

Develop the web application using Node-RED Steps Followed

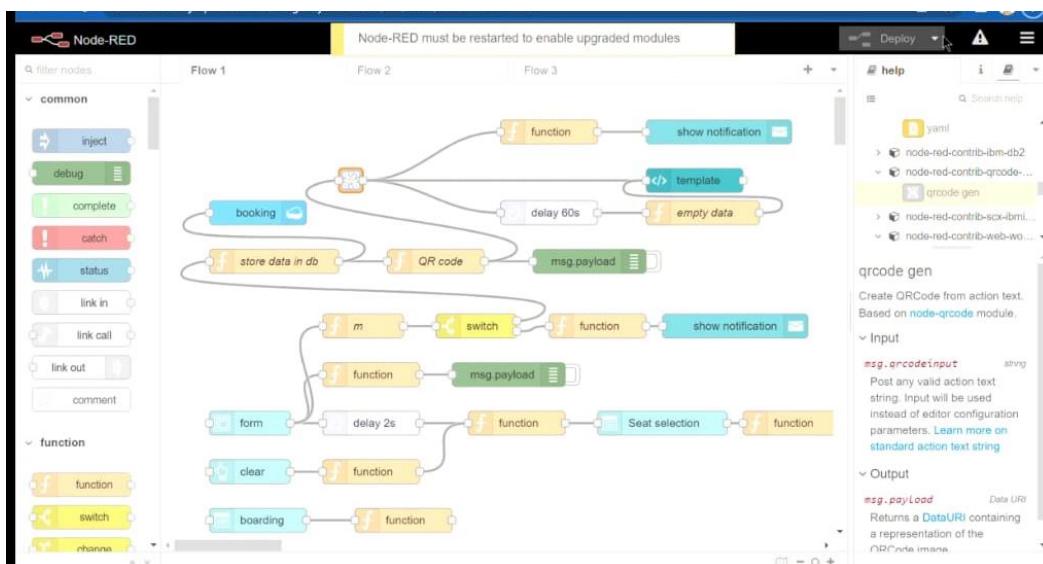
Date	10 November 2022
Team ID	PNT2022TMID01176
Project Name	Project - Smart Solution For Railways

Develop The Web Application Using Node-RED

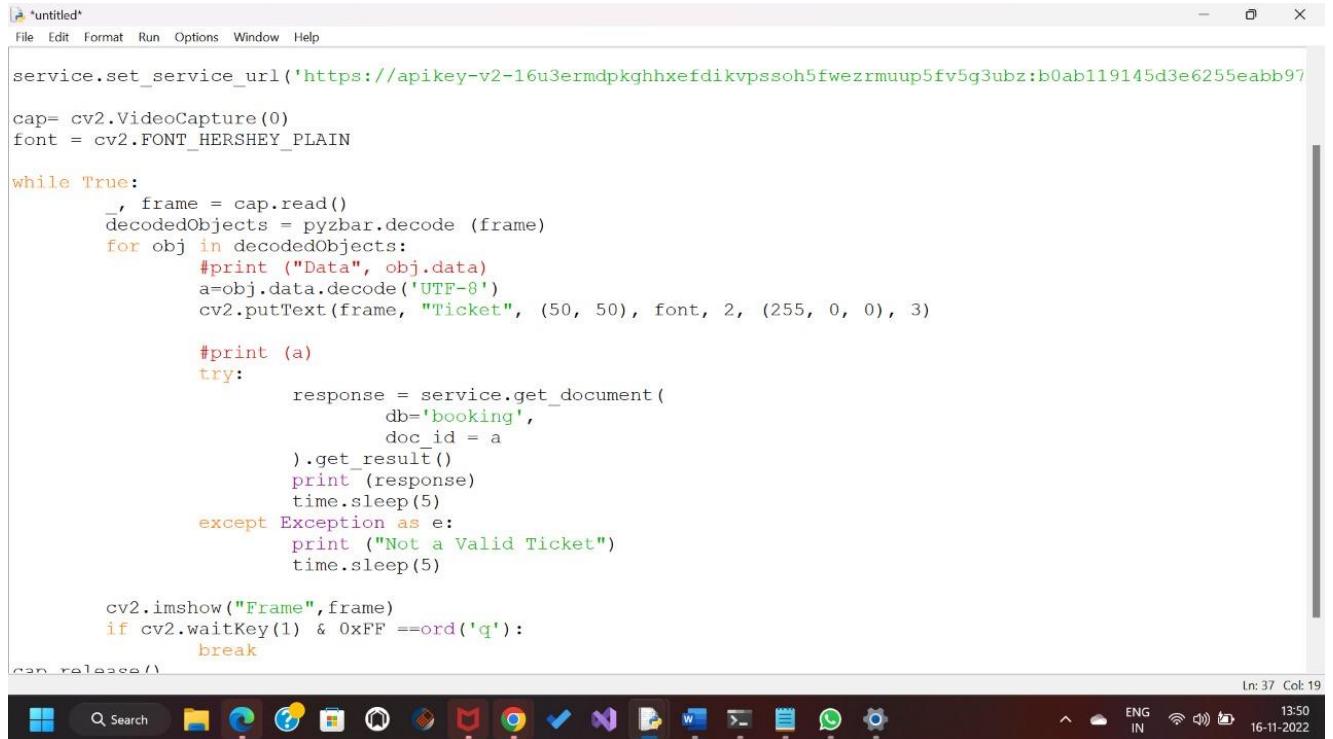
Location Tracking:

- A web application is developed using node red to track the live location of the train by receiving the latitude and longitude value of the location through GPS module.
- The received values will be updated in the IBM Watson IoT Platform through a python code.
- A node red is used to obtain the value from the IBM Watson IoT and to locate the values in the Map.
- When the python code is made to Run the Location is updated in IBM Watson IoT Platform with a delay of 3 seconds.

- Open the Node-RED project:



- Added code to get QR Generator:



The screenshot shows a code editor window titled "untitled". The code is written in Python and performs the following tasks:

- Imports the cv2 and pyzbar modules.
- Sets the service URL to a specific API endpoint.
- Creates a VideoCapture object for the default camera (0).
- Specifies a plain Hershey font for text rendering.
- Enters a loop that reads frames from the camera.
- Decodes QR codes from the frames using pyzbar.
- For each decoded object, prints its data and decodes it to UTF-8.
- Places the text "Ticket" at (50, 50) in the frame.
- Prints the decoded data.
- Tries to get a document from a database named "booking" with the ID "a".
- If successful, prints the response and sleeps for 5 seconds.
- If there's an exception, prints "Not a Valid Ticket" and sleeps for 5 seconds.
- Shows the frame using cv2.imshow("Frame", frame).
- Breaks the loop if the user presses 'q'.
- Closes the video capture device.

The status bar at the bottom right indicates "Ln: 37 Col: 19", "13:50", "ENG IN", and the date "16-11-2022".

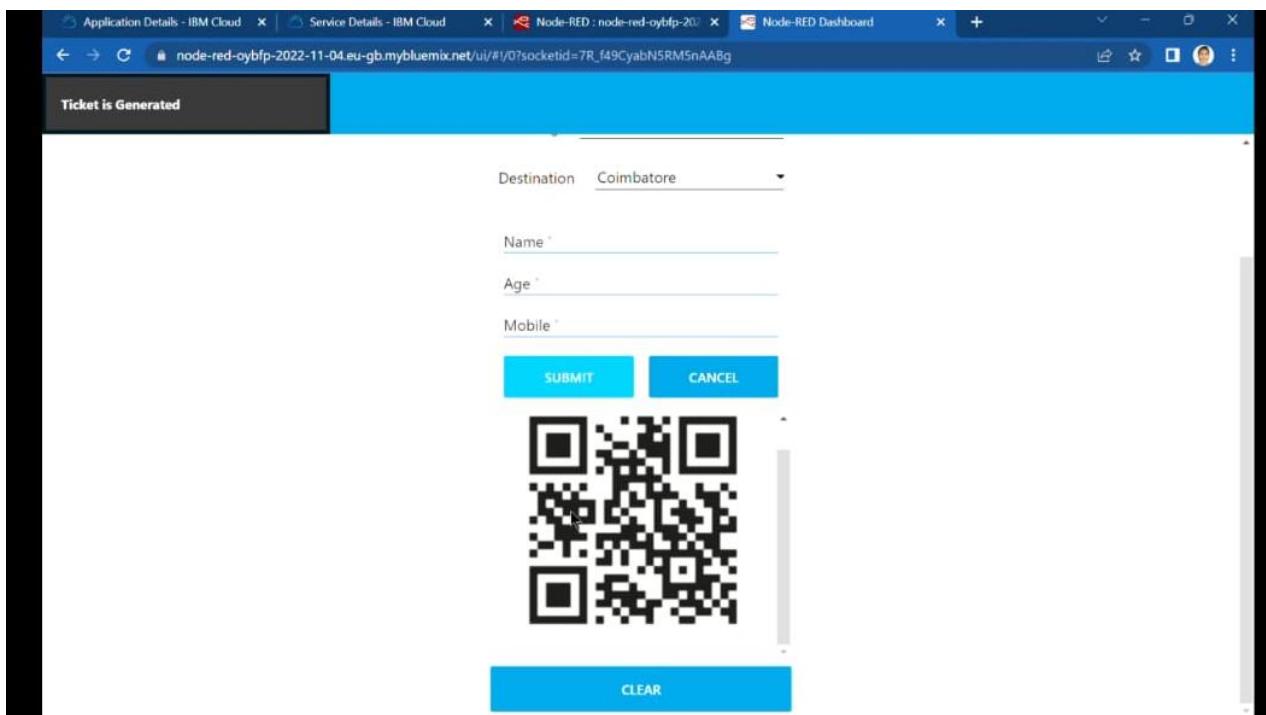
```
File Edit Format Run Options Window Help
service.set_service_url('https://apikey-v2-16u3ermdpkghhxefdikvpssoh5fwezrmuup5fv5g3ubz:b0ab119145d3e6255eabb97
cap= cv2.VideoCapture(0)
font = cv2.FONT_HERSHEY_PLAIN

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)

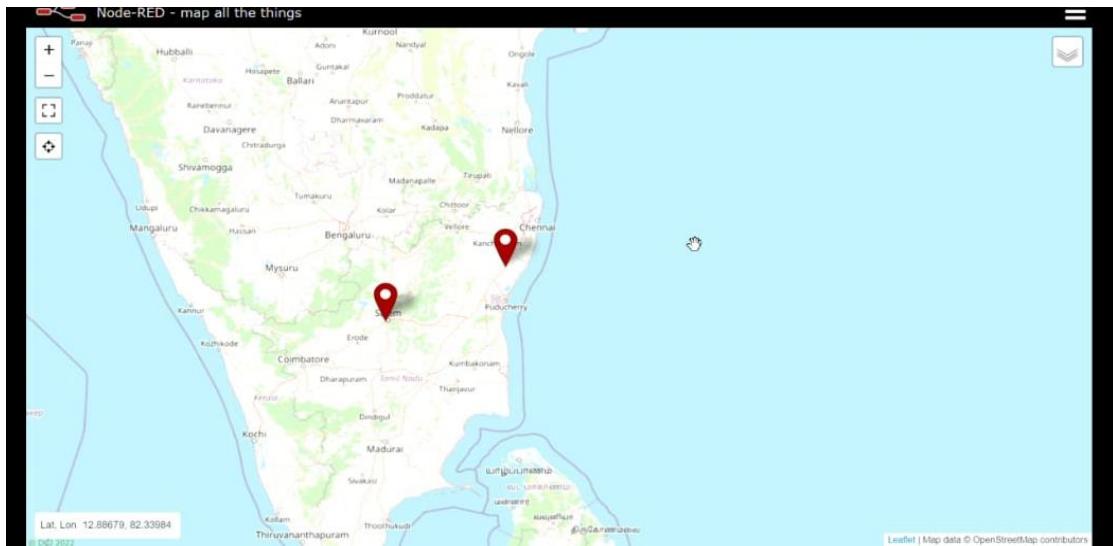
        #print (a)
        try:
            response = service.get_document(
                db='booking',
                doc_id = a
            ).get_result()
            print (response)
            time.sleep(5)
        except Exception as e:
            print ("Not a Valid Ticket")
            time.sleep(5)

    cv2.imshow("Frame",frame)
    if cv2.waitKey(1) & 0xFF ==ord('q'):
        break
cap.release()
```

- Create the QR Generator :



- Showing the destination to users:



Result: Successfully developed the web application using Node-RED