Kathmandu University

Department of Computer Science and Engineering

Dhulikhel, Kavre

A Project Proposal
On
"MovieMaestro"

[Code No: ENGG 102]

(For partial fulfillment of I Year/II Semester in Computer Engineering)

Submitted by

Aditya Pokharel (32)
Abhiyan Regmi (42)
Sameer Kumar Sah (43)
Parikchit Sen (45)

Submitted to

Mr. Dhiraj Shrestha

Department of Computer Science and Engineering

Submission Date: 30th May, 2024

Abstract

In an era where the abundance of movie content can overwhelm viewers, an efficient movie recommendation system is essential to enhance user experience. The purpose of this project is to develop a simple, yet effective movie recommendation system that provides users with tailored movie suggestions based on their preferences. Utilizing QT Creator, C++ along with Object-Oriented Programming, and SQLite, we will create a GUI application that allows users to interact with the system and receive movie recommendations. The system will store and manage user preferences and movie data using SQLite, and apply straightforward algorithms to suggest movies. Expected outcomes include a user-friendly interface and satisfactory movie recommendations, enhancing the overall viewing experience. This project will demonstrate the effectiveness of integrating user preferences with basic recommendation algorithms in a GUI application.

Keywords: Object-Oriented Programming, SQLite, QT Creator, GUI.

Table of Contents

List of Figures

Acronyms/Abbreviations

UI	User Interface
GUI	Graphical User Interface
SQLite	Structured Query Language Lite
RDBMS	Relational Database Management System

Chapter 1   Introduction

In the digital age, the availability of countless movies all over the internet can be a bit overwhelming for viewers to choose. Traditionally, movie recommendations were based on critics' reviews or word-of-mouth, but with the advent of technology, recommendation systems have revolutionized the way people discover movies. These systems use algorithms to filter and predict user preferences, making it easier for individuals to find movies that match their tastes. The need for effective recommendation systems has become more pronounced with the rise of streaming platforms, where users have access to extensive libraries of movies.

Our project, MovieMaestro, aims to simplify and improve the movie selection process by providing them a better user-friendly experience. The development of this project is just a small initiative to make people's life easier by saving their valuable time.

1.1   Background

In recent years, the development of these systems has seen significant advancements, particularly with the integration of machine learning algorithms, collaborative filtering, and content-based filtering techniques. Platforms like Netflix, Amazon Prime, and Hulu have set the standard by utilizing sophisticated algorithms to deliver highly accurate recommendations, thereby increasing user satisfaction and retention.

MovieMaestro is an application which offer a robust and efficient platform for users to discover movies that match their taste. The simplicity and effectiveness of MovieMaestro will make it a valuable tool for users seeking personalized movie recommendations while maintaining control over their personal data. It will leverage a combination of user preference analysis and an intuitive user-friendly interface to

deliver tailored movie recommendations. With a straightforward approach, it is light platform developed to provide quality content to the user.

However, MovieMaestro lacks machine learning algorithms in its recommendation process. Machine learning has ability to analyze vast amounts of data & identify complex patterns. But, without utilizing machine learning, MovieMaestro may struggle to deliver highly personalized and accurate recommendations compared to systems that do. Without these advanced techniques, MovieMaestro might rely on simpler, less sophisticated methods, potentially leading to less precise recommendations and a less engaging user experience.

Another significant drawback is the challenge posed by insufficient data, particularly in the early stages of a user's interaction with the system, known as the cold start problem. When a new user begins using MovieMaestro, the system lacks sufficient information about their preferences, making it difficult to provide accurate recommendations. Similarly, when new movies are added to the database, the system may not have enough user interaction data to effectively recommend these titles to the right audience which may result in user disengagement.

Although MovieMaestro can cause less satisfying experience due to its drawbacks, it is a simple and efficient platform which still provides valuable recommendations based on user preferences and available data.

1.2 Objectives

The major objectives of MovieMaestro are listed below:

- To save time by eliminating the need to manually search for movies
- To ensure that personal information is handled responsibly and transparently
- To offer an intuitive and easy-to-use user-friendly interface
- To empower effortless movie selection tailored to individual tastes

1.3   Motivation and Significance

Users often struggle to discover new movies that align with their preferences. In order to solve this issue, we selected movie recommendation system, MovieMaestro, as our project work. Unlike some systems that prioritize advanced algorithms at the expense of user experience, MovieMaestro prioritizes simplicity and accessibility while still delivering reliable recommendations. Additionally, MovieMaestro stands out by offering a privacy-conscious platform that respects user data and preferences, fostering trust and confidence among users.

1.4   Expected Outcomes

Through MovieMaestro's recommendation algorithms, it aims to promote diversity in movie consumption by recommending movies on the basis of a wide range of genres, languages, date of release, user ratings, and etc. ensuring users stay up to date with current trends and releases.

Chapter 2   Related Works/ Existing Works

Some existing platforms related to our project are mentioned below:

2.1   Netflix

Netflix is an online platform which utilizes a sophisticated movie recommendation system to personalize the user experience by helping users discover new content.
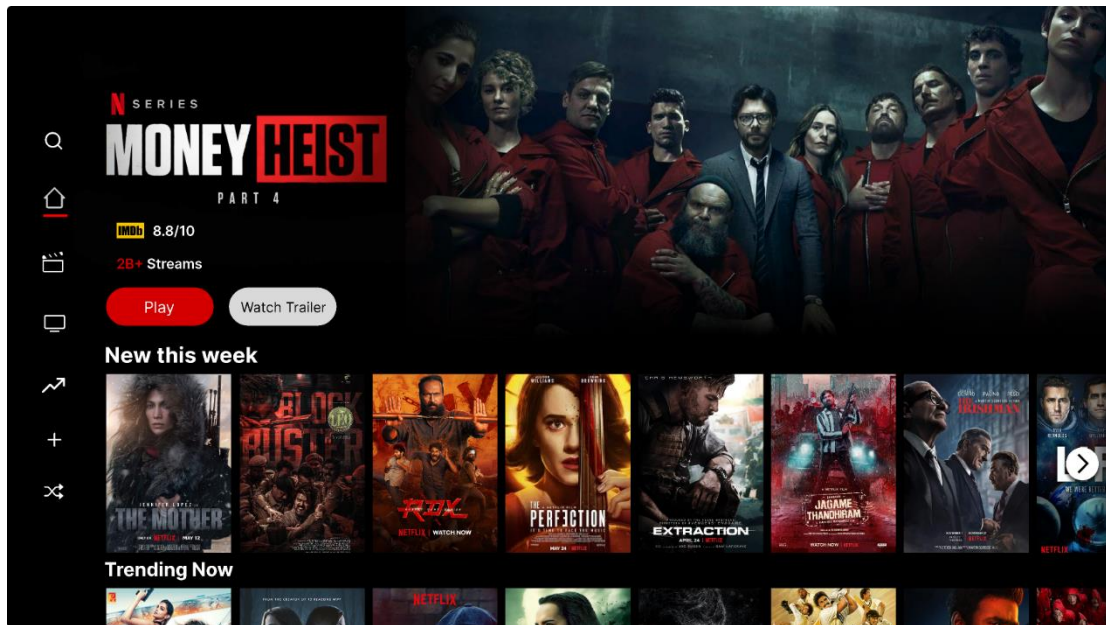

Figure 2.1.1: Netflix Home Page

2.2 Hulu

Hulu is also a movie recommendation system platform designed to personalize the viewing experience by analyzing their preferences and viewing habits.
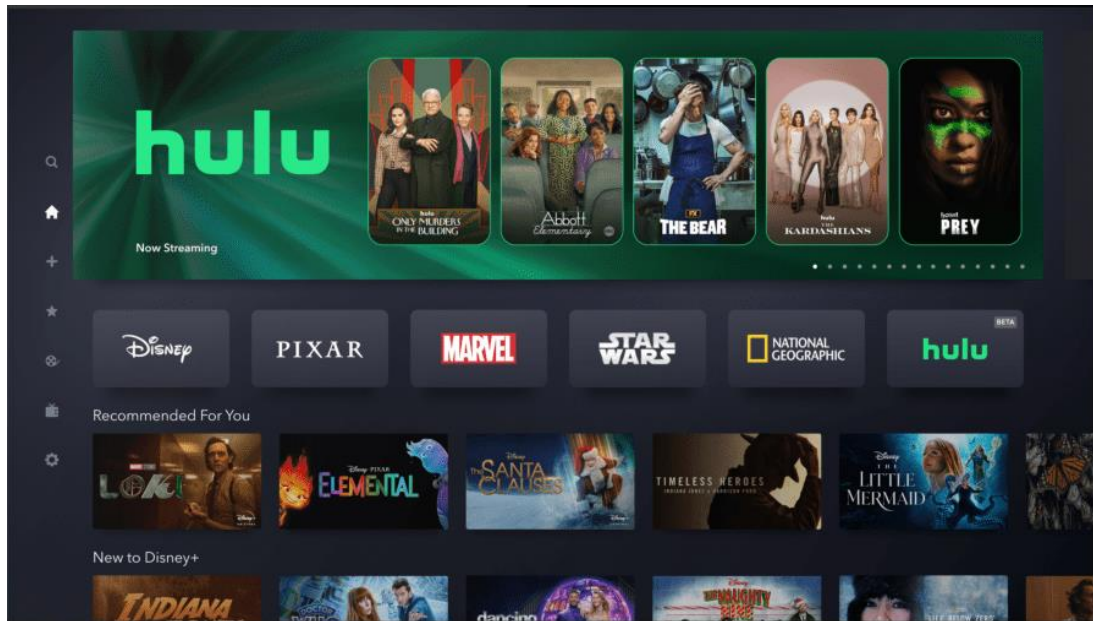

Figure 2.2.1: Hulu Home Page

Chapter 3   Procedure and Methods

- **Research & Proposal**

  We explored various project ideas and analyzed them. We finalized one of them and prepared a proposal.

- **Planning & Work Distribution**

  We will plan the entire project's outline and assign tasks to team members.

- **Set Up & Learning**

  All the requirements for developing the project will be set up. We will learn and familiarize ourselves with the basics of QT Creator, C++ & SQLite. We will take Netflix and Hulu as our reference.

- **Front-end & Back-end Development**

  We will design the user-friendly GUI, develop the backend logic & database using QT Creator, C++ & SQLite.

- **Testing**

  We will test the application to identify any issues and fix the bugs. The functionality and security of the application will also be tested.

- **Documentation**

  We will document the project's outline, functionality and implementation for future reference and maintenance.
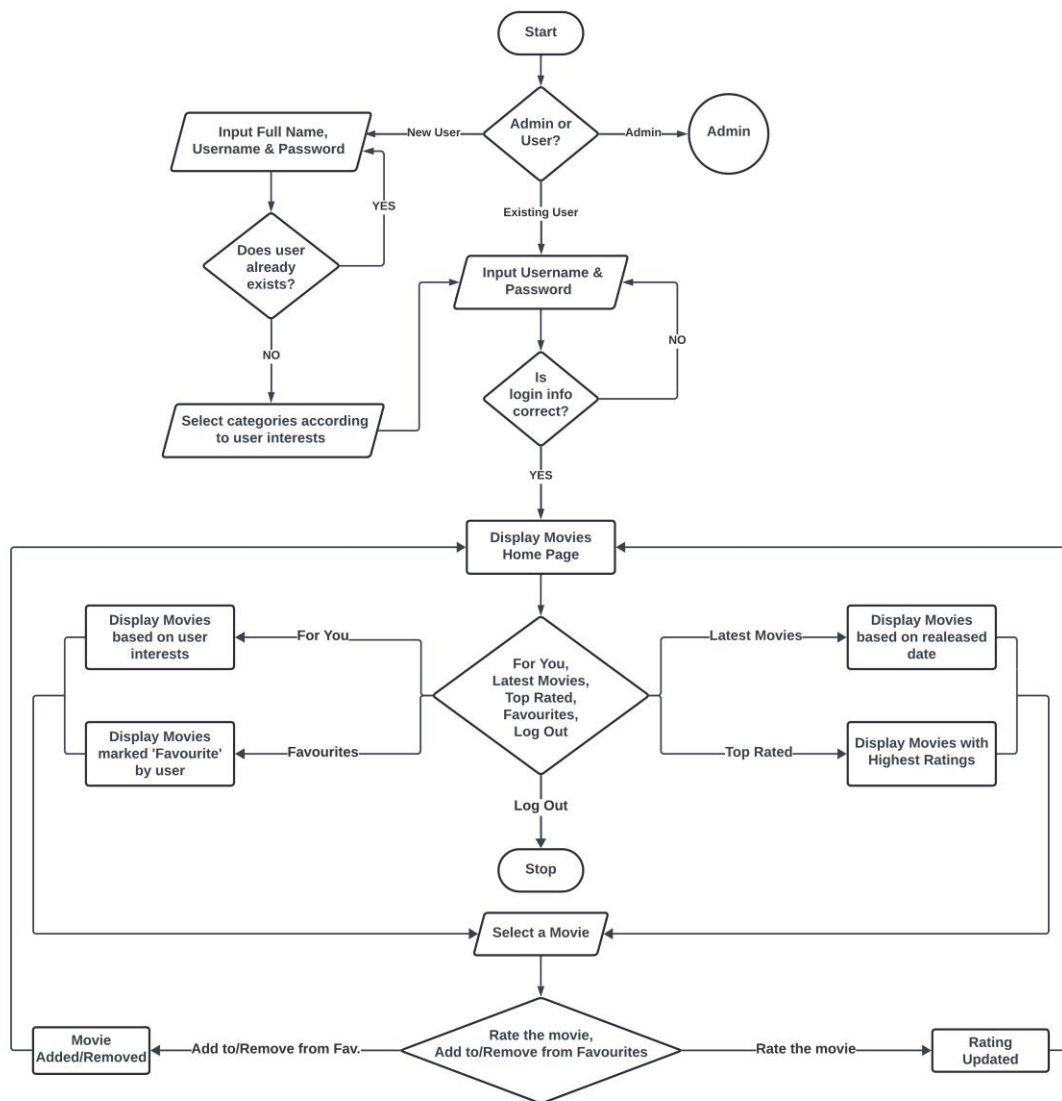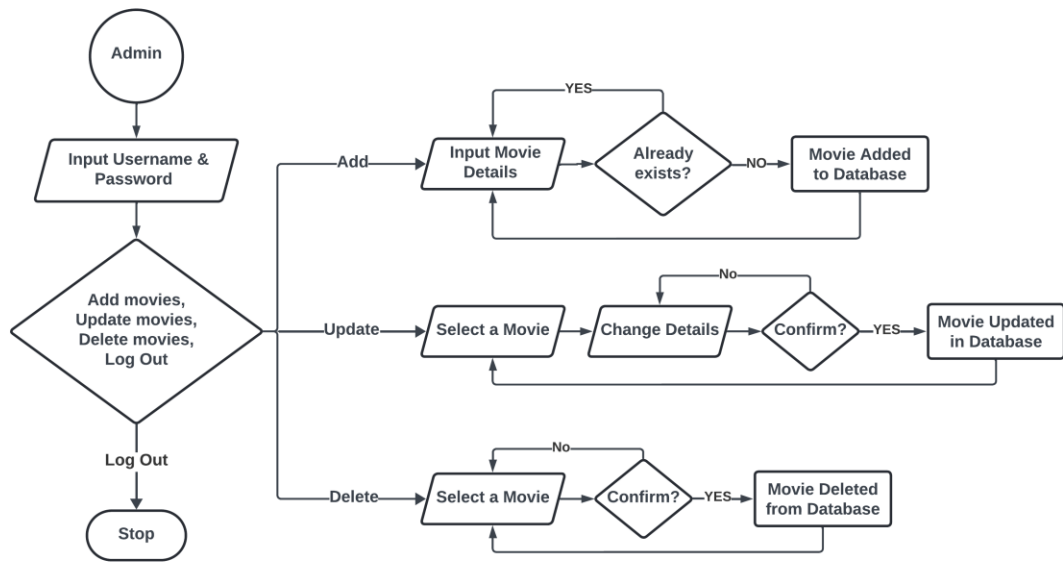
Figure 3.1: Flowchart for New/Existing User

Figure 3.2: Flowchart for Admin

Chapter 4   System Requirement Specifications

The system requirements of MovieMaestro are given below:

4.1   Software Specifications

4.1.1   Front-end Technologies

- **QT Creator**

  QT Creator is a cross-platform integrated development environment (IDE) designed for building applications using the QT framework. QT Creator provides developers with a comprehensive set of tools for designing, coding, debugging, and deploying software applications, particularly those that utilize the QT application framework.

4.1.2   Back-end Technologies

- **SQLite**

  SQLite is a lightweight, server less, self-contained, and open-source RDBMS. Unlike traditional client-server database engines, SQLite is embedded directly into the application, requiring minimal configuration and administration.

- **C++ Programming Language**

  C++ is a cross-platform language developed as an extension to the C language that can be used to create high-performance applications. It is an object-oriented programming language that supports features such as classes, objects, encapsulation, abstraction, inheritance, and polymorphism.

4.2   Hardware Specifications

Any system having Windows is sufficient to run this application.

Chapter 5   Project Planning and Scheduling

The dedication and contribution of each member is required for the completion of this project. Teamwork plays a significant role to achieve this goal. In order to develop the application in the given time, which is limited, we decided to break it down. It will help us to accomplish the tasks of this project work smoothly and calmly without having to rush at the end, avoiding last-minute stress. The rough estimation of the time allocation for different tasks is illustrated below in the Gantt chart.
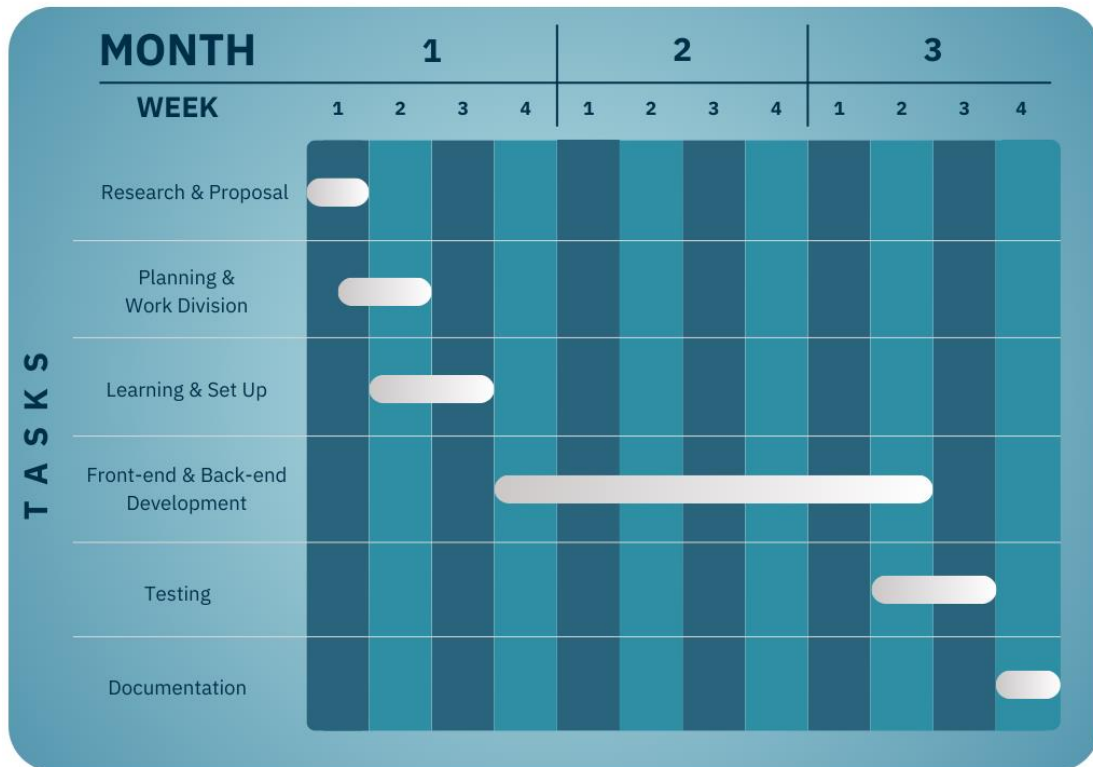

Figure 5.1: Gantt Chart

References

- Netflix

  [Netflix Nepal - Watch TV Shows Online, Watch Movies Online](#)

- Hulu

  [Stream TV and Movies Live and Online | Hulu](#)