**Giải thích ý nghĩa code trong app.py**

import streamlit as st

import pickle

import requests

#Hàm fetch\_poster(movie\_id) là một hàm tùy chỉnh được sử dụng trong ứng dụng Streamlit để lấy URL poster của một bộ phim cụ thể

#từ The Movie Database (TMDb) API. Hàm này trả về URL đầy đủ cho hình ảnh poster của bộ phim được chỉ định bởi movie\_id.

def fetch\_poster(movie\_id):

     url = "https://api.themoviedb.org/3/movie/{}?api\_key=c7ec19ffdd3279641fb606d19ceb9bb1&language=en-US".format(movie\_id)

     data=requests.get(url)

     data=data.json()

     poster\_path = data['poster\_path']

     full\_path = "https://image.tmdb.org/t/p/w500/"+poster\_path

     return full\_path

movies = pickle.load(open("C:/Users/THUY TRANG/Pictures/NCKH/movie\_recommender\_system-main/movie\_recommender\_system-main/movies\_list.pkl", 'rb'))

similarity = pickle.load(open("C:/Users/THUY TRANG/Pictures/NCKH/movie\_recommender\_system-main/movie\_recommender\_system-main/similarity.pkl", 'rb'))

movies\_list=movies['title'].values

#Câu lệnh movies\_list=movies['title'].values được sử dụng để tạo ra một danh sách các bộ phim trong tập dữ liệu.

#Danh sách này được lấy từ cột “title” của DataFrame movies.

st.header("Movie Recommender System")

import streamlit.components.v1 as components

#Câu lệnh import streamlit.components.v1 as components được sử dụng để import module components của Streamlit.

#Module này cung cấp các công cụ để tạo các thành phần tùy chỉnh trong ứng dụng Streamlit.

imageCarouselComponent = components.declare\_component("image-carousel-component", path="frontend/public")

#Câu lệnh trên được sử dụng để khai báo một thành phần tùy chỉnh trong ứng dụng Streamlit.

#Thành phần này được gọi là “image-carousel-component” và được định nghĩa trong thư mục “frontend/public”.

imageUrls = [

    fetch\_poster(1632),

    fetch\_poster(299536),

    fetch\_poster(17455),

    fetch\_poster(2830),

    fetch\_poster(429422),

    fetch\_poster(9722),

    fetch\_poster(13972),

    fetch\_poster(240),

    fetch\_poster(155),

    fetch\_poster(598),

    fetch\_poster(914),

    fetch\_poster(255709),

    fetch\_poster(572154)

    ]

#Câu lệnh trên là một phần của ứng dụng Streamlit để tạo ra một hệ thống gợi ý phim.

#Biến imageUrls chứa một danh sách các URL hình ảnh của các bộ phim được gợi ý.

imageCarouselComponent(imageUrls=imageUrls, height=200)

selectvalue=st.selectbox("Select movie from dropdown", movies\_list)

#Hàm imageCarouselComponent được sử dụng để hiển thị một carousel hình ảnh.

#Biến imageUrls chứa danh sách các URL hình ảnh của các bộ phim được gợi ý.

#Biến selectvalue chứa tên bộ phim được chọn từ menu thả xuống.

def recommend(movie):

    index=movies[movies['title']==movie].index[0]

    distance = sorted(list(enumerate(similarity[index])), reverse=True, key=lambda vector:vector[1])

    #Đoạn code trên là một phần của ứng dụng Streamlit để tạo ra một hệ thống gợi ý phim.

    #Dòng lệnh distance = sorted(list(enumerate(similarity[index])), reverse=True, key=lambda vector:vector[1])

    # có chức năng tính toán độ tương đồng giữa các bộ phim trong tập dữ liệu.

    # Cụ thể, đầu tiên, vị trí của bộ phim được chọn từ menu thả xuống được xác định trong tập dữ liệu

    # bằng cách sử dụng movies[movies['title']==movie].index[0]. Sau đó, độ tương đồng giữa bộ phim này và

    # các bộ phim khác trong tập dữ liệu được tính toán bằng cách sử dụng similarity[index].

    # Kết quả trả về là một danh sách các cặp (vị trí, độ tương đồng) được liệt kê theo thứ tự giảm dần của độ tương đồng.

    # Cuối cùng, danh sách này được sắp xếp theo thứ tự giảm dần của độ tương đồng bằng cách sử dụng sorted và lưu vào biến distance.

    recommend\_movie=[]

    recommend\_poster=[]

    for i in distance[1:6]:

        movies\_id=movies.iloc[i[0]].id

        recommend\_movie.append(movies.iloc[i[0]].title)

        recommend\_poster.append(fetch\_poster(movies\_id))

    return recommend\_movie, recommend\_poster

    #Đoạn code trên là một phần của ứng dụng Streamlit để tạo ra một hệ thống gợi ý phim.

    # Nó có chức năng lấy danh sách các bộ phim tương tự dựa trên bộ phim được chọn từ menu thả xuống.

    # Cụ thể, đầu tiên, vị trí của bộ phim được chọn từ menu thả xuống được xác định trong tập dữ liệu

    # bằng cách sử dụng new\_df[new\_df['title']==movie].index[0]. Sau đó, độ tương đồng giữa bộ phim này

    # và các bộ phim khác trong tập dữ liệu được tính toán bằng cách sử dụng similarity[movie\_index].

    # Cuối cùng, danh sách các bộ phim được lưu vào biến movies\_list.

if st.button("Show Recommend"):

    movie\_name, movie\_poster = recommend(selectvalue)

    col1,col2,col3,col4,col5=st.columns(5)

    with col1:

        st.text(movie\_name[0])

        st.image(movie\_poster[0])

    with col2:

        st.text(movie\_name[1])

        st.image(movie\_poster[1])

    with col3:

        st.text(movie\_name[2])

        st.image(movie\_poster[2])

    with col4:

        st.text(movie\_name[3])

        st.image(movie\_poster[3])

    with col5:

        st.text(movie\_name[4])

        st.image(movie\_poster[4])

#Đoạn code trên là một phần của ứng dụng Streamlit để tạo ra một hệ thống gợi ý phim.

# Khi người dùng nhấn vào nút “Show Recommend”, danh sách các bộ phim được lấy từ hàm recommend sẽ được hiển thị trong một bảng

# gồm năm cột và một số hàng, mỗi hàng chứa tên của một bộ phim và hình ảnh của nó. Mỗi cột chứa thông tin về một bộ phim khác nhau.