app.py

```python
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=438d0ad84ccfae1a3da1eb0b2be8dffa&language=en-US'.format(movie_id))
    data = response.json()
    print(data)
    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_poster=[]
    for i in movies_list:
        movie_id = movies.iloc[i[0]].movie_id
        recommended_movies.append(movies.iloc[i[0]].title)
        recommended_movies_poster.append(fetch_poster(movie_id))
    return recommended_movies,recommended_movies_poster

movie_dict =  pickle.load(open('movie_dict.pkl','rb'))
movies = pd.DataFrame(movie_dict)
similarity =  pickle.load(open('similarity.pkl','rb'))

st.title('FILMELPER')
selected_movie_name = st.selectbox(
    'MOVIE RECOMMENDER SYSTEM',
movies['title'].values)

if st.button('Recommend'):
    names,posters = recommend(selected_movie_name)
    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])
    # for i in recommendations:
    #     st.write(i)
```