

PROJECT DEVELOPMENT PHASE

Sprint - IV

Date	17 November 2022
Team ID	PNT2022TMID49469
Project Name	AI-Powered Nutrition Analyzer For Fitness Enthusiasts
Maximum Marks	8 Marks

Integration the Deployed Model with Flask

Web Application

App.py

```
from flask import Flask,render_template,request
# Flask-It is our framework which we are going to use to run/serve our application.
#request-for accessing file which was uploaded by the user on our application.
import os
import numpy as np #used for numerical analysis
from tensorflow.keras.models import load_model#to load our trained model
from tensorflow.keras.preprocessing import image
import requests
import mysql.connector

app = Flask(__name__,template_folder="templates") # initializing a flask app
# Loading the model
model=load_model('scanmynutri.h5')
print("Loaded model from disk")

conn=mysql.connector.connect(host="localhost", user="root", password="", database="login")
cursor=conn.cursor()

@app.route('/')# route to display the home page
def home():
    return render_template('home.html')#rendering the home page

@app.route('/image1',methods=['GET','POST'])# routes to the index html
def image1():
    return render_template("image.html")

@app.route('/personalworkout')# route to display the home page
def personalworkout():
```

```

return render_template('personalworkout.html')#rendering the home page

@app.route('/Fleiblediet')# route to display the home page
def Fleiblediet():
    return render_template('Fleiblediet.html')#rendering the home page

@app.route('/login')
def login(): # put application's code here
    return render_template('login.html')

@app.route('/register')
def register():
    return render_template('register.html')

@app.route('/login_validation', methods=['POST'])
def login_validation():
    email=request.form.get('email')
    password=request.form.get('password')

    cursor.execute("""SELECT * FROM `users` WHERE `email` LIKE '{}' AND `password` LIKE
'{}'""".format(email,password))
    users = cursor.fetchall()

    if len(users)>0:
        return render_template('image.html')
    else:
        return render_template('login.html', prediction_text = "1" )

@app.route('/add_user', methods=['POST'])
def add_user():
    name= request.form.get('name')
    email = request.form.get('email')
    password = request.form.get('password')

    cursor.execute("""INSERT INTO `users`(`id`, `name`, `email`, `password`) VALUES
(NULL, '{}', '{}', '{}')""".format(name,email,password))
    conn.commit()
    return render_template('login.html', prediction_text = "0")

@app.route('/predict', methods=['GET', 'POST']) # route to show the predictions in a web UI
def launch():
    if request.method == 'POST':
        f = request.files['file'] # requesting the file
        basepath = os.path.dirname('__file__') # storing the file directory
        filepath = os.path.join(basepath, "uploads", f.filename) # storing the file in uploads folder
        f.save(filepath) # saving the file

```

```

img = image.load_img(filepath, target_size=(64, 64)) # load and reshaping the image
x = image.img_to_array(img) # converting image to an array
x = np.expand_dims(x, axis=0) # changing the dimensions of the image

pred = np.argmax(model.predict(x), axis=1)
print("prediction", pred) # printing the prediction
index = ['Banana', 'Beetroot', 'Blueberry', 'Cauliflower', 'Cherry 1', 'Cocos', 'Eggplan', 'Ginger
Root', 'Grape Blue', 'Grapefruit Pink', 'Guava', 'Kiwi', 'Lemon', 'Mango', 'Onion Red', 'Orange',
'Papaya', 'Pepper Green', 'Pineapple', 'Plum', 'Potato Red', 'Raspberry', 'Strawberry', 'Tomato 1',
'Watermelon', 'burger', 'butter_naan', 'chai', 'chapati', 'chole_bhature', 'dal_makhani', 'dhokla',
'fried_rice', 'idli', 'jalebi', 'kaathi_rolls', 'kadai_paneer', 'kulfi', 'masala_dosa', 'momos', 'paani_puri',
'pakode', 'pav_bhaji', 'pizza', 'samosa']

result = str(index[pred[0]])

x = result
print(x)
result = scanmynutri(result)
print(result)

return render_template("0.html", showcase=(result), showcase1=(x))

def nutrition(index):
    url = "https://calorieninjas.p.rapidapi.com/v1/nutrition"

    querystring = {"query": index}

    headers = {
        'x-rapidapi-key': "5d797ab107mshe668f26bd044e64p1ffd34jsnf47bfa9a8ee4",
        'x-rapidapi-host': "calorieninjas.p.rapidapi.com"
    }

    response = requests.request("GET", url, headers=headers, params=querystring)

    print(response.text)
    return response.json()['items']

if __name__ == "__main__":
    # running the app
    app.run(debug=False)

```