import streamlit as st  
import pickle  
import pandas as pd  
import requests  
  
def fetch\_poster(movie\_id):  
 response = requests.get('https://api.themoviedb.org/3/movie/{}?api\_key=d1aa405518bcd86047c245cdc2bfe4e5'.format(movie\_id))  
 data = response.json()  
 return "https://image.tmdb.org/t/p/w500/" + data['poster\_path']  
  
  
  
def recommend(movie):  
 movie\_index = movies[movies['title'] == movie].index[0]  
 distances = similarity[movie\_index]  
 movies\_list = sorted(list(enumerate(distances)), reverse=True, key=lambda x: x[1])[1:6]  
  
 recommended\_movies = []  
 recommended\_movies\_posters = []  
 for i in movies\_list:  
 movie\_id = movies.iloc[i[0]].movie\_id  
 #fetch poster from API  
 recommended\_movies.append(movies.iloc[i[0]].title)  
 recommended\_movies\_posters.append(fetch\_poster(movie\_id))  
 return recommended\_movies,recommended\_movies\_posters  
  
  
movies\_dict = pickle.load(open('movie\_dict.pkl','rb'))  
movies= pd.DataFrame(movies\_dict)  
  
  
st.title('Movie Reccomender System')  
  
similarity = pickle.load(open('similarity.pkl','rb'))  
  
  
option = st.selectbox(  
'Enter movie name?',  
 movies['title'].values)  
  
if st.button('Recommend'):  
 names,posters= recommend(option)  
  
 col1, col2, col3, col4, col5 = st.columns(5)  
 with col1:  
 st.text(names[0])  
 st.image(posters[0])  
  
 with col2:  
 st.text(names[1])  
 st.image(posters[1])  
  
 with col3:  
 st.text(names[2])  
 st.image(posters[2])  
  
 with col4:  
 st.text(names[3])  
 st.image(posters[3])  
  
  
 with col5:  
 st.text(names[4])  
 st.image(posters[4])