A MINI PROJECT REPORT

ON

" Movie Recommendation System"

Submitted to

SAVITRIBAI PHULE PUNE UNIVERSITY

in completion of

Lab Practical-II

(B.E Computer Engineering)

BY

Group No :- 37

ANIKET GHOLVE	**405B027**
RITIK RAJ	**405B064**
OM SHARMA	**405B073**
RAJA KUMAR	**405A069**

Department of Computer Engineering

Sinhgad College of Engineering, Pune-41

Accredited by NAAC with grade 'A'

YEAR 2021-2022

CERTIFICATE

Sinhgad Technical Education Society,
Department of Computer Engineering
Sinhgad College of Engineering, Pune-41
Accredited by NAAC with grade 'A'

"PROJECT TITLE"

Submitted to

SAVITRIBAI PHULE PUNE UNIVERSITY

in completion of

Lab Practical-II

(B.E Computer Engineering)

BY

ANIKET GHOLVE	**405B027**
RITIK RAJ	**405B064**
OM SHARMA	**405B073**
RAJA KUMAR	**405A069**

Shweta Kambare
Department of Computer Engineering

M.P.Wankhade
Department of Computer Engineering

Dr. S.D. Lokhande
Principal
SCOE, Pune

CONTENTS

Acknowledgement

It is indeed a great pleasure and moment of immense satisfaction for us to present a project report on "Movie Recommendation System" amongst a wide panorama that provided us inspiring guidance and encouragement, we take the opportunity to thanks those who gave us their indebted assistance. We wish to extend our cordial gratitude with profound thanks to our internal guide for his everlasting guidance. It was his inspiration and encouragement which helped us in completing our project.

Our sincere thanks and deep gratitude to Head of Department, Dr. M.P Wankhade and other faculty member; but also to all those individuals involved both directly and indirectly for their help in all aspect of the project.

At last but not least we express our sincere gratitude to our Institute's Principal Dr. Dr.S.D.Lokhande, for providing us infrastructure and technical environment.

- ANIKET GHOLVE

- RITIK RAJ

- OM SHARMA

- RAJA KUMAR

- (GROUP NO 37)

Abstract

Over the past years, the internet has broadened the horizon of various domains to interact and share meaningful information. As it is said that everything has its pros and cons therefore, along with the expansion of domain comes information overload and difficulty in extraction of data. To overcome this problem the recommendation system plays a vital role. It is used to enhance the user experience by giving fast and coherent suggestions. This paper describes an approach which offers generalized recommendations to every user, based on movie popularity and/or genre. Content-Based Recommender System is implemented using various deep learning approaches. This paper also gives an insight into problems which are faced in content-based recommendation system and we have made an effort to rectify them.

1. Background And Basics:

Advancement in technology is reaching new heights every day and due to which we can see enormous growth in information. To deal with such large data we use machine learning that automates analytical model building [1]. The early classification of machine learning is divided into three broad categories: Supervised learning, Unsupervised learning and Reinforcement learning [2]. We use computers to make predictions to help us achieve better results using various computational statistics. Tasks can be performed without being explicitly programmed to do so [3]. It becomes a tedious task to extract the relevant information. Search engines solve the problem to some extent but it does not solve the personalization problem. Recommendation System framework plays a vital role in today's internet surfing, be it buying a product from an e-commerce site or watching a movie on some video-on-demand service [4]. In our everyday life, we depend on recommendations given by other people either by word of mouth or reviews of general surveys. People often use recommender systems over the web to make decisions for the items related to their choice. Recommendation systems are software tools and techniques whose goal is to make useful and sensible recommendations to a collection of users for items or products that might interest them [5]. In other words, the recommender system or recommendation systems belongs to a class of information filtering system that aims at predicting the 'preference' or 'rating' given to an item.

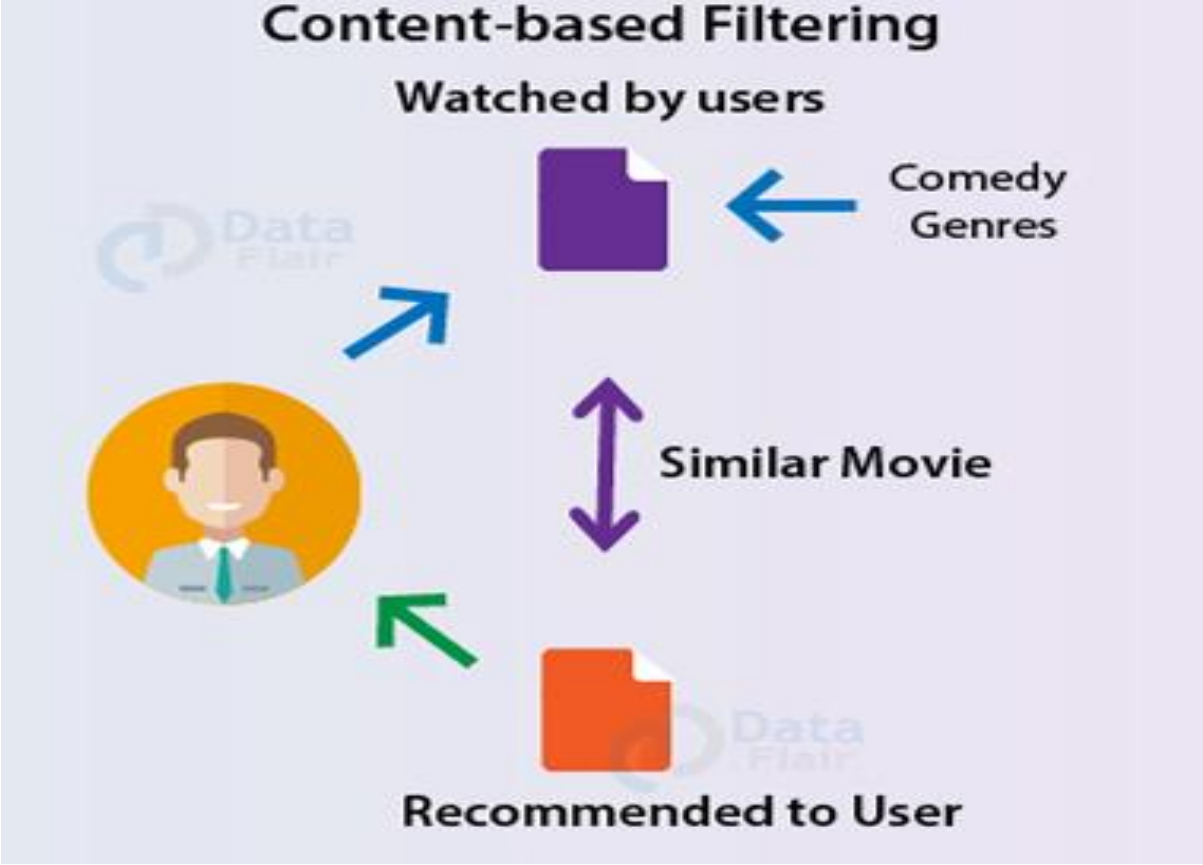
2. PROBLEM STATEMENT

Given a set of users with their previous ratings for a set of movies, can we predict the rating they will assign to a movie they have not previously rated? Ex. "Which movie will you like" given that you have seen X-Men, X-Men II, X-Men : The Last Stand and users who saw these movies also liked "X-Men Origins : Wolverine"

3. SYSTEM REQUIREMENTS:

Jupyter, Matplotlib, Numpy, Pandas, Sklearn, 64 bit OS, 8 GB

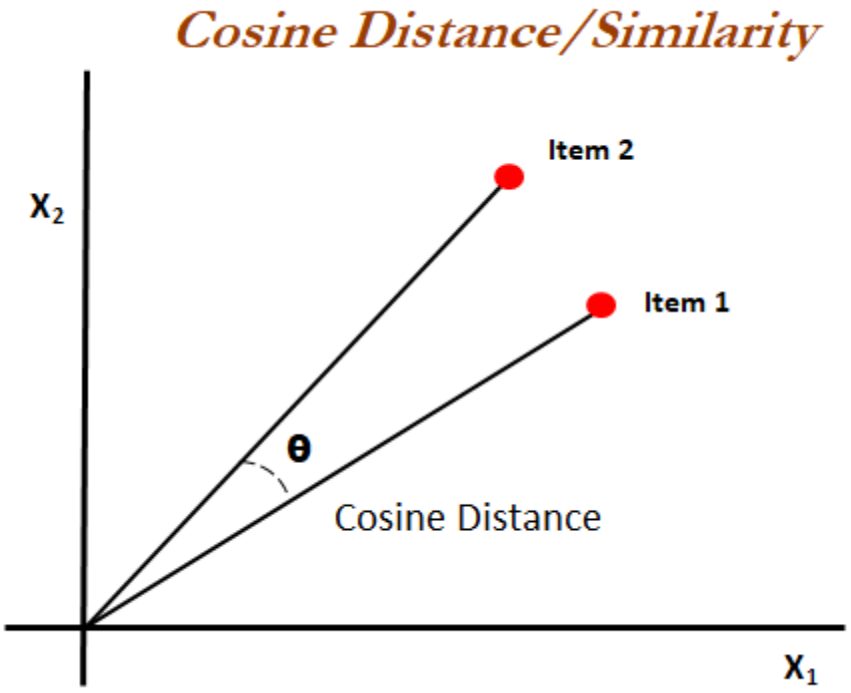
4. Use case diagram



5.Method Used

COSINE SIMILARITY

Cosine similarity among two objects measures the angle of cosine between the two objects. It compares two documents on a normalized scale. It can be done by finding the dot product between the two identities



As the above diagram shows, the angle between v1 and v2 is. Lesser the angle between the two vectors more is the similarity. It means if the angle between two vectors is small, they are almost alike each other and if the angle between the two vectors is large then the vectors are very different from each other

6.Software Testing

Selenium:

Selenium is a free (open source) automated testing suite for web applications across different browsers and platforms. Selenium is a suite ofsoftware tools to automate Web Browsers. It is an Open source suite of tools mainly used for Functional and Regression Test Automation.
Selenium is a free (open source)automated testing suite for web applications across different browsers and platforms.It is quite similar to HPQuick Test Pro (QTP now UFT)
Selenium focuses on automating web-based applications. Testing done using a Selenium tool is usually referred as Selenium Testing.

Selenium IDE:

Selenium IDE (Integrated Development Environment) is primarily a record/ runtool that a test case developer uses to develop Selenium Test cases. Selenium IDE is an easy to use tool from the Selenium Test Suite and can even be used by someone new to developing automated test cases for their web applications. One does not require any special setup to get started with Selenium IDE. You just need to add the extension of your specific browser. Selenium IDE provides you with a GUI (Graphical User Interface) for easily recording your interactions with the Website. Selenium IDE allows a user or a test case developer to create the test cases and test suites and edit it later as per their requirements. The development environment also provides the capability of converting test cases to different programming languages, which makes it easier for the user and does not mandate the need for

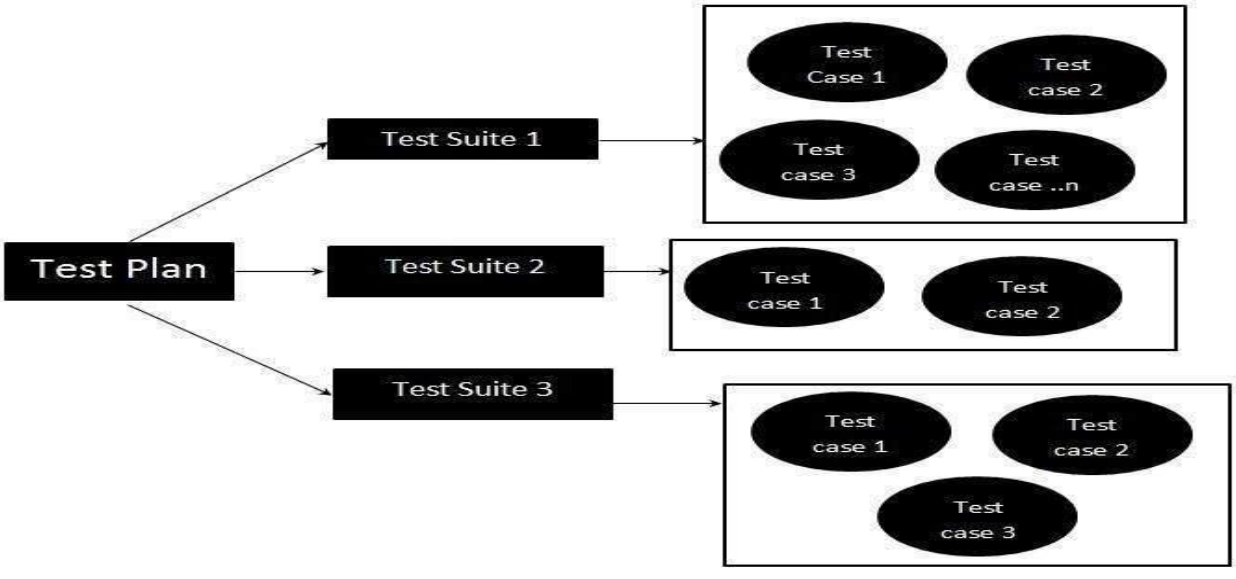
knowing a specific programming language.Sample Screenshots of application

7. Manual Testing

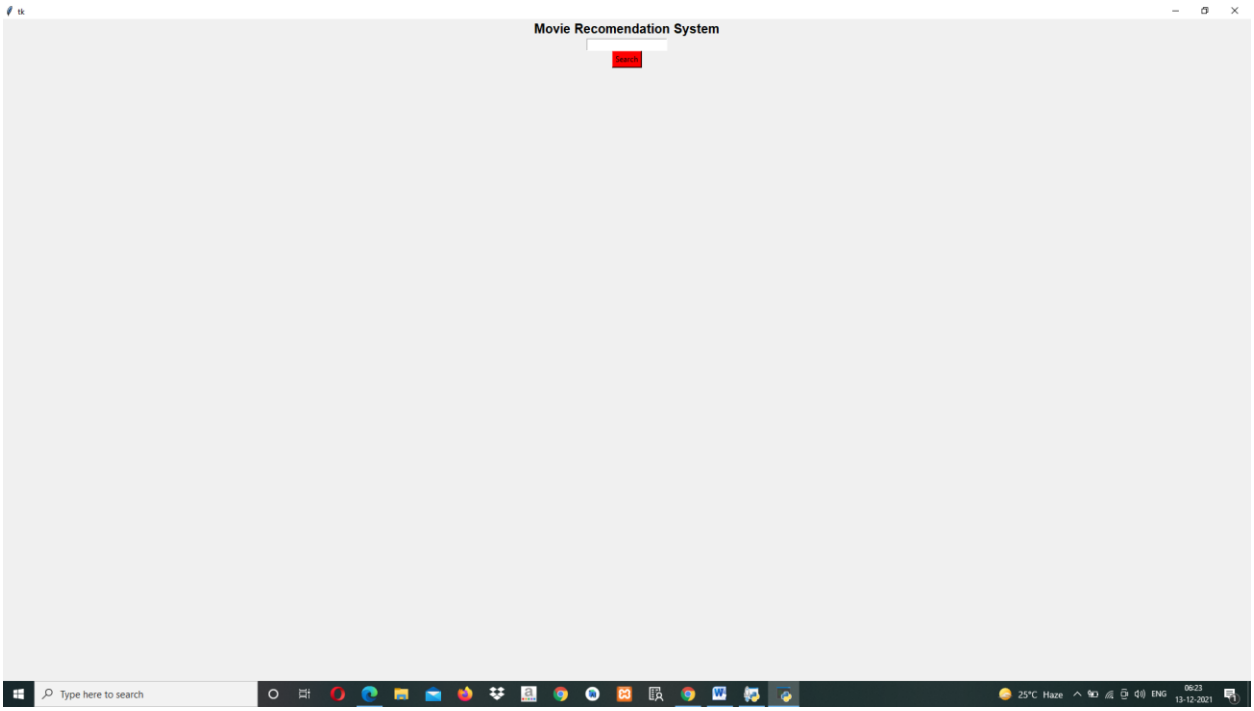
Manual Testing is a type of software testing in which test cases are executed manually by a tester without using any automated tools. The purpose of Manual Testing is to identify the bugs, issues, and defects in the software application. Manual software testing is the most primitive technique of all testing types and it helps to findcritical bugs in the software application.

Any new application must be manually tested before its testing can be automated. Manual Software Testing requires more effort but is necessary to check automation feasibility. Manual Testing concepts does not require knowledge of any testing tool. One of the Software Testing Fundamentals is "100% Automation is not possible". This makes Manual Testing imperative.

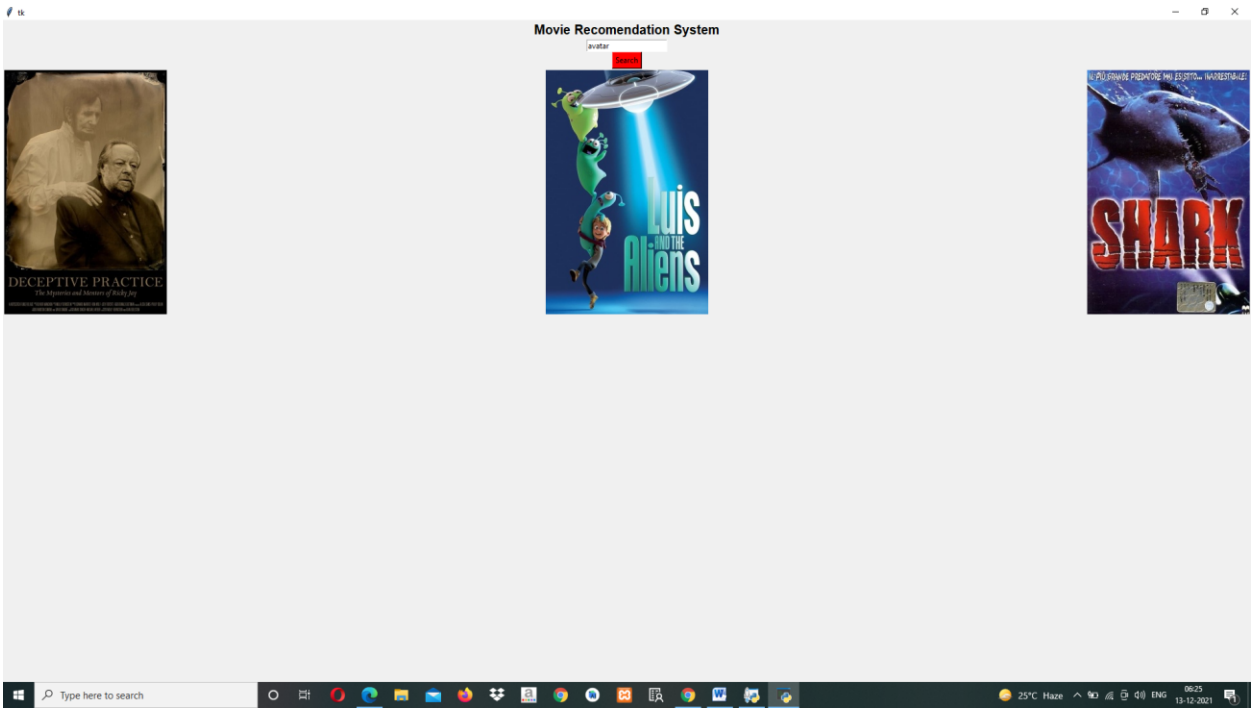
A test plan document is prepared that acts as a guide to the testing process in order to have the complete test coverage.



Before Recommendation:



After Recommendation:



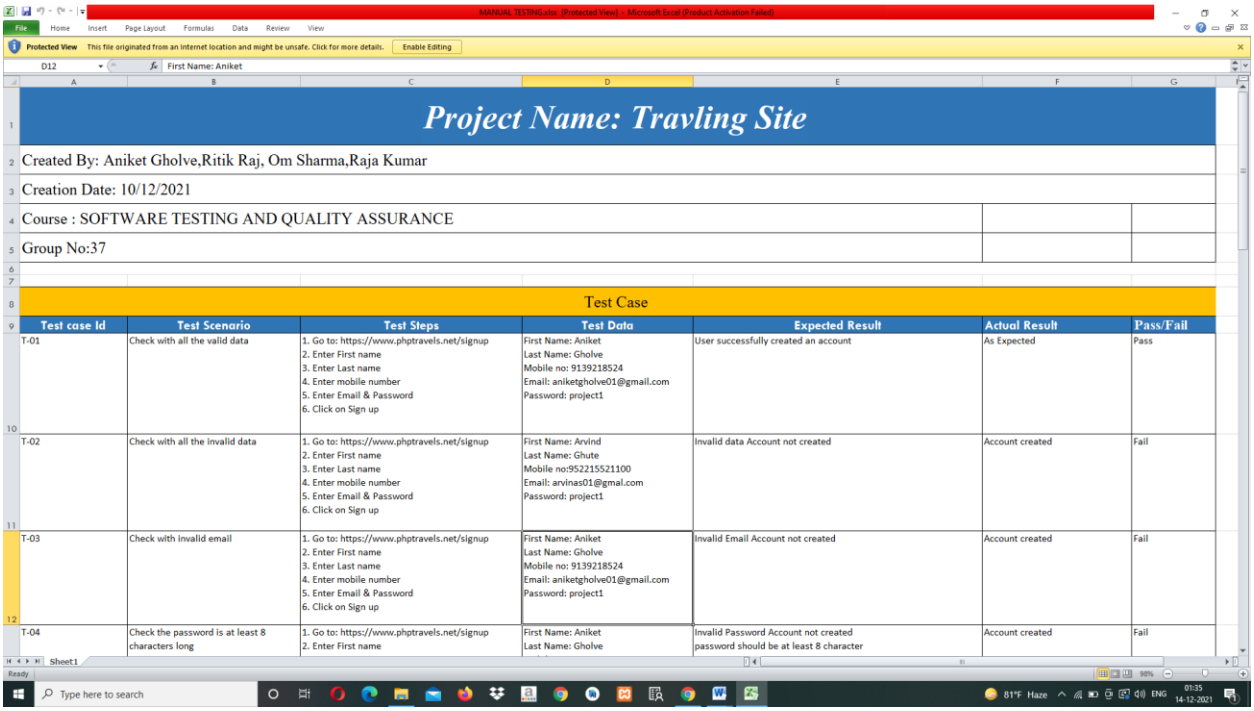
Source Code:

```
from tkinter import *
import pandas as pd
import numpy as np
import wget
import requests
import pprint
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.metrics.pairwise import cosine_similarity
from PIL import Image, ImageTk
import cv2

def Recomend():
    def get_title_from_index(index):
        return df[df.index == index]["movie_title"].values[0]
    def get_index_from_title(title):
        title=title.lower()
        try:
            str1=df[df.movie_title==title]["index"].values[0]
            return str1
        except Exception as e:
            txt.insert(0.0,"no movie in data base")
    df=pd.read_csv("new_data.csv")
    features=['genres','director_name','actor_1_name','actor_2_name','actor_3_name']

    for feature in features:
        df[feature]=df[feature].fillna('')
    def combine_row(row):
        return row['genres']+" "+row['director_name']+" "+row['actor_1_name']+" "+row['actor_2_name']+" "+row['actor_3_name'];

    df["combine_features"]=df.apply(combine_row,axis=1)
    cv=CountVectorizer()
    count_matrix=cv.fit_transform(df["combine_features"])
    cosine_sim=cosine_similarity(count_matrix)
    movie_user_likes=moviename.get()


    movie_index=get_index_from_title(movie_user_likes)

    similar_movies=list(enumerate(cosine_sim[movie_index]))
    sorted_similar_movie=sorted(similar_movies,key=lambda x:x[1] , reverse=True)
    i=0
    for movie in sorted_similar_movie:
        ins=get_title_from_index(movie[0])
        if(i>0):
            photo(ins,i)
        i=i+1
        if i>3:
            break;



def photo(mname,i):
    api_key = "c6d9b9c2ffe1197d725e93bfdcf4325c"
    api_base_url = f"https://api.themoviedb.org/3"
    endpoint_path = f"/search/movie"
    search_query = mname
    endpoint = f"{api_base_url}{endpoint_path}?api_key={api_key}&query={search_query}"
    # print(endpoint)
    r = requests.get(endpoint)
    # pprint.pprint(r.json())
    if r.status_code in range(200, 299):
```

```
        if len(results) > 0:
            # print(results[0].keys())
            movie_ids = set()
            for result in results:
                _id = result['id']
                # print(result['title'], _id)
                movie_ids.add(_id)
            # print(list(movie_ids))
    for movie_id in movie_ids:
        api_version = 3
        api_base_url = f"https://api.themoviedb.org/{api_version}"
        endpoint_path = f"/movie/{movie_id}/images"
        endpoint = f"{api_base_url}{endpoint_path}?api_key={api_key}"
        r = requests.get(endpoint)
    if r.status_code in range(200, 299):
       data = r.json()
       fetch=data['posters']
       url=fetch[0]['file_path']
       print(url)
       image_url=f"https://image.tmdb.org/t/p/w500{url}"
       image_filename = wget.download(image_url)
       print('Image Successfully Downloaded: ', image_filename)
       str1="D:\\Recomendation System\\"+image_filename
       image=cv2.imread(str1)
       scale_percentage=.50
       wid=int(image.shape[1]*scale_percentage)
       hei=int(image.shape[0]*scale_percentage)
       dimension=(wid,hei)
       resized=cv2.resize(image,dimension,interpolation=cv2.INTER_AREA)
       cv2.imwrite(image_filename,resized)
       image = Image.open(str1)
       photo = ImageTk.PhotoImage(image)
       label = Label(root, image = photo)
       label.image = photo
       if i==1:
           label.grid(row=7,sticky=EW)
       elif i==2:
           label.grid(row=7,sticky=W)
       elif i==3:
           label.grid(row=7,sticky=E)
```

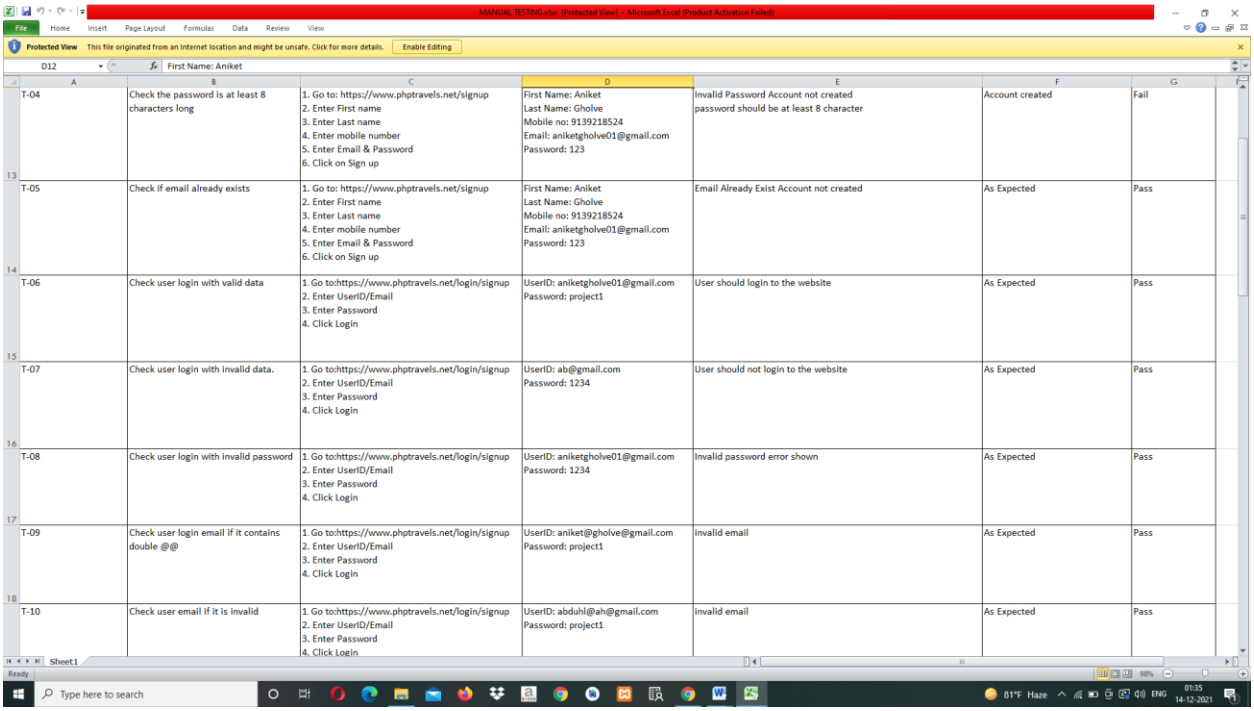
```
root= Tk()
root.geometry("1200x900")
SSSSSSSSSSSSSSmoviename=StringVar()
label=Label(root, text="Movie Recomendation System",font=('Aerial 15 bold'))
label.grid(row=1, column=0)
label.grid_rowconfigure(1, weight=1)
movie=Entry(root,textvariable=moviename)
movie.grid(row=2,column=0)
button=Button(bg="red",text="Search",command=Recomend)
button.grid(row=4,column=0)
root.grid_rowconfigure(1, weight=0)
root.grid_columnconfigure(0, weight=1)
root.mainloop()
```

Manual Testing:



Project Name: Travling Site						
Created By: Aniket Gholve,Ritik Raj, Om Sharma,Raja Kumar						
Creation Date: 10/12/2021						
Course : SOFTWARE TESTING AND QUALITY ASSURANCE						
Group No:37						
Test Case						
Test case Id	**Test Scenario**	**Test Steps**	**Test Data**	**Expected Result**	**Actual Result**	**Pass/Fail**
T-01	Check with all the valid data	1. Go to: https://www.phptravels.net/signup 2. Enter First name 3. Enter Last name 4. Enter mobile number 5. Enter Email & Password 6. Click on Sign up	First Name: Aniket Last Name: Gholve Mobile no: 9139218524 Email: aniketgholve01@gmail.com Password: project1	User successfully created an account	As Expected	Pass
T-02	Check with all the invalid data	1. Go to: https://www.phptravels.net/signup 2. Enter First name 3. Enter Last name 4. Enter mobile number 5. Enter Email & Password 6. Click on Sign up	First Name: Arvind Last Name: Ghute Mobile no:952215521100 Email: arvinas01@gmal.com Password: project1	Invalid data Account not created	Account created	Fail
T-03	Check with invalid email	1. Go to: https://www.phptravels.net/signup 2. Enter First name 3. Enter Last name 4. Enter mobile number 5. Enter Email & Password 6. Click on Sign up	First Name: Aniket Last Name: Gholve Mobile no: 9139218524 Email: aniketgholve01@gmail.com Password: project1	Invalid Email Account not created	Account created	Fail
T-04	Check the password is at least 8 characters long	1. Go to: https://www.phptravels.net/signup 2. Enter First name	First Name: Aniket Last Name: Gholve	Invalid Password Account not created password should be at least 8 character	Account created	Fail

Test cases sample 1



T-04	Check the password is at least 8 characters long	1. Go to: https://www.phptravels.net/signup 2. Enter First name 3. Enter Last name 4. Enter mobile number 5. Enter Email & Password 6. Click on Sign up	First Name: Aniket Last Name: Gholve Mobile no: 9139218524 Email: aniketgholve01@gmail.com Password: 123	Invalid Password Account not created password should be at least 8 character	Account created	Fail
T-05	Check if email already exists	1. Go to: https://www.phptravels.net/signup 2. Enter First name 3. Enter Last name 4. Enter mobile number 5. Enter Email & Password 6. Click on Sign up	First Name: Aniket Last Name: Gholve Mobile no: 9139218524 Email: aniketgholve01@gmail.com Password: 123	Email Already Exist Account not created	As Expected	Pass
T-06	Check user login with valid data	1. Go to:https://www.phptravels.net/login/signup 2. Enter UserID/Email 3. Enter Password 4. Click Login	UserID: aniketgholve01@gmail.com Password: project1	User should login to the website	As Expected	Pass
T-07	Check user login with invalid data.	1. Go to:https://www.phptravels.net/login/signup 2. Enter UserID/Email 3. Enter Password 4. Click Login	UserID: ab@gmail.com Password: 1234	User should not login to the website	As Expected	Pass
T-08	Check user login with invalid password	1. Go to:https://www.phptravels.net/login/signup 2. Enter UserID/Email 3. Enter Password 4. Click Login	UserID: aniketgholve01@gmail.com Password: 1234	Invalid password error shown	As Expected	Pass
T-09	Check user login email if it contains double @@	1. Go to:https://www.phptravels.net/login/signup 2. Enter UserID/Email 3. Enter Password 4. Click Login	UserID: aniket@gholve@gmail.com Password: project1	invalid email	As Expected	Pass
T-10	Check user email if it is invalid	1. Go to:https://www.phptravels.net/login/signup 2. Enter UserID/Email 3. Enter Password 4. Click Login	UserID: abduhl@ah@gmail.com Password: project1	invalid email	As Expected	Pass

Test cases sample 2

AUTOMATION TESTING:

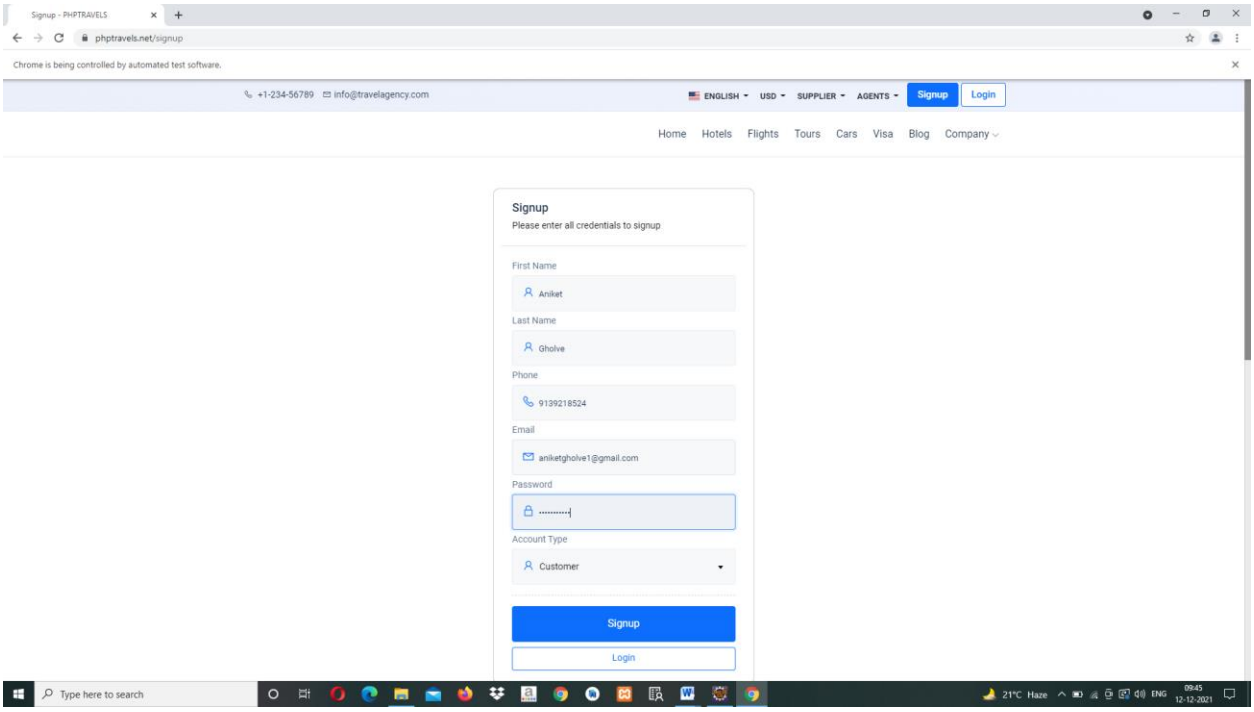


Fig. 1: Signup Up Test Case

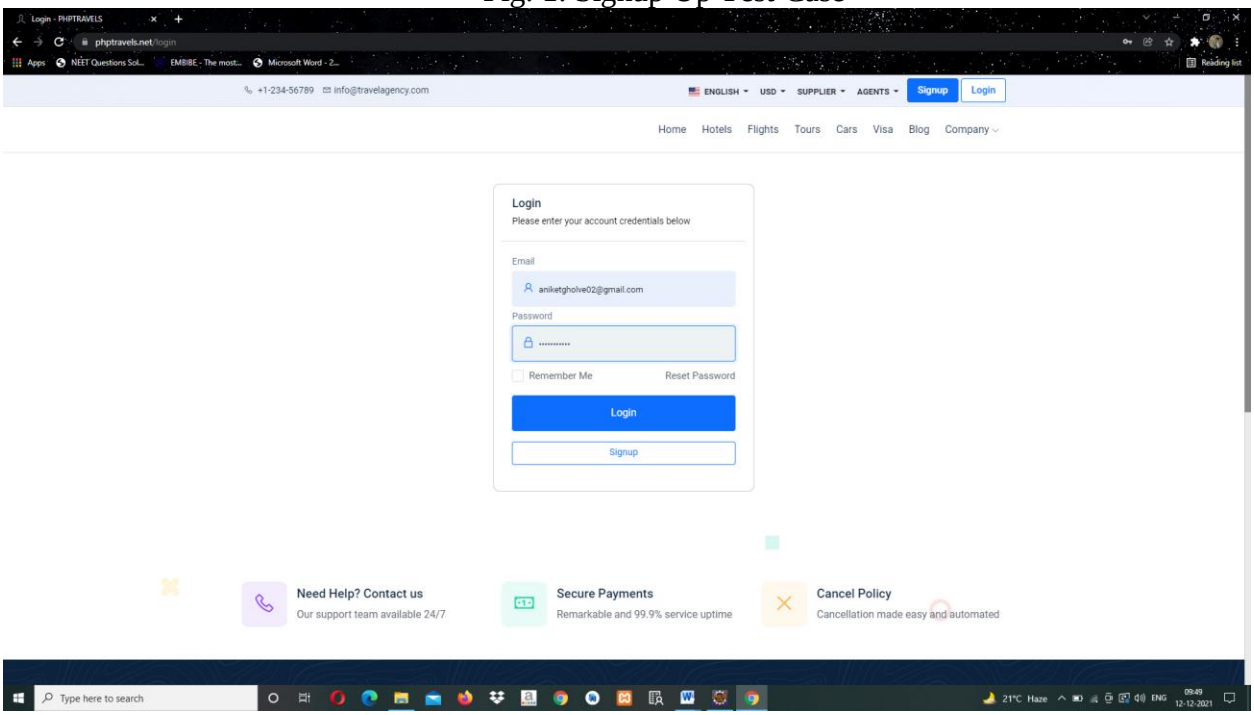


Fig. 2 : Login Test Case

REFERENCES

Gediminas Adomavicius and Alexander Tuzhilin. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. Knowledge and Data Engineering, IEEE Transactions on, 17(6):734–749, 2005.

2. Ricardo Baeza-Yates, Berthier Ribeiro-Neto, et al. Modern information retrieval, volume 463. ACM Press New York, 1999

3. ShumeetBaluja, Rohan Seth, D Sivakumar, Yushi Jing, Jay Yagnik, Shankar Kumar, Deepak Ravichandran, and Mohamed Aly. Video suggestion and discovery for youtube: taking random walks through the view graph. In Proceedings of the 17th international conference on World Wide Web, pages 895–904. ACM, 2008.

4. Xu Hailing, Wu Xiao, Li Xiaodong, and Yan Baoping. Comparison study of internet recommendation system. Journal of Software, 20(2):350–362, 2009.

5. T. E. D. Mining, "Enhancing teaching and learning through educational data mining and learning analytics: An issue brief," in Proceedings of a conference on advanced technology for education, 2012.

6. Nakagawa and T. Ito, "An implementation of a knowledge recommendation system based on similarity among users' profiles," in Sice 2002. roceedings of the Sice Conference, 2002, pp. 326–327 vol.1.

7. T. K. Quan, I. Fuyuki, and H. Shinichi, "Improving the accuracy of recommender system by clustering items based on the stability of user similarity," in International Conference on Computational Intelligence for Modelling Control and Automation, 2006, p. 61

8. M. Muozorganero, G. A. Ramezgonzlez, P. J. Muozmerino, and C. D Kloos, "A collaborative recommender system based on space-time similarities," vol.

9, no. 3, pp. 81–87, 2010. 9. B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Item-based collaborative filtering recommendation algorithms," in Proceedings of the 10th international conference on World Wide Web. ACM, 2001, pp. 285–295. 10. G. Wang, "Survey of personalized recommendation system," Computer Engineering & Applications, 2012.

CONCLUSION:

Successfully implemented Movie Recommendation System using cosine similarity and have done Manual testing and Automation Testing