

Final Report for IBM Project

Title	Project Report
Team ID	PNT2022TMID53688
Project Name	Smart Solutions For Railways

Team Members:

Biplab Das S 714019106015

Harsha K 714019106031

Jaffer Sathick HM 714019106035

Jawahar Prasad S 714019106036

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 (Explain the features added in the project along with code)

- 7.1 Feature 1
- 7.2 Feature 2
- 7.3 Database Schema (if Applicable)

8. TESTING

- 8.1 Test Cases
- 8.2 User Acceptance Testing

9. RESULTS

- 9.1 Performance Metrics

10. ADVANTAGES & DISADVANTAGES

11. CONCLUSION

12. FUTURE SCOPE

13. APPENDIX

Source Code

GitHub & Project Demo Link

SMART SOLUTION FOR RAILWAYS

1.INTRODUCTION :

Smart Solutions for railways is designed to reduced the work load of the user and also the use of paper.

1.1 PROJECT OVERVIEW :

- 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.

1.2 PURPOSE :

Smart Solutions for railways is designed to reduced the work load of the user and also the use of paper.

ABSTRACT

From years, Indian Railways has been the most prominent, convenient and affordable means of transport. In this project, our motive is to digitize the ticket checking and validation system along with allocation of vacant berths, respective to gender safety in an ongoing journey, which will eliminate the major work of TTE (Train Ticket Examiner). Our proposed system helps to allocate seats to the passengers. The purpose of this project is growth of digitization with QR code and the proposed application. This digitization will eradicate corruption as the system will maintain all the records in database. Also, study explored the reservation status will assure safety of passenger.

2.LITERATURE SURVEY

2.1 EXISTING PROBLEM :

The seats of their choice are not available to passengers. They must bring a physical ticket with them, which could be misplaced. Quick-moving passengers might not have enough time to wait for the train for an extended period of time. They can decide whether to wait or use another form of transportation by knowing the train's current location.

2.2 APP REFERENCES

TITLE	DESCRIPTION
An intelligent ticket checker application for train using QR code	We have precisely got our two of views fulfilled in this project: 1. QR code working. 2. Ticket system for trains.
Android application generating QR code as Railway Ticket	With the help of this project we came up with the idea to print QR code on tickets.
Dynamic and transparent seat allocation for each gender safety using QR code in mobile application	From this project we came up with the idea to make an android app for ticket checker.
Android application for local railway ticketing using GPS validation	They proposed the various techniques for buying metro tickets or local railway tickets through their Smartphone application and introduced ticket checker.
Android application for local railway ticketing using GPS validation	They proposed the various techniques for buying metro tickets or local railway tickets through their Smartphone application and introduced ticket checker.

REFERENCES

[1] Smita Patil, Shruti Desurkar, Deepali Sanaskar (2016) “An intelligent ticket checker application for train using QR code.”

[2] Karhikram, Sarvanan, Madhavan (2014) “Android application generating QR code as Railway Ticket.”

[3] Joydeep Singh, Vaibhav Shukla (2015) “Dynamic and transparent seat allocation using QR code in Mobile application.”

[4] Tushar Dongare, Akshay Babar (2014) “Android application for ticket reservation with GPS as ticket validation.”

[5] Snehal Kalbhor, Ashwini Mangulkar, Mrs. Snehal Kulkarni (2014) “Android application for local railway ticketing using GPS validation.”

IOT REFERENCES

S.No	TITLE	PROPOSED WORK	TOOLS USED/ ALGORITHM	TECHN OLOGY	ADVANTAGES/ DISADVANTAGE S
1.	smart railway systems of layer applications based on internet of things	Railway networks across the world are getting busier and busier; trains travel at higher speeds and carry more passengers or heavier axle loads than ever before. Accordingly, the railway industry calls for new information technologies (ITs) to meet its development. Railway systems have already relied on ITs almost as much as they rely on physical assets, and this dependence is growing as these systems face burgeoning demands .	big data, sensors, data mining,intelligent systems	Internet of Things	This chapter will discuss the framework and for a smart railway based on the Internet of Things and big data, we present the architecture of a smart railway, which is divided into four layers perception and action layer, transfer layer, data engine layer, application layer, and discuss the advanced technologies in each layer.

S.No	TITLE	PROPOSED WORK	TOOLS USED/ ALGORITHM	TECHNOLOGY	ADVANTAGES/ DISADVANTAGES
2.	A Novel Approach for Big Data Classification and Transportation in Rail Networks	A new framework into future data-driven railway condition monitoring systems (RCM). For this proposed an edge processing unit that includes two main parts: a data classification model that classifies Internet of Things (IoT) data into maintenance-critical data (MCD) and maintenance-non-critical data (MNCD) and a data transmission .	CBM, RCM, MCD,MNCD	Internet of Things	The development of condition based monitoring CBN systems in the railway industry has received the highest investment policy will deal with of big data problem in the future because these have velocity, and volume . RCM will be strongly reliant on data received from heterogeneous IOT devices .

S.No	TITLE	PROPOSED WORK	TOOLS USED/ ALGORITHM	TECHNOLOGY	ADVANTAGES/ DISADVANTAGES
3.	Remote sensor networks for condition monitoring: An application on railway industry	In recent years, the range of sensing technologies has expanded rapidly, whereas sensor devices have become cheaperThis has prompted to a fast extension in condition checking of frameworks, structures, vehicles, and hardware utilizing sensors. Key components are the current advances in systems administration	Remote sensor systems (WSNs) , LPWAN,RFID.	Internet of Things	This is indispensable for the advancement, redesigning, and extension of railroad systems. This venture studies the remote sensors arrange innovation for checking in the railroad business for dissecting frameworks, structures, vehicles, and apparatus.

S.No	TITLE	PROPOSED WORK	TOOLS USED/ ALGORITHM	TECHNOLOGY	ADVANTAGES/ DISADVANTAGE
4.	5G key technologies for smart railways	Railway communications has attracted significant attention from both academia and industries due to the booming development of railways, especially high-speed railways (HSRs). To be in line with the vision of future smart rail communications, the rail transport industry needs to develop innovative communication network architectures and key technologies	SDN , SD-WAN ,5G edge, digital and hybrid multi cloud	Internet of Things	we have identified significant 5G-based key technologies for HSRs, such as spatial modulation, fast channel estimation, cell-free massive multiple-input-multiple-output (MIMO), mmWave, efficient beamforming, wireless backhaul, reliable low latency communications, and enhanced handover strategies.

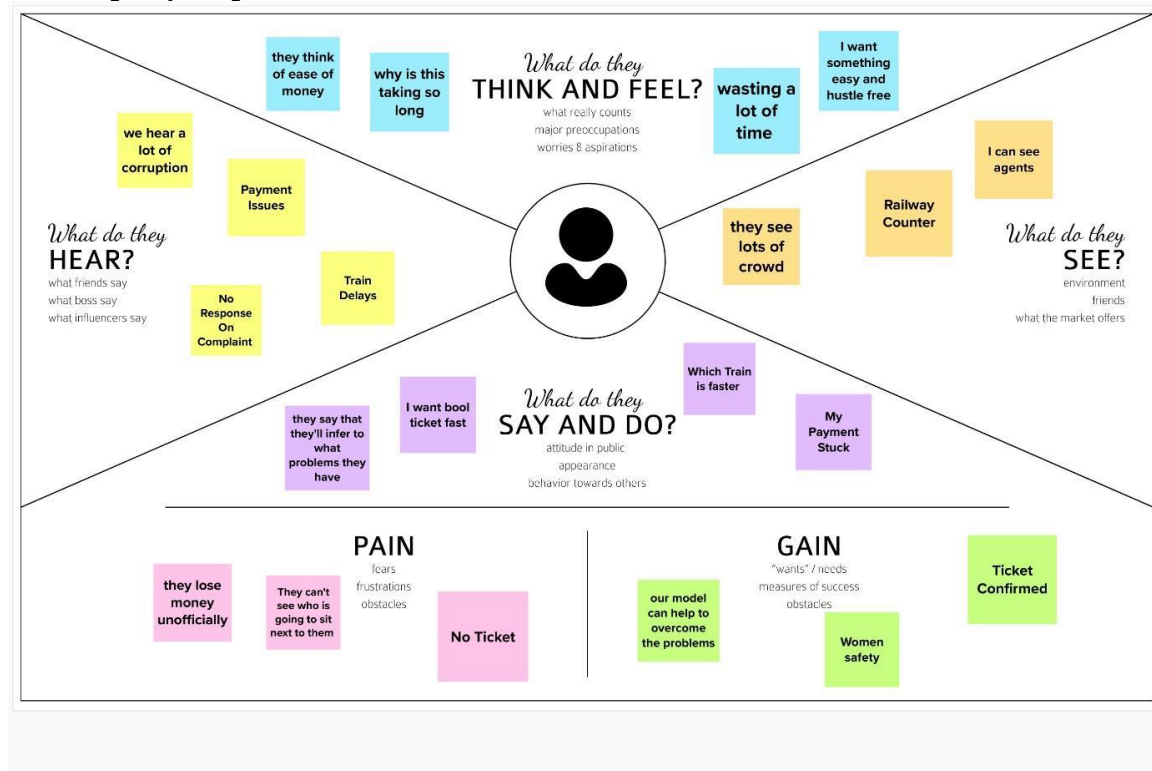
S.No	TITLE	PROPOSED WORK	TOOLS USED/ ALGORITHM	TECHNOLOGY	ADVANTAGES/ DISADVANTAGES
5.	OTFS-TSMA for Massive Internet of Things in High-Speed Railway	Massive internet of things (mIoT) could play an important role in the future smart high-speed railway (HSR), where grant-free multiple access technologies are required. Recently, tandem spreading multiple access (TSMA) has been raised for mIoT without mobility which achieves high connectivity and reliability.	MIOT , TSMA , OTFS Transceiver	Internet of Things	the four typical smart railways services, including railway safety-critical service, passenger oriented service , decision making smart HSR to enable environment sensing of IOT service in greater ways.

2.3.PROBLEM STATEMENT DEFINITION:

QUESTION	DESCRIPTION
Who does the problem affect?	Voyagers, travelers & tourists.
What are the boundaries of the problem?	Mechanism for purchasing train tickets that generates a special QR code for each ticket. GPS tracking of a train's present location.
What is the issue?	The seats of their choice are not available to passengers. They must bring a physical ticket with them, which could be misplaced. Quick-moving passengers might not have enough time to wait for the train for an extended period of time. They can decide whether to wait or use another form of transportation by knowing the train's current location.
When does the issue occur?	All the time.
Where is the issue occurring?	Train and in Railway Stations currently available train reservation systems.
Why is it important that we fix the problem?	Railway is one of the most common modes of transport. Improving the user experience is very important. An efficient way to check the tickets is of top priority.

3.IDEATION & PROPOSED SOLUTION

3.1.Empathy Map Canvas



3.2 Ideation & Brainstorming :



3.3 Proposed Solution :

S.NO.	PARAMETER	DESCRIPTION
1.	Problem Statement (Problem to be solved)	<ul style="list-style-type: none">● The goal of smart solutions for railways is to lessen user workload, reduce paper use, and deliver real-time train location information.● Due to their hectic schedules and rapid global travel, online booking process is required. The lines at the area in front of the ticket windows in railroad has to be reduced.● The amount has significantly increased over time.● "Ticket reservations made at the counter are insufficient", very practical for the travellers.● The travellers are having difficulty obtaining tickets in time for the even counters. Therefore, people prefer to use online tickets booking.
2.	Idea / Solution description	<ul style="list-style-type: none">● The user can book tickets on a website, respective to the allocations of each gender safety where they will also receive a QR code that they can display to the ticket collector so that the ticket collector can scan it to retrieve the user's information.● By installing a GPS module inside the train, the website also displays the train's real-time positions. The journey's location will be regularly updated on the website.● The database will contain the user's booking information, which may be retrieved at any time.

3.	Novelty / Uniqueness	<ul style="list-style-type: none"> ●The homepage will offer the customer a QR code, which will cut down on paper effort. ●When the ticket collector scans the QR Code, all of the client booking information will be stored in the database with a special ID and be retrievable. ●Also included is an interactive seat map.
4.	Social Impact / Customer Satisfaction	<ul style="list-style-type: none"> ●There is no need to go to the station to purchase tickets, and the transaction procedure is made simple. The system is also easy to use and trustworthy. ●One can manage online ticket reservations and submit a cancellation request in the event of a change in plans. ●On all confirmations and cancellations, the customer will receive notifications through email and mobile phone.
5.	Business Model (Revenue Model)	<ul style="list-style-type: none"> ● Model (Revenue Model) - With the help of this application, the client may plan their trip, check the availability of seats, examine an interactive seat map, and choose a seat that is most convenient for them. ● Additionally, it makes it simple for your clients to schedule daily shuttles and journeys, and it minimizes carrying around tickets. ● The consumer may also view the train's current location. Without this arrangement, individuals would have to travel to the station to purchase

		<p>tickets and bring those tickets with them to the ticket collector.</p>
6.	Scalability of the Solution	<ul style="list-style-type: none"> ● Counter tickets must be handled carefully, but text messages on a phone are more than sufficient. ● Printouts are unnecessary, by disregarding printouts, you are becoming more environmentally conscious and helping to create a greener planet. ● Tell TTR your name and that you are a passenger with a legitimate evidence without having to take out your wallet or show them your ticket. ● While purchasing an E-ticket online straight from your bank makes your work easier, purchasing a counter ticket required you to have cash.

3.4 Problem Solution fit :



4 REQUIREMENT ANALYSIS

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Web page
FR-2	User Confirmation	Confirmation via OTP through SMS or OTP through gmail
FR-3	User QR code generation	QR code generator
FR-4	GPS tracker	Current location of train is tracked
FR-5	Available of seat	Based on the availability of the seat respective to gender, we can be book the ticket with own convenience.
FR-6	Testing	UQR =scanned QR from the user ; CQR= the code already obtained by the ticket collector ; if(UQR==CQR) { Print(“PASS”); } else { Print(“FAIL”); }

Non-functional Requirements:

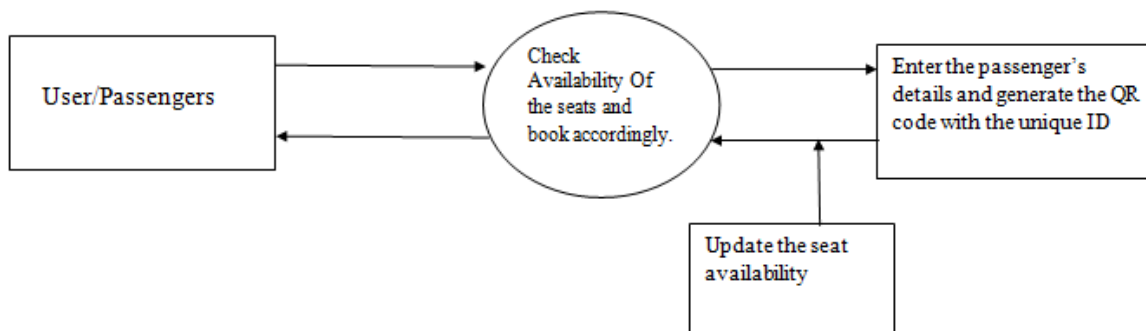
Following are the non-functional requirements of the proposed solution.

FR No.	Non-Functional Requirement	Description
NFR-1	Usability	User can navigate and easily to find the location of the train
NFR-2	Security	Each individual is safe and details are secured in the cloud database
NFR-3	Reliability	Reliable to the user without any failure.
NFR-4	Performance	User friendly and easy to use.
NFR-5	Availability	Availability we can access it 24*7 using the internet.

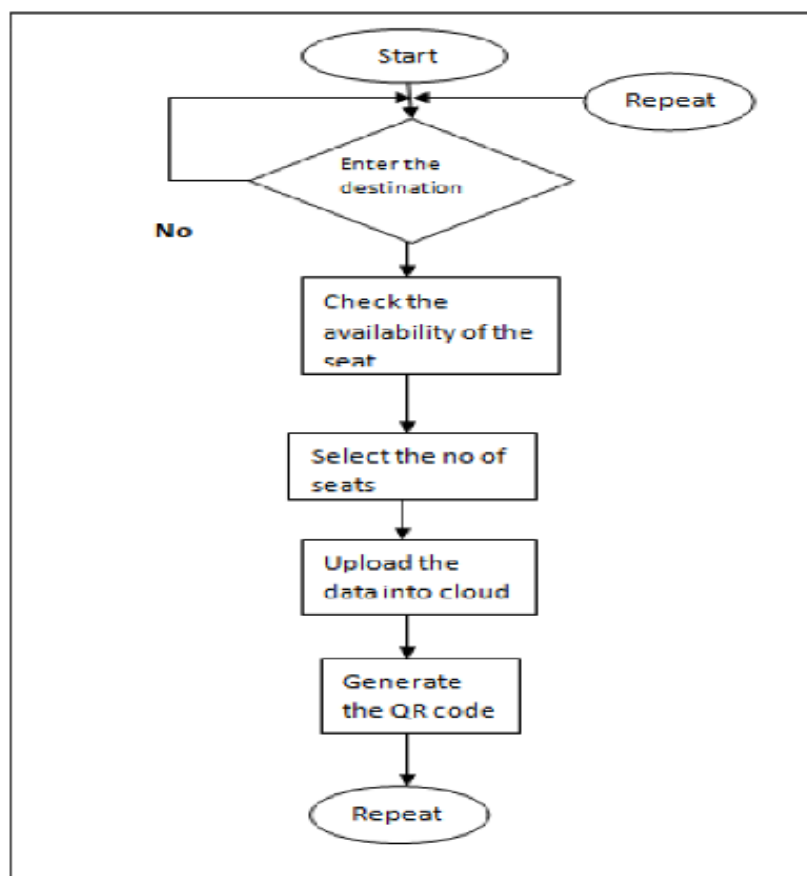
NFR-6	Scalability	Support the user cost of the reservation, available seats and track the train's status.
-------	-------------	---

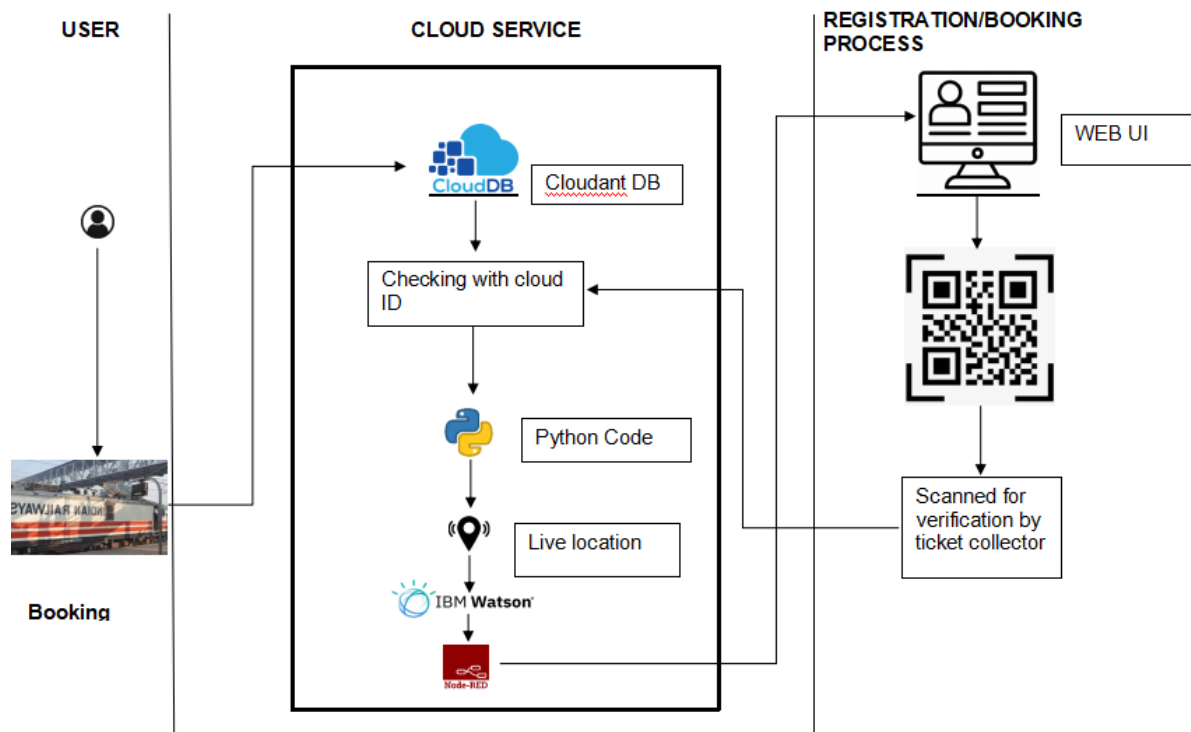
5 PROJECT DESIGN

5.2 Data Flow Diagrams



Solution & Technical Architecture





5.3 User Stories

+ Application Roles:

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High	Sprint-1
		USN-2	As a user, I will receive confirmation SMS/ email once I have registered for the application.	I can receive confirmation email/SMS & click confirm	High	Sprint-1
		USN-3	As a user, I can register for the application through Facebook/LinkedIn.	I can register & access the dashboard with Facebook/LinkedIn Login	Low	Sprint-2
		USN-4	As a user, I can register for the application through Gmail.	I can register & access the dashboard with Gmail Login	Medium	Sprint-1
	Login	USN-5	As a user, I can log into the application by entering email & password credentials.	I can access easily with email & password to go to dashboard	High	Sprint-1
	Interface	USN-6	As a user, the interface should be user friendly manner.	I can access the application easily		
Customer (Web user)	Dashboard	WUSN-1	As a Web User, I can register and then login to specific information.	I can easily track the train	High	Sprint-1
Customer Care Executive	View manner	CCE-1	As a customer care, I can view the data whenever needed.	I can easy to view the passengers detail when TTR scans the QR code	High	Sprint-1
Reviews	Experiences	RV-1	As an user, people can give their feedbacks and reviews.	I can make even better app with the people reviews and experiences	High	Sprint-2

6 PROJECT PLANNING & SCHEDULING

6.1. Sprint Planning & Estimation

6.2.





7 Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint - 1	User Registration	USN - 1	Registration through app, Setting up User Id and Password	2	High	Jaffer Sathick HM, Harsha K
Sprint - 1	User Login	USN - 2	Login using User Id and Password	2	High	Jaffer Sathick HM, Harsha K
Sprint - 2	App Permission	USN - 3	Need to fill some details before accessing the application and some Id for confirmation details	2	Medium	Biplab Das S, Harsha K, Jaffer Sathick HM, Jawahar Prasad S
Sprint - 3	Booking tickets	USN - 4	User can select the best seat available for their travel and timing	2	High	Biplab Das S, Jawahar prasad S
Sprint - 4	Confirm the tickets	USN - 5	User needs to confirm the Train tickets and proceed for the payment through QR code	3	Medium	Jawahar Prasad, Jaffer Sathick HM
Sprint - 4	Feedback	USN - 6	User can also give feedback about the journey and overall experience	1	Low	Jaffer Sathick HM, Biplab Das S

7.1. Sprint Delivery Schedule



















Sprint	Total Story Points	Duration	Sprint StartDate	Sprint EndDate (Planned)	Story Points Completed (as on planned End Date)	Sprint Release Date (actual)
Sprint – 1	20	5 days	1 Nov 2022	5 Nov 2022	20	1 Nov 2022
Sprint – 2	20	2 days	5 Nov 2022	7 Nov 2022	20	5 Nov 2022
Sprint – 3	20	6 days	7 Nov 2022	12 Nov 2022	20	7 Nov 2022
Sprint – 4	20	6 days	14 Nov 2022	19 Nov 2022	20	14 Nov 2022

7.2. Reports from JIRA:

The total JIRA sprints included in a proposed project:

Sprints		
>	 IBM-5 Superfast Sprint 1	DONE
>	 IBM-6 Sprint 2- Creating databases	DONE
>	 IBM-42 Sprint-3	DONE
>	 IBM-43 Sprint-4	DONE

The sprints includes:

Sprints		
▼	 IBM-5 Superfast Sprint 1	DONE
	 IBM-7 As a user, I can register through phone and get otp code	DONE 
	 IBM-8 As a user, I can verify the otp code	DONE 
	 IBM-9 As a user, I can login with ID and Password	DONE 
	 IBM-10 As a user, I can enter the starting location and destina...	DONE 
▼	 IBM-6 Sprint 2- Creating databases	DONE
	 IBM-15 As a user, I can provide basic details as name, date of ...	DONE 
	 IBM-16 As a user, I can select the class and coach	DONE 
	 IBM-17 As a user, I can choose to pay through credit card, De...	DONE 
	 IBM-34 As a user, I will be redirected to select seats	DONE 

▼ ⚡ IBM-42 Sprint-3	DONE
📌 IBM-19 As a user, I can download the generated ticket	DONE J
📌 IBM-20 As a user, I can see the selected seats	DONE
📌 IBM-21 As a user, I get reminder notification of the booked tic...	DONE HK
📌 IBM-22 As a user, I can track the live location of the train	DONE JH
▼ ⚡ IBM-43 Sprint-4	DONE
📌 IBM-23 As a user, I can raise queries	DONE JH
📌 IBM-24 As a user, I can get answer to the queries	DONE
📌 IBM-25 As a user, I can feed information through the	DONE HK
📌 IBM-26 As a user, I can cancel the ticket	DONE J

8 CODING & SOLUTIONING :

traingps.py:

```
import wiotp.sdk.device
import time
import random
myConfig = {
    "identity": {
        "orgId": "iqhx38",
        "typeId": "iot",
        "deviceId": "iot"
    },
    "auth": {
        "token": "714019106015"
    }
}

def myCommandCallback (cmd):
    print ("Message received from IBM IoT Platform: %s" % cmd.data['command'])
    m=cmd.data['command']

client = wiotp.sdk.device.DeviceClient(config=myConfig, logHandlers=None)
client.connect()

def pub (data):
    client.publishEvent(eventId="status", msgFormat="json", data=myData,onPublish=None)
    print ("Published data Successfully: %s", myData)
while True:
    myData={'name': 'Ernakulam Exp', 'lat': 10.9973, 'lon': 76.9664}
    pub (myData)
    print("Coimbatore station")
    time.sleep(4)
    myData={'name': 'Ernakulam Exp', 'lat': 13.0827, 'lon': 80.2755}
    pub(myData)
    print("Chennai MGR station")
    time.sleep(4)
    myData={'name': 'Ernakulam Exp', 'lat': 16.5180, 'lon': 80.6195}
    pub (myData)
    print("Vijayawada station")
    time.sleep(5)
    myData={'name': 'Ernakulam Exp', 'lat': 20.266762, 'lon': 85.843425}
    pub (myData)
    print("Bhubaneswar station")
    time.sleep(3)
    myData={'name': 'Ernakulam Exp', 'lat': 22.5839, 'lon': 88.3434}
    pub (myData)
    print("Howrah station")
    time.sleep(5)
    client.commandCallback = myCommandCallback
```

```
client.disconnect ()
```

Scanningqr.py

```
import wiotp.sdk.device
```

```
import time
```

```
import random
```

```
myConfig = {  
    "identity": {  
        "orgId": "iqhx38",  
        "typeId": "iot",  
        "deviceId": "iot"  
    },  
    "auth": {  
        "token": "714019106015"  
    }  
}
```

```
def myCommandCallback (cmd):  
    print ("Message received from IBM IoT Platform: %s" % cmd.data['command'])  
    m=cmd.data['command']
```

```
client = wiotp.sdk.device.DeviceClient(config=myConfig, logHandlers=None)  
client.connect()
```

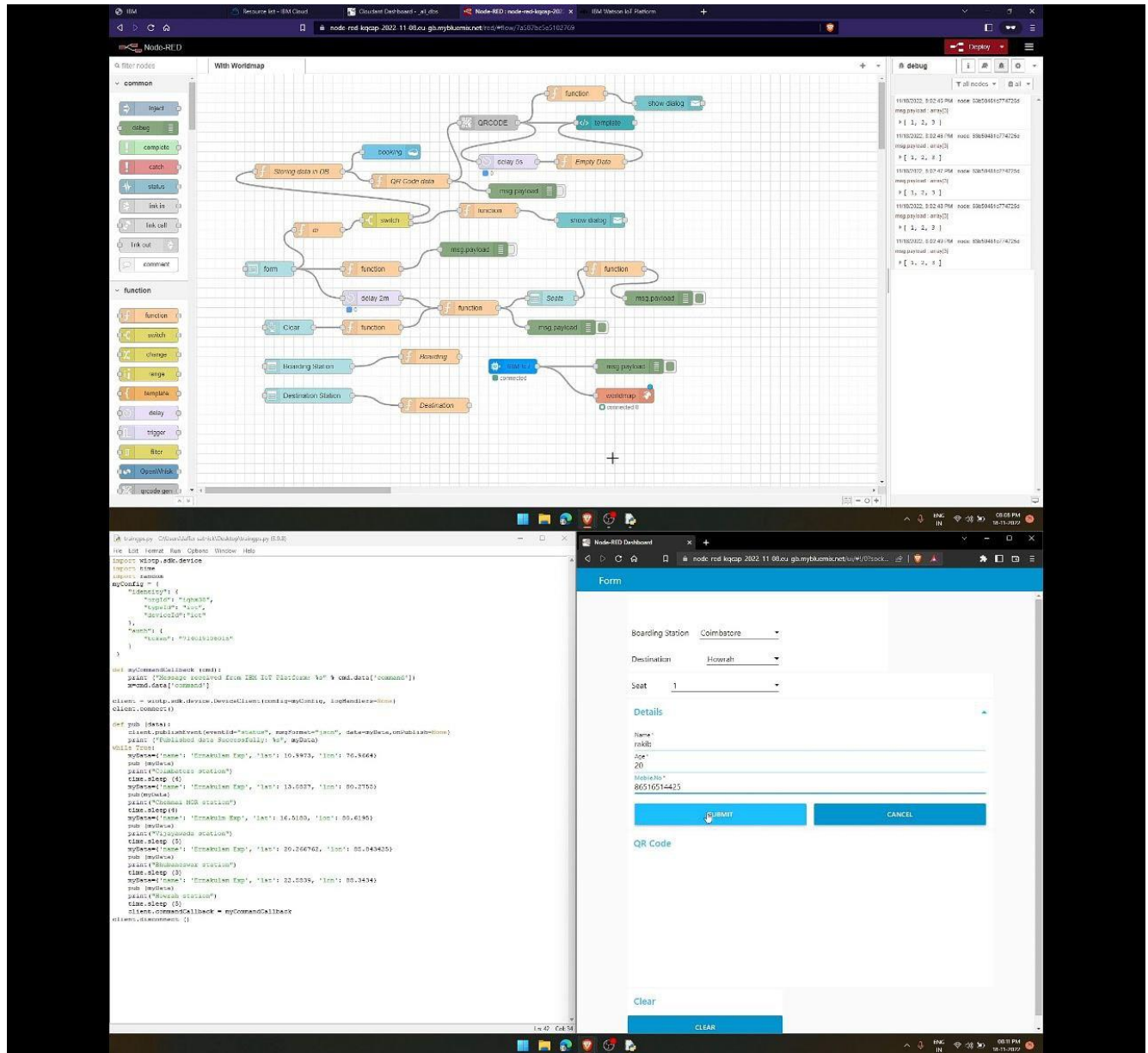
```
def pub (data):  
    client.publishEvent(eventId="status", msgFormat="json", data=myData,onPublish=None)  
    print ("Published data Successfully: %s", myData)  
while True:  
    myData={'name': 'Ernakulam Exp', 'lat': 10.9973, 'lon': 76.9664}  
    pub (myData)  
    print("Coimbatore station")  
    time.sleep (4)  
    myData={'name': 'Ernakulam Exp', 'lat': 13.0827, 'lon': 80.2755}  
    pub(myData)
```



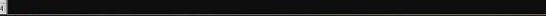
```
print("Chennai MGR station")
time.sleep(4)
myData={'name': 'Ernakulm Exp', 'lat': 16.5180, 'lon': 80.6195}
pub (myData)
print("Vijayawada station")
time.sleep (5)
myData={'name': 'Ernakulam Exp', 'lat': 20.266762, 'lon': 85.843425}
pub (myData)
print("Bhubaneswar station")
time.sleep (3)
myData={'name': 'Ernakulam Exp', 'lat': 22.5839, 'lon': 88.3434}
pub (myData)
print("Howrah station")
time.sleep (5)
client.commandCallback = myCommandCallback
client.disconnect ()
```

8.2 Test Cases

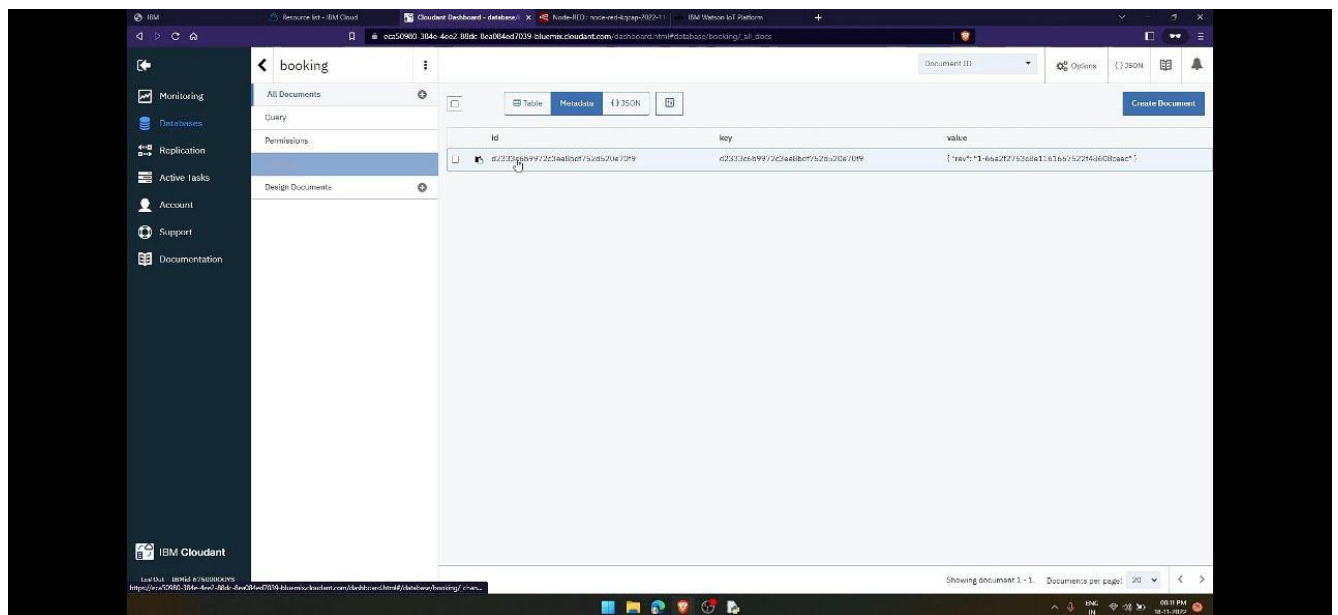
The node red is deployed giving the IBM Watson API and cloudant details



After submitting it, the UI will show a dialog box showing the ticket is generated with a QR code below.

Line 42 Col 5

The `scanningqr.py` is executed after the submission to scan the ticket and it retrieves the data from the IBM Cloudant.

[illegible]

9.1 Performance Metrics

[illegible]

10 CONCLUSION

Using the Web application, a user books a ticket based on the availability of the seats by giving the general required information. Once a user clicks on the submit button, a QR code is generated with a Unique ID and the data is stored in the Cloudant DB with that Unique ID. Users can save the QR code for further process. In python code, a Ticket collector can scan the QR code and extract the information from the QR Code i.e., Unique ID. With that Unique ID, data is fetched from the Cloudant DB, if it is not found, then it displays Not a Valid Ticket. Also, the live location of the train will be published to IBM IoT platform using python code . The train location can be tracked from a Web Application.

11 FUTURE SCOPE

Improving and increasing customer experience ,vehicle tracking system.IOT is used along with AI which Provides enhanced features in finding out delays. Predicting delay and detecting the train arrival time so that help the passenger to act accordingly and keep tracking the location of the train and travel in easy and modern way. To maintain the tracks, repairs and services to avoid accident, safeguard of things, track the running status of the train in smart railway system and reach the destination place on time due to train delay.

12 APPENDIX

Source Code:

<https://github.com/IBM-EPBL/IBM-Project-29993-1660137434/tree/main/Final%20Deliverables/Demo%20video%20&%20Final%20code>

GitHub & Project Demo Link:

<https://github.com/IBM-EPBL/IBM-Project-29993-1660137434>

<https://drive.google.com/drive/folders/1usjp71-5TX1pPFZWxkS4JQcnVVi1GqSo>