

CINEPHILE CORNER - MOVIE MANAGEMENT SYSTEM

A MINI-PROJECT REPORT

Submitted by

EASHAAN MANOHAR

200701070

HARRISH VIJAY G

200701080

in partial fulfillment of the award of the degree

of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING



**RAJALAKSHMI
ENGINEERING COLLEGE**
An AUTONOMOUS Institution
Affiliated to ANNA UNIVERSITY, Chennai

RAJALAKSHMI ENGINEERING COLLEGE, THANDALAM

JUNE - 2022

BONAFIDE CERTIFICATE

Certified that this project **“CINEPHILE CORNER – MOVIE MANAGEMENT SYSTEM”** is the bonafide work of **“Eashaan Manohar 200701070, Harrish Vijay G 200701080”** who carried out the project work under my supervision.

SIGNATURE

Mrs. JANANEE V

SUBJECT INCHARGE

ASSISTANT PROFESSOR

Dept. of Computer Science and
Engineering,

Rajalakshmi Engineering College,
Chennai

This mini project report is submitted for the viva voce examination to be held on

INTERNAL EXAMINER

EXTERNAL EXAMINER

ABSTRACT

Cinema is an excellent escape from reality and is highly instrumental in giving us a break from our humdrum life. It is an excellent media to convey powerful messages to the society. In today's time people are bombarded with huge content, leaving them confused on which movies to watch. Thus, there is an increased need to maintain a database system containing movie details.

The system houses the details of the movies with all the necessary attributes and the same is updated into the database in the backend. Thus this eases the choice of the users on which movie to watch and also the movies will get their due recognition.

SQLite is used to store the DB of Movie details and Python is used for programming the working front end. This necessarily doesn't need any kind of security authentication as it's supposed to be free for all to use and doesn't specifically contain any sensitive or confidential information.

The Administrator can only login to make the changes in the Data Base and manage it. The database contains the movie ID, movie name, release date, director, cast, budget and rating. The application interacts with the user to get the details of the movie of his/her choice and the information will be stored in the database which can be later viewed, updated, deleted and cleared from the database.

ACKNOWLEDGEMENT

We express our sincere thanks to our beloved and honorable chairman **MR. S. MEGANATHAN** and the chairperson **DR. M. THANGAM MEGANATHAN** for their timely support and encouragement.

We are greatly indebted to our respected and honorable principal **Dr. S.N. MURUGESAN** for his able support and guidance

No words of gratitude will suffice for the unquestioning support extended to us by our Head of The Department **Dr.P. REVATHY M.E Ph.D.**, and our Academic Head **Mr. K.E. NARAYANA**, for being ever supporting force during our project work

We also extend our sincere and hearty thanks to our internal guide **Mrs. JANANEE.V**, for her valuable guidance and motivation during the completion of this project.

Our sincere thanks to our family members, friends and other staff members of computer science engineering.

1. Eashaan Manohar 200701070

2. Harrish Vijay G 200701080

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO
	ABSTRACT	3
1	INTRODUCTION	8
1.1	INTRODUCTION	8
1.2	SCOPE OF THE WORK	8
1.3	PROBLEM STATEMENT	8
1.4	AIM AND OBJECTIVES OF THE PROJECT	9
2	SYSTEM SPECIFICATIONS	10
2.1	HARDWARE SPECIFICATIONS	10
2.2	SOFTWARE SPECIFICATIONS	10
3	MODULE DESCRIPTION	11
4	SYSTEM DESIGN	12
4.1	ARCHITECTURE DIAGRAM	12
4.2	E-R MODEL	13
5	CODING	14
6	SCREENSHOTS	19
7	CONCLUSION AND FUTURE ENHANCEMENT	22
	REFERENCES	23

LIST OF FIGURES

FIGURE NO.	TITLE	PAGE NO.
4.1	ARCHITECHTURE DIAGRAM	12
4.2	E-R MODEL	

LIST OF TABLES

TABLE NO.	TITLE	PAGE NO.
2.1	HARDWARE SPECIFICATIONS	10
2.2	SOFTWARE SPECIFICATIONS	10

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

Cinema is an integral medium of communication which has the potential to grab the attention of millions across the globe. There is a plethora of genres for every culture and tongue that has walked the earth. Cinema illuminates the mind and can be a balm as well as antitode to various stress and problems that envelope us.

1.2 SCOPE OF THE WORK

Through the movie management system, we can scale this to bigger and complex cineplexes as cinema has now become a major earner in the entertainment industry with lots of scope and potential for growth and development. The complex task of keeping a record of all the movies is a tedious task and this is where database system can help in providing an efficient solution

1.3 PROBLEM STATEMENT

The need for this project is due to the fact that people nowadays are bombarded with thousands of movies, and they are left with confusion and might forget to keep track of all the movies out there. In such cases, the film might go unrecognized despite having strong content and solid entertainment quotient. Thus, for the movie to resonate with the audience and a solution for all cinephiles, there must be a database system that can record the movie details along with ratings.

1.4 AIM AND OBJECTIVES OF THE PROJECT

Through this mini project we aim to cater all the cinephiles out there with the aim of helping them in making better decision with regards to the movies they choose. This will enable them to watch all the movies they wish too as they can store its details before they forget. The objective is to ease the work of not only the user but also various film circuits to gain insights on which movie is better in terms of filmmaking quality.

CHAPTER 2

SYSTEM SPECIFICATIONS

2.1 HARDWARE SPECIFICATIONS

Processor	:	Intel i5
Memory Size	:	8GB (Minimum)
HDD	:	1 TB (Minimum)

2.2 SOFTWARE SPECIFICATIONS

Operating System	:	WINDOWS 10
Front – End	:	Python
Back - End	:	SQLite
Language	:	Python, SQL

CHAPTER 3

MODULE DESCRIPTION

This application consists of various modules depicting the CRUD operations. When the programs run, the user can enter the movie details and click Add New option to add the movie to the database. There are also provisions for search, update, clear, delete and exit. The description of the modules are as follows:

1. Add New

When the person who interacts tries to insert a new movie into the database, he can make use of the Add New option. This inserts the movie to the database in the backend.

2. Clear

With the help of clear, the user can erase the entire details (not from the database) present in the screen. This is particularly useful when the user finished inserting one film details and wishes to enter the next one.

3. Display

This module displays the entire contents from the database from which the user can select the desired movies either for knowing about it or to do some changes or to delete it from the database

4. Delete

This module deletes the particular movie details from the database. The user can make use of this if he has watched it or does not want to watch it anymore.

5. Search

With the help of this module, we can search for a particular movie having the desired rating we want. This proves as a measure to weigh the movie's worth

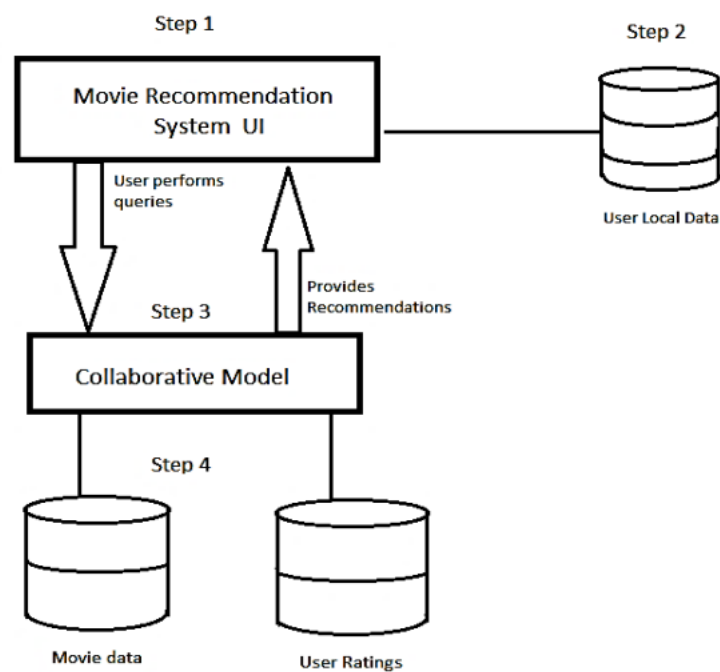
6. Update

If the user has any corrections to make, they can select the particular movie and after making changes can give the update option to change the desired attributes. This flexibility makes the system easier to handle

CHAPTER 4

SYSTEM DESIGN

4.1 ARCHITECTURE DIAGRAM



The diagram describes the working of the system in a dynamic way. The movie details entered by the user are stored in a database for future reference and makes their watch better and easier.

4.2 E-R MODEL

CHAPTER 5

SAMPLE CODING

```
#backend

import sqlite3

def modifyAfterUpdateTrigger():

    con = sqlite3.connect("movie1.db")

    cur = con.cursor()

    try:

        cur.execute("CREATE TRIGGER modifyAfterupdate AFTER UPDATE

            ON book

            BEGIN

                UPDATE book set Rating = 0.5 WHERE Rating <= 0;

                UPDATE book SET Rating = 5 WHERE Rating >= 6;

            END;

        ")

    except Exception as e:

        print(e)

    con.commit()
```

```
con.close()
```

```
def reduceRatingTrigger():
```

```
    con = sqlite3.connect("movie1.db")
```

```
    cur = con.cursor()
```

```
    try:
```

```
        cur.execute("""CREATE TRIGGER reduceRating AFTER INSERT
```

```
                        On book
```

```
                        BEGIN
```

```
                            UPDATE book SET Rating = 5 WHERE Rating >= 6;
```

```
                        END;
```

```
                    """)
```

```
    except Exception as error:
```

```
        print(error)
```

```
    con.commit()
```

```
    con.close()
```

```
def modifyRatingTrigger():
```

```
    con = sqlite3.connect("movie1.db")
```

```
    cur = con.cursor()
```

try:

```
cur.execute("CREATE TRIGGER modifyRating AFTER INSERT
```

```
ON book
```

```
BEGIN
```

```
UPDATE book set Rating = 0.5 WHERE Rating <= 0;
```

```
END;
```

```
");
```

except Exception as e:

```
print(e)
```

```
con.commit()
```

```
con.close()
```

def MovieData():

```
con=sqlite3.connect("movie1.db")
```

```
cur=con.cursor()
```

```
cur.execute("CREATE TABLE IF NOT EXISTS book(id INTEGER PRIMARY  
KEY,Movie_ID text,Movie_Name text,Release_Date text,Director text,Cast  
text,Budget text,Duration text,Rating text)")
```

```
con.commit()
```

```
con.close()
```



```
def
AddMovieRec(Movie_ID,Movie_Name,Release_Date,Director,Cast,Budget,Duratio
n,Rating):
```

```
    con=sqlite3.connect("movie1.db")
```

```
    cur=con.cursor()
```

```
    cur.execute("INSERT INTO book VALUES (NULL,?,?,?,?,?,?,?)",
(Movie_ID,Movie_Name,Release_Date,Director,Cast,Budget,Duration,Rating))
```

```
    con.commit()
```

```
    con.close()
```

```
def ViewMovieData():
```

```
    con=sqlite3.connect("movie1.db")
```

```
    cur=con.cursor()
```

```
    cur.execute("SELECT * FROM book")
```

```
    rows=cur.fetchall()
```

```
    con.close()
```

```
    return rows
```

```
def DeleteMovieRec(id):
```

```
    con=sqlite3.connect("movie1.db")
```

```
    cur=con.cursor()
```

```
cur.execute("DELETE FROM book WHERE Movie_ID= ?", (id,))
```

```
con.commit()
```

```
con.close()
```

```
def
```

```
SearchMovieData(Movie_ID="",Movie_Name="",Release_Date="",Director="",Cast="",Budget="",Duration="",Rating=""):
```

```
con=sqlite3.connect("movie1.db")
```

```
cur=con.cursor()
```

```
cur.execute("SELECT * FROM book WHERE Rating = ?",(Rating))
```

```
rows=cur.fetchall()
```

```
con.close()
```

```
return rows
```

```
def
```

```
UpdateMovieData(id,Movie_ID="",Movie_Name="",Release_Date="",Director="",Cast="",Budget="",Duration="",Rating=""):
```

```
con=sqlite3.connect("movie1.db")
```

```
cur=con.cursor()
```

```
cur.execute("UPDATE book SET  
Movie_ID,Movie_Name=?,Release_Date=?,Director=?,Cast=?,Budget=?,Duration=  
?,Rating=? WHERE  
id=?", (Movie_ID,Movie_Name,Release_Date,Director,Cast,Budget,Duration,Rating  
)
```

```
con.commit()
```

```
con.close()
```

CHAPTER 6

SCREEN SHOTS

Movie Management System

CINEPHILE CORNER

Movie Info_

Movie ID:

Movie Name:

Release Date:

Director:

Cast:

Budget (Crores INR):

Duration (Hrs):

Rating (Out of 5):

Movie Details_

Add New Display Clear Search Delete Update Exit

Fig 6.1 Landing Page

Movie Management System

CINEPHILE CORNER

Movie Info_

Movie ID: M001

Movie Name: Vikram

Release Date: 03/06/2002

Director: Lokesh Kanagaraj

Cast: Kamal Haasan, VJS, FaFa

Budget (Crores INR): 110

Duration (Hrs): 2.53

Rating (Out of 5): 4.5

Movie Details_

M001 Vikram 03/06/2002 {Lokesh Kanagaraj} (Ka

Add New Display Clear Search Delete Update Exit

Fig 6.2 Movie Details Insertion

Movie Management System

CINEPHILE CORNER

Movie Info_

Movie ID:	M001
Movie Name:	Vikram
Release Date:	03/06/2002
Director:	Lokesh Kanagaraj
Cast:	Kamal Haasan, Vijay Sethupathi
Budget (Crores INR):	110
Duration (Hrs):	2.53
Rating (Out of 5):	4.5

Movie Details_

M001 Vikram 03/06/2002 {Lokesh Kanagaraj} {Ka

Add New
Display
Clear
Search
Delete
Update
Exit

Fig 6.3 Movie Details Updating

Movie Management System

CINEPHILE CORNER

Movie Info_

Movie ID:	
Movie Name:	
Release Date:	
Director:	
Cast:	
Budget (Crores INR):	
Duration (Hrs):	
Rating (Out of 5):	4.5

Movie Details_

M001 Vikram 03/06/2002 {Lokesh Kanagaraj} {Ka

Add New
Display
Clear
Search
Delete
Update
Exit

Fig 6.4 Searching a Movie

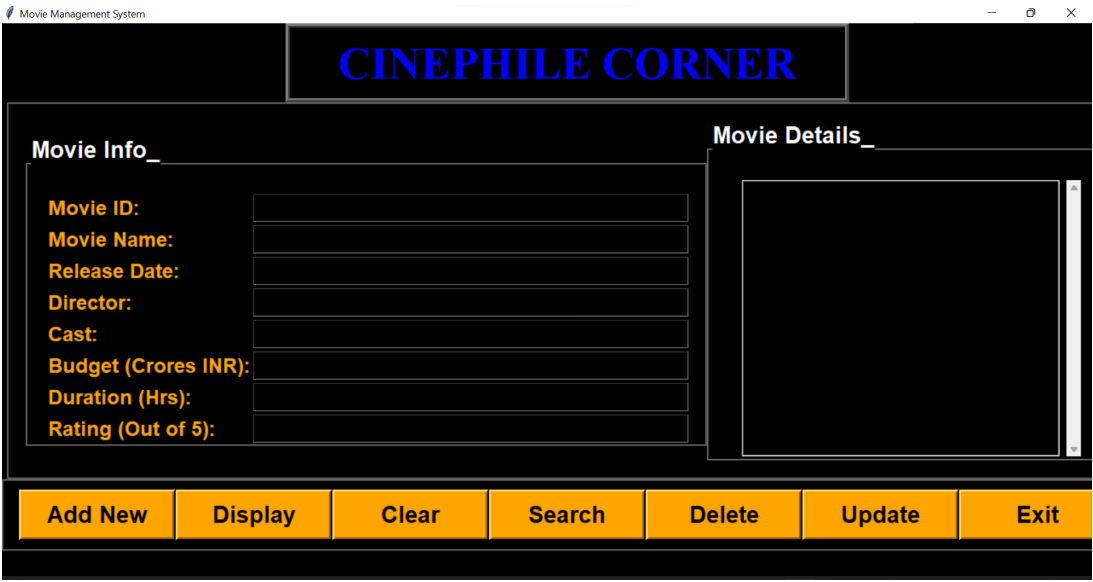


Fig 6.5 Deletion of Movie Details

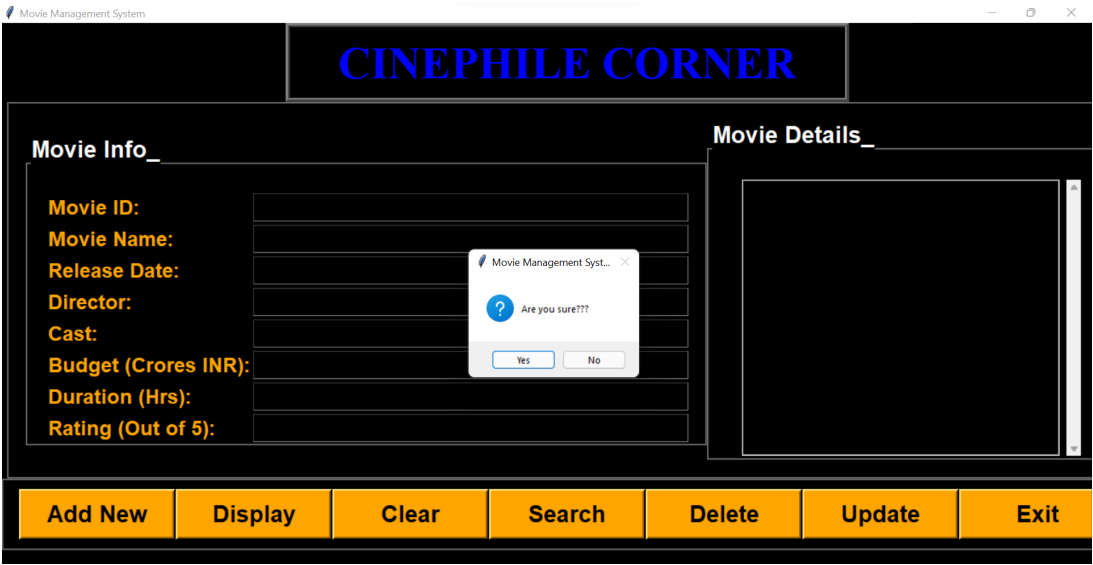


Fig 6.6 Exit from Application

CHAPTER 7

CONCLUSION AND FUTURE ENHANCEMENT

The project is titled “Cinephile Corner – Movie Management System”. It stores the movie details which the user wishes to and offers them an optimal way to keep track of which movies to watch. This project includes all the concepts of Data Base Management System.

However, this project can be improved in such a way that it turns into a real time application. It can be created into an App that could be helpful and beneficial to many people. The App could also be scaled to bigger system by bringing in the film circuits so that they can also gain insights on which film is being appreciated and provide them with due recognition and awards. The system can also be enhanced using machine learning and we can come up with models which according to the watch history and ratings of movies given by the user can recommend movies accordingly.

REFERENCES

1. <https://www.w3schools.com/>
2. <https://www.tutorialspoint.com/>
3. <https://www.wikipedia.org/>
4. <https://www.learnpython.org/>
5. <https://www.codecademy.com/learn/learn-python>