

# **SMART SOLUTIONS FOR RAILWAYS**

***SUMBITTED BY***

**TEAM ID: PNT2022TMID00940**

<b>Aravind G</b>	<b>- 211419106030</b>
<b>Abinesh B</b>	<b>- 211419106005</b>
<b>Avinash AJ</b>	<b>- 211419106037</b>
<b>Allan Germanus S</b>	<b>- 211419106025</b>

***In the partial fulfillment of the award of the degree of***

**BACHELOR OF ENGINEERING**

**In**

**Electronics And Communication Engineering**

**Panimalar Engineering College**

**Chennai-6000123**

**NOV-2022**

# **ANNA UNIVERSITY : CHENNAI**

## **BONAFIDE CERTIFICATE**

Certified that this project report “**SMART SOLUTIONS FOR RAILWAYS**” is the bonafide work of “**ARAVIND G (211419106030) , ABINESH B (211419106005) , AVINASH AJ(211419106037) , ALLAN GERMANUS S (211419106025)**” who carried out the project work under supervision of Faculty Mentors and Industry Mentors .

# CONTENTS

## **1. INTRODUCTION**

Project Overview

Purpose

## **2. LITERATURE SURVEY**

Existing problem

References

Problem Statement Definition

## **3. IDEATION & PROPOSED SOLUTION**

Empathy Map Canvas

Ideation & Brainstorming

Proposed Solution

Problem Solution fit

## **4. REQUIREMENT ANALYSIS**

Functional requirement

Non-Functional requirements

## **5. PROJECT DESIGN**

Data Flow Diagrams

Solution & Technical Architecture

User Stories

## **6. PROJECT PLANNING & SCHEDULING**

Sprint Planning & Estimation

Sprint Delivery Schedule

Reports from JIRA

## **7. CODING & SOLUTIONING (Explain the features added in the project along with code)**

Feature 1

Feature 2

Database Schema (if Applicable)

## **8. TESTING**

Test Cases

User Acceptance Testing

## **9. RESULTS**

Performance Metrics

## **10.ADVANTAGES & DISADVANTAGES**

## **11.CONCLUSION**

## **12.FUTURE SCOPE**

## **13.APPENDIX**

Source Code

GitHub & Project Demo Link

# INTRODUCTION

# **1. INTRODUCTION**

## **1.1 PROJECT OVERVIEW**

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.

## **1.2. PURPOSE**

Internet is basically system of interconnected computers through network. But now its use is changing with changing world and it is not just confined to emails or web browsing. Today's internet also deals with embedded sensors and has led to development of smart homes, smart rural area, e-health care's etc. and

this introduced the concept of IoT . Internet of Things refers to interconnection or communication between two or more devices without human to-human and human-to-computer interaction. Connected devices are equipped with sensors or actuators perceive their surroundings. IoT has four major components which include sensing the device, accessing the device, processing the information of the device, and provides application and services. In addition to this it also provides security and privacy of data . Automation has affected every aspect of our daily lives. More improvements are being introduced in almost all fields to reduce human effort and save time. Thinking of the same is trying to introduce automation in the field of track testing. Railroad track is an integral part of any company's asset base, since it provides them with the necessary business functionality. Problems that occur due to problems in rail roads need to be overcome. The latest method used by the Indian railroad is the tracking of the train track which requires a lot of manpower and is time-consuming.

# **LITERATURE SURVEY**



## **LITERATURE SURVEY**

### **2.1 EXISTING SYSTEM**

In the Existing train tracks are manually researched. LED (Light Emitting Diode) and LDR (Light Dependent Resistor) sensors cannot be implemented on the block of the tracks ]. The input image processing is a clamorous system with high cost and does not give the exact result. The Automated Visual Test Method is a complicated method as the video color inspection is implemented to examine the cracks in rail track which does not give accurate result in bad weather. This traditional system delays transfer of information. Srivastava et al., (2017) proposed a moving gadget to detect the cracks with the help of an array of IR sensors to identify the actual position of the cracks as well as notify to nearest railway station . Mishra et al., (2019) developed a system to track the cracks with the help of Arduino mega power using solar energy and laser. AGSM along with a GPS module was implemented to get the actual location of the faulty tracks to inform the authorities using SMS via a link to find actual location on Google Maps. Rizvi Aliza Raza presented a prototype in that is capable of capturing photos of the track and compare it with the old database and sends a message to the authorities regarding the crack detected. The detailed analysis of traditional railway track fault detection techniques is explained in table.

## 2.2 REFERENCES

1. D. Hesse, "Rail Inspection Using Ultrasonic Surface Waves" Thesis, Imperial College of London, 2007.
2. Md. Reya Shad Azim<sup>1</sup> , Khizir Mahmud<sup>2</sup> and C. K. Das. Automatic railway track switching system, International Journal of Advanced Technology, Volume 54, 2014.
3. S. Somalraju, V. Murali, G. saha and V. Vaidehi, "Title-robust railway crack detection scheme using LED (Light Emitting Diode) - LDR (Light Dependent Resistor) assembly IEEE 2012.
4. S. Srivastava, R. P. Chourasia, P. Sharma, S. I. Abbas, N. K. Singh, "Railway Track Crack detection vehicle", IARJSET, Vol. 4, pp. 145-148, Issued in 2, Feb 2017.
5. U. Mishra, V. Gupta, S. M. Ahzam and S. M. Tripathi, "Google Map Based Railway Track Fault Detection Over the Internet", International Journal of Applied Engineering Research, Vol. 14, pp. 20-23, Number 2, 2019.
6. R. A. Raza, K. P. Rauf, A. Shafeeq, "Crack detection in Railway track using Image processing", IJARIT, Vol. 3, pp. 489-496, Issue 4, 2017.

7. N. Bhargav, A. Gupta, M. Khirwar, S. Yadav, and V. Sahu, “Automatic Fault Detection of Railway Track System Based on PLC (ADORTAST)”, International Journal of Recent Research Aspects, Vol. 3, pp. 91-94, 2016.

## **2.3 PROBLEM STATEMENT DEFINITION**

Among the various modes of transport, railways is one of the biggest modes of transport in the world. Though there are competitive threats from airlines, luxury buses, public transports, and personalized transports the problem statement is to answer the question “What are the problems faced by the passengers while travelling by train at station and on board”.

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

# **IDEATION AND PROPOSED SOLUTION**

### **3. IDEATION AND PROPOSED SOLUTION**

#### **3.1 EMPATHY MAP CANVAS**

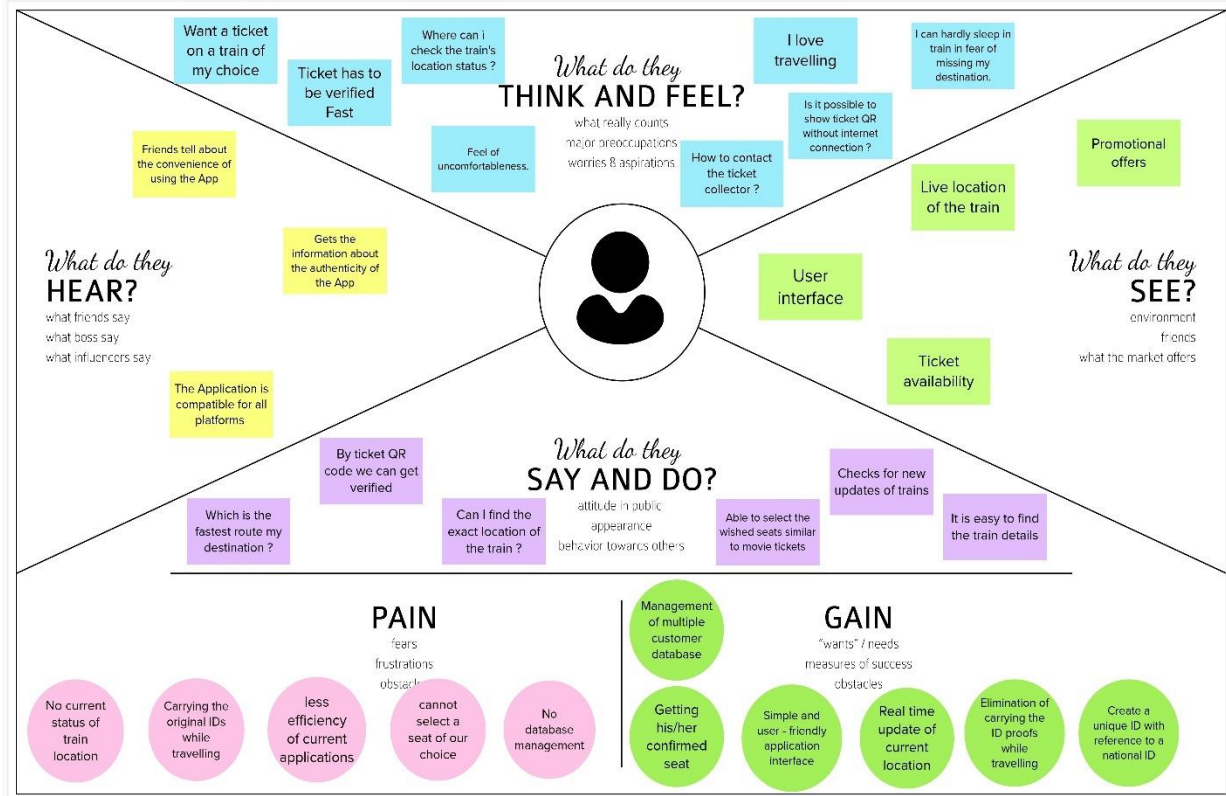
# Empathy Map Canvas

Gain insight and understanding on solving customer problems.

## Smart Solutions For Railways

1

Build empathy and keep your focus on the user by putting yourself in their shoes.



## 3.2 IDEATION & BRAINSTORMING



## Brainstorm & idea prioritization

Use it's versatile in your own  
but not in your own  
can't think of their imagination and  
start shaping concepts even if you're  
not sitting in the same room.

- 1. Main idea or topic
- 2. Think of ideas
- 3. 20 possible ideas

Brainstorming tool

1

### Before you collaborate

All the steps are essential and  
• Brainstorming is not a one-time  
activity.

### Team gathering

- Alan Gernius S
- Anish B
- Anish A
- Anish G

### Set the goal

To build on the Train Ticket Booking  
platform with advanced features that serve  
the needs of end user

Open article

1

### Define your problem statement

- Many passengers request to know the  
availability of seats on a train and  
the status of the train.
- To book seats on a train based on the availability.
- To manage the train and the passengers in the  
system.
- To manage the train and the passengers in the  
system.
- To manage the train and the passengers in the  
system.

#### SOLUTION

- To build on the Train Ticket Booking  
platform with advanced features that serve  
the needs of end user
- Creating a platform for  
• User interface
- User interface
- User interface

#### KEY FEATURES

- Easy Verification
- Unique QR Ticket
- Live Location Status
- Seat Availability
- Data management
- Time management

2

## Brainstorm

ALLAN :

Access to  
train's live  
location

Access to  
train's live  
location

Access to  
train's live  
location

Access to  
train's live  
location

Access to  
train's live  
location

Access to  
train's live  
location

Access to  
train's live  
location

Access to  
train's live  
location

ABINISH :

Access to  
train's live  
location

Access to  
train's live  
location

Access to  
train's live  
location

Access to  
train's live  
location

Access to  
train's live  
location

Access to  
train's live  
location

Access to  
train's live  
location

Access to  
train's live  
location

ABINISH :

Access to  
train's live  
location

Access to  
train's live  
location

Access to  
train's live  
location

Access to  
train's live  
location

Access to  
train's live  
location

Access to  
train's live  
location

Access to  
train's live  
location

Access to  
train's live  
location

3

## Group ideas

### QR GENERATION

- Train's live location
- Train's live location
- Train's live location
- Train's live location

### DATABASE MANAGEMENT

- Train's live location
- Train's live location
- Train's live location
- Train's live location

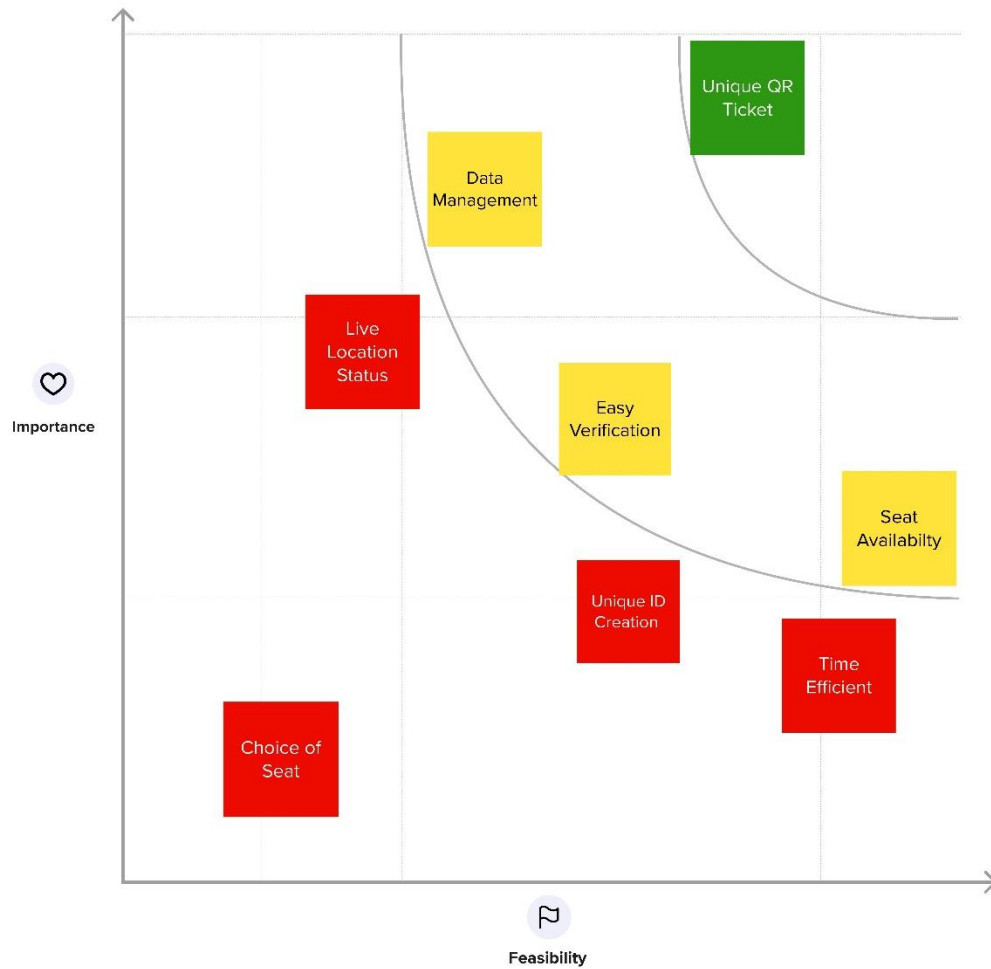
### SEAT AVAILABILITY

- Train's live location
- Train's live location
- Train's live location
- Train's live location

### LIVE LOCATION STATUS

- Train's live location
- Train's live location
- Train's live location
- Train's live location

# Prioritize





### 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 lot 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

Define CS, fit into CL	<b>1. CUSTOMER SEGMENT(S)</b> <b>CS</b>  Passengers who uses railways is our customer.	<b>6. CUSTOMER LIMITATIONS</b> <small>EG. BUDGET, DEVICES</small> <b>CL</b>  Network Connection, Getting familiar with the digitized process.	<b>5. AVAILABLE SOLUTIONS</b> <small>PLUSES &amp; MINUSES</small> <b>AS</b>  Digitizing the booking and verification process & alert passenger before their destination arrives. Before times ticket booking was in person and verification was paaper pen work & passenger were unaware of timings. Digitizing the works reduces manual paper pen work and it becomes easier and time saving.	Explore AS, differentiate
	<b>2. PROBLEMS / PAINS + ITS FREQUENCY</b> <b>PR</b>  No current status of train location. Carrying the original ID's while travelling. Less efficiency of current application. Cannot select exactly the choice of seat due to mass booking. Huge database management is required. Server and connectivity issues.	<b>9. PROBLEM ROOT / CAUSE</b> <b>RC</b>  Paper pen works take time and can be time consuming. People in fast world wont like to still stand in a queue and book ticket.	<b>7. BEHAVIOR + ITS INTENSITY</b> <b>BE</b>  Passengers open website books ticket andd gets QR Code and it is just scanned by TTR while boarding.	
Focus on PR, tap into BE, understand RC	<b>3. TRIGGERS TO ACT</b> <b>TR</b>  Neighbour who boooked their tickets through website and said about paperless verificaton . Know about new smart sytems in railways through news.	<b>10. YOUR SOLUTION</b> <b>SL</b>  Our solution is to design a website where we can book ticket and receive QR Code which can be scanned during boarding. Passengers can also monitor the train status and as well as they are alerted through mobile before their destination arrives.	<b>8. CHANNELS of BEHAVIOR</b> <b>CH</b>  ONLINE  Passengers book on their own.	Extract online & offline CH of BE
	<b>4. EMOTIONS</b> <small>BEFORE / AFTER</small> <b>EM</b>  <b>Before:</b> Unaware, Time consuming, Difficulty <b>After:</b> Aware, Time saving , Easy.		  OFFLINE  Passenger book through service centers or at railways.	
Identify strong TR & EA				

# **REQUIREMENT ANALYSIS**

## 4. REQUIREMENT ANALYSIS

### 4.1. FUNCTIONAL REQUIREMENTS

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	Passenger Registration	Registration through Application by providing the required Details
FR-2	Passenger Login	Login Using the Unique Username and corresponding Password
FR-3	Admin Login	Login Using the Admin Username and Password
FR-4	Passenger Books Ticket	Books the Ticket through app by providing the Details required
FR-5	Selecting the Seat	While booking passenger should select which seat is comfortable for him/her.
FR-6	QR Code Generation	A QR Code is generated following a successful booking
FR-7	Admin Cancel the Booking	If the information is inappropriate or the passenger is suspected to be an inappropriate person, admin can cancel his/her ticket.
FR-8	Tracking the location of Train	Passenger can view the current location of his/her Train.
FR-9	TTR Verifies the Passenger	TTR Scans the QR Code shown by the Passenger which gives the passenger Information that have to be verified.

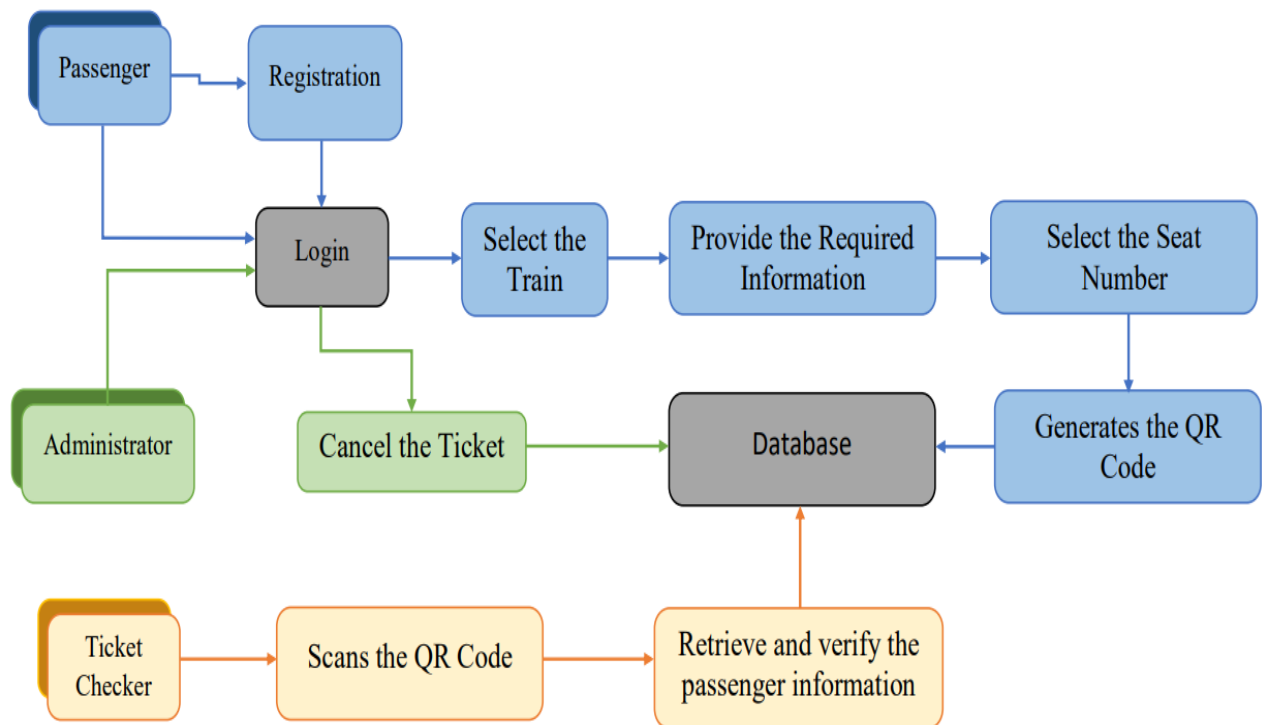
## 4.2 NON-FUNCTIONAL REQUIREMENTS

FR No.	Non-Functional Requirement	Description
NFR-1	Usability	People with no high knowledge on using mobile can easily handle the application.
NFR-2	Security	Access permissions for the particular system information may only be changed by the system's data administrator.
NFR-3	Reliability	The database update process must roll back all related updates when any update fails.
NFR-4	Performance	The front-page load time must be no more than 2 seconds for users that access the website using an LTE mobile connection.
NFR-5	Availability	New module deployment mustn't impact front page, product pages, and check out pages availability and mustn't take longer than one hour. The rest of the pages that may experience problems must display a notification with a timer showing when the system is going to be up again.
NFR-6	Scalability	The website attendance limit must be scalable enough to support 10,000 users at a time

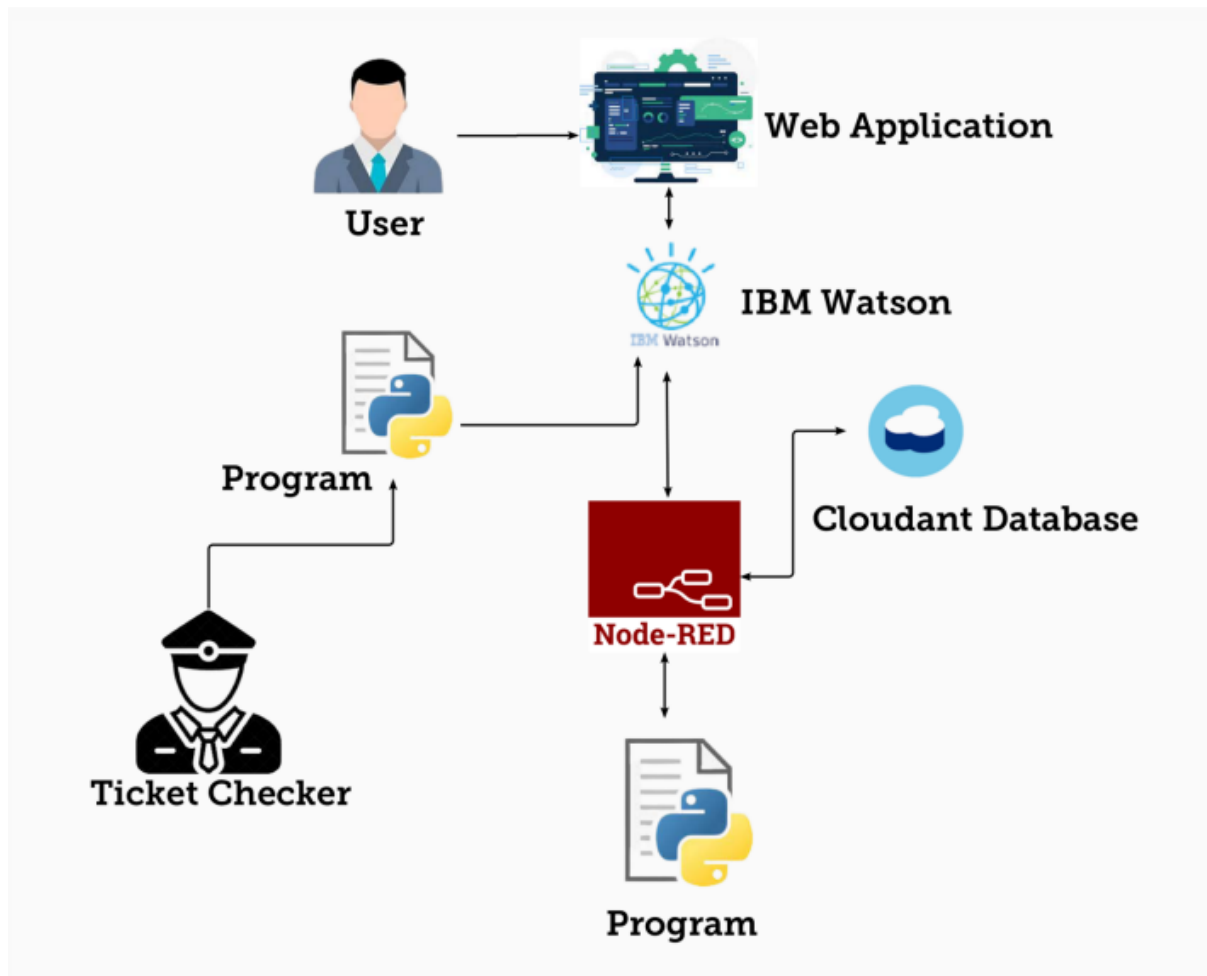
# PROJECT DESIGN

## 5. PROJECT DESIGN

### 5.1 DATA FLOW DIAGRAMS



## 5.2 SOLUTION & TECHNICAL ARCHITECTURE





## 5.3 USER STORIES

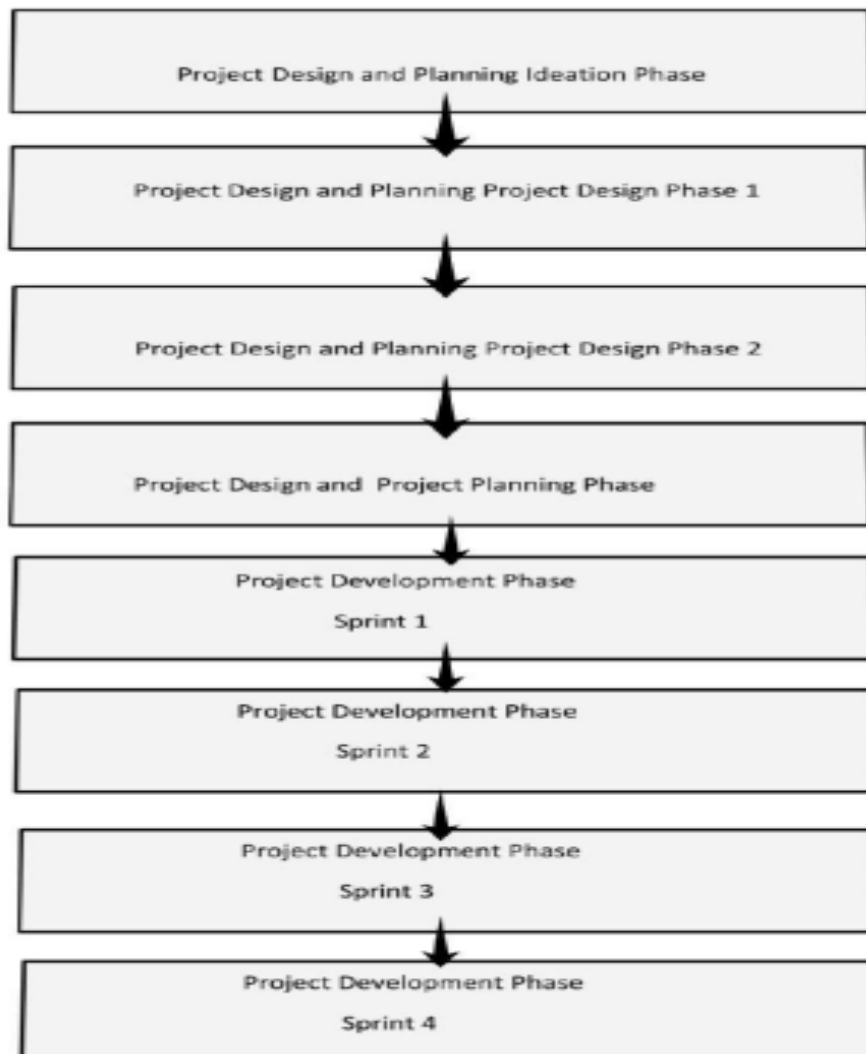
User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority
Passenger	Registration	USN-1	As a Passenger, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High
		USN-2	As a Passenger, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	High
	Login	USN-3	As a Passenger, I can log into the application by entering email & password	I can access my account / dashboard	High
	Books Ticket	USN-4	I can select the Train and the train route to be travelled.	I can see the Train Name and Its route in the form.	Medium
		USN-5	I provide the basic details such as name, age, mobile number, etc.	I can see the Basic Information in the form.	High
	Selecting the Seat	USN-6	After providing the basic information, I can select the desired seat I wanted if it is in available state.	I can see that the seat is selected for booking.	Medium
	QR Code Generation	USN-7	At last the QR Code is generated which contains the unique id through which the passenger information can be retrieved.	I can receive the QR Code which I can be screenshotted.	High
	Tracking the location of Train	USN-8	As a Passenger, I can track the exact current location of the train.	I can view the exact location of the train.	Medium
Administrator	Login	USN-9	As a Administrator, I can log into the application by entering email & password	I can access my account/ dashboard	Medium
	Cancel the Booking	USN-10	As a Administrator, I can Cancel the Ticket if the information of the passenger is inappropriate.	I can receive confirmation message about the cancellation of the ticket.	Low
Ticket Checker	TTR Verifies the Passenger	USN-11	As a Ticket Checker, I can scan the QR Code shown by the passenger.	I can see the passenger information in application.	High

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority
		USN-12	As a Ticket Checker, I can verify the passenger using the information that displayed after scanning the QR Code.	I can verify that the passenger information is matching with the information shown by the QR Code.	High

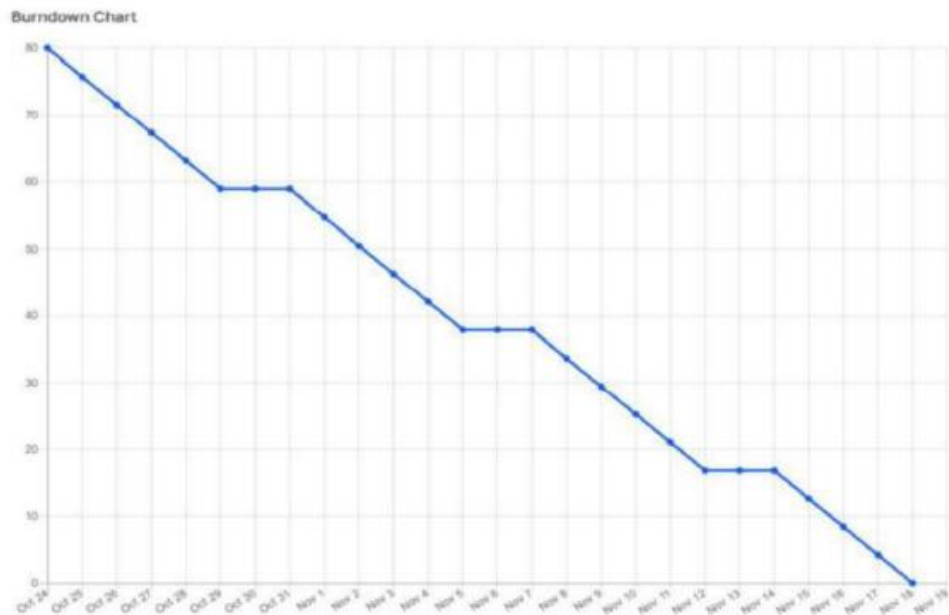
# **PROJECT PLANNING AND SCHEDULING**

## 6. PROJECT PLANNING AND SCHEDULING

### 6.1. SPRINT PLANNING & ESTIMATION



### Burndown Chart:



## 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

### Project Tracker, Velocity & Burndown Chart: (4 Marks)

#### Velocity:

Imagine we have a 10-day sprint duration, and the velocity of the team is 20 (points per sprint). Let's calculate the team's average velocity (AV) per iteration unit (story points per day)

$$AV = \frac{\text{sprint duration}}{\text{velocity}} = \frac{20}{10} = 2$$

# **CODING AND SOLUTIONING**

## **7. CODING AND SOLUTIONING**

### **7.1. FEATURE**

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

### **7.2. FEATURE**

- Verification
- Ticket Booking
- Ticket Cancellation
- Know your location
- Available Seats
- QR Generation
- TC Verifiaction made easy
- Details stored at Database (DB)

## PYTHON CODE FOR TRAIN LOCATION:

```
import wiotp.sdk.device
import time
import random
myConfig = {
    "identity": {
        "orgId": "ojhlri",
        "typeId": "GPS",
        "deviceId": "12345"
    },
    "auth": {
        "token": "12345678"
    }
}

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

client =
wiotp.sdk.device.DeviceClient(config=myCon
fig, 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 (4)
    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 ()

```

## PYTHON CODE FOR SCANNING THE QR:

```

import cv2
import numpy as np
import time

```

```

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

authenticator =
BasicAuthenticator('apikey-
267fe83f32ec443b804f3c76d688d2e3',
'b1456cbe3c712c2d555623fe092c074478521354'
)
service =
CloudantV1(authenticator=authenticator)

service.set_service_url('https://4f470103-
dd2c-4f50-94e8-3439395a7935-
bluemix.cloudant.com')

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(

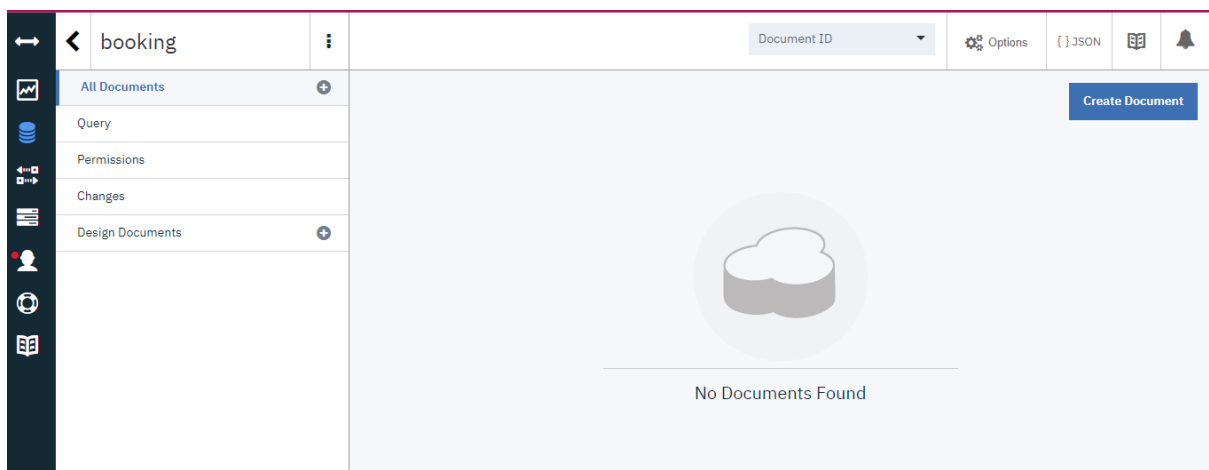
```

```
        db='booking',
        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()
```

**TESTING**

### Node-Red Connection for Ticket Booking and QR Generation:



## WEB Application created from Node-Red:

Booking Corner

Boarding Station Chennai ▼

Destination Bangalore ▼

Seat Select o... ▼

Name \*

Age \*

Mobile Number \*

SUBMIT CANCEL

CLEAR

## LINK FOR Booking Corner:

<https://node-red-gitmx-2022-11-08.eu-gb.mybluemix.net/ui/>

## Booking Happens :

≡ QR Code Gen

Booking Corner

Boarding Station

Destination

Seat

Name \*

Age \*

Mobile Number \*

SUBMIT

CANCEL

CLEAR

Hyderabad

Vijayawada

Chennai

Bangalore

Arun

21

753216497

≡ QR Code Gen

Booking Corner

Boarding Station

Destination

Seat

Name \*

Age \*

Mobile Number \*

SUBMIT

CANCEL

CLEAR

Hyderabad

Chennai

Select o...

Ticket is Generated

OK

## QR Generation:

≡ QR Code Gen


Age \*

Mobile Number \*

SUBMIT

CANCEL

CLEAR





## Details will be stored at Cloudant Database:

↔

<

booking

⋮

Document ID

Options

{ } JSON

🔔

All Documents

Query

Permissions

Changes

Design Documents

Table

Metadata

{ } JSON

Create Document

Boarding

Destination

Name

Age

Mobile Number

☐

📍 Hyderabad

Chennai

Arun

21

753216497

↔

booking > 2022-11-12,15:27:37

{ } JSON

🔔

Save Changes

Cancel

Upload Attachment

Clone Document

Delete

1

2

3

4

5

6

7

8

9

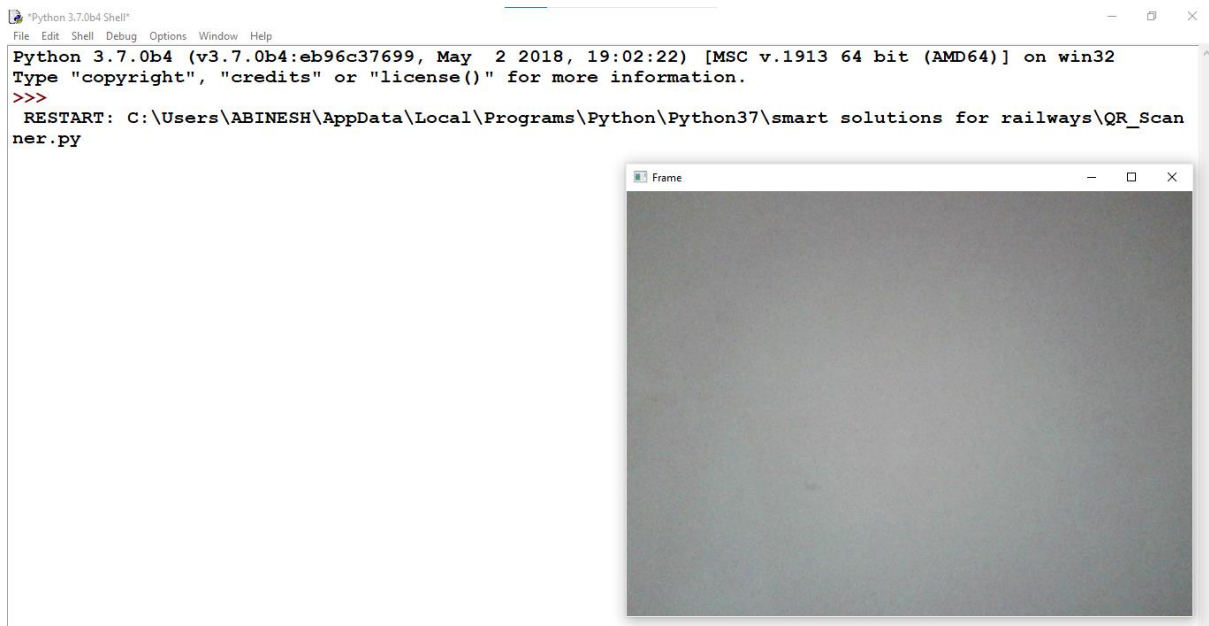
```
"_id": "2022-11-12,15:27:37",
"_rev": "1-c31e1d7186bd8ccf6104252d74a3e8bd",
"name": "Arun",
"Age": 21,
"Mobile Number": 753216497,
"Boarding": "Hyderabad",
"Destination": "Chennai"
```



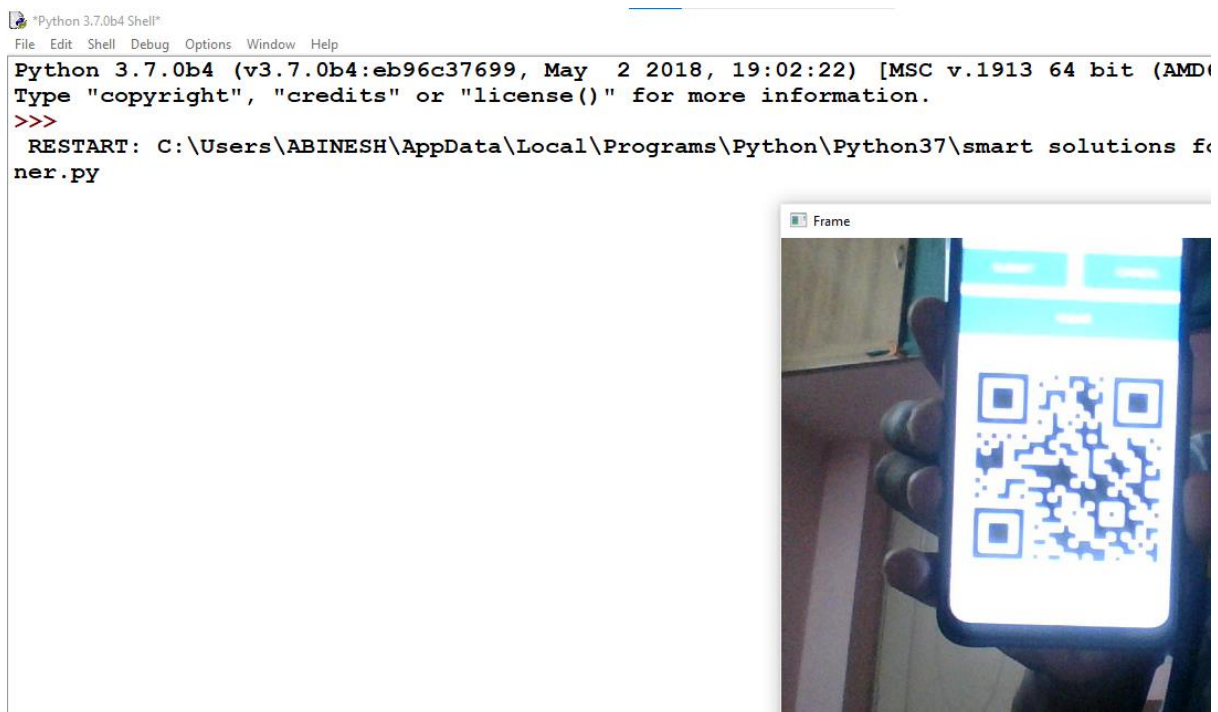
# RESULT

**Executing the program:**

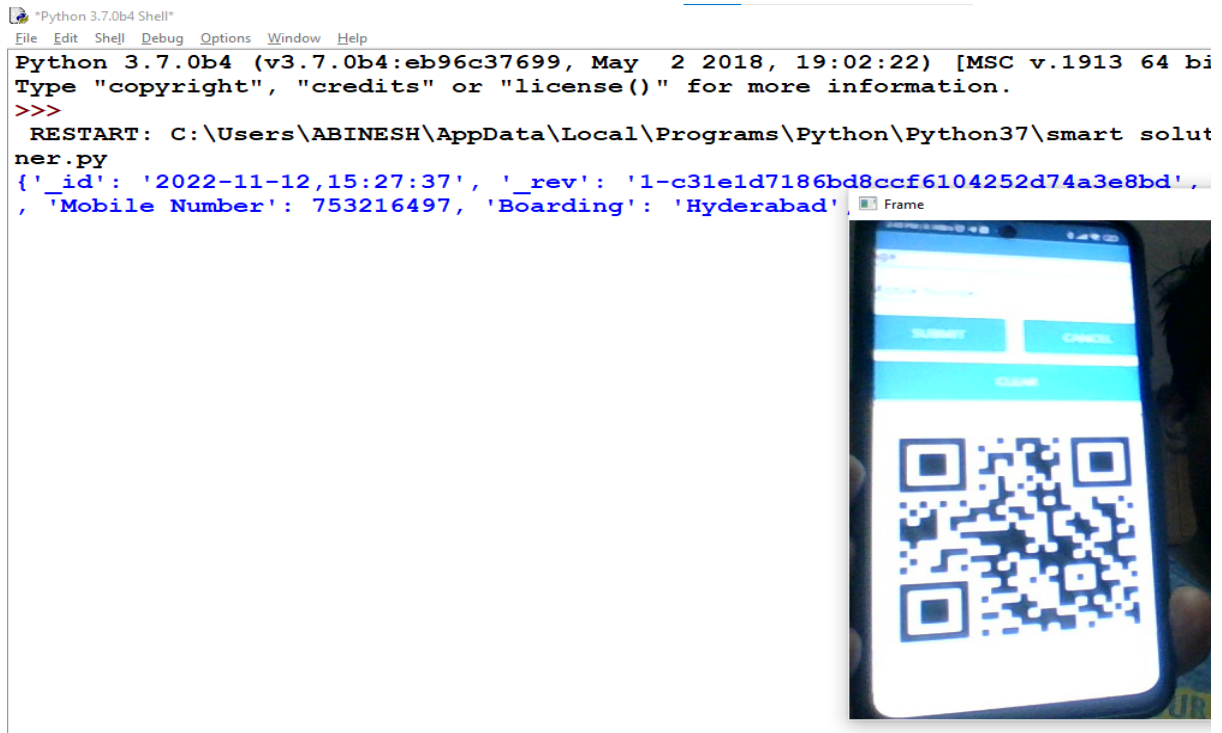
**Scanner Opens :**



**QR is shown:**



**Data Fetched from Cloud by scanning the QR:**



Showing a Random Qr From Google:



Output: (Shows “Not a Valid Ticket”)

Python 3.7.0b4 Shell

File Edit Shell Debug Options Window Help

Python 3.7.0b4 (v3.7.0b4:eb96c37699, May 2 2018, 19:02:22) [MSC v.1913 64 b  
Type "copyright", "credits" or "license()" for more information.  
>>>  
RESTART: C:\Users\ABINESH\AppData\Local\Programs\Python\Python37\smart solu  
ner.py  
{'\_id': '2022-11-12,15:27:37', '\_rev': '1-c31e1d7186bd8ccf6104252d74a3e8bd',  
, 'Mobile Number': 753216497, 'Boarding': 'Hyderabad',  
Not a Valid Ticket



Python 3.7.0b4 Shell

File Edit Shell Debug Options Window Help

Python 3.7.0b4 (v3.7.0b4:eb96c37699, May 2 2018, 19:02:22) [MSC v.1913 64 bit (AMD64)] on win32  
Type "copyright", "credits" or "license()" for more information.  
>>>  
RESTART: C:\Users\ABINESH\AppData\Local\Programs\Python\Python37\smart solutions for railways\QR\_Scan  
ner.py  
{'\_id': '2022-11-12,15:27:37', '\_rev': '1-c31e1d7186bd8ccf6104252d74a3e8bd', 'Name': 'Arun', 'Age': 21  
, 'Mobile Number': 753216497, 'Boarding': 'Hyderabad', 'Destination': 'Chennai'}  
Not a Valid Ticket

## 9.1. PERFORMANCE METRICS



# **ADVANTAGES & DISADVANTAGES**

## **10. ADVANTAGES & DISADVANTAGES**

### **10.1 ADVANTAGES**

- λ Openness – compatibility between different system modules, potentially from different vendors;
- λ Orchestration – ability to manage large numbers of devices, with full visibility over them.
- λ Dynamic scaling – ability to scale the system according to the application needs, through resource virtualization and cloud operation
- λ Automation – ability to automate parts of the system monitoring application, leading to better performance and lower operation costs.

### **10.2. DISADVANTAGES**

- λ Approaches to flexible, effective, efficient, and low-cost data collection for both railway vehicles and infrastructure monitoring, using regular trains
- λ Data processing, reduction, and analysis in local controllers, and subsequent sending of that data to the cloud, for further processing
- λ Online data processing systems, for real-time monitoring, using emerging
- λ communication technologies
- λ Integrated, interoperable, and scalable solutions for railway systems preventive maintenance.

# CONCLUSION



## **11. CONCLUSION**

Accidents occurring in Railway transportation system cost a large number of lives. So this system helps us to prevent accidents and giving information about faults or cracks in advance to railway authorities. So that they can fix them and accidents cases become less. This project is cost effective. By using more techniques they can be modified and developed according to their applications. By this system many lives can be saved by avoiding accidents. The idea can be implemented in large scale in the long run to facilitate better safety standards for rail tracks and provide effective testing infrastructure for achieving better results in the future.

# **FUTURE SCOPE**

## **12. FUTURE SCOPE**

In future CCTV systems with IP based camera can be used for monitoring the visual videos captured from the track. It will also increase security for both passengers and railways. GPS can also be used to detect exact location of track fault area, IP cameras can also be used to show fault with the help of video. Locations on Google maps with the help of sensors can be used to detect in which area track is broken.

# APPENDIX

## **13. APPENDIX**

### **13.1. SOURCE PROGRAM**

#### **LOCATION TRACKER:**

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

myConfig = {
    "identity": {
        "orgId": "ojhlri",
        "typeId": "GPS",
        "deviceId": "12345"
    },
    "auth": {
        "token": "12345678"
    }
}

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, 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 ()
```

## **PYTHON CODE FOR QR SCANNER:**

```
import cv2
import numpy as np
import time
import pyzbar.pyzbar as pyzbar
from ibmcloudant.cloudant_v1 import CloudantV1
from ibmcloudant import CouchDbSessionAuthenticator
from ibm_cloud_sdk_core.authenticators import BasicAuthenticator

authenticator = BasicAuthenticator('apikey-
267fe83f32ec443b804f3c76d688d2e3',
'b1456cbe3c712c2d555623fe092c074478521354')
service = CloudantV1(authenticator=authenticator)

service.set_service_url('https://4f470103-dd2c-4f50-94e8-3439395a7935-
bluemix.cloudant.com')

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='booking',
            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()

```

## 13.2 GITHUB LINK

<https://github.com/IBM-EPBL/IBM-Project-3047-1658498553>

## 13.3 DEMO LINK

[https://drive.google.com/file/d/1S2F-prciLIp0tscZ0r1m4cfbqhEVhrsR/view?usp=share\\_link](https://drive.google.com/file/d/1S2F-prciLIp0tscZ0r1m4cfbqhEVhrsR/view?usp=share_link)