By:Shruthi Mohan	**MLOps-Recommendation System Using Flask**
Date: 23-03-2023	

Aim:
To build a flask model that implements a recommendation system.

Algorithm:

- Import the necessary modules and the necessary datasets.
- Fill the null values present in the data and pre-process the data.
- Create a recommendation model that receives user and input and recommends items and rename the file as model.py.
- Connect mysql using pymysql library to verify the login credentials.
- Create html page to display the output, receive input and a login page and render all the html pages using appropriate functions.
- Pass the input values to python from html to check the login credentials using sql.
- Pass the input for recommendation and print the recommendations in the html page.

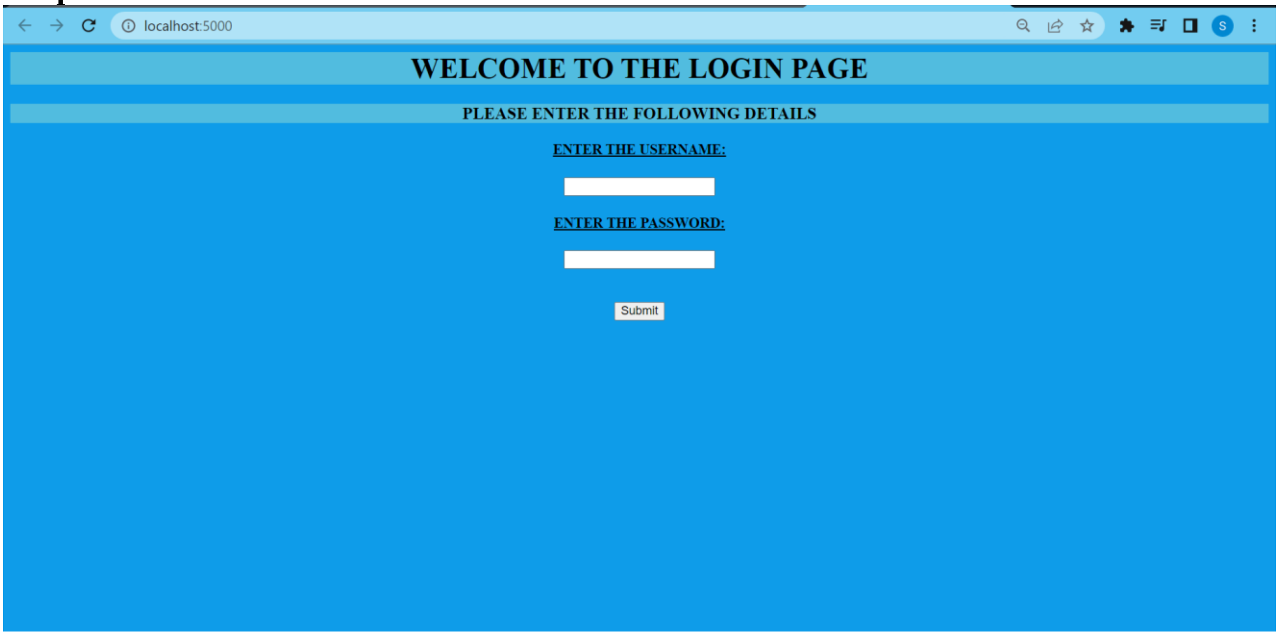
Program:

```
from flask import Flask, request, render_template
#import pickle
from rsmodel import *
import pymysql as pms
app = Flask(__name__)

@app.route("/")
def main():
    return render_template("login_page.html")

@app.route("/check", methods=['Post'])
def check():
    username = request.form['username']
    password = request.form['password']
    #return "Success!"
    f=open('ps.txt','rb')
    conn=pms.connect(host="localhost",port=3306,user="root",password="Ms03#shrusps",db=
"employees")
    cur=conn.cursor() #object for accessing queries
    cur.execute("select * from login")
    for i in cur.fetchall():
        if i[0]==username and i[1]==password:
            return render_template("success.html")
    return render_template("login_page.html",data="INVALID USER CREDENTIALS")

@app.route("/recommend", methods=['Post'])
def recommend():
#    model = pickle.load(open(r'C:\Users\Shruthi
Mohan\OneDrive\Desktop\College_Stuff\Sem4\ML\CIA2- FLASK\Solution1\model.pkl','rb'))
    input_series = request.form['series']
    rec = get_recommendations_new(input_series)
    return render_template("result.html",data=rec)

if __name__=='__main__':
    app.run(host='localhost',port=5000)
```

Recommendation system:

```
import pandas as pd
import numpy as np
#import pickle

series = pd.read_csv(r"C:\Users\Shruthi
Mohan\OneDrive\Desktop\College_Stuff\Sem4\ML\CIA2- FLASK\Solution1\TV Series.csv")
series_data = pd.read_csv(r"C:\Users\Shruthi
Mohan\OneDrive\Desktop\College_Stuff\Sem4\ML\CIA2- FLASK\Solution1\TV Series.csv")

series_data=series_data.iloc[:10000]

#EDA
series_data = series_data.fillna('')
series_data.drop('Runtime',axis=1,inplace=True)

def clean_data(x):
    return str.lower(x.replace(" ", ""))

for i in series_data.columns:
    series_data[i] = series_data[i].apply(clean_data)

def create_soup(x):
    return x['Series Title']+ ' ' + x['Release Year'] + ' ' + x['Genre'] + ' '
+x['Rating']+' '+ x['Cast']+' '+ x['Synopsis']
series_data['soup'] = series_data.apply(create_soup, axis=1)

#tf-idf
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.metrics.pairwise import cosine_similarity
count = CountVectorizer(stop_words='english')
count_matrix = count.fit_transform(series_data['soup'])
cosine_sim = cosine_similarity(count_matrix, count_matrix)

series_data=series_data.reset_index()
indices = pd.Series(series_data.index, index=series_data['Series Title'])

def get_recommendations_new(title):
    global cosine_sim
    title=title.replace(' ','').lower()
    idx = indices[title]

    # Get the pairwise similarity scores of all movies with that series
    sim_scores = list(enumerate(cosine_sim[idx]))

    # Sort the series based on the similarity scores
    sim_scores = sorted(sim_scores, key=lambda x: x[1], reverse=True)

    # Get the scores of the 10 most similar series
    sim_scores = sim_scores[1:11]
    # Get the series indices
    movie_indices = [i[0] for i in sim_scores]

    # Return the top 10 most similar series
    result =  series['Series Title'].iloc[movie_indices]
    result = result.to_frame()
    result = result.set_index(np.arange(1,len(result)+1))
    return result

#pickle.dump(get_recommendations_new,open('rsmodel.pkl','wb'))
```

Output Screenshot:



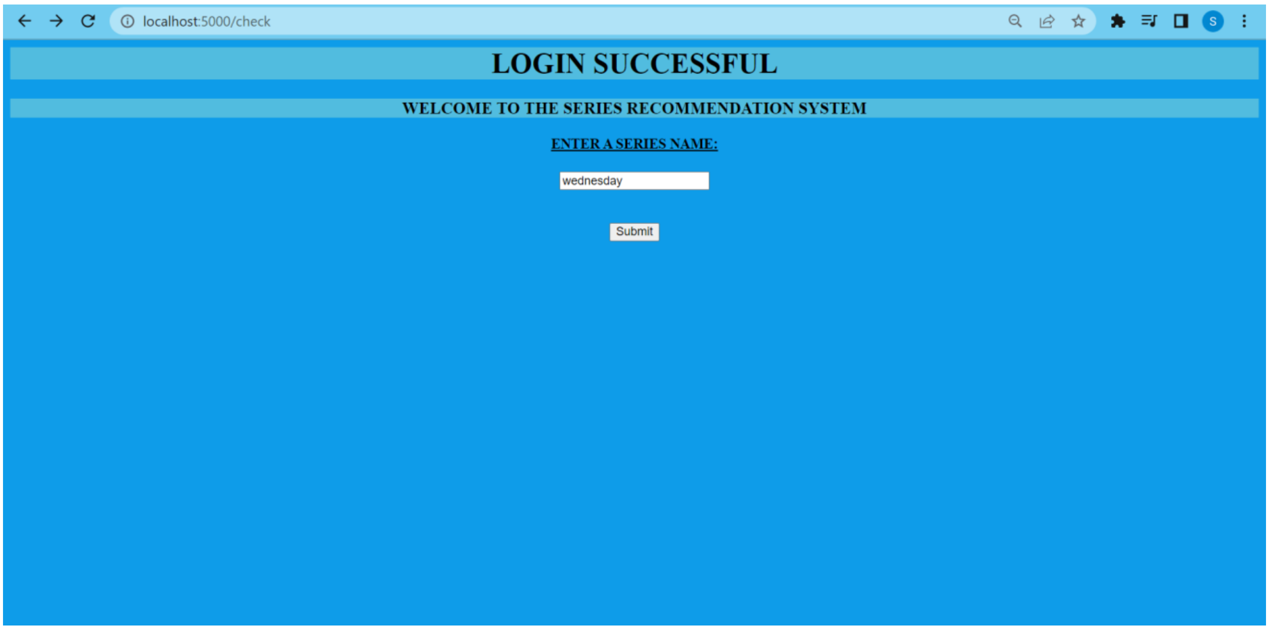
localhost:5000

WELCOME TO THE LOGIN PAGE

PLEASE ENTER THE FOLLOWING DETAILS

ENTER THE USERNAME:

ENTER THE PASSWORD:



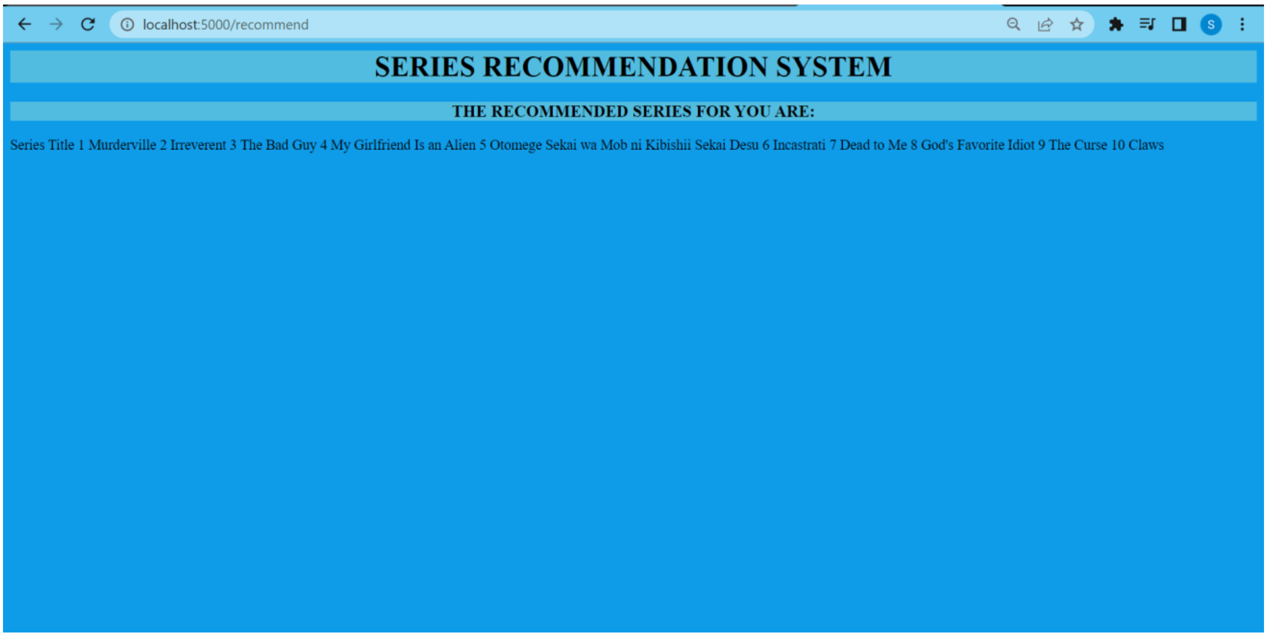
localhost:5000/check

LOGIN SUCCESSFUL

WELCOME TO THE SERIES RECOMMENDATION SYSTEM

ENTER A SERIES NAME:

wednesday



SERIES RECOMMENDATION SYSTEM

THE RECOMMENDED SERIES FOR YOU ARE:

```
Series Title 1 Murderville 2 Irreverent 3 The Bad Guy 4 My Girlfriend Is an Alien 5 Otomege Sekai wa Mob ni Kibishii Sekai Desu 6 Incastrati 7 Dead to Me 8 God's Favorite Idiot 9 The Curse 10 Claws
```

Result:

Hence, the recommendation system is built and implemented using flask.