**CODE FOR EXECUTION OF PROJECT**

# import Dependencies

import os

import numpy as np

import pandas as pd

from six import reraise

import tensorflow as tf

from tensorflow.keras.applications.inception\_v3 import preprocess\_input

from tensorflow.keras.models import load\_model

from tensorflow.keras.preprocessing import image

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

from werkzeug.utils import secure\_filename

# Create a Flask App

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

# load the model

Model\_path = "model\_inceptionV3.h5"

model = load\_model(Model\_path)

itempredictname="NIL"

def search\_and\_read\_text\_file(file\_name):

# Get the current working directory

current\_directory = os.getcwd()

# Check if the folder exists

folder\_path = os.path.join(current\_directory, 'itemdescription')

if not os.path.exists(folder\_path):

print(f"The folder 'itemdescription' does not exist in the current directory.")

return

# Check if the file exists in the folder

file\_path = os.path.join(folder\_path, file\_name)

if not os.path.exists(file\_path):

print(f"The file '{file\_name}' does not exist in the folder 'itemdescription'.")

retdata="The file does not exist in the folder 'itemdescription'."

return

# Read the contents of the file

with open(file\_path, 'r') as file:

contents = file.read()

retdata=contents

print(f"Contents of '{file\_name}':\n{contents}")

return retdata

# Create a function to take and image and predict the class

def model\_predict(img\_path , model):

global itempredictname

print(img\_path)

img = image.load\_img(img\_path , target\_size=(299 , 299))

x = image.img\_to\_array(img)

x = x / 255

x = np.expand\_dims(x , axis = 0)

preds = model.predict(x)

preds = np.argmax(preds , axis = 1)

if preds == 0:

preds = "This Item is Burger"

itempredictname="Burger"

elif preds == 1:

preds = "This Item is Butter Naan"

itempredictname="ButterNaan"

elif preds == 2:

preds = "This Item is Chai"

itempredictname="Chai"

elif preds == 3:

preds = "This Item is Chapati"

itempredictname="Chapati"

elif preds == 4:

preds = "This Item is Chole Bhature"

itempredictname="CholeBhature"

elif preds == 5:

preds = "This Item is Dal Makhani"

itempredictname="DalMakhani"

elif preds == 6:

preds = "This Item is Dhokla"

itempredictname="Dhokla"

elif preds == 7:

preds = "This Item is Fried Rice"

itempredictname="FriedRice"

elif preds == 8:

preds = "This Item is Idli"

itempredictname="Idli"

elif preds == 9:

preds = "This Item is Jalebi"

itempredictname="Jalebi"

elif preds == 10:

preds = "This Item is Kaathi Rolls"

itempredictname="KaathiRolls"

elif preds == 11:

preds = "This Item is Kadai Paneer"

itempredictname="KadaiPaneer"

elif preds == 12:

preds = "This Item is Kulfi"

itempredictname="Kulfi"

elif preds == 13:

preds = "This Item is Masala Dosa"

itempredictname="MasalaDosa"

elif preds == 14:

preds = "This Item is Momos"

itempredictname="Momos"

elif preds == 15:

preds = "This Item is Paani Puri"

itempredictname="PaaniPuri"

elif preds == 16:

preds = "This Item is Pakode"

itempredictname="Pakode"

elif preds == 17:

preds = "This Item is Pav Bhaji"

itempredictname="PavBhaji"

elif preds == 18:

preds = "This Item is Pizza"

itempredictname="Pizza"

else:

preds = "This Item is Samosa"

itempredictname="Samosa"

return preds

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

def index(): # Main Page

global itempredictname

return render\_template('index.html')

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

def uploads():

global itempredictname

if request.method == 'POST':

# Get the File from post request

f = request.files['file']

# Save the file to ./uploads

basepath = os.path.dirname(\_\_file\_\_)

file\_path = os.path.join(basepath , 'uploads' , secure\_filename(f.filename))

f.save(file\_path)

# Make Prediction

preds = model\_predict(file\_path , model)

result = preds

'''

if itempredictname=="Burger":

print("---- BURGER DETECTED -----")

file\_name = itempredictname+'.txt'

print(file\_name)

datareturned=search\_and\_read\_text\_file(file\_name)

result =result+"\n"+datareturned

'''

print("PREDICTED ITEM:",itempredictname)

try:

file\_name = itempredictname+'.txt'

print(file\_name)

datareturned=search\_and\_read\_text\_file(file\_name)

result =result+"\n"+datareturned

except:

result =result+"\n"+" NO INFORMATION FOUND"

return result

return None

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

app.run()