SMART SOLUTIONS FOR RAILWAYS

A PROJECT REPORT

Submitted by

SHAMPRASATH A S

SEENUVASAN M

TAMILSELVAN M

THARUNKUMAR M

ABSTRACT

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.

TABLE OF CONTENTS

CHAPTER	TITLE	NO.
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 Feature 1

7.2 Feature 2

7.3 Database Schema (if Applicable)

8. TESTING

8.1 Test Cases

9. RESULTS

9.1 Performance Metrics

10. ADVANTAGES & DISADVANTAGES

11. CONCLUSION

12. FUTURE SCOPE

13. APPENDIX

Source Code

GitHub & Project Demo Link

CHAPTER 1

INTRODUCTION

1.1 OVERVIEW

This project aims at development of a Railway Reservation System that facilitates the Railway customers to manage their reservations and the Railway administrators to modify the backend databases in a UserFriendly manner. This project includes the following functions: 1) Create new database 2) Add new Record 3) Modify 4) Display record 5) Ticket reservation 6) Ticket Modification 7) Ticket Cancellation 8) Ticket printing 9)Train tracking .

Our project introduces railway reservation system to make the reservation system more efficient, easier and fast. This project explores how computer technology can be used to solve the problem of user. The main objectives provided by this software are as follows: We can enquire about availability of trains We can reserve and cancel their seats We can modify the information related to Trains 1 , Timetable , Train Name , Train Number , Ticket Fare. This project is dedicated to model existing railway reservation systems that aim at development of Railway Reservation System that facilitates the railway customer to manage

their reservations and the railway administrator to modify the backend database in a user-friendly manner.
In this emerging world of computers, almost all-manual system has switched to automated and computerized system. Therefore, we are developing the software for “Railway Reservation System” to model the present system and to remove the drawbacks of the present system. This project explores how computer technology can be used to solve the problem of user. This being a big step in terms of improvement in the railway system it is widely accepted across the country. Rather than designing manually, we have made use of computer.

1.2 Purpose

The purpose of this software is to describe the Railway Reservation System which provides the rail timing details, reservation, enquiry, billing and cancellation on various types of reservation namely:

1.Confirm reservation for confirm seat

2.Online Payments

3.Train tracking

4.Reservation against cancellation

This project is dedicated to model the existing railway reservation system that aims at development of Railway Reservation System that facilitates the railway customer to manage their reservations and the railway administrator to modify the backend database in a user-friendly manner. The customer and the railway administrator are two parties that interact with the database, who have different ‘view level schemas’ to the database information. The software provides a comprehensive set of features to enhance the operational limits. Now one can easily plan the journey comfortably as the process is efficient and fast with being easy to access. The efficiency of the railway will increase result of computerization.

CHAPTER 2

LITERATURE SURVEY

2.1 Existing problem

The biggest obstacle is poor connectivity in the net. When that is the case, it is advisable to stop trying at all. Because even if somehow you almost book the desired ticket, the payment part can frustrate the whole activity. Then your ticket as well as money is stuck. There are innumerable problems with Tatkal in busy season on busy routes. There is no way to beat the agents because they use every available legal and other way to get you the ticket. You should learn ways to save time, and you should try to plan in advance. That is the real alternative.Sometimes your money will get stuck even when everything is ok .Then the problem is with your bank's server. Here again there is little you can do. Now, it seems to be a challenge to develop an online booking system that offers multiple types of bookings on a single platform. It's always been challenging to manage online booking cancellations. If you provide the booking cancellation option to customers, you will frequently receive tons of booking cancellation requests from the customers that are too complex to manage. On the other hand, if you will not provide the cancel booking option to your customers, you will be flooded with bookings or probably start receiving spams. This may also result in a large number of no-shows if the customers find no option to cancel bookings. It's quite complex to manually set up a different price for each booking slot. In order to have a flexible price for each booking, your online booking system should be capable enough to show a different slot price of the same booking product on different dates. If you are in an eCommerce business, you have to deal with customer complaints. Your customer may have a bad experience with your service but the main challenge is to handle such situations with grace.

2.2 References

1.S.D.T. Kelly N.K. Suryadevara and S.C.Mukhopadhyay "Towards the implementation of IoT for environmental condition monitoring in homes"

In this paper, we have reported an effective implementation for Internet of Things used for monitoring regular domestic conditions by means of low cost ubiquitous sensing system. The description about the integrated network architecture and the interconnecting mechanisms for the reliable measurement of parameters by smart sensors and transmission of data via internet is being presented. The longitudinal learning system was able to provide a self-control mechanism for better operation of the devices in monitoring stage. The framework of the monitoring system is based on a combination of pervasive distributed sensing units, information system for data aggregation, and reasoning and context awareness. Results are encouraging as the reliability of sensing information transmission through the proposed integrated network architecture is 97%. The prototype was tested to generate real-time graphical information rather than a test bed scenario.

2. Y.S.Song J.Kim S.W. Choi and Y.K. Kim "Long term evolution for wireless communications: Testbed deployment and performance evolution

In this article, we show the feasibility of the LTE-R testbed with essentially IP-based network architecture. Specifically, we discuss procedures of deploying LTE-R by describing our construction of a testbed in a commercial railway through cell planning and optimization. Then we demonstrate the performance enabled by the implementation of a testbed for LTE-R. We confirm that not only reliable communications but also multimedia services requiring high data rates are feasible, which gives us some guarantee of the prosperity of various advanced train services. We also discuss a number of valuable technical communication issues related to inherent characteristics of railway communications that are unlike those of commercial wireless communications.

3. J. Kim S.W. Choi Y.S. Song Y.K Yoon and Y.K Kim "Automatic train control LTE: Design and performance evolution"

Due to technical advances in train control and wireless communications, unmanned train operation has gained in popularity of late. On the other hand, any errors involved in managing the QoS of train control traffic will cause negative consequences such as possible loss of life. Operators therefore naturally wish to scrutinize the specifications so that the wireless communications system is capable of guaranteeing the QoS of the train control traffic. In this article, we propose a feasible QoS management scheme for train control traffic based on the methodology used in a conventional LTE system. Based on the proposed scheme, we evaluate the feasibility of the LTE system using a testbed built in a commercial railway region. The key issues to support the train control services by the LTE system are the design of a QoS policy based on analyzing the characteristics of the train control traffic and the appropriate adjustment of the cell parameters during the cell planning and optimization procedures in order to resolve any network issues that may cause problems with data pause.

4. B. Martinez M. Monton I. Vilajosana and J.D. Prades "The power of models: Modeling power consumption for IoT devices"

Low-energy technologies in the Internet of Things (IoTs) era are still unable to provide the reliability needed by the industrial world, particularly in terms of the wireless operation that pervasive deployments demand. While the industrial wireless performance has achieved an acceptable degree in communications, it is no easy task to determine an efficient energy-dimensioning of the device in order to meet the application requirements. This is especially true in the face of the uncertainty inherent in energy harvesting. Thus, it is of utmost importance to model and dimension the energy consumption of the IoT applications at the pre-deployment or pre-production stages, especially when considering critical factors, such as reduced cost, life-time, and available energy. This paper presents a comprehensive model for the power consumption of wireless sensor nodes. The model takes a system-level perspective to account for all energy expenditures: communications, acquisition and processing. Furthermore, it is based only on parameters that can empirically be quantified once the platform (i.e., technology) and the application (i.e., operating conditions) are defined. This results in a new framework for studying and analyzing the energy life-cycles in applications, and it is suitable for determining in advance the specific weight of application parameters, as well as for understanding the tolerance margins and tradeoffs in the system.

5. J. Kim J. Lee J. Kim and J. Yun "M2M service platforms: Survey issues and enabling technologies"

Machine-to-Machine (M2M) refers to technologies with various applications. In order to provide the vision and goals of M2M, an M2M ecosystem with a service platform must be established by the key players in industrial domains so as to substantially reduce development costs and improve time to market of M2M devices and services. The service platform must be supported by M2M enabling technologies and standardization. In this paper, we present a survey of existing M2M service platforms and explore the various research issues and challenges involved in enabling an M2M service platform. We first classify M2M nodes according to their characteristics and required functions, and we then highlight the features of M2M traffic. With these in mind, we discuss the necessity of M2M platforms. By comparing and analyzing the existing approaches and solutions of M2M platforms, we identify the requirements and functionalities of the ideal M2M service platform. Based on these, we propose an M2M service platform (M2SP) architecture and its functionalities, and present the M2M ecosystem with this platform. Different application scenarios are given to illustrate the interaction between the components of the proposed platform. In addition, we discuss the issues and challenges of enabling technologies and standardization activities, and outline future research directions for the M2M network.

6. A. Gluhak etal , "A survey on facilities for experimental Internet of Things research"

The initial vision of the Internet of Things was of a world in which all physical objects are tagged and uniquely identified by RFID transponders. However, the concept has grown into multiple dimensions, encompassing sensor networks able to provide realworld intelligence and goal-oriented collaboration of distributed smart objects via local networks or global interconnections such as the Internet. Despite significant technological advances, difficulties associated with the evaluation of IoT solutions under realistic conditions in real-world experimental deployments still hamper their maturation and significant rollout. In this article we identify requirements for the next generation of IoT experimental facilities. While providing a taxonomy, we also survey currently available research testbeds, identify existing gaps, and suggest new directions based on experience from recent efforts in this field.

2.3 Problem Statement Definition

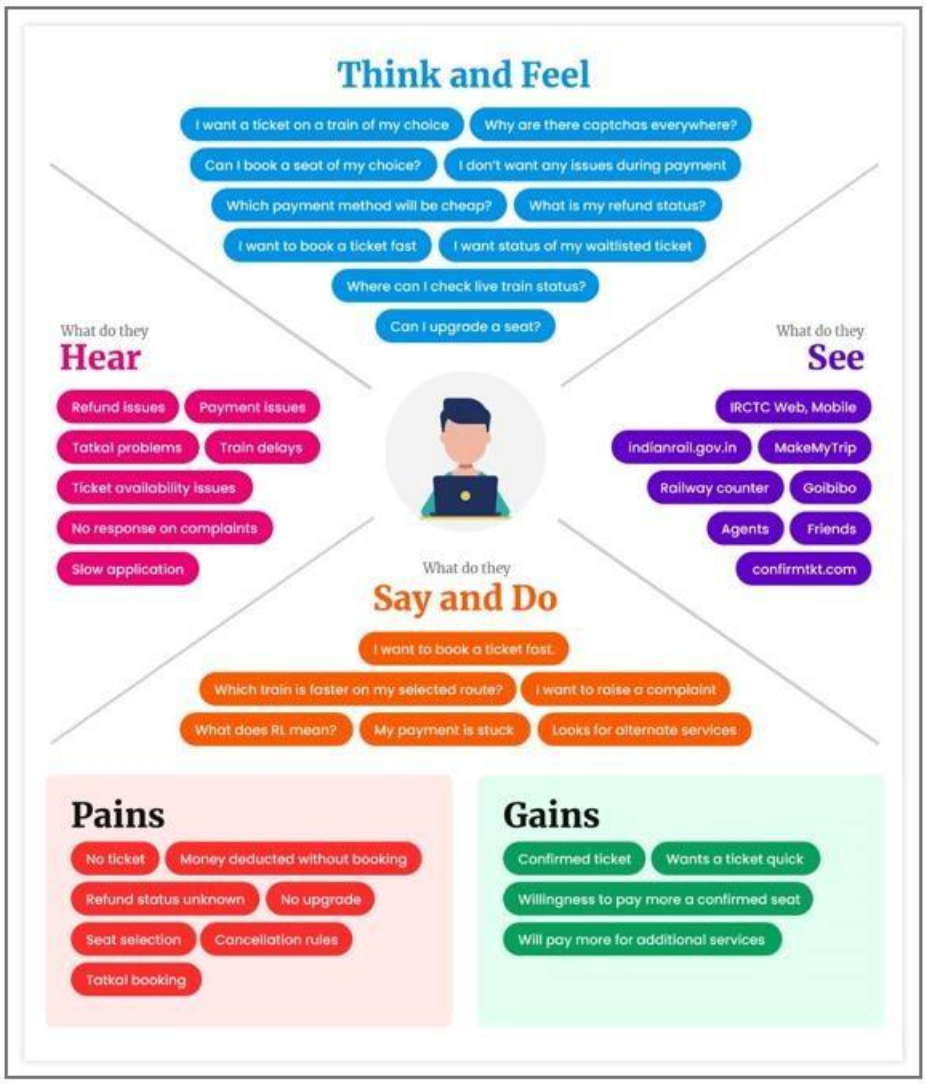
Online railway reservation is an efficient way to reserve tickets not by standing in the railway station queue .Now all railways has their own website for online reservation provide better customer service. The manual filling of reservation form cannot be changed once the details had been entered. The goal of online railway reservation is easing the tedious task of railway activity. Initially the customer has to create an ID in the appropriate website, so that the user can log into the system for doing further activities. An online manager will maintain a database. To do login process the customer has to fill a registration form that contains the username, password, first name and last name etc. After submitting the form to the server a customer ID is created with username and password thereby the customer with only the appropriate ID can reserve the tickets.

RESERVATION The customer can reserve ticket by filling the reservation form present in the website. The reservation form present contain passenger name, sex, age, address, credit no, bank name, class through which the passenger is to travel etc. The online manager will verify the detail and provide PNR number to the customers who reserve the ticket. CANCELLATION The user can also perform cancellation of the ticket which he she had reserved earlier by entering PNR no. This PNR no will be checked with the PNR no in the database. If it exists then it will be cancelled. After cancellation process the conformation message will be send by the server. TICKET STATUS The customer can also view the ticket status by entering the PNR number on the ticket status icon. To view the train details the user should click the train details icon in the homepage. TRAIN DETAILS It provide information about arrival and departure trains along with information about stations through which it passes search about train passing through stations can be obtained either by means of train no, train name or specifying the source and destination station. While displaying information about train it has to provide following information. Stations through which train passes along with arrival and departure time. Availability of seats in different classes along with waiting list. Before issuing ticket the amount from customer has to be transferred to the railway account. Thus this has simplified the task of reservation, cancellation of tickets in railways.

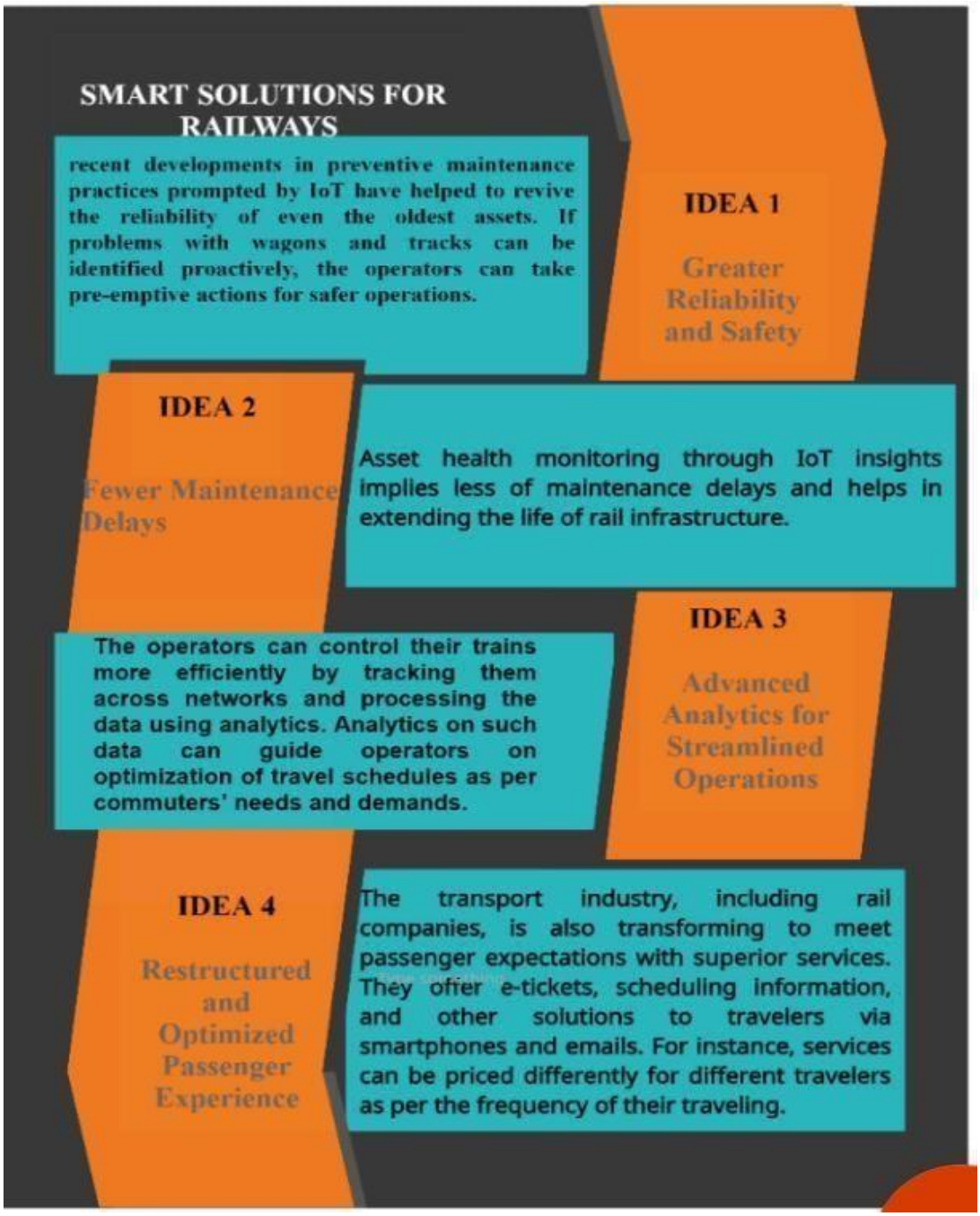
CHAPTER 3

IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas



3.2 Ideation & Brainstorming

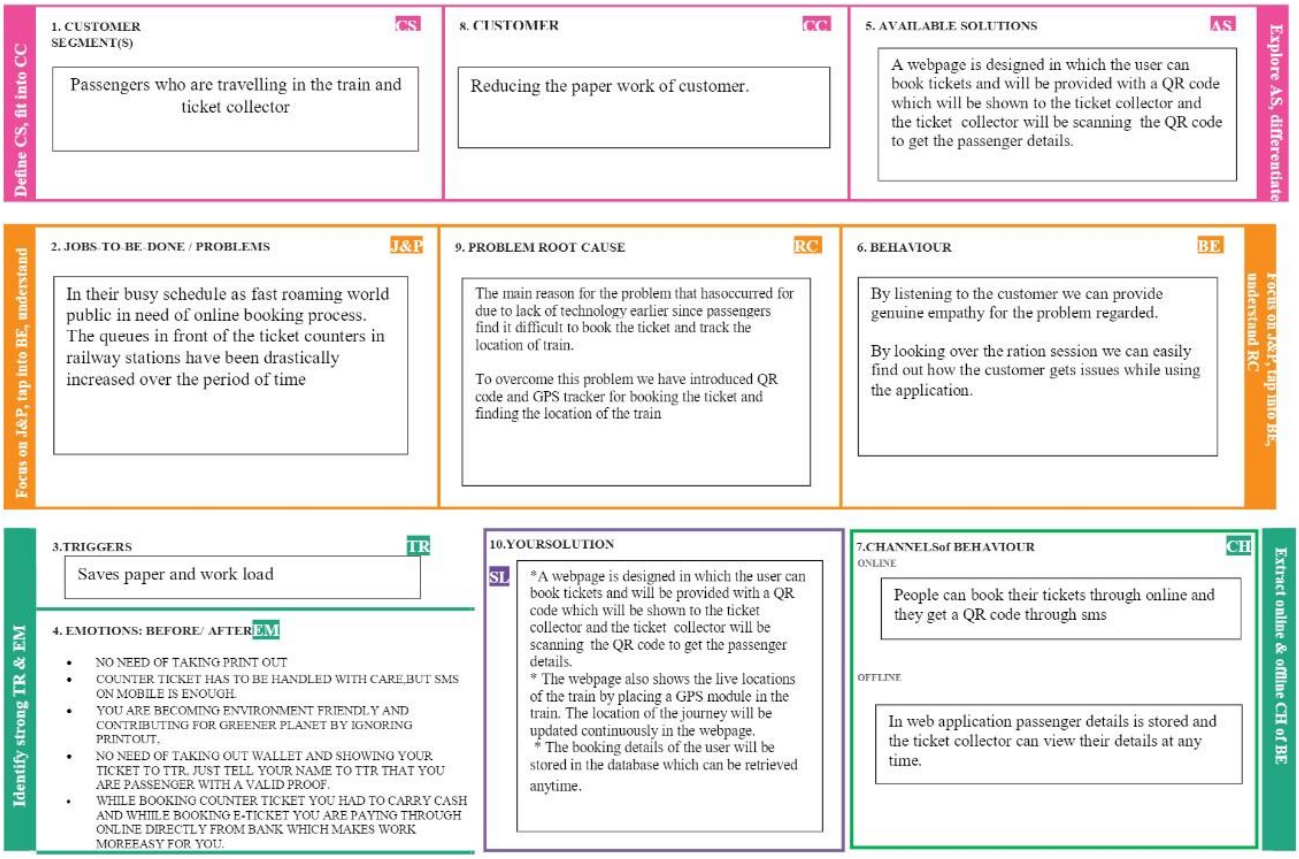


3.3 Proposed Solution

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Smart Solutions For Railways
2.	Idea / Solution description	Smart railway described the passenger ticket generation, ticket validation, with Unique Identification Authority of India (UIDAI) under the smart train transportation the vision of India 2022 and the experimental result proved that IoT system is effective than well known System. Handling the passenger reservation data has been a key point of consideration in most railway services. The smart railways research report also provides an in-depth analysis of proposed.
3.	Novelty / Uniqueness	• Reduce operation and maintenance costs. • It is fast process. • Data efficiency is more. • Information is accurate.
4.	Social Impact / Customer Satisfaction	Smart railway can have a better understanding of customers' needs. In terms of comfort, availability of seats, comfortable in the units of carriage, temperature, smoothness of ride, and punctuality are the aspects in the dimensions, whereas connection as the second dimension measures adequacy of support services, such as parking facilities, easy accessibility, frequency of trains, and suitable time to board on train, while the last dimension, which is convenience, measures travel information, online counter service, such as ease of buying tickets and convenient office hours at the terminal.
5.	Business Model (Revenue Model)	• Global System Mobile Communications- Railway (GSM-R) • Long Term Evaluation (LTE) • fifth generation (5G)

		• IEEE 802.11 • Wireless Sensor Networks (WSN)
6.	Scalability of the Solution	The Smart railways database maintained by two data bases one is railway data base and another one is UIDAI. The passenger details as input that will match with the UIDAI data base and that information stored in railway reservation database. The Simulation environment is tested with one hundred stored UIDAI passengers information and developed coding smart railway system. We obtained the experimental results on PYTHON programming language and proved the effective result on IoT system. The existing PRS model is a omniscient system it can be modified to IoT system. It IoT system In build UIDAI based verification scheme. The total time is required to book the ticket using IoT system is not take more than 4 mile seconds. The smart reservation system is provides efficient searching and indexing operations are needed for fast query processing.

3.4 Problem Solution Fit



CHAPTER 4

REQUIREMENT ANALYSIS

4.1 Functional Requirement

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Form Registration through Gmail Registration through App
FR-2	Login	Login the app using username and password Search the train details
FR-3	Booking Ticket	Choose the train based on their own journey Give the personal details Book the ticket via online
FR-4	QR Generation	After booking the ticket it will generate the qr code It contains all the information
FR-6	Ticket Scanner	It is used to scan the ticket it will retrieve the data from the database

4.2 Non-functional Requirements:

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	It can be assessed from efficiency of use , intuitive , low perceived work .
NFR-2	**Security**	Security requirements ensure that the software is protected from unauthorized access to the system and its stored data.
NFR-3	**Reliability**	Reliability defines how likely it is for the software to work without failure for a given period of time.
NFR-4	**Performance**	Performance is a quality attribute that describes the responsiveness of the system to various user interactions with it.
NFR-5	**Availability**	Availability is gauged by the period of time that the system's functionality and services are available for use with all operations.
NFR-6	**Scalability**	serving more users, processing more data, and doing more transactions.

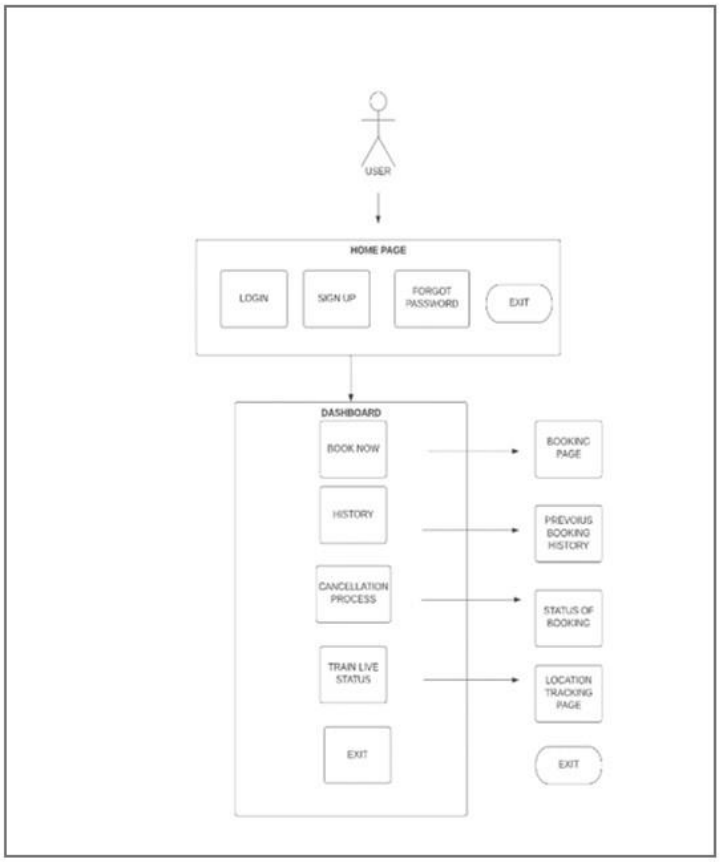
CHAPTER 5

PROJECT DESIGN

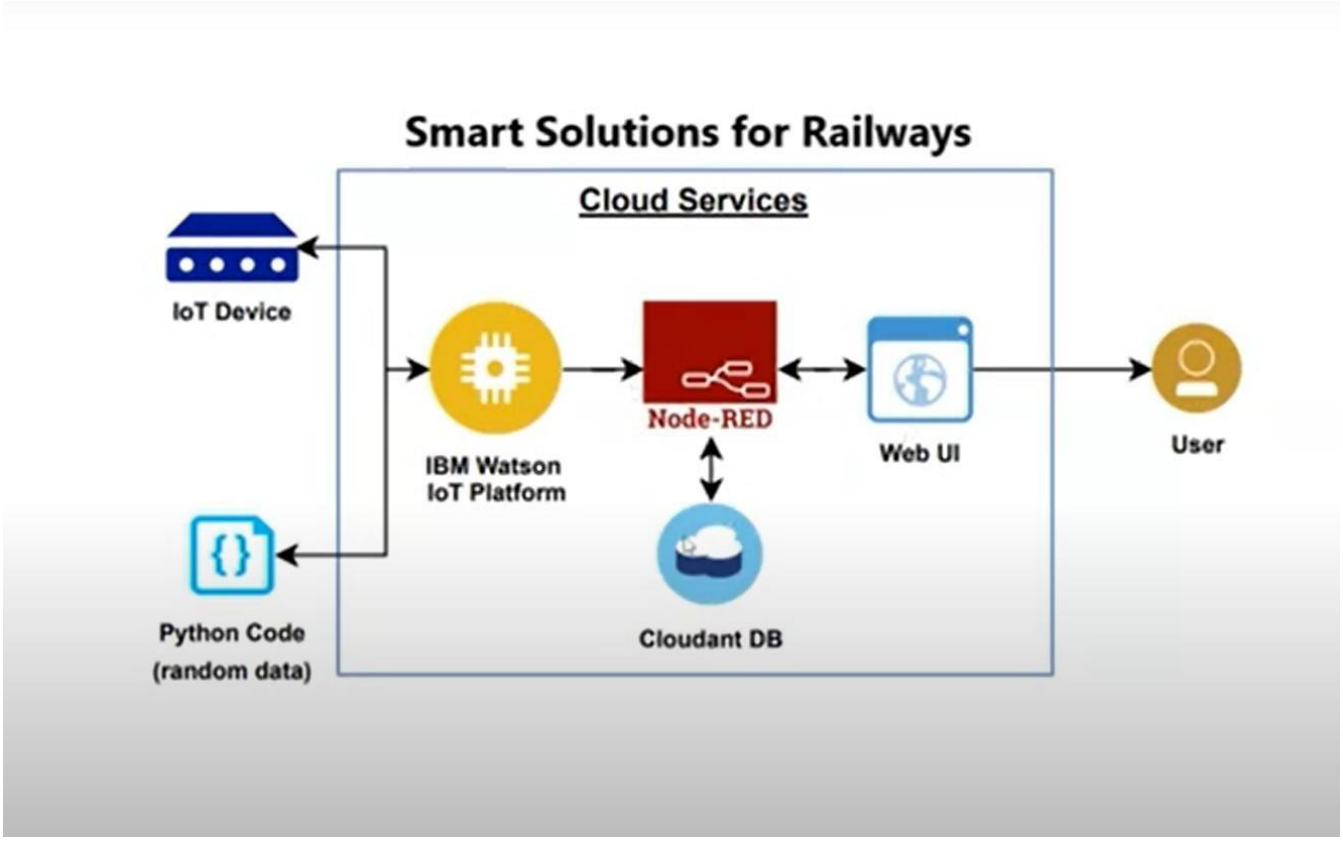
5.1 Data Flow Diagram

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.

Data flow diagram:



5.2 Solution and architecture :



5.3 User Stories

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release

Passenger (email validation)	Registration	USN-1	First of all we need to register. Here well need	It will generate registration successful message.	High	Sprint1
Passenger	Login	USN-1	As a user, I can log into the Website by entering email & password.	It will generate login successful message.	High	Sprint1
Passenger	Dashboard	USN-1	As a passenger,I can click the link to open the ticket booking page.	We book the tickets after successful login authentication.	Medium	Sprint2

Database Administrated	Track the location	USN-7	As an administrator ,used to track the current status and location of the time.		Medium	Sprint2

CHAPTER 6

PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority
Sprint-1	Registration	USN-1	As a user, I can register for the application by entering my email, password	2	High
Sprint-2	Booking	USN-3	As a user, I can book the ticket by entering the personal details.	2	Low
Sprint-2	Online Payment	USN-4	After booking, the user should pay the amount through various online payment method. And then the ticket will be booked.	2	Medium
Sprint-3	Tracking	USN-5	As a user, I can track the train by using my unique id through app.	1	High

6.2 Sprint Delivery Schedule

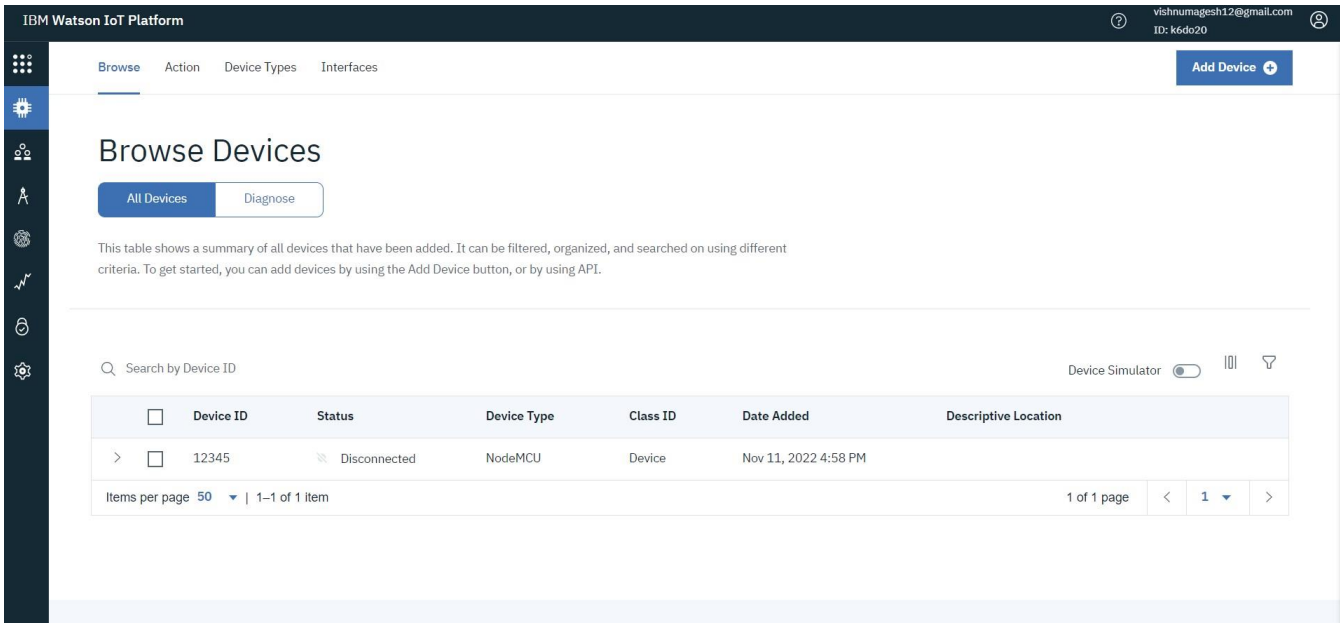
Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022		05 Oct 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022		12 Oct 2022

CHAPTER 7

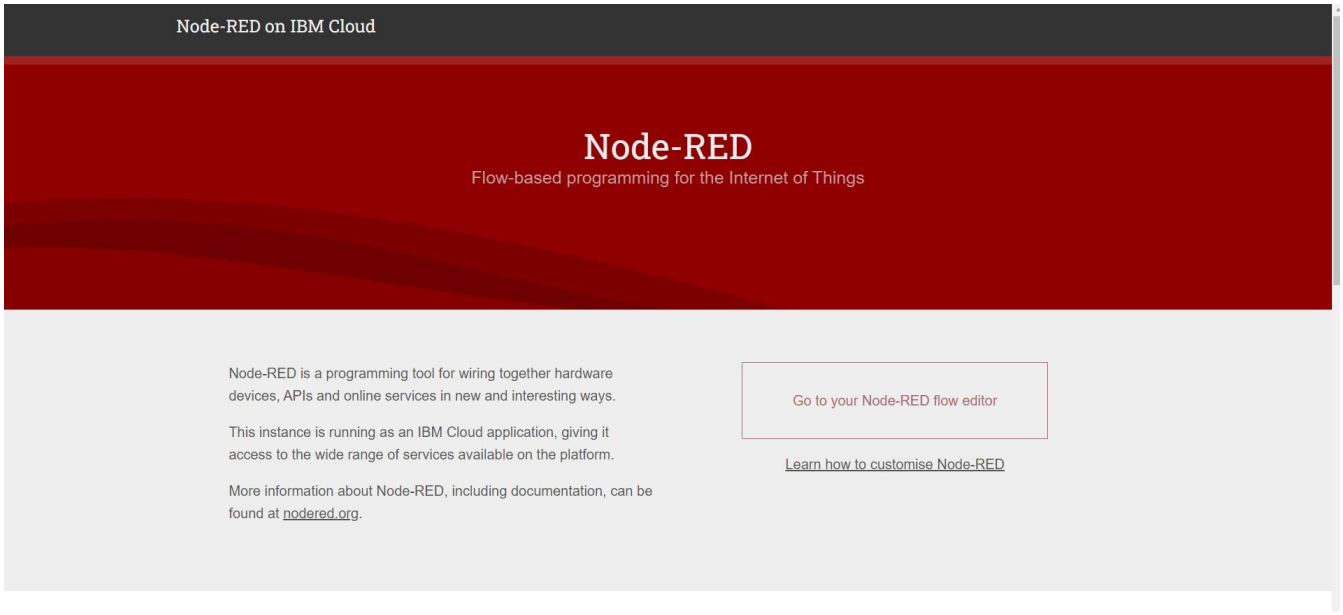
CODING & SOLUTIONING

7.1 Feature 1

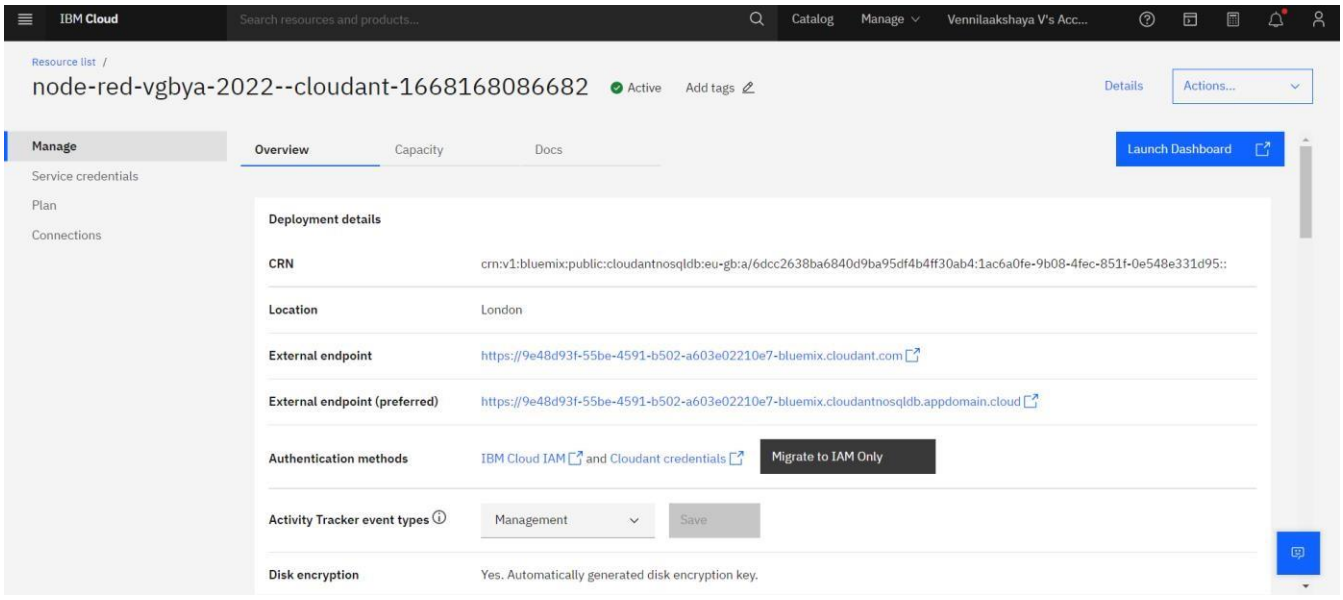
IBM Watson Platform :



Node Red :

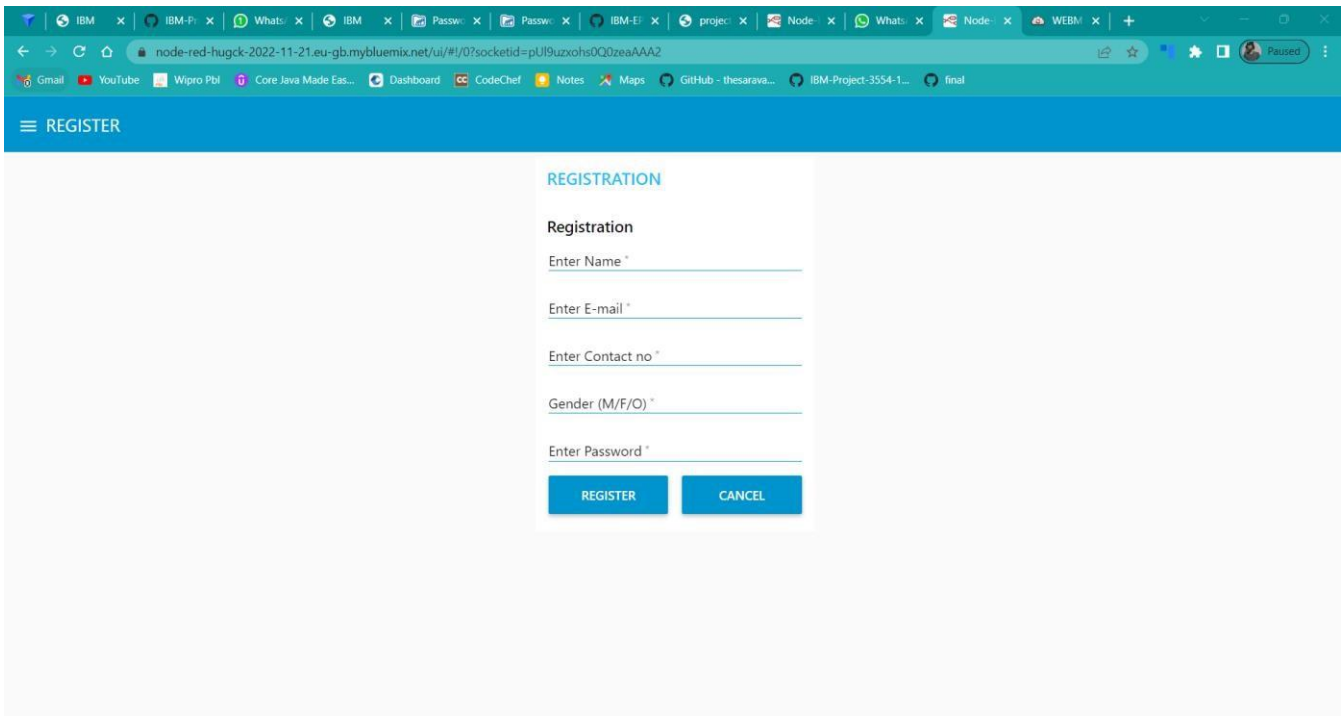


Cloudant DB :

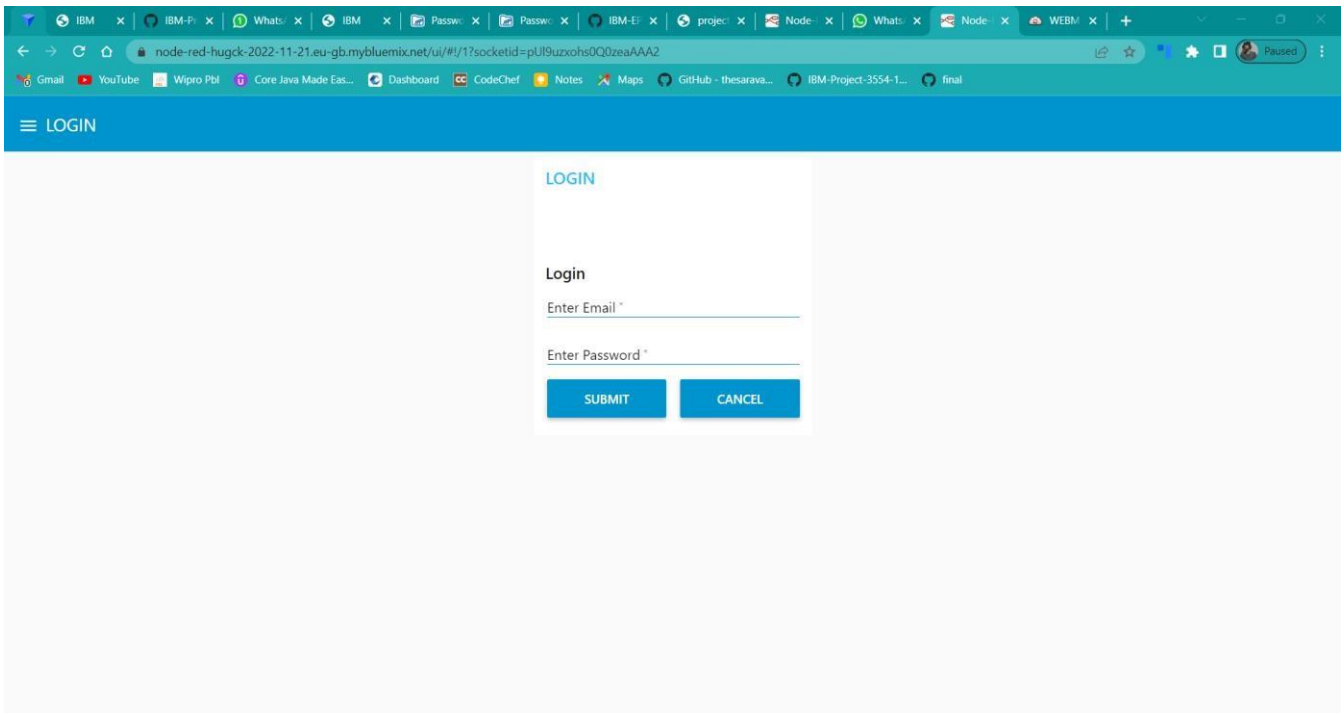


7.2 Features 2 :

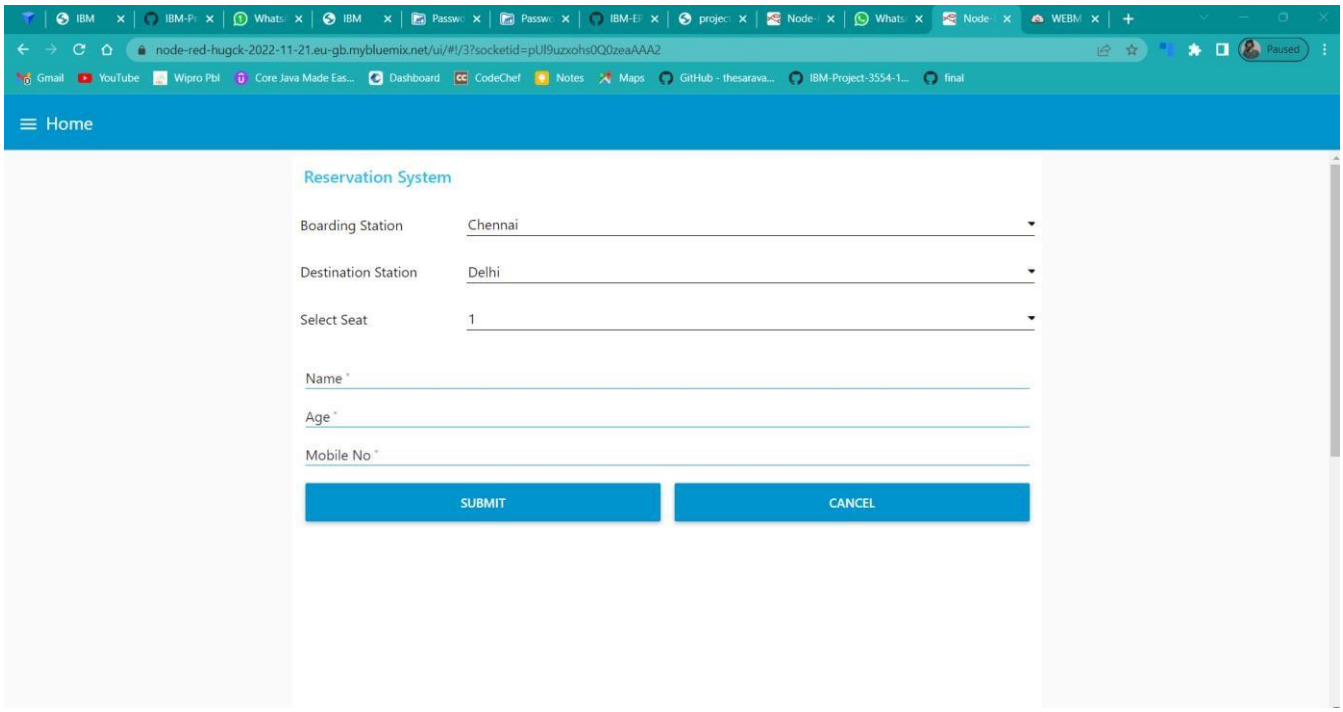
Registration Page:



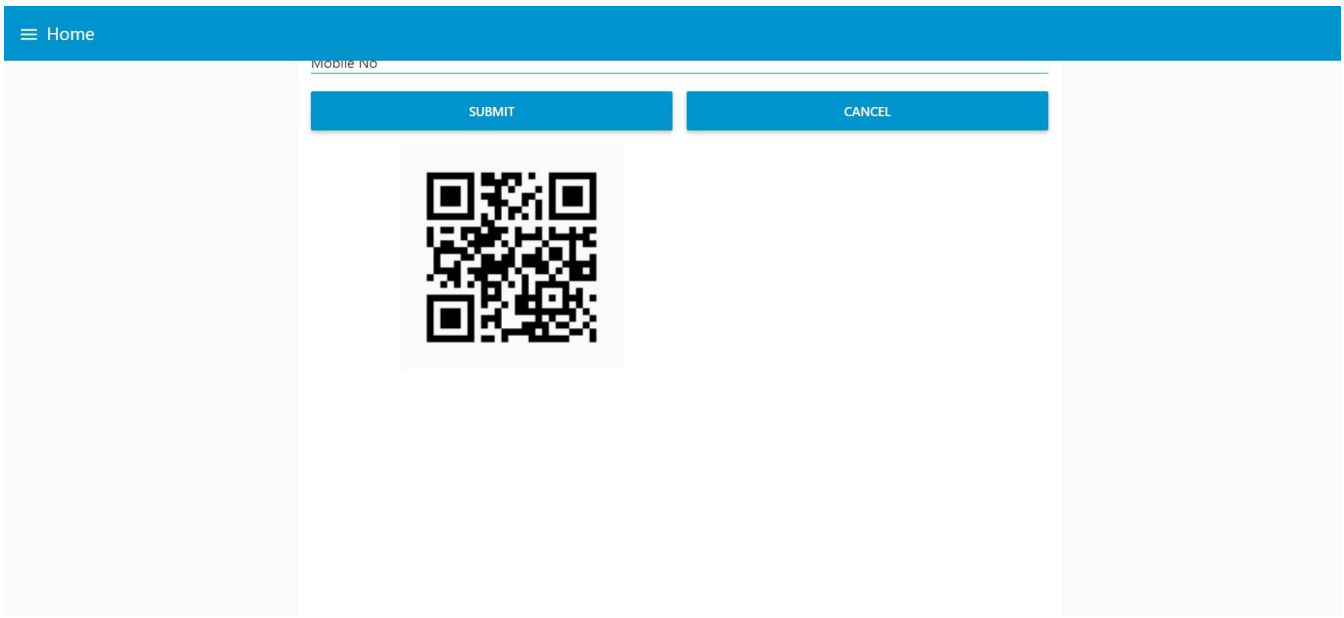
Login Page:



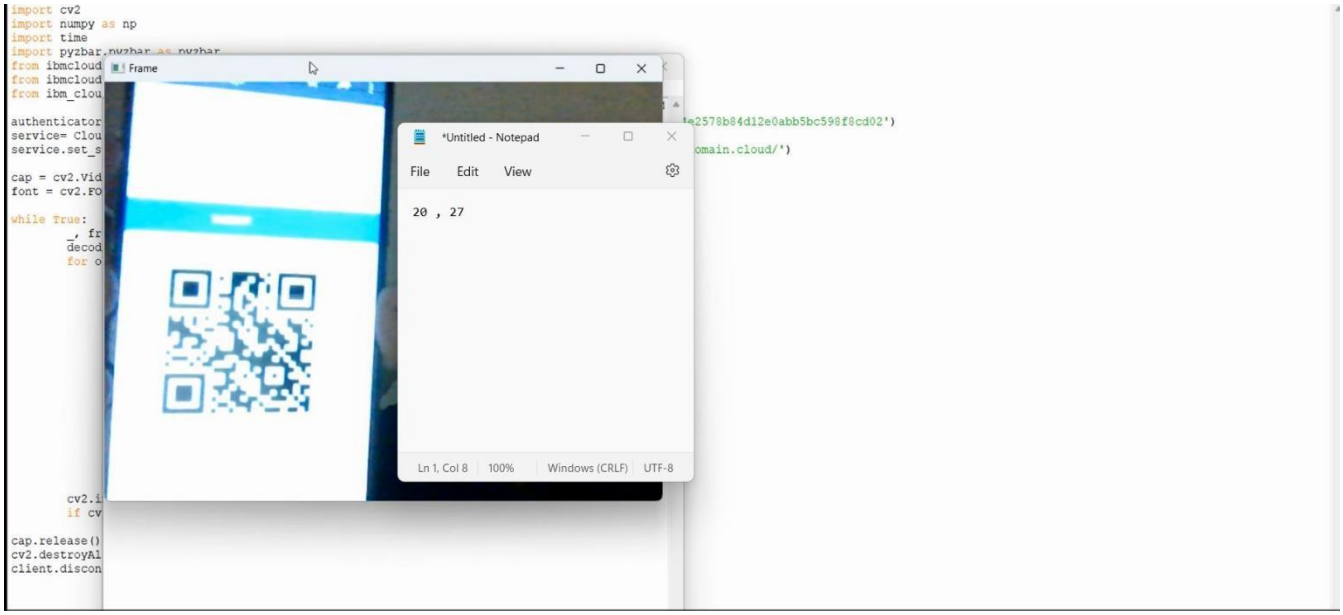
Reservation Page:



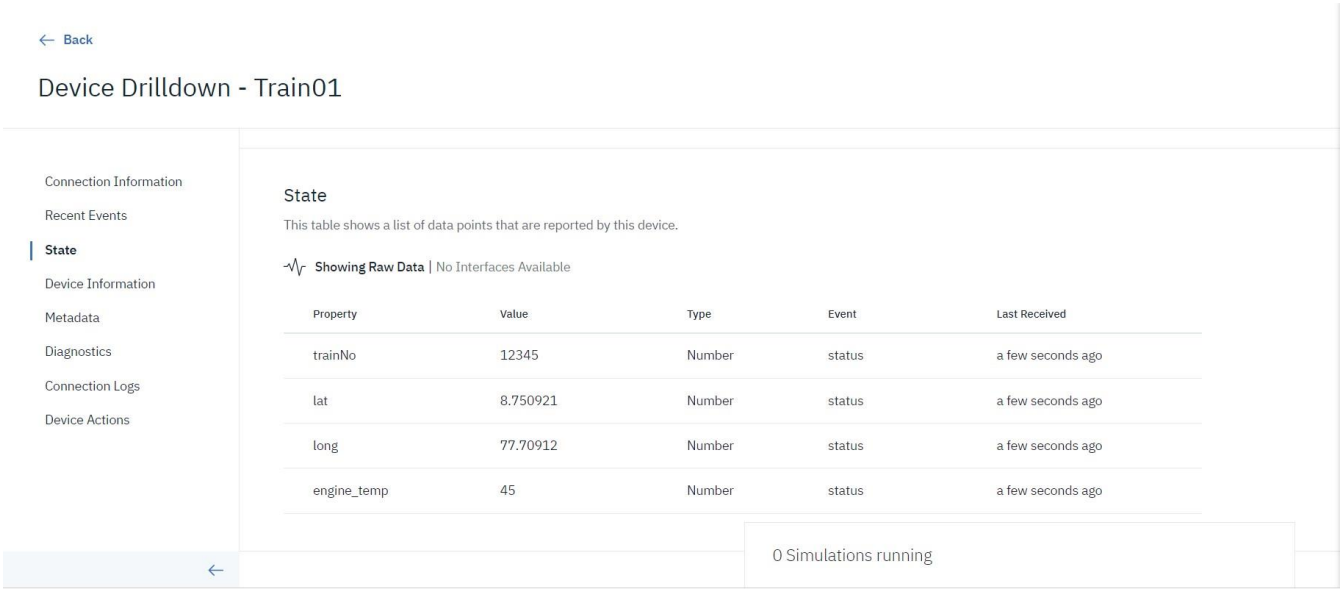
QR code generation:



QR code Scanner :



Live Train Status:



CHAPTER 8

TESTING

8.1 Test cases

Date	18-Nov-22
Team ID	PNT2022TMID18625
Project Name	Smart Solutions for Railways
Maximum Marks	4 marks

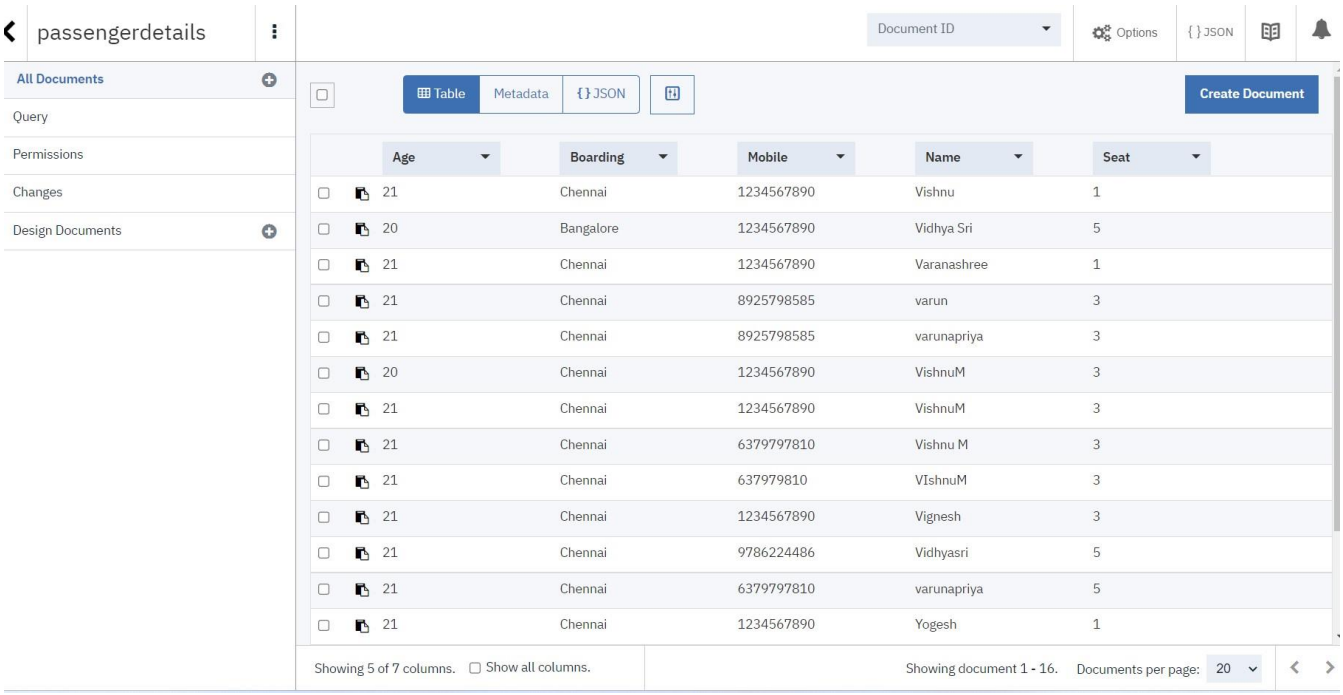
Test case ID	Feature Type	Component	Test Scenario	Pre-Requisite	Steps To Execute	Test Data	Expected Result	Actual Result	Status	Co
1	Functional	Home Page	Verify user is able to see the Login/Signup popup when user clicked on My account button		1.Enter URL and click go 2.Click on My Account dropdown button 3.Verify login/Singnup displayed or not	https://smart-railway-app-1610.mybluemix.net/ui	Login/Signup popup should display	TRUE		
2	UI	Home Page	Verify the UI elements in Login/Signup popup		1.Enter URL and click go 2.Fill the details 3.Click the Login/sign button	https://smart-railway-app-1610.mybluemix.net/ui	Login/Signup popup should display	TRUE		
3	Functional	Home page	Verify user is able to log into application with Valid credentials	user name and password	1.Enter URL(https://smart-railway-app-1610.mybluemix.net/ui) and click go 2.Enter Valid username/email in Email text box 3.Enter valid password in password text box 4.Click on login button	Username: vishnumagesh12@gmail.com	Application should show below UI elements: a.email text box b.password text box c.singin/signup button with blue colorcolour	TRUE		
4	Functional	Login page	Verify user is able to log into application with InValid credentials	user name and wrong password	1.Enter URL(https://smart-railway-app-1610.mybluemix.net/ui) and click go 2.Enter Valid username/email in Email text box 3.Enter Invalid password in password text box 4.Click on login button	username:vishnumagesh11@gmail.com	Application should show below UI elements: a.email text box b.password text box c.singin/signup button with blue colorcolour d.popup for enter wrong credentials	TRUE		

Test case ID	Feature Type	Component	Test Scenario	Pre-Requisite	Steps To Execute	Test Data	Expected Result	Actual Result	Status	Co
5	Functional	Home Page	Verify user is able to move from login page to Booking page		1.Enter URL(https://smart-railway-app-1610.mybluemix.net/ui) and click go 2.Enter Valid username/email in Email text box 3.Enter valid password in password text box 4.Click on login button 5.Scroll down the home page 6.Select the Booking Page		Application should show below UI elements: a.email text box b.password text box c.singin/signup button with blue colorcolour	TRUE		
6	Functional	Home Page			1.Enter URL(https://smart-railway-app-1610.mybluemix.net/ui) and click go 2.Enter Valid username/email in Email text box 3.Enter valid password in password text box 4.Click on login button 5.Scroll down the home page 6.Select the Tracking Page		Application should show below UI elements: 1.It will display map for tracking train	TRUE		
7	UI	Home Page	Verify user is able to track the location of train in every 3 minutes		2.Enter Valid username/email in Email text box 3.Enter valid password in password text box 4.Click on login button 5.Scroll down the home page 6.Select the Tracking Page					
8	Functional	Login page	Verify user is able to log into application with Valid credentials	user name and password	1.Enter URL(https://smart-railway-app-1610.mybluemix.net/ui) and click go 2.Enter Valid username/email in Email text box 3.Enter valid password in password text box 4.Click on login button	username: vijay@gmail.com sameple password: vijay17	Application should show below UI elements: a.email text box b.password text box c.singin/signup button with blue colorcolour	TRUE		
9	Functional	BookingPage	Verify user is able to enter the datails for booking train		1.Enter URL() and click go 2.Enter Valid username/email in Email text box 3.Enter valid password in password text box 4.Click on login button 5.scroll down to homepage 6.select Bookingpage in homepage 7.Enter the details for booking train 8.Click Book Button for Booking Ticket	sample train no : 12632 sample name: shane sample tickets: 1 sample date: 19/11/2022	Application should show below UI elements: a.Name text box b. Train no text box c.Book/cancel button with blue colorcolour	TRUE		
10	Functional	Booking Page	It will retrive the data form database using unique id	qrcode	scanning using our application	id:1234567814124 name:vishnu boaring:chennai seat:1	application will return the	TRUE		

CHAPTER 9

RESULTS

9.1 Performance Metrics



CHAPTER 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, andsubsequent sending of that data to the cloud, for further processing; ● Online data processing systems, for realtime monitoring, usingemerging
- communication technologies;
- Integrated, interoperable, and scalable solutions for railway systems preventive maintenance.

CHAPTER 11

CONCLUSION

Using this system every data is safe so they cannot modify the data because inside QR code there is only one unique id.

CHAPTER 12

FUTURE SCOPE :

If anyone wants to extend this project then he/she can make an additional database of Train Fare. And database for updated availability of seats which is available after the cancellation of ticket on that specific train etc. He/she can also add some more command buttons in the existing software and extend working of the existing software. Implementations of this project idea are in industrial use. Hence, this can be used for suggesting improvements in design, performance and greater usability. Apart from the industrial applications, it is a research-oriented project as well, the task of performance evaluation of different database designs, for efficiency, is in this spirit. He/she also add some online payments method into this project.

CHAPTER 13
APPENDIX

Source code ：

```
import wiotp.sdk.device import
time import random
myConfig={
   "identity": {
      "orgId": "kbzwt4",
      "typeId": "NodeMCU",
```

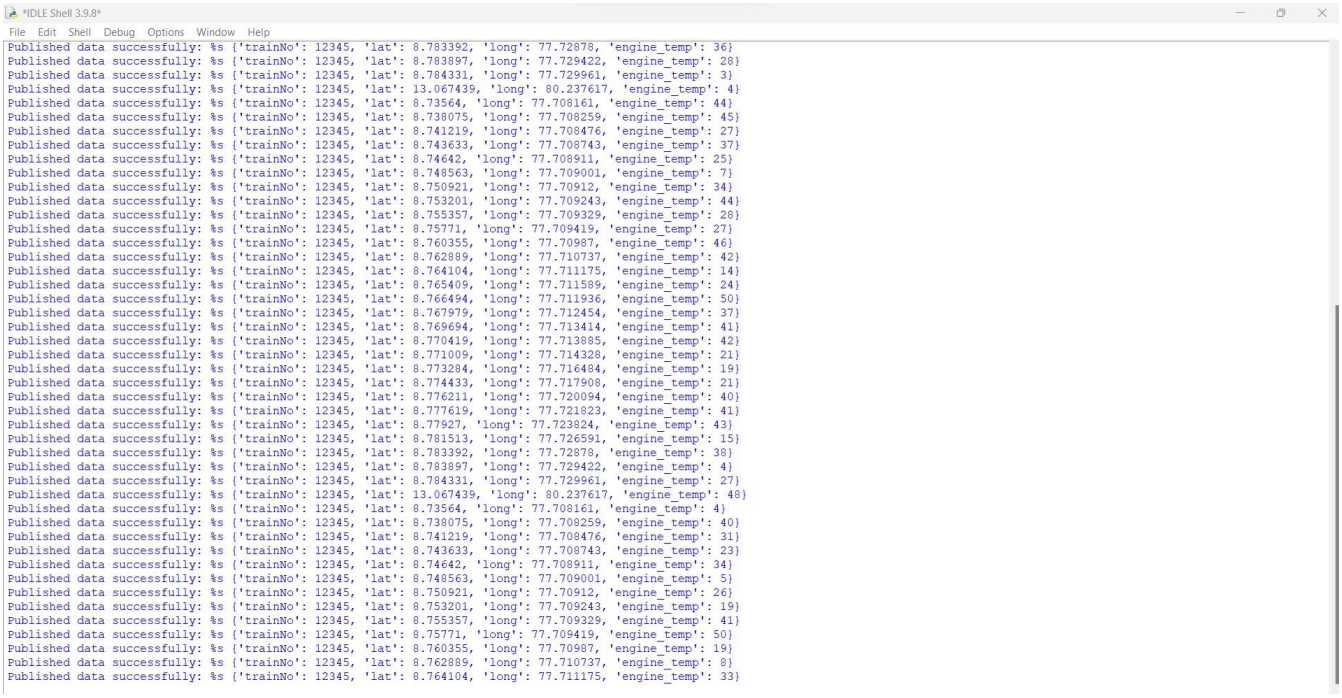
```
    "deviceId": "12345"
  },
  "auth": {
    "token": "12345678"
  }
}
def myCommandCallback(cmd):
  print("msg received from IBM IOT platform: %s" %cmd.data['command'])
m=cmd.data['command']  client=wiotp.sdk.device.DeviceClient(config=myConfig,logHandlers=None)
client.connect() def pub(data):

client.publishEvent(eventId="status",msgFormat="json",data=myData,qos=0,onPublish=
None)    print("Published data successfully:%s",myData) while True:
myData={'name':'train1','lat':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()
```

Train01.py - C:\Users\Vennila\Desktop\Project\Train01.py (3.9.8)

File Edit Format Run Options Window Help

```
import wiotp.sdk.device
import time
import random

myConfig = {
    "identity": {
        "orgId": "rc6ado",
        "typeId": "Train01",
        "deviceId": "Train01",
    },
    "auth": {
        "token": "12345678910"
    }

}

train_detial = {12345: [[13.067439,80.237617],
                        [8.735640, 77.708161],
                        [8.738075, 77.708259],
                        [8.741219, 77.708476],
                        [8.743633, 77.708743],
                        [8.746420, 77.708911],
                        [8.748563, 77.709001],
                        [8.750921, 77.709120],
                        [8.753201, 77.709243],
                        [8.755357, 77.709329],
                        [8.757710, 77.709419],
                        [8.760355, 77.709870],
                        [8.762889, 77.710737],
                        [8.764104, 77.711175],
                        [8.765409, 77.711589],
                        [8.766494, 77.711936],
                        [8.767979, 77.712454],
                        [8.769694, 77.713414],
                        [8.770419, 77.713885],
                        [8.771009, 77.714328],
                        [8.773284, 77.716484],
                        [8.774433, 77.717908],
                        [8.776211, 77.720094],
                        [8.777619, 77.721823],
                        [8.779270, 77.723824],
                        [8.781513, 77.726591],
                        [8.783392, 77.728780],
                        [8.783897, 77.729422],
                        [8.784331, 77.729961]
                        ], 12765: [12.972442,77.580643], 14375: [9.939093,78.121719]}
```

OUTPUT:



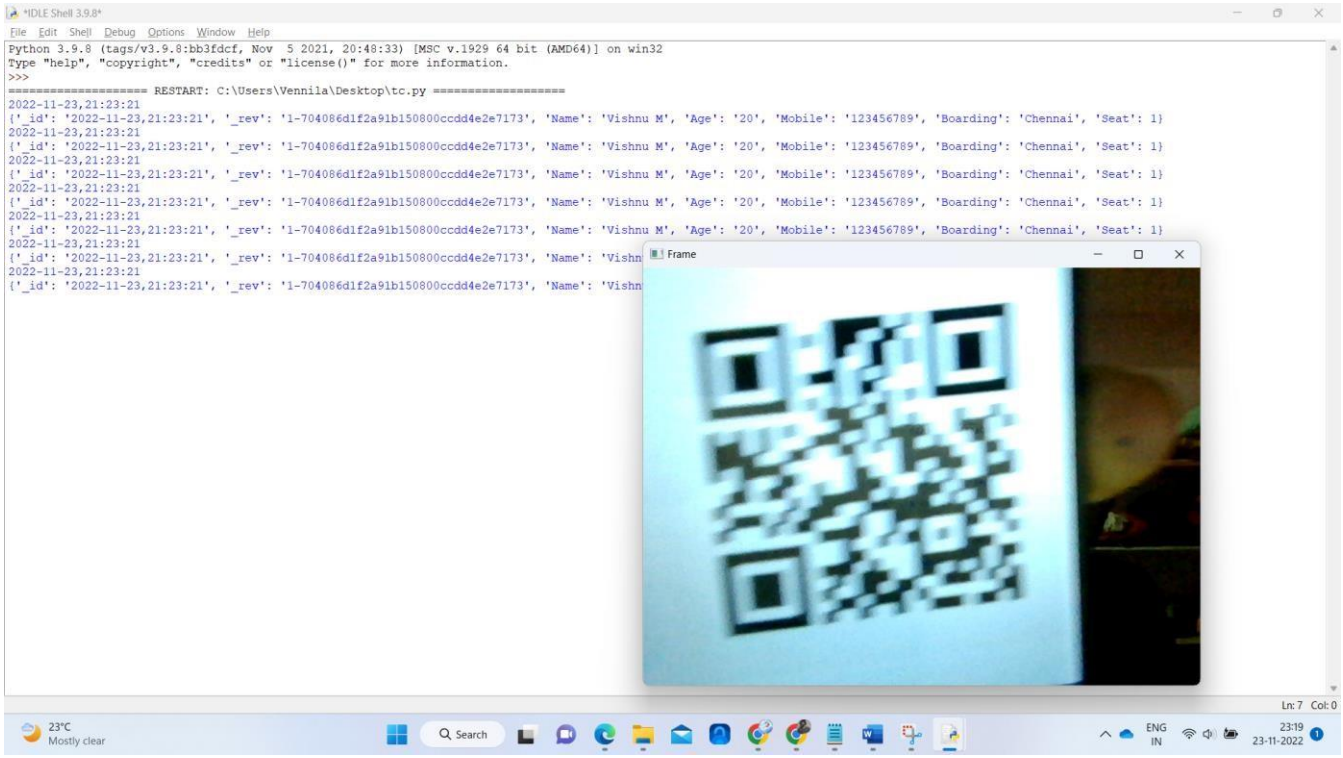
QR SCANNER CODE :

```
from http import client import cv2 import

pyzbar  from pyzbar.pyzbar import decode

import time


from ibmcloudant.cloudant_v1 import CloudantV1  from ibmcloudant import

CouchDbSessionAuthenticator from ibm_cloud_sdk_core.authenticators import BasicAuthenticator


authenticator = BasicAuthenticator('apikey-v2-
rsy830cz1zi58n2c6r65zltdnil5hsvuehtcomsrbbe', '8c8217f7524c8e496de81adc45fd866d') service =

CloudantV1(authenticator=authenticator)


service.set_service_url('https://apikey-v2-
rsy830cz1zi58n2c6r65zltdnil5hsvuehtcomsrbbe:8c8217f7524c8e496de81adc45fd866d@b a67c7fa-
520d4bdc-9344-0a240f78077b-bluemix.cloudantnosqldb.appdomain.cloud')
```

```
cap= cv2.VideoCapture(0) font =
cv2.FONT_HERSHEY_PLAIN

while True:
      _, frame = cap.read()
decodedObjects = 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='ibm_railways',              doc_id = a
).get_result()              print (response)
time.sleep(5)          except Exception as e:
            print(a)
            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()
```

```
*tc.py - C:\Users\Vennila\Desktop\tc.py (3.9.8)*
File  Edit  Format  Run  Options  Window  Help
import cv2
import numpy as np
import time
import pyzbar.pyzbar as pyzbar
from ibmcloudant import CouchDbSessionAuthenticator
from ibmcloudant.cloudant_v1 import CloudantV1
from ibm_cloud_sdk_core.authenticators import BasicAuthenticator

authenticator = BasicAuthenticator('apikey-v2-1o48jnpbynon2fygu5zws0pcep7l6xunbxh2z0c46a90','0334e2578b84d12e0abb5bc598f8cd02')
service= CloudantV1 (authenticator=authenticator)
service.set_service_url('https://9e48d93f-55be-4591-b502-a603e02210e7-bluemix.cloudantnosqldb.appdomain.cloud/')

cap = cv2.VideoCapture(0)
font = cv2.FONT_HERSHEY_PLAIN

while True:
        _, frame = cap.read()
        decodedObjects = pyzbar.decode (frame)
        for obj in decodedObjects:
                #print ("Data", obj.data)
                a=obj.data.decode('UTF-8')
                cv2.putText(frame, "Ticket", (50, 50), font, 2,
                            (255, 0, 0), 3)
                print(a)
                try:
                        response = service.get_document(
                            db='passengerdetails',
                            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()
```

OUTPUT:


IDLE Shell 3.9.8
File Edit Shell Debug Options Window Help
Python 3.9.8 (tags/v3.9.8:bb3fdcf, Nov 5 2021, 20:48:33) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
========== RESTART: C:\Users\Vennila\Desktop\tc.py ==========
2022-11-23,21:23:21
{'_id': '2022-11-23,21:23:21', '_rev': '1-704086d1f2a91b150800ccdd4e2e7173', 'Name': 'Vishnu M', 'Age': '20', 'Mobile': '123456789', 'Boarding': 'Chennai', 'Seat': 1}
2022-11-23,21:23:21
{'_id': '2022-11-23,21:23:21', '_rev': '1-704086d1f2a91b150800ccdd4e2e7173', 'Name': 'Vishnu M', 'Age': '20', 'Mobile': '123456789', 'Boarding': 'Chennai', 'Seat': 1}
2022-11-23,21:23:21
{'_id': '2022-11-23,21:23:21', '_rev': '1-704086d1f2a91b150800ccdd4e2e7173', 'Name': 'Vishnu M', 'Age': '20', 'Mobile': '123456789', 'Boarding': 'Chennai', 'Seat': 1}
2022-11-23,21:23:21
{'_id': '2022-11-23,21:23:21', '_rev': '1-704086d1f2a91b150800ccdd4e2e7173', 'Name': 'Vishnu M', 'Age': '20', 'Mobile': '123456789', 'Boarding': 'Chennai', 'Seat': 1}
2022-11-23,21:23:21
{'_id': '2022-11-23,21:23:21', '_rev': '1-704086d1f2a91b150800ccdd4e2e7173', 'Name': 'Vishnu M', 'Age': '20', 'Mobile': '123456789', 'Boarding': 'Chennai', 'Seat': 1}
2022-11-23,21:23:21
{'_id': '2022-11-23,21:23:21', '_rev': '1-704086d1f2a91b150800ccdd4e2e7173', 'Name': 'Vishn
2022-11-23,21:23:21
{'_id': '2022-11-23,21:23:21', '_rev': '1-704086d1f2a91b150800ccdd4e2e7173', 'Name': 'Vishn
Frame
Ln: 7 Col: 0
23°C
Mostly clear
Search
ENG
IN
23:19
23-11-2022

GitHub & Project Demo Link :

GitHub Link - https://github.com/IBM-EPBL/IBM-Project-26445-1660026927