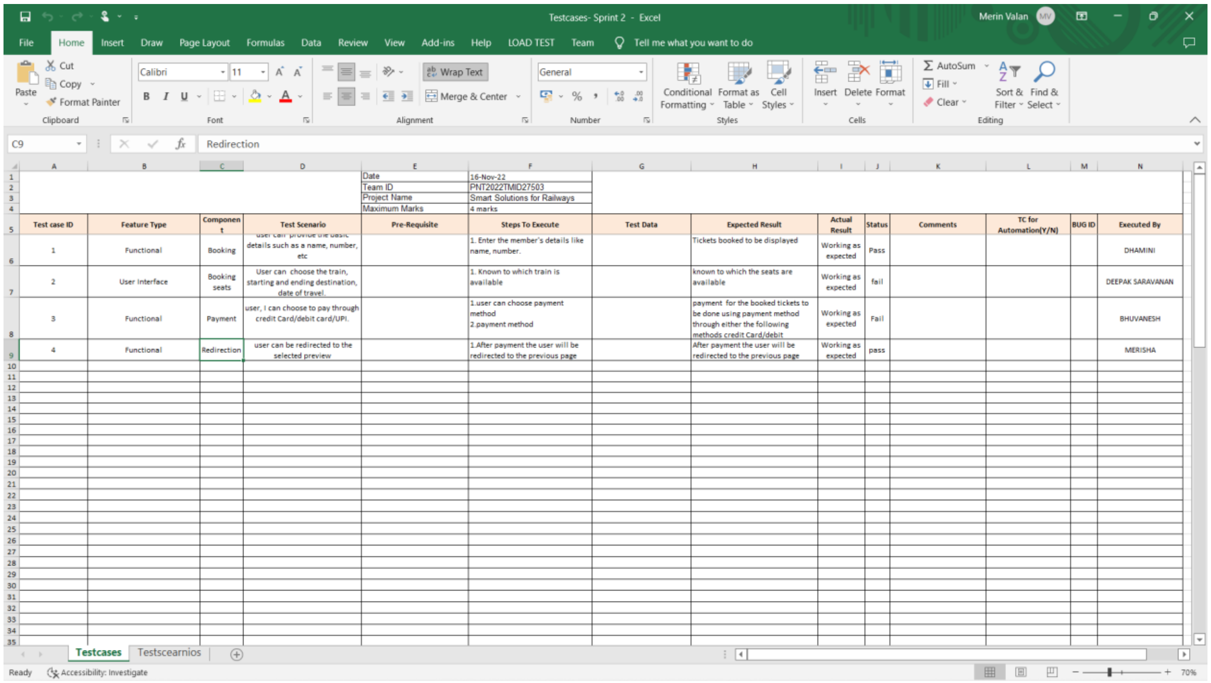
## Test Case 1



| | |
|---|---|
| Date | 16-Nov-22 |
| Team ID | PNT2022TMID27503 |
| Project Name | Smart Solutions for Railways |
| Maximum Marks | 4 marks |

| Test case ID | Feature Type | Component | Test Scenario | Pre-Requisite | Steps To Execute | Test Data | Expected Result | Actual Result | Status | Commments | TC for Automation(Y/N) | BUG ID | Executed By |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | Functional | Registration | Registration through the form by Filling in my details | | 1.Click on register<br>2.Fill the registration form<br>3.click Register | | Registration form which is to be filled is displayed | Working as expected | PASS | | | | DEEPAK SARAVANAN |
| 2 | UI | Generating OTP | Generating the OTP for further process | | 1.Generating of OTP number | | Users can register through mobile phone and receive OTP | Working as expected | PASS | | | | BHUVANESH |
| 3 | Functional | OTP verification | Verify user OTP using the given mail | | 1.Enter gmail id and enter password | Username: railways<br>password: train | After OTP verification it is to be displayed | Working as expected | FAIL | | | | MERISHA |
| 4 | Functional | Login page | Verify user is able to log into application with InValid credentials | | 1.Enter into log in page<br>2.Click on My Account dropdown button<br>3.Enter InValid username/email in Email text box<br>4.Enter valid password in password text box<br>5.Click on login button | Username: railways<br>password: train | The application should show 'Incorrect email or password ' validation message. | Working as expected | FAIL | | | | DHAMINI |
| 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: railways<br>password: train | A user can view about the available trains and also about the start and destination details | Working as expected | PASS | | | | JUDE ANTO BENHUR |

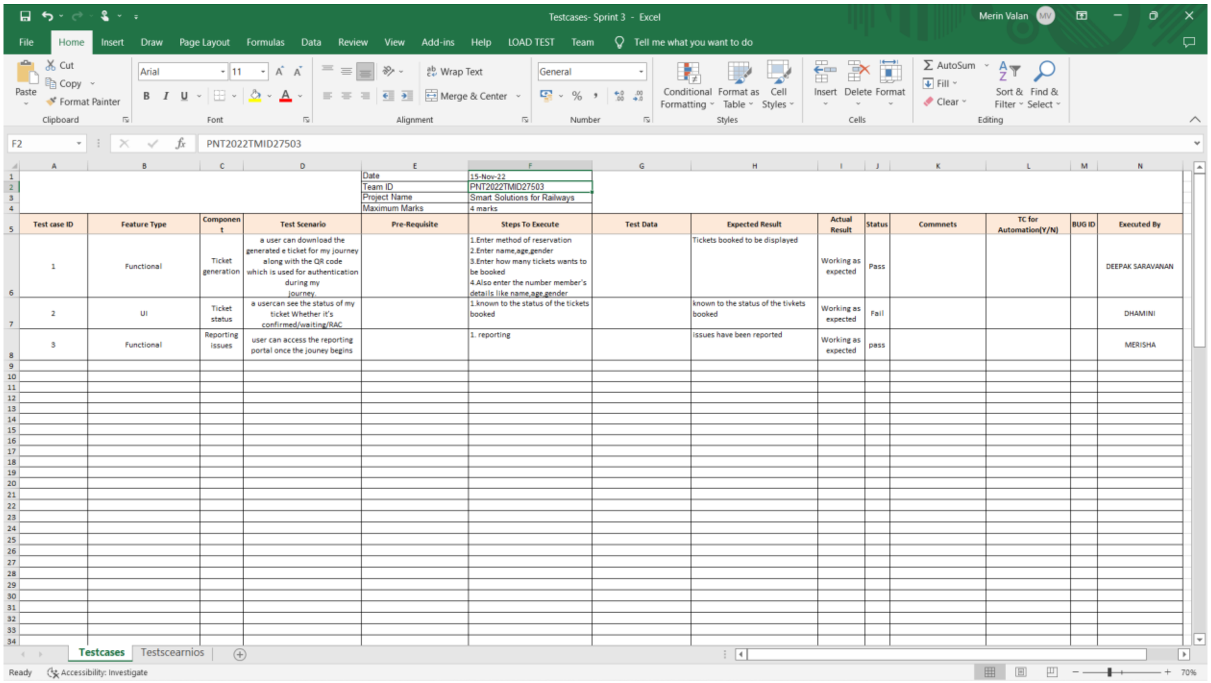
## Test Case 2


Testcases- Sprint 2 - Excel
Merin Valan
File Home Insert Draw Page Layout Formulas Data Review View Add-ins Help LOAD TEST Team Tell me what you want to do
Cut Copy Format Painter Paste Clipboard
Calibri 11 Font
Wrap Text Merge & Center Alignment
General Number
Conditional Formatting Format as Table Cell Styles Styles
Insert Delete Format Cells
AutoSum Fill Clear Sort & Filter Find & Select Editing
C9
Redirection
Testcases Testscearnios
Ready Accessibility: Investigate 70%

| | |
|---|---|
| Date | 16-Nov-22 |
| Team ID | PNT2022TMID27503 |
| Project Name | Smart Solutions for Railways |
| Maximum Marks | 4 marks |

| 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, number, etc | | 1. Enter the member's details like name, number. | | Tickets booked to be displayed | Working as expected | Pass | | | | DHAMINI |
| 2 | User Interface | Booking seats | User can choose the train, starting and ending destination, date of travel. | | 1. Known to which train is available | | known to which the seats are available | Working as expected | fail | | | | DEEPAK SARAVANAN |
| 3 | Functional | Payment | user, I can choose to pay through credit Card/debit card/UPI. | | 1.user can choose payment method<br>2.payment method | | payment for the booked tickets to be done using payment method through either the following methods credit Card/debit | Working as expected | Fail | | | | BHUVANESH |
| 4 | Functional | Redirection | user can be redirected to the selected preview | | 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 | | | | MERISHA |

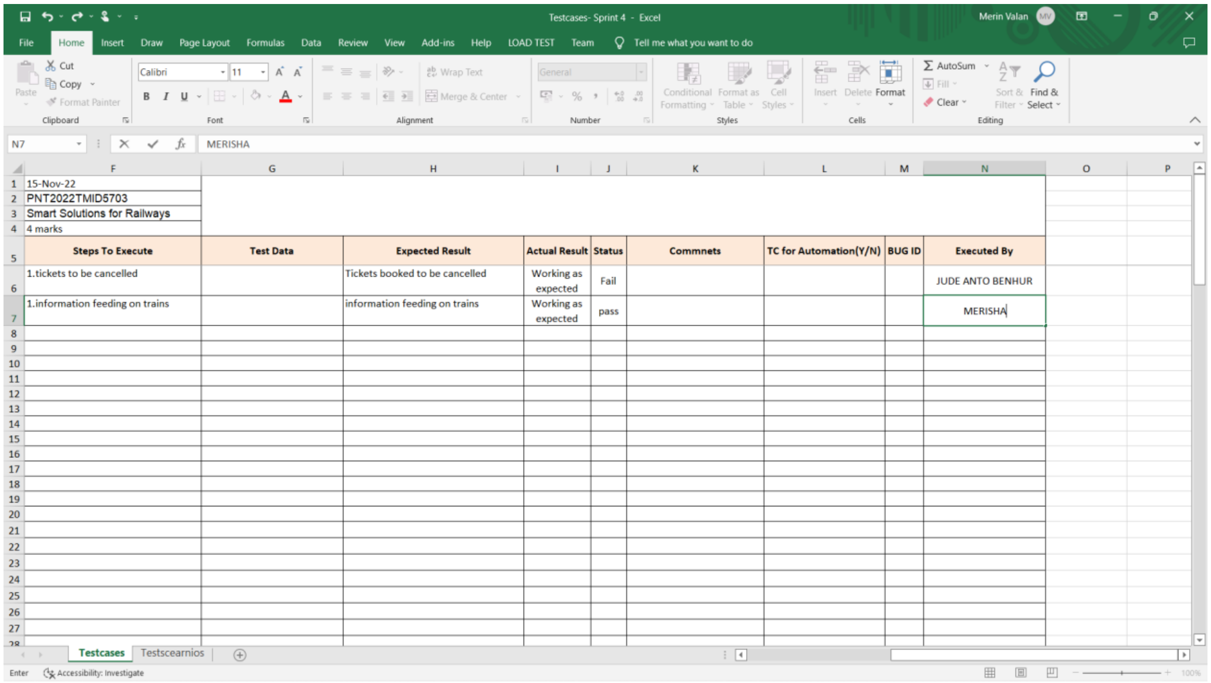
## Test Case 3


Testcases- Sprint 3 - Excel
Merin Valan
File Home Insert Draw Page Layout Formulas Data Review View Add-ins Help LOAD TEST Team Tell me what you want to do
Cut Copy Format Painter Paste Clipboard
Arial 11 Font
Wrap Text Merge & Center Alignment
General Number
Conditional Formatting Format as Table Cell Styles Styles
Insert Delete Format Cells
AutoSum Fill Clear Sort & Filter Find & Select Editing
F2
PNT2022TMID27503
Testcases Testscearnios
Ready Accessibility: Investigate 70%

| | |
|---|---|
| Date | 15-Nov-22 |
| Team ID | PNT2022TMID27503 |
| Project Name | Smart Solutions for Railways |
| Maximum Marks | 4 marks |

| 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/N) | BUG ID | Executed By |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 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<br>2.Enter name,age,gender<br>3.Enter how many tickets wants to be booked<br>4.Also enter the number member's details like name,age,gender | | Tickets booked to be displayed | Working as expected | Pass | | | | DEEPAK SARAVANAN |
| 2 | UI | Ticket status | a usercan 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 tivkets booked | Working as expected | Fail | | | | DHAMINI |
| 3 | Functional | Reporting issues | user can access the reporting portal once the jouney begins | | 1. reporting | | issues have been reported | Working as expected | pass | | | | MERISHA |

## Test Case 4



MERISHA

15-Nov-22
PNT2022TMID5703
Smart Solutions for Railways
4 marks

| Steps To Execute | Test Data | Expected Result | Actual Result | Status | Commnets | TC for Automation(Y/N) | BUG ID | Executed By |
|---|---|---|---|---|---|---|---|---|
| 1.tickets to be cancelled | | Tickets booked to be cancelled | Working as expected | Fail | | | | JUDE ANTO BENHUR |
| 1.information feeding on trains | | information feeding on trains | Working as expected | pass | | | | MERISHA |

## 9.ADVANTAGES

- For safety and protection passengers can use this app
- It is very to use as it is very simple
- It has minimized error rate than other applications which are in use

## 10.DISADVANTAGES

- Network issue may arise

## 11.CONCLUSION

Almost all the countries across the globe strive to meet the demand for safe, fast, and reliable rail services. Lack of operational efficiency and reliability, safety, and security issues, besides aging railway systems and practices are haunting various countries to bring about a change in their existing rail infrastructure. The global rail industry struggles to meet the increasing demand for freight and passenger transportation due to lack of optimized use of rail network and inefficient use of rail assets. Often, they suffer from the lack in smart technologies and latest technological updates to provide the most efficient passenger services. This is expected to induce rail executives to build rail systems that are smarter and more efficient. The passenger reservation system of Indian Railways is one of the world's largest reservation models. Daily about one million passengers travel in reserved accommodation with Indian Railways. Another sixteen million travel with unreserved tickets in Indian Railways. In this vast system, it is a herculean task to efficiently handle the passenger data, which is a key point of consideration now-a-days. But the implementation of the latest technological updates in this system gradually turns inevitable due to increasing demand for providing the most efficient passenger services. Handling the passenger data efficiently backed by intelligent processing and timely retrieval would help backing up the security breaches. Here we've explored different issues of implementing smart computing in railway systems pertaining to reservation models besides pointing out some future scopes of advancement. Most significant improvements have been evidenced by more informative and user-friendly websites, mobile applications for real-time information about vehicles in motion, and e-ticket purchases and timetable information implemented at stations and stops. With the rise of Industry, railway companies can now ensure that they are prepared to avoid the surprise of equipment downtime. Like above mentioned, the developed application of our project can lead the passenger who travel can travel safely without any fear.

## 12.FUTURE SCOPE

This application is ensured for safety for the passengers while they are travelling alone as well as they travel with their family or friends. In future, this application may also be used by passengers who travel through bus. By further enhancement of the application the passengers can explore more features regarding their safety.

## 13.APPENDIX

### 13.1 SOURCE CODE

### DESTINATION:

```
#import module
 import requests
 from bs4 import BeautifulSoup
 # user define function
 # Scrape the data
 def getdata(url):
```

```
r = requests.get(url)
return r.text
# input by geek
from_Station_code = "GAYA"
from_Station_name = "GAYA"
To_station_code = "PNBE"
To_station_name = "PATNA"
# url
url = "https://www.railyatri.in/booking/trains-between-
stations?from_code="+from_Station_code+"&from_name="+from_Station_name+"+JN
+ \
To_station_code+"&to_name="+To_station_name + \
"+JN+&user_id=-1603228437&user_token=355740&utm_source=dwebsearch_tbs_searc
# pass the url
# into getdata function
htmldata = getdata(url)
soup = BeautifulSoup(htmldata, 'html.parser')
# find the Html tag
# with find()
# and convert into string
data_str = ""
for item in soup.find_all("div", class_="col-xs-12 TrainSearchSection"):
data_str = data_str + item.get_text()
result = data_str.split("\n")
print("Train between "+from_Station_name+" and "+To_station_name)
print("")
# Display the result
for item in result:
if item != "":
print(item)
```

**Registration**

```
From tkinter import*
base = Tk()
base.geometry("600x600")
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="Name", width=10, font=("Times New Roman",12))
lb1.place(x=20, y=120)
```

```
en1= Entry(base)
en1.place(x=200, y=120)
lb3= Label(base, text="Email", width=10, font=("Times New Roman",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=("Times New
Roman",12))
lb4.place(x=19, y=200)
en4= Entry(base)
en4.place(x=200, y=200)
lb5= Label(base, text="Select Gender", width=15, font=("Times New
Roman",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_states = ("Andhra Pradesh","Arunchal
Pradesh","Assam","Bihar","Chhattisgarh","Goa",
"Gujarat","Haryana","Himachal Pradesh","Jharkhand","Karnataka",
"Kerala","Madhya Pradesh","Maharashtra","Manipur","Meghalaya",
"Mizoram","Nagaland","Odisha","Punjab","Rajasthan","Sikkim",
"Tamil Nadu", "Telangana","Tripura","Uttarkhand","Uttar pradesh",
"West Bengal")
cv = StringVar()
drplist= OptionMenu(base, cv, *list_of_states)
drplist.config(width=15)
cv.set("Andhra Pradesh")
lb2= Label(base, text="Select State", width=13,font=("Times New
Roman",12))
lb2.place(x=14,y=280)
drplist.place(x=200, y=275)


lb6= Label(base, text="Enter Password", width=13,font=("Times New
Roman",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=("Times New Roman",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()
```

**Seats Booking**

```
Defberth_type(s):
            if s>0 and s<73:
            if s % 8 == 1 or s % 8 == 4:
            print (s), "is lower berth"
            elif s % 8 == 2 or s % 8 == 5:
            print (s), "is middle berth"
            elif s % 8 == 3 or s % 8 == 6:
            print (s), "is upper berth"
            elif s % 8 == 7:
            print (s), "is side lower berth"
            else:
            print (s), "is side upper berth"
            else:
            print (s), "invalid seat number"

            # Driver code
            s = 15
            berth_type(s)

            s = 0
            berth_type(s)

            s = 45
            berth_type(s)
```

## Ticket Booking

```
Print(“\nTicket Booking System\n”)
while restart != ('N','NO','n','no'):
print("1.Check PNR status")
print("2.Ticket Reservation")
option = int(input("\nEnter your option : "))

if option == 1:
print("Your PNR status is t3")
exit(0)

elif option == 2:
people = int(input("\nEnter no. of Ticket you want : "))
name_l = []
age_l = []
sex_l = []
for p in range(people):
name = str(input("\nName : "))
name_l.append(name)
age = int(input("\nAge : "))
age_l.append(age)
sex = str(input("\nMale or Female : "))
sex_l.append(sex)

restart = str(input("\nDid you forgot someone? y/n: "))
if restart in ('y','YES','yes','Yes'):
restart = ('Y')
else :
x = 0
print("\nTotal Ticket : ",people)
for p in range(1,people+1):
print("Ticket : ",p)
print("Name : ", name_l[x])
print("Age : ", age_l[x])
print("Sex : ",sex_l[x])
x += 1
```

## Confirmation

```
Import requests json
pnr_no = "4465877280"
```

```
a = "https://indianrailapi.com/api/v2/PNRCheck/apikey
/375b8caa6a27e3d1b9922c851245c93f/PNRNumber/"+ pnr_no + "/"

dk = requests.get(a)
result = dk.json()

if result["ResponseCode"] == '200':

pnr_number = result['PnrNumber']
train_name = result["TrainNumber"]
Journey_class = result["JourneyClass"]
Chat_Prepared = result["ChatPrepared"]
from_station = result["From"]
to_station = result["To"]
dateof_journey = result["JourneyDate"]
passengers_list = result["Passangers"]

print(f"PnrNumber {pnr_number}\nTrain Name {train_name}\
nJourney Class {Journey_class}\nChart Preadared {Chat_Prepared}\n
From Station {from_station} To {to_station}\nJourney Date
 {dateof_journey}")

for passenger in passengers_list:

passenger_num = passenger["Passenger"]

current_status = passenger["CurrentStatus"]

booking_status = passenger["BookingStatus"]

print(" passenger number : " + str(passenger_num)
+ "\n current status : " + str(current_status)
+ "\n booking_status : " + str(booking_status))
else:
print("Wrong Pnr Number")
```

**Ticket Generation**

```
Class Ticket:
counter=0
def __init__(self,passenger_name,source,destination):
self.__passenger_name=passenger_name
self.__source=source
self.__destination=destination
self.Counter=Ticket.counter
Ticket.counter+=1
def validate_source_destination(self):
if (self.__source=="Delhi" and (self.__destination=="Pune" or
self.__destination=="Mumbai" or self.__destination=="Chennai" or
self.__destination=="Kolkata")):
return True
else:
return False

def generate_ticket(self ):
if True:
__ticket_id=self.__source[0]+self.__destination[0]+"0"+str(self.Counter)
print( "Ticket id will be:",__ticket_id)
else:
return False
def get_ticket_id(self):
return self.ticket_id
def get_passenger_name(self):
return self.__passenger_name
def get_source(self):
if self.__source=="Delhi":
return self.__source
else:
print("you have written invalid soure option")
return None
def get_destination(self):
if self.__destination=="Pune":
return self.__destination
elif self.__destination=="Mumbai":
return self.__destination
elif self.__destination=="Chennai":
return self.__destination
elif self.__destination=="Kolkata":
return self.__destination

else:
return None
```

**OTP Generation**

Import library

```
import math, random

# function to generate OTP
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())
```

**OTP Verification**

```
Import os
import math
import random
import smtplib

digits = "0123456789"
OTP = ""

for i in range (6):
OTP += digits[math.floor(random.random()*10)]
otp = OTP + " is your OTP"
message = otp
s = smtplib.SMTP('smtp.gmail.com', 587)
s.starttls()

emailid = input("Enter your email: ")
s.login("YOUR Gmail ID", "YOUR APP PASSWORD")
s.sendmail('&&&&&&',emailid,message)

a = input("Enter your OTP >>: ")
if a == OTP:
print("Verified")
else:
        print("Please Check your OTP again")
```

**GITHUB LINK:**

**https://github.com/IBM-EPBL/IBM-Project-37065-1660300130**