

SPRINT -2 DOCUMENT

Date	11-11-2022
Team ID	PNT2022TMID34503
Project Name	Smart Solutions for Railways

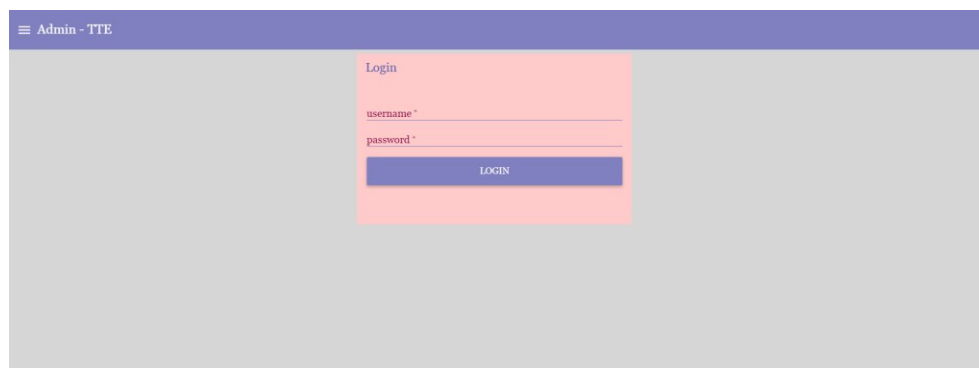
DESCRIPTION:

In this sprint, we have designed a webpage for TTE to verify the train ticket easily. Login page is designed for the TTE to login and after successful login they are redirected for ticket verification. In this form, TTE can scan the QR code and verify the users's personal details, boarding, destination details and seats. If it is valid, the verified users details are displayed in web UI.

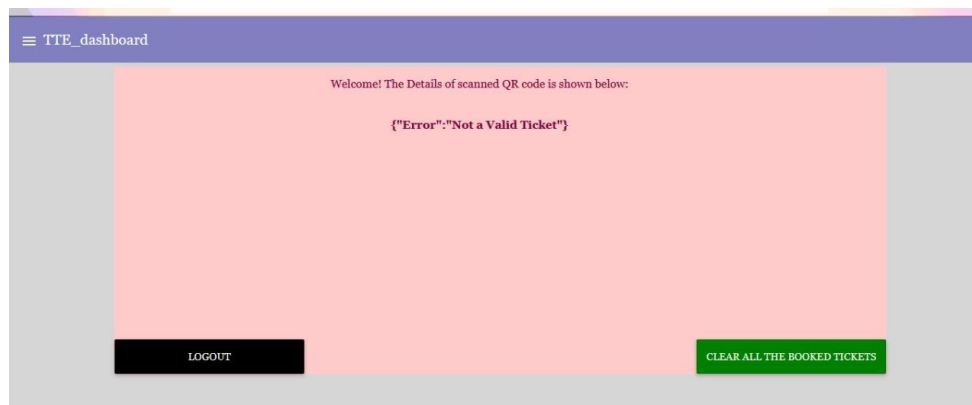
FEATURES:

- 1) Login page
- 2) TTE dashboard which contains clear button
- 3) Ticket verification
- 4) Displaying verified user details in web ui

LOGIN PAGE:



TTE dashboard which contains clear button:



Ticket verification:



```
Published data Successfully: %s {'Error': 'Not a Valid Ticket'}  
  
Not a Valid Ticket
```

Displaying verified user details in web ui:

Python code

```
import cv2 as cv  
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  
import wiotp.sdk.device  
  
authenticator=BasicAuthenticator('apikey-v2-  
1w8tqt2prt3j7qz9d1rgrxhar3w9v43i2359u79ut5jb','86181a38eca19ae487f512b10aca0c80')  
service=CloudantV1(authenticator=authenticator)  
  
service.set_service_url('https://apikey-v2-  
1w8tqt2prt3j7qz9d1rgrxhar3w9v43i2359u79ut5jb:86181a38eca19ae487f512b10aca0c80@9163f25a-10b8-4374-  
a8de-cb92e4357567-bluemix.cloudantnosqldb.appdomain.cloud')  
  
cap = cv.VideoCapture(0)  
font = cv.FONT_HERSHEY_PLAIN  
if not cap.isOpened():  
    print("Cannot open camera")
```

```
exit()
```

```
myConfig = {  
    "identity":{  
        "orgId":"78kd0j",  
        "typeId":"ltx33",  
        "deviceId":"12345678"  
    },  
    "auth":{  
        "token":"LTx@33333"  
    }  
}
```

```
def myCommandCallback(cmd):  
    print("Message received fromIBM 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=response, qos=0, onPublish=None)  
    print("Published data Successfully: %s",response)  
    print("\n")
```

```
while True:  
    ret, frame=cap.read()  
    decodedObjects = pyzbar.decode(frame)  
    if not ret:  
        print("Can't receive frame (stream end?). Exiting ...")  
        break  
    for obj in decodedObjects:
```

```
a=obj.data.decode('UTF-8')
cv.putText(frame, "Ticket", (50,50),font,2,
           (255 ,0, 0),3)
```

```
try:
```

```
    response=service.get_document(
        db='bookingdetails',
        doc_id = a
    ).get_result()
```

```
    print(response)
```

```
    print("\n\n")
```

```
    pub(response)
```

```
    time.sleep(5)
```

```
except Exception as e:
```

```
    response={'Error':'Not a Valid Ticket'}
```

```
    pub(response)
```

```
    print("Not a Valid Ticket")
```

```
    print("\n\n")
```

```
    time.sleep(5)
```

```
cv.imshow("Frame" ,frame)
```

```
if cv.waitKey(1) & 0xFF == ord('q'):
```

```
    break
```

```
client.commandCallback = myCommandCallback
```

```
cap.release()
```

```
cv.destroyAllWindows()
```

```
client.disconnect()
```

NODE-RED FLOW:

