

DELIVERY OF SPRINT-1

TEAM ID	PNT2022TMID25722
PROJECT NAME	SMART SOLUTION FOR RAILWAYS
DATE	06/11/2022

SPRINT-1

```
#include <LiquidCrystal.h>
```

```
LiquidCrystal
```

```
lcd(5,6,8,9,10,11); int redled =
```

```
2; int greenled = 3; int buzzer
```

```
= 4; int sensor = A0;
```

```
int sensorThresh = 400;
```

```
void setup()
```

```
{
```

```
pinMode(redled,OUTPUT);
```

```
pinMode(greenled,OUTPUT);
```

```
pinMode(buzzer,OUTPUT);
```

```
pinMode(sensor,INPUT);
```

```
serial.begin(9600);  
lcd.begin(16,2);  
  
}  
void loop()  
{  
  
    int analogValue = analogRead(sensor);  
    Serial.print(analogvalue);  
    if(analogValue>sensorThresh)  
    {  
  
        digitalWrite(redled,HIGH);  
        digitalWrite(greenled,LOW);  
        tone(buzzer,1000,10000); lcd.clear();  
        lcd.setCursor(0,1);  
        lcd.print("RAILWAYS"); delay(1000);  
        lcd.clear(); lcd.setCursor(0,1);  
        lcd.print("SMART SOLUTIONS FOR  
RAILWAYS");  
        delay(1000);  
    }  
    else  
    {
```

```
        digitalWrite(greenled
        ,HIGH);
        digitalWrite(redled,L
        OW);
        noTone(buzzer);
        lcd.clear();
        lcd.setCursor(0,0);
        lcd.print("SAFE");
        delay(1000);

    lcd.clear();

    lcd.setCursor(0,1);
    lcd.print("ALL CLEAR");

        delay(1000);
    }
}
```

QR CODE:

```
from ibmcloudant import CouchDbSessionAuthenticator
from ibm_cloud_sdk_core.authenticators import
BasicAuthenticator authenticator =
BasicAuthenticator('apikey-v2-
```

```
16u3crmdpkghhxefdikvpssoh5fwezrmuup5fv5g3ubz',  
'b0ab119f45d3e6255eabb978')
```

```
service = CloudantV1(authenticator=authenticator)
```

```
service.set_service_url('https://apikey-v2-
```

```
16u3ermdpkghhxefdikvpssoh5fwezrmuup5fv5g3ubz:b0ab  
119145d3e6255ea bb978e7e2f0')
```

```
cap=
```

```
cv2.VideoCapture(0) font
```

```
=
```

```
cv2.FONT_HERSHEY_PLAI
```

```
N
```

```
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()