Title	Project Report
Team ID	PNT2022TMID19465
Project Name	Smart Solution For Railways
Date	16/11/2022

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.

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 REFERENCES

S.No	TITLE	PROPOSED WORK	TOOLS USED/ ALGORITHM	TECHNOLOGY	ADVANTAGES/ DISADVANTAGES
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 tyical 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.

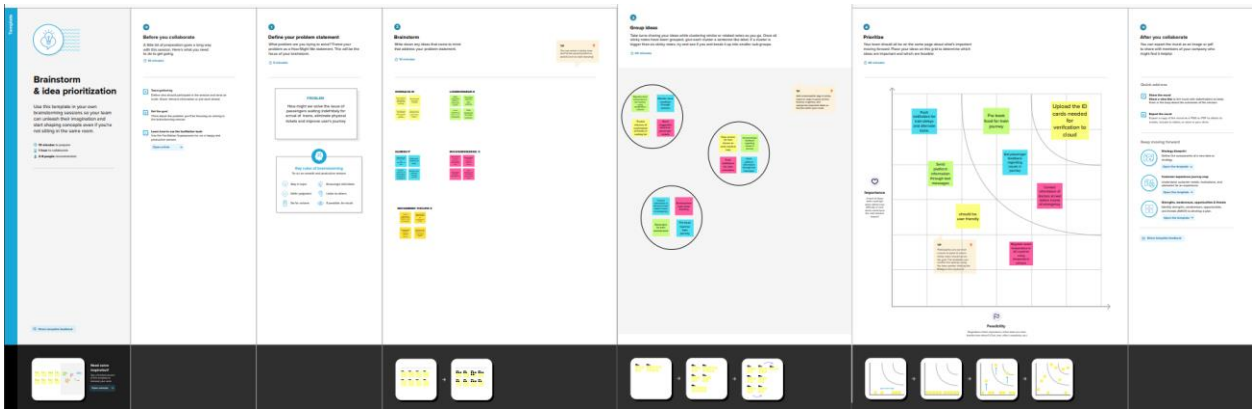
2.3.PROBLEM STATEMENT DEFINITION:

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

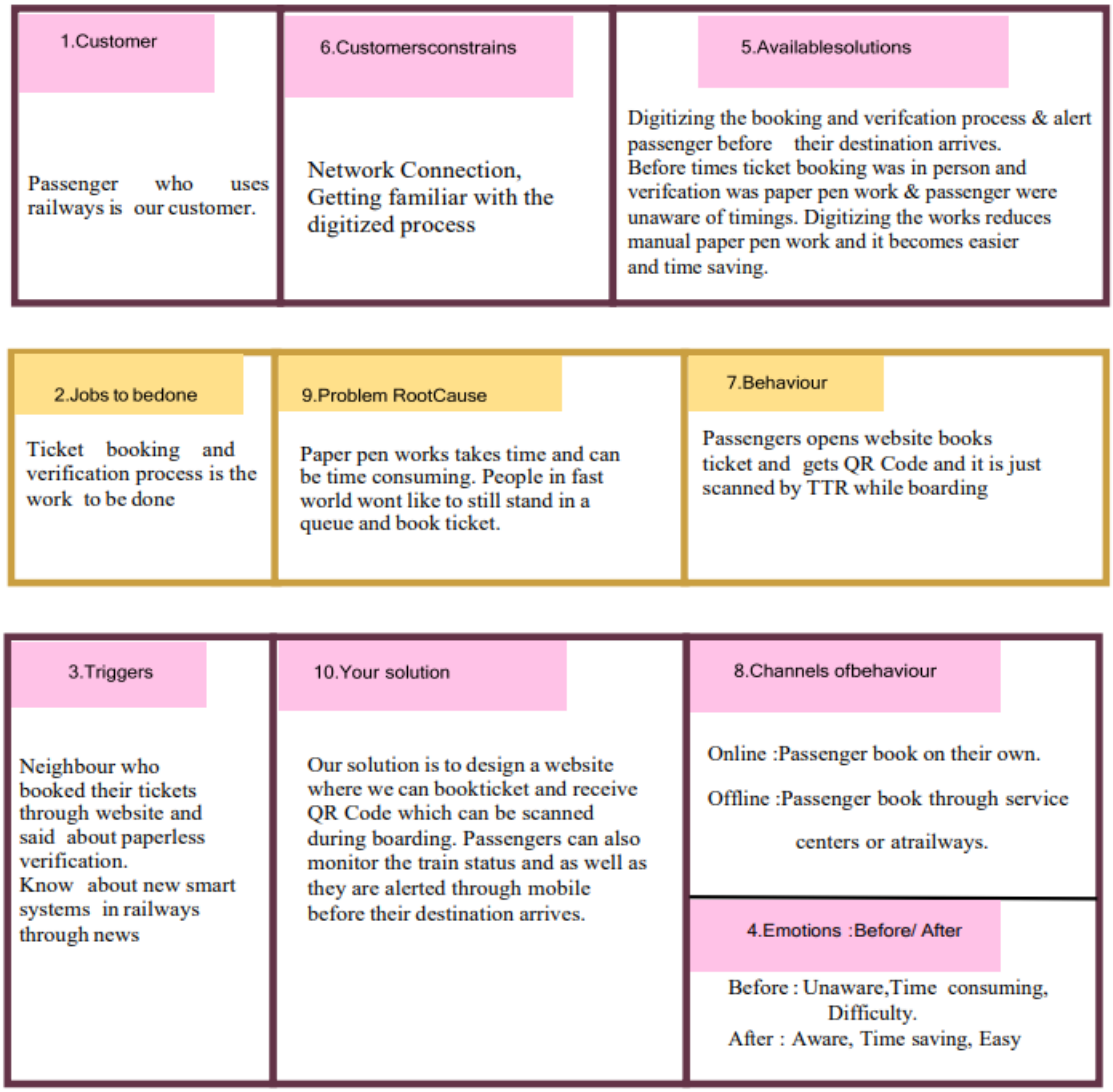
3.2 Ideation & Brainstorming :



3.3 Proposed Solution :

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	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.
2.	Idea / Solution description	The idea is able to predict the delays in prior and helps to decide best alternatives to reduce delays.With the help of sensor to detect the crack in rail track with measuring the distance from the track to sensor to reduce the accidents.
3.	Novelty / Uniqueness	Improving and increasing customer experience ,vehicle tracking system.IOT is used along with AI which Provides enhanced features in finding out delays.
4.	Social Impact / Customer Satisfaction	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
5.	Business Model (Revenue Model)	Product and service sales to the railways system .User booking and Scheduling Service it makes passengers to avoid delays which will make more number of people to shift to railway mode of transportation which increases the revenue of railways
6.	Scalability of the Solution	Using IoT in railways, increased the use of trains among people due to its convenient usage.So it will automatically increase the both revenue and expenses ,but the revenue will chase the expenses and will be boosting .

3.4 Problem Solution fit :



4 REQUIREMENT ANALYSIS

Functional Requirements:

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 Gmail Registration through Mobile number
FR-2	User Confirmation	Confirmation via EmailConfirmation via OTP Confirmation via call
FR-3	Journey details	Provides From and To information and date of travel.
FR-4	Select Trains	Select the appropriate trains among the list and alsobased on the seat availability.
FR-5	Book and add passenger	Fill the essential details such as name, contact details age, sex.
FR-6	Proceed to pay	Select an appropriate payment options among UPI, Internet banking, credit card, debit card.
FR-7	Ticket confirmation and Invoices	Ticket confirmation status is send to their registered email id.
FR-8	GPS	Tracking the live location and the status will be updated to the passengers.
FR-9	PIR	To get the information of detecting the motions in Railway Track.
FR-10	Database management	Entire Journey details will be stored in the server.
FR-11	E-catering	Foods are available for the registered passengers in an effective manner.
FR-12	Health monitoring	Monitor the health of the loco pilot and alert when any emergency.
FR-13	Ultrasonic Sensor	To calculate the distance between the Train and Station.

Non-functional Requirements:

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

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

5 PROJECT DESIGN

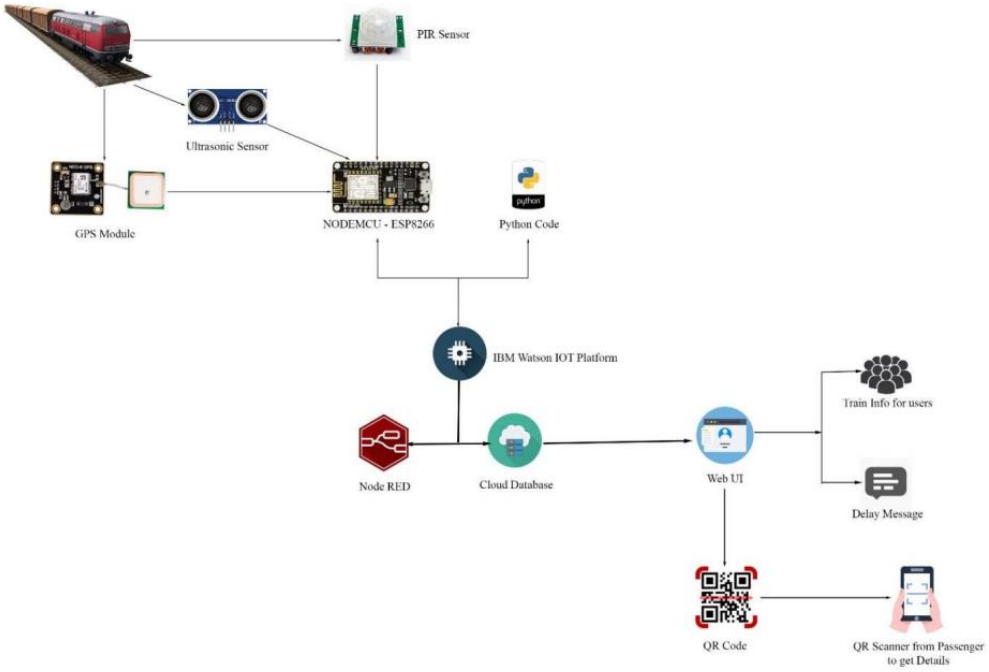
5.2 Data Flow Diagrams

Data Flow Diagrams:

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

Example:

5.3 Solution & Technical Architecture



Technical Architecture :

Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	Web UI	User can login and book their ticket through the website based on the availability of the seats.	HTML, CSS, JavaScript
2.	Cloud Services	Requirements filled by the passenger is stored in the cloud database.	Python
3.	GPS Tracking	Live Location details shared through the code to share the location in the website	IBM Watson Service
4.	External API-1	Used for rail schedule, ticketing and travel documents generation, cancellation.	Sabre API
5.	External API-2	Used for combining carriers and ticket types, Multilanguage & currency support.	Trainline B2B API
6.	Data Processing	Ticket is verified with the unique ID generated with the cloudant DB	Python, IBM cloud

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	CSS, Backend framework,	Python, IBM cloudant DB
2.	Security Implementations	Data entered are encrypted, Continuous Location Tracking	Python, Cloud service
3.	Scalable Architecture	The scanner and the codes written are highly scalable where any implementation can be done anytime needed	Python
4.	Availability	Any time available system. The ticket can be verified by the ticket collector from anywhere.	IBM Load Balancer
5.	Performance	Though the details are get stored in the cloud the system crash will not affect the data. The data can be retrieved from anywhere with a scanner. And the GPS states the exact location of the train.	Distributed Services, GPS Tracker

5.4 User Stories

User Stories

Use the below template to list all the user stories for the product.

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	Reserving ticket	USN-1	As a user, I can reserve for the ticket 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 email once I have registered for the account	I can receive confirmation email & click confirm	High	Sprint-1
		USN-3	As a user, I can register for the account and enter the details for reserving the ticket and get information about Train and Receive Alerts.	I can register & access the dashboard with Login Credentials.	Low	Sprint-1
	Dashboard	USN-4	As a user, I can view the reserved ticket in the dashboard.	I can access it usingdatabase	Medium	Sprint-3
Customer (Webuser)	Reserving ticket	USN-1	As a user, I can register to creating account by entering email, password.	I can access my account	High	Sprint-1
		USN-2	As a user, I will receive confirmation email once I have registered for the web user.	I can receive confirmation email.	High	Sprint-1
		USN-3	As a user, I can log into the website entering email & password.	Only valid credentials must be acceptable.	High	Sprint-1
	Dashboard	USN-4	As a user, I can register for the account and enter the details for reserving the ticket and get information about Train	I can register & access the dashboard with Login Credentials.	Low	Sprint-1
Customer Care Executive	Customer	USN-1	Connects with the service by logging in and get alert through it.	Can get connected withthe server	Medium	Sprint-1
Administrator	Admin	USN-1	As a admin, He/She can monitor real time and send alerts.	The admin can monitor the process by 24/7 hrs.	High	Sprint-1

6 PROJECT PLANNING & SCHEDULING

6.1.Sprint Planning & Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	As a passenger, I want to create a login credentials so I can securely access myselfservice online account.	15	High	KOWSALYA M LOGESHWARA S KAMESH P BHUVANESHWARAN K MOHAMMED THOUFIK S
Sprint-1	Ticket Conformation	USN-2	As a passenger, I want to check my ticketwhether it is conformed or not.	5	Medium	KOWSALYA M LOGESHWARA S KAMESH P BHUVANESHWARAN K MOHAMMED THOUFIK S
Sprint-2	Payment	USN-3	As a passenger, I want to pay my ticket cost inonline payment	15	High	KOWSALYA M LOGESHWARA S KAMESH P BHUVANESHWARAN K MOHAMMED THOUFIK S
Sprint-3	Booking Status	USN-4	As a passenger, I want to check my ticket onceit is conformed.	5	Medium	KOWSALYA M LOGESHWARA S KAMESH P BHUVANESHWARAN K MOHAMMED THOUFIK S
Sprint-4	Updating Train Information	USN-5	As an admin, I want to check the trains details like when will train reach stations and updateTrain information.	10	Medium	KOWSALYA M LOGESHWARA S KAMESH P BHUVANESHWARAN K MOHAMMED THOUFIK S

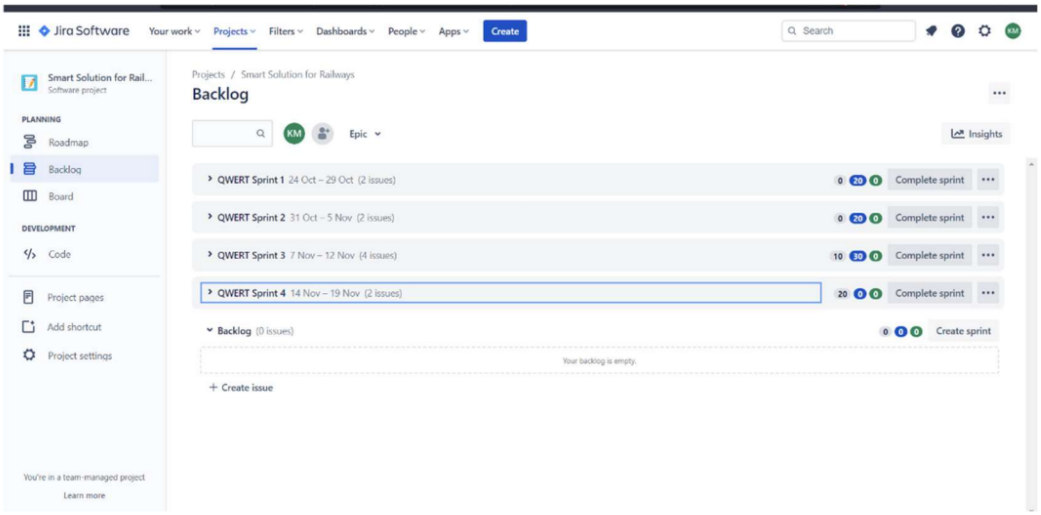
Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-3	Verifying Tickets	USN-6	As a TC, I want to check the users whether he/she have tickets or not with scanning the QR Code	15	High	KOWSALYA M LOGESHWARA S KAMESH P BHUVANESHWARAN K MOHAMMED THOUFIK S
Sprint-2	Knowing Current Location details	USN-7	As a passenger, I want to know the train current location.	5	Low	KOWSALYA M LOGESHWARA S KAMESH P BHUVANESHWARAN K MOHAMMED THOUFIK S
Sprint-4	Raise a compliant	USN-8	As a user, I should able to raise a ticket if something is wrong	10	Medium	KOWSALYA M LOGESHWARA S KAMESH P BHUVANESHWARAN K MOHAMMED THOUFIK S

6.2.Sprint Delivery Schedule

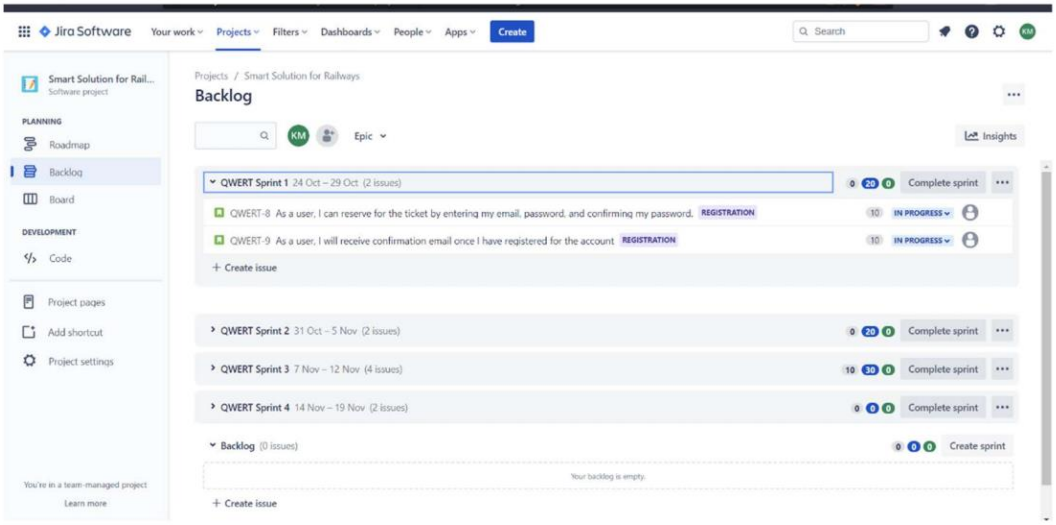
Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022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

6.3.Reports from JIRA:

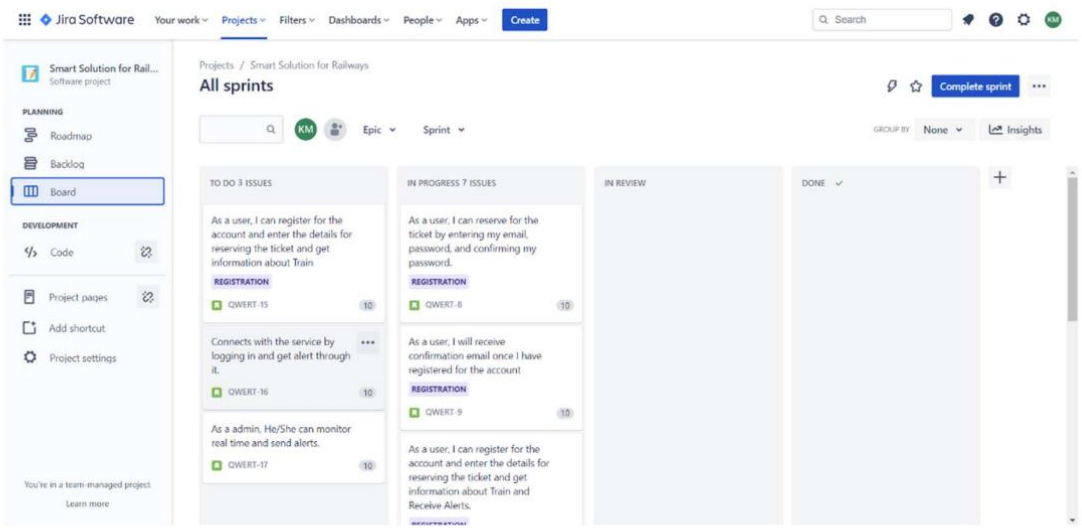
1. Image showing the backlogs create in the JIRA Software (PETA Spring1, PETA Spring2, PETA Spring3, PETA Spring4)



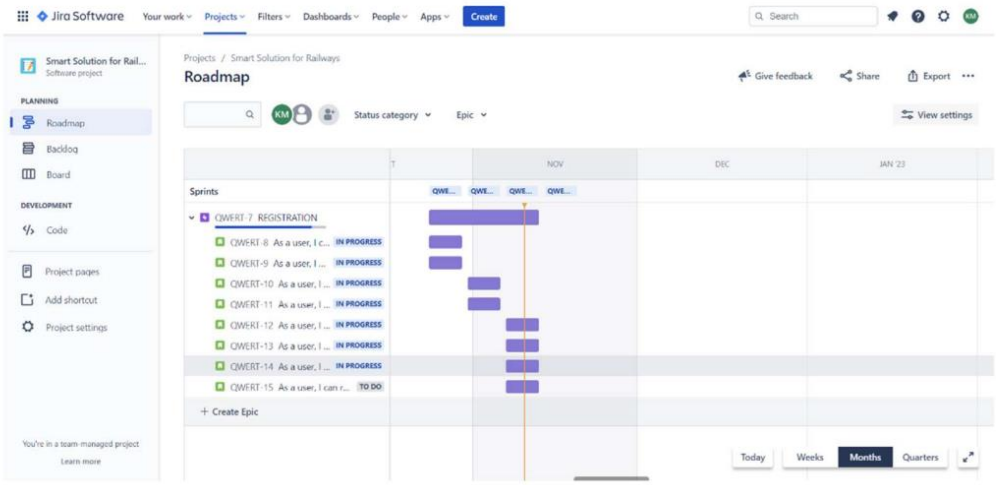
2. Image Showing the User Stories in the respective sprints.



3. Image showing the workspace progress in the sprint



4. Image Showing the RoadMap of Sprint



7 CODING & SOLUTIONING :

Python.py:

```
import wiotp.sdk.device
import time
import random
myConfig = {
    "identity": {
        "orgId": "xfxj98",
        "typeId": "railway23",
        "deviceId":"Device1"
    },
    "auth": {
        "token": "987456321"
    }
 }

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': 'Train1', 'lat': 17.6387448, 'lon': 78.4754336}
    pub (myData)
    time.sleep (3)
    #myData={'name': 'Train2', 'lat': 17.6387448, 'lon': 78.4754336)
    #pub (myData)
    #time.sleep (3)
    myData={'name': 'Train1', 'lat': 17.6341908, 'lon': 78.4744722}
```

```
    pub(myData)
    time.sleep(3)
    myData={'name': 'Train1', 'lat': 17.6340889, 'lon': 78.4745052}
    pub (myData)
    time.sleep (3)
    myData={'name': 'Train1', 'lat': 17.6248626, 'lon': 78.4720259}
    pub (myData)
    time.sleep (3)
    myData={'name': 'Train1', 'lat': 17.6188577, 'lon': 78.4698726}
    pub (myData)
    time.sleep (3)
    myData={'name': 'Train1', 'lat': 17.6132382, 'lon': 78.4707318}
    pub (myData)
    time.sleep (3)
    client.commandCallback = myCommandCallback
client.disconnect ()
```

Scanner.py

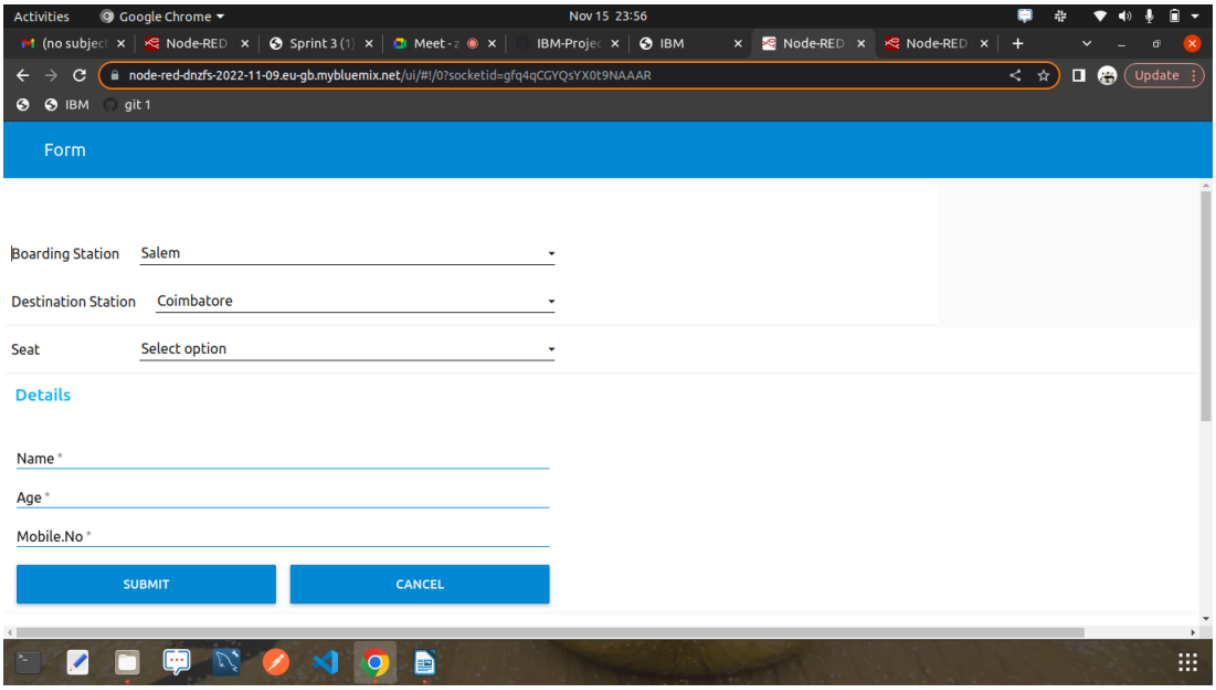
```
from http import client
import cv2
import pyzbar
from pyzbar.pyzbar import decode
import time

from ibmcloudant.cloudant_v1 import CloudantV1
from ibmcloudant import CouchDbSessionAuthenticator
from ibm_cloud_sdk_core.authenticators import BasicAuthenticator

authenticator = BasicAuthenticator('apikey-v2-1oj043bu90m78ng4h2j27w5nob2nvcma6xanc6bk0a7m',
'daf3c00c2cc182af425a5691a07f7b93')
service = CloudantV1(authenticator=authenticator)
```

```
service.set_service_url('https://apikey-v2-
1oj043bu90m78ng4h2j27w5nob2nvcma6xanc6bk0a7m:daf3c00c2cc182af425a5691a07f7b93@932393aa
-9f82-4144-9251-2c519fb30962-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='booking',
            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()
```

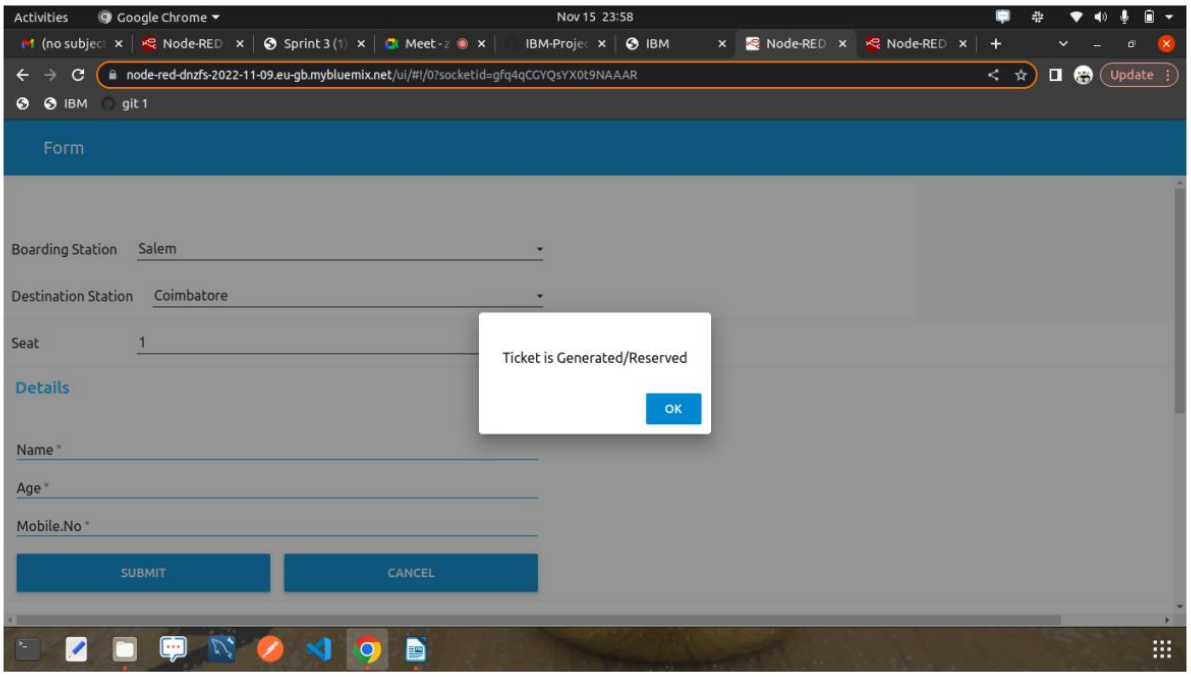
8 TESTING

8.2 Test Cases

Checking the web UI

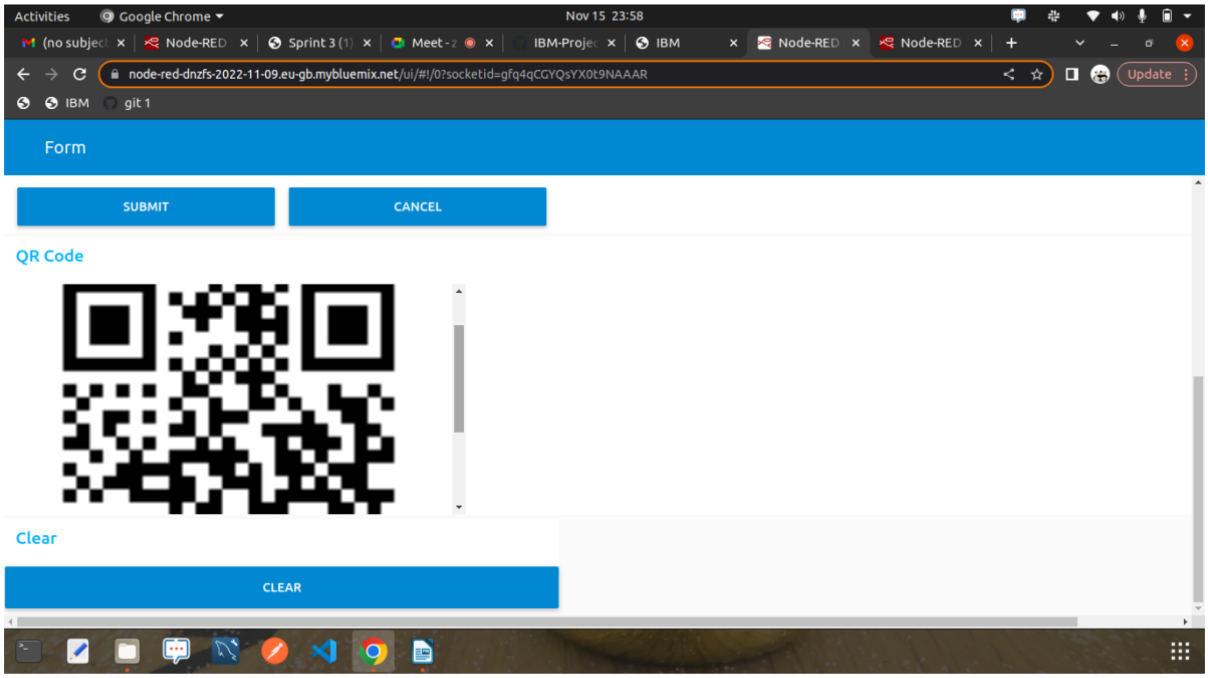


Testing for ticket registration



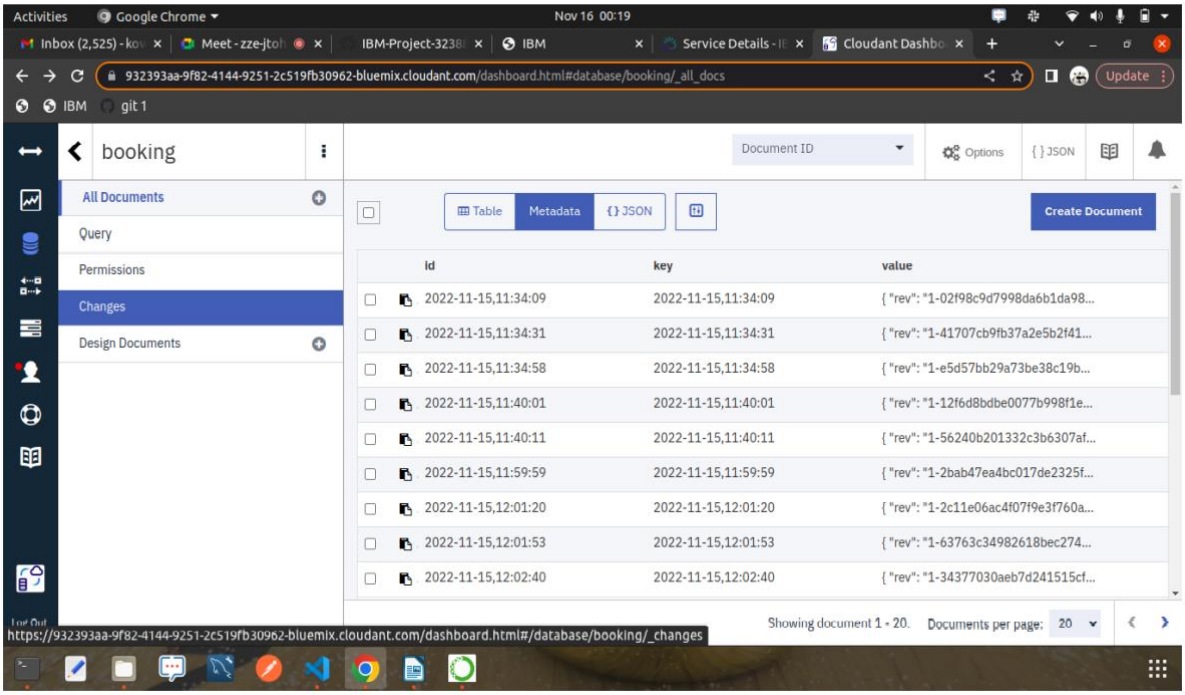
8.3 User Acceptance Testing

Testing for QR code generation



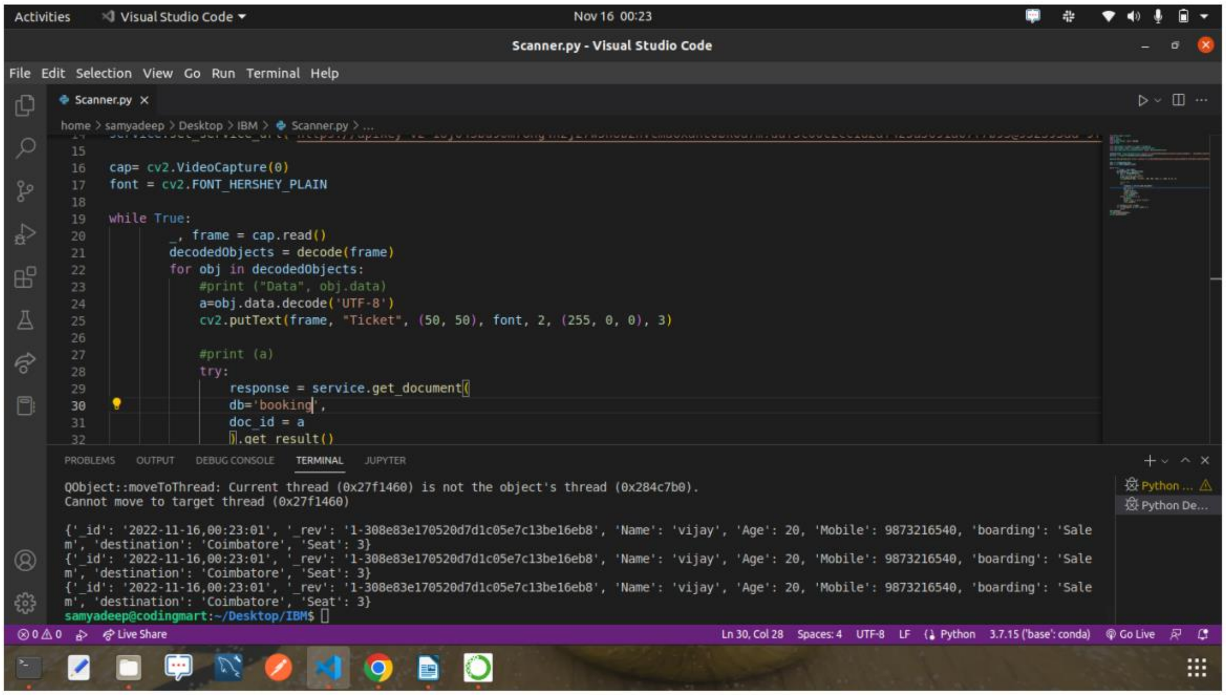
Generated successfully-test case passed

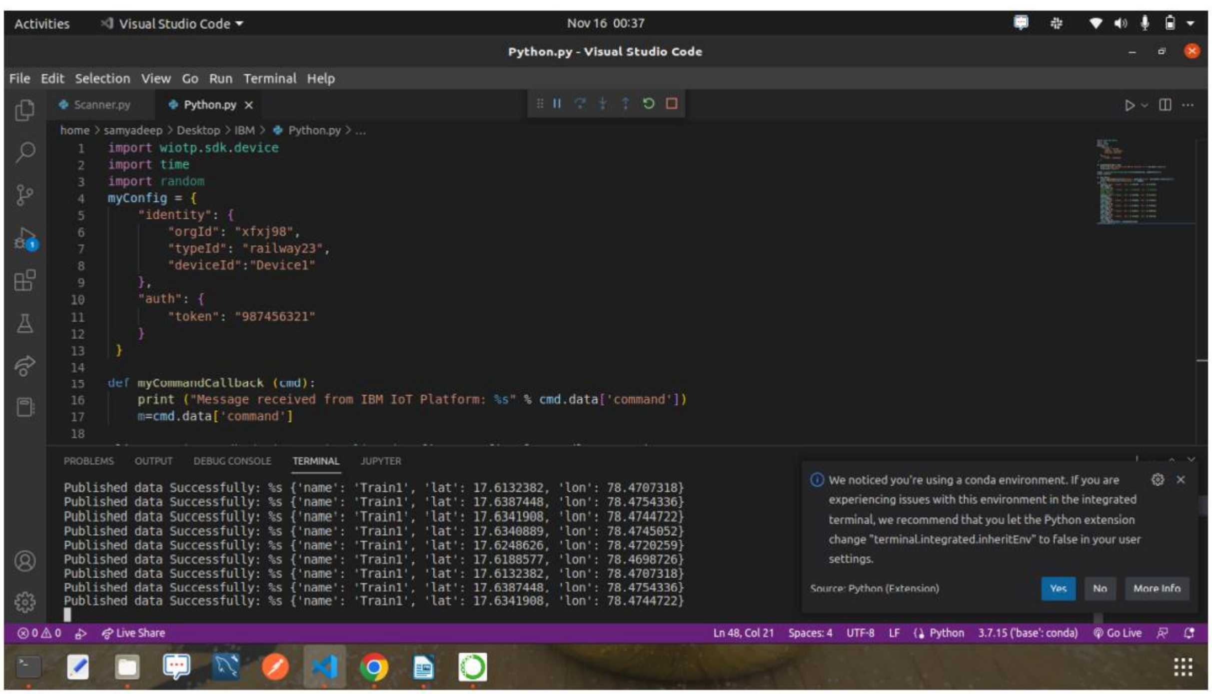
Testing for Storing in database



9 RESULTS

9.2 Performance Metrics





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-32388-1660209483/blob/main/Final%20Deliverables/Final%20Code/flows.json

GitHub & Project Demo Link:

https://github.com/IBM-EPBL/IBM-Project-32388-1660209483

https://github.com/IBM-EPBL/IBM-Project-32388-1660209483/blob/main/Final%20Deliverables/Demo%20Video/Demo%20video.mkv