PROJECT DEVELOPMENT PHASE

(DELIVERY OF SPRINT-3)

Date	**10 November 2022**
Team ID	**PNT2022TMID13795**
Project Name	**Car Resale Value prediction**
Maximum marks	**4 Marks**

● **Flask Framework:**

```python
# Import Libraries

import pandas as
pd import numpy as
np
from flask import Flask, render_template, Response,
request import pickle
from sklearn.preprocessing import LabelEncoder

app = Flask(_name_)#initiate flask app

def load_model(file='model.sav'):#load the saved model
    return pickle.load(open(file, 'rb'))

@app.route('/')
def index():#main page
    return render_template('car.html')

@app.route('/predict_page')
def predict_page():#predicting page
    return
    render_template('value.html')

@app.route('/predict',
methods=['GET','POST']) def predict():
    reg_year = int(request.args.get('regyear'))
    powerps = float(request.args.get('powerps'))
    kms= float(request.args.get('kms'))
    reg_month =
    int(request.args.get('regmonth'))

    gearbox =
    request.args.get('geartype') damage =
    request.args.get('damage') model =
```

```python
request.args.get('model') brand =
request.args.get('brand')
fuel_type = request.args.get('fuelType')
veh_type =
request.args.get('vehicletype')
```

```python
    new_row = {'yearOfReg':reg_year, 'powerPS':powerps,
               'kilometer':kms, 'monthOfRegistration':reg_month,
               'gearbox':gearbox, 'notRepairedDamage':damage,
               'model':model, 'brand':brand, 'fuelType':fuel_type,
               'vehicletype':veh_type}

    print(new_row)

    new_df                                                    =
        pd.DataFrame(columns=['vehicletype','yearOfReg','gearbox',
        'powerPS','model','kilometer','monthOfRegistration','fuelType'
        , 'brand','notRepairedDamage'])
    new_df             =            new_df.append(new_row,
    ignore_index=True) labels =
['gearbox','notRepairedDamage','model','brand','fuelType','vehicletype'
    ] mapper = {}

    for i in labels:
        mapper[i] = LabelEncoder()
        mapper[i].classes =
np.load(str('classes'+i+'.npy'), allow_pickle=True)
        transform = mapper[i].fit_transform(new_df[i])
        new_df.loc[:,i+'_labels'] = pd.Series(transform, index=new_df.index)
    labeled = new_df[['yearOfReg','powerPS','kilometer','monthOfRegistration']
+ [x+'_labels' for x in labels]]

    X = labeled.values.tolist()
    print('\n\n', X)
    predict = reg_model.predict(X)


    #predict =
    predictions['predictions'][0]['values'][0][0]
    print("Final prediction :",predict)

    return render_template('predict.html',predict=predict)

if_name_=='_main_':
    reg_model = load_model()#load the saved
    model app.run(debug=True)
```

HTML Design Code:

```html
<!DOCTYPE html>
<nav lang="en" dir="ltr">
  <head>
    <style>
      :root {

  --typewriterSpeed: 6s;
}

body
{
    margin:0;


    background-color:peachpuff;
}
h1
{
    position:relative
    ; font-size:4rem;
    position:relative
    ;

}
a{
  text-decoration: none;
}
.bg{

  margin:50px 170px;

}
button{
  margin-left: 650px;
  background-color:
  lightgreen;
}
h1::before,
h1::after
{
    content:"";
    position:absolute;
    top:0;
    bottom:0
    ;
    left:0;
    right:0;
}


h1::before
{
```

```css
    background:peachpuff;
    animation:typewritter 6s steps(28) 2s forwards;
}
```

```css
h1::after
{
    width:0.125em;
    background:black;
    animation:
    typewritter 6s steps(28) 2s
    forwards, blink 730ms steps(28)
    infinite, blinks 12s steps(28)
    forwards;
}
@keyframes blinks
{
    to{                                    background-color:peachpuff;

    }
}

@keyframes typewritter
{
    to{left:100%;}
}

@keyframes blink
{
    to{
    background-color:transparent;
    }
}


.subtitle
{
  display:grid;
  place-content:center
  ; text-align:center;
    color:hsl(0 0%
    0%/0.7);
    font-size:2rem;
    font-weight:400
    ; opacity:0;
    transform: translateY(3rem);
    animation: fade 2s ease 8s
    forwards;


}


@keyframes fade {
  to {
    opacity: 1;
    transform: translateY(0);
  }
}
```

```css
    div {text-align: center;}
</style>
```

```
<meta charset="utf-8">
<title>Car resale value </title>
<link rel="stylesheet" href="../static/css/style.css">
    <link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/4.7.0/css/font-awesome.min.css">

</head>
<body>

  <section class="header">
    <div class="text-box">
<h1 class="bg">CAR RESALE VALUE PREDICTION</h1>
<p class="subtitle">welcome to our webpage</a></p>
<br>
        <button class="subtitle "><a href="./predict_page" style="font-
size:30px" >Check price</a></button>
        <a href='https://postimg.cc/kB3n3Ss8'
target='_blank'><img
src='https://i.postimg.cc/kB3n3Ss8/drift2.jpg'
border='0' style="width:600px"alt='drift2'/></a>


    </div>
   </section>
 </nav>
   </body>
   </body>
 </body>
</htm
  l>
```
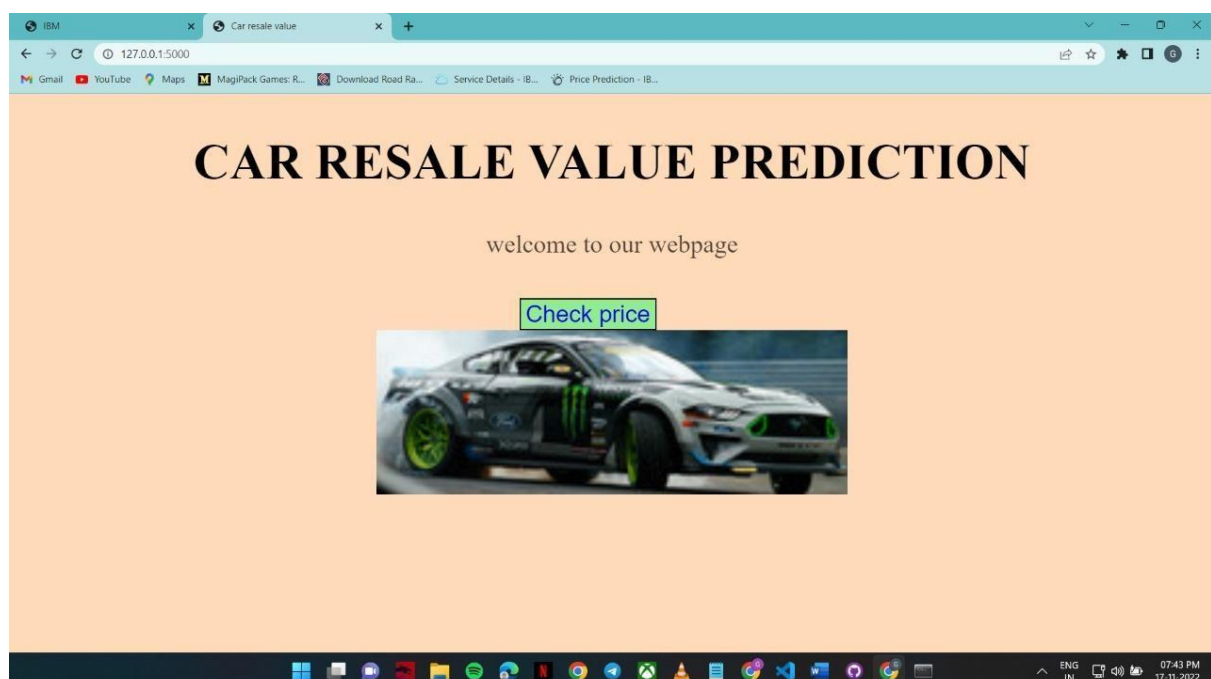
Application Webpage:

IBM | Car Resale Predicted Value | +

← → C ⓘ 127.0.0.1:5000/predict?regyear=2011&regmonth=3&powerps=190&kms=70000&geartype=manual&damage=no&model=navara&brand=nissan&fuelType=diesel&vehicletype=suv&Submi...

M Gmail ▶ YouTube ♥ Maps M MagiPack Games: R... Download Road Ra... Service Details - IB... Price Prediction - IB...
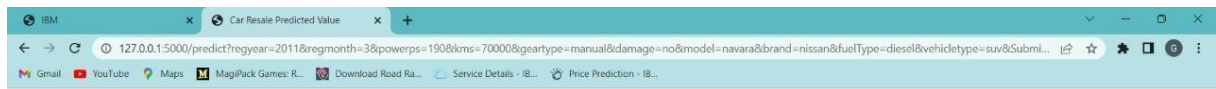
The Predicted Car Resale Value is

₹[16984.07610861]