|  |  |
| --- | --- |
| **TEAM ID** | PNT2022TMID12053 |
| **DATE** | 26/10/2022 |
| **PROJECT NAME** | Intelligent Vehicle Damage Assessment and Cost Estimator for Insurance Companies |
| **TEAM MEMBERS** | Aravintan.T  Diwahar.S.J  Sharan.s  Sneha.Y |

|  |
| --- |
| #Importing  Libraries |
| import re |
| import numpy as np |
| import os |
| from flask import Flask, app, request, render\_template |
| from keras import models |
| from keras.models import load\_model |
| from keras.preprocessing import image |
| from tensorflow.python.ops.gen\_array\_ops import concat |
| from keras.applications.inception\_v3 import preprocess\_input |
| import requests |
| from flask import Flask, request, render\_template, redirect, url\_for |
| from cloudant.client import Cloudant |
| #Create Database |
| client = Cloudant.iam('00cba18f-2150-4961-9102-f29b9aee35de-  bluemix','ht\_ByiEjrGeaitIZJTC-ri5\_8Oq-dxTNHLGho1mpt0d5', connect=True) |
| my\_database = client.create\_database('my\_database') |
| #Loading the Model |
| model1 = load\_model('Model/level.h5') |
| model2 = load\_model('Model/body.h5') |
| app = Flask( name ) |
| @app.route('/') |
| def index(): |
| return render\_template('index.html') |
| @app.route('/index.html') |
| def home(): |
| return render\_template('index.html') |
| @app.route('/register.html') |
| def register(): |
| return render\_template('register.html') |
| @app.route('/afterreg', methods=['POST']) |
| def afterreg(): |
| x = [x for x in request.form.values()] |
| print(x) |
| data = { |
| '\_id': x[1], |
| 'name': x[0], |
| 'psw': x[2] |

|  |
| --- |
| } |
| print(data) |
| query = {'\_id': {'$eq': data['\_id']}} |
| docs = my\_database.get\_query\_result(query) |
| print(docs) |
| print(len(docs.all())) |
| if(len(docs.all())==0): |
| url = my\_database.create\_document(data) |
| response = request.get(url) |
| return render\_template('login.html', pred="Registration Successful,  Please login using your details") |
| else: |
| return render\_template('register.html', pred="You are already a member, Please login using your details") |
| @app.route('/login.html') |
| def login(): |
| return render\_template('login.html') |
| @app.route('/afterlogin', methods=['POST']) |
| def afterlogin(): |
| user = request.form['\_id'] |
| passw = request.form['psw'] |
| print(user,passw) |
| query = {'\_id': {'$eq': user}} |
| docs = my\_database.get\_query\_result(query) |
| print(docs) |
| print(len(docs.all())) |
| if(len(docs.all())==0): |
| return render\_template('login.html', pred="The Username is not  found") |
| else: |
| if((user==docs[0][0]['\_id'] and passw==docs[0][0]['psw'])): |
| return redirect(url\_for('prediction')) |
| else: |
| print('Invalid User') |
| @app.route('/logout.html') |

|  |
| --- |
| def logout(): |
| return render\_template('logout.html') |
| @app.route('/prediction.html') |
| def prediction(): |
| return render\_template('prediction.html') |
| @app.route('/result') |
| def res(): |
| if request.methods=="POST": |
| f=request.files['image'] |
| basepath=os.path.dirname( file ) |
| filepath=os.path.join(basepath,'uploads',f.filename) |
| f.save(filepath) |
| img=image.load\_img(filepath,target\_size=(256,256)) |
| x=image.img\_to\_array(img) |
| x=np.expand\_dims(x,axis=0) |
| img\_data=preprocess\_input(x) |
| prediction1=np.argmax(model1.predict(img\_data)) |
| prediction2=np.argmax(model2.predict(img\_data)) |
| index1=['front','rear','side'] |
| index2=['minor','moderate','severe'] |
| result1 = index1[prediction1] |
| result2 = index2[prediction2] |
| if(result1 == "front" and result2 == "minor"): |
| value = "3000 - 5000 INR" |
| elif(result1 == "front" and result2 == "moderate"): |
| value = "6000 - 8000 INR" |
| elif(result1 == "front" and result2 == "severe"): |
| value = "9000 - 11000 INR" |
| elif(result1 == "rear" and result2 == "minor"): |
| value = "4000 - 6000 INR" |
| elif(result1 == "rear" and result2 == "moderate"): |
| value = "7000 - 9000 INR" |
| elif(result1 == "rear" and result2 == "severe"): |
| value = "11000 - 13000 INR" |
| elif(result1 == "side" and result2 == "minor"): |
| value = "6000 - 8000 INR" |
| elif(result1 == "side" and result2 == "moderate"): |
| value = "9000 - 11000 INR" |
| elif(result1 == "side" and result2 == "severe"): |
| value = "12000 - 15000 INR" |

|  |
| --- |
| else: |
| value = "16000 - 50000 INR" |
| return render\_template('prediction.html',prediction=value) |
| if name ==" main ": |
| app.run(debug = False,port = 8080) |