# Car Price Prediction System

Submitted in partial fulfillment of the requirements of the degree

**BACHELOR OF ENGINEERING** IN **INFORMATION TECHNOLOGY**

By

**Aryan Garate – 13**

**Akash Kulal – 25**

**Sushant Singh – 45**

**Naman Suthar - 46**

Supervisor

**Prof.** Prajkta Khaire





**Department of Information Technology**

Shivajirao S. Jondhale College of Engineering.

Dombivli (E)

(Affiliated to University of Mumbai)

(AY 2021-22)

# CERTIFICATE

This is to certify that the Sensor Lab Mini Project entitled “**Car Price Prediction System”** is a bonafide work of **Aryan Garate (13), Akash Kulal (25), Sushant Singh (45) ,Naman Suthar (49)** submitted to the University of Mumbai in partial fulfillment of the requirement for the award of the degree of **“Bachelor of Engineering”** in **“Information Technology” .**

### (Prof. Prajkta Khaire)

Supervisor

Dr. Savita Sangam Dr. P.R. Rodge

Head of Department Principal

**Code :**

**Ouick Analysis.ipynb :**

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated A screenshot of a computer

Description automatically generatedGraphical user interface

Description automatically generatedA screenshot of a computer

Description automatically generatedGraphical user interface, text, website

Description automatically generatedA screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated

### Application.py

from flask import Flask,render\_template,request,redirect

from flask\_cors import CORS,cross\_origin

import pickle

import pandas as pd

import numpy as np

app=Flask(\_\_name\_\_)

cors=CORS(app)

model=pickle.load(open('LinearRegressionModel.pkl','rb'))

car=pd.read\_csv('Cleaned\_Car\_data.csv')

@app.route('/',methods=['GET','POST'])

def index():

    companies=sorted(car['company'].unique())

    car\_models=sorted(car['name'].unique())

    year=sorted(car['year'].unique(),reverse=True)

    fuel\_type=car['fuel\_type'].unique()

    companies.insert(0,'Select Company')

    return render\_template('index.html',companies=companies, car\_models=car\_models, years=year,fuel\_types=fuel\_type)

@app.route('/predict',methods=['POST'])

@cross\_origin()

def predict():

    company=request.form.get('company')

    car\_model=request.form.get('car\_models')

    year=request.form.get('year')

    fuel\_type=request.form.get('fuel\_type')

    driven=request.form.get('kilo\_driven')

    prediction=model.predict(pd.DataFrame(columns=['name', 'company', 'year', 'kms\_driven', 'fuel\_type'],

                              data=np.array([car\_model,company,year,driven,fuel\_type]).reshape(1, 5)))

    print(prediction)

    return str(np.round(prediction[0],2))

if \_\_name\_\_=='\_\_main\_\_':

    app.run()

**Output :**

**Graphical user interface, application

Description automatically generatedGraphical user interface, application

Description automatically generated**