Institute of Arts and Sciences (Chiniot Campus)
Govt College University Faisalabad

Software Requirement Specification

(SRS Document)

For

Cinema Complex System

By

Umer Jahangir **225356**

Instructor

Dr. Waqar Hussain

Bachelor in Computer Science (2022-2026)

Table of Contents

Revision History

Name	Date	Reason For Changes	Version

1. Introduction

1.1 Purpose

The purpose of this document is to provide a detailed description of the Cinema complex System, outlining its features, functionalities, and requirements. It serves as a guide for developers, testers, and other stakeholders involved in the development and maintenance of the system.

1.2 Intended Audience

This document is intended for:

- *Developers*
- *Testers*
- *Project Managers*
- *Stakeholders involved in the Cinema Complex System*

Readers are advised to review the document thoroughly to understand the requirements and constraints of the system.

1.3 Project Scope

The Cinema Complex System aims to provide a comprehensive solution for managing the operations of a cinema complex. The system covers various functionalities to enhance the overall efficiency of the cinema's day-to-day activities.

1.4 Definitions

Managerial Staff

As cinema managers, our job involves handling movie listings, making sure they're up-to-date and appealing. We also take care of employee-related tasks like registration, attendance tracking, and staff changes. We keep an eye on how things are going, looking at numbers and reports to improve how the cinema works.
Our goal is to make sure people have a great time watching movies at our cinema.

Front Desk Personnel

Front desk staff at the cinema are the team members directly interacting with customers, handling various responsibilities to ensure a smooth and enjoyable experience.
Their main tasks include processing customer reservations, generating tickets, and addressing any inquiries or concerns customers may have.

Marketing Team

The specialized team dedicated to cinema promotions and customer engagement is focused on enhancing the overall experience for moviegoers. Their responsibilities encompass monitoring the popularity of movies, collaborating on promotional activities to attract audiences, and analyzing customer feedback to improve offerings.

Customer

As end-users utilizing cinema services, our responsibilities revolve around making the most of the movie-watching experience. This includes reserving movie slots to secure our preferred showtimes, purchasing tickets for the chosen films, and regularly checking movie schedules and availability. We play a crucial role in the cinema community by providing valuable feedback on our movie experiences, contributing to the continuous improvement of services and ensuring that our cinematic outings are enjoyable and tailored to our preferences.

Input

Input is the data or information that a system receives from the external environment.
It serves as the raw material for processing within a system.
In a computer program, user keyboard entries, mouse clicks, or data from sensors can be considered as input.

Output

Output is the result or information produced by a system or process after the input has been processed.
In a computer program, the output could be displayed on a screen, printed on paper, or saved in a file.

Prompt

Prompt refers to a message or indicator that prompts the user to provide input. It's a way for the program to communicate with the user.

2. Overall Description

2.1 Product Perspective

The Cinema Complex System operates independently as a standalone product, managing movie-related activities within the cinema. It interfaces with external databases or services for additional functionalities, such as movie data retrieval or user authentication. The system doesn't rely on integration with other external systems, ensuring autonomy and efficient cinema management.

2.2 Product Features

1. ***Movie Management***

   - *Add, modify, and delete movie records*
   - *Display the list of movies with details*
   - *Sort movies based on name, genre, or price*

2. ***Employee Management***

   - *Register, modify, and delete employee records*
   - *Display the list of employees with details*
   - *Mark attendance and manage attendance records*

3. ***Customer Reservation System***

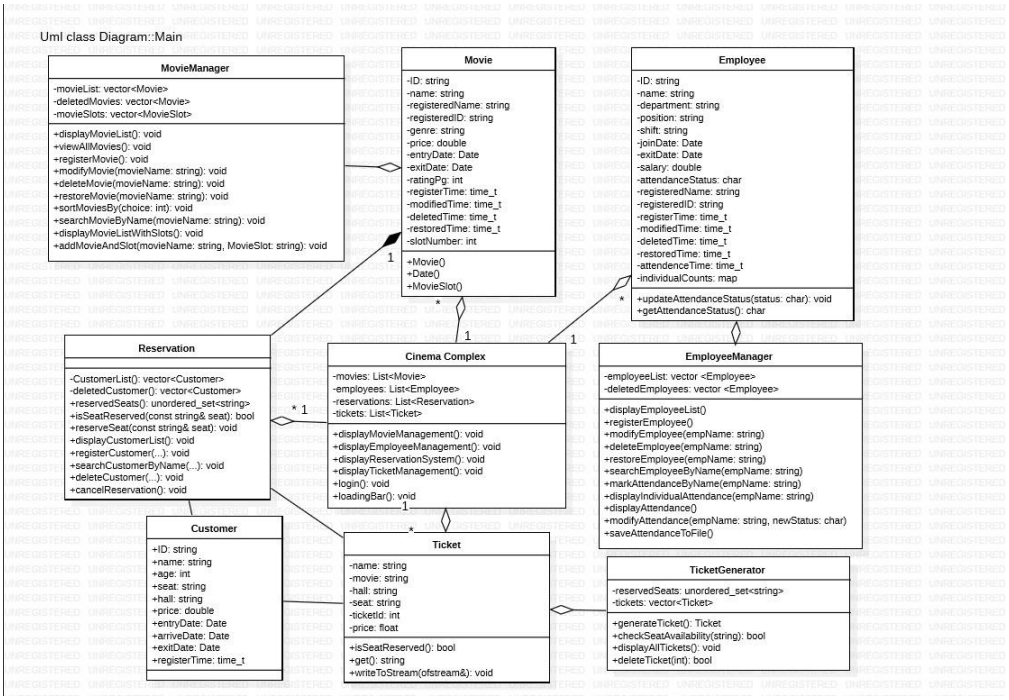
   - *Register and search customer reservations*
   - *Display customer lists, individual reservations, and attendance*
   - *Cancel customer reservations*

4. ***Customer Ticket System***

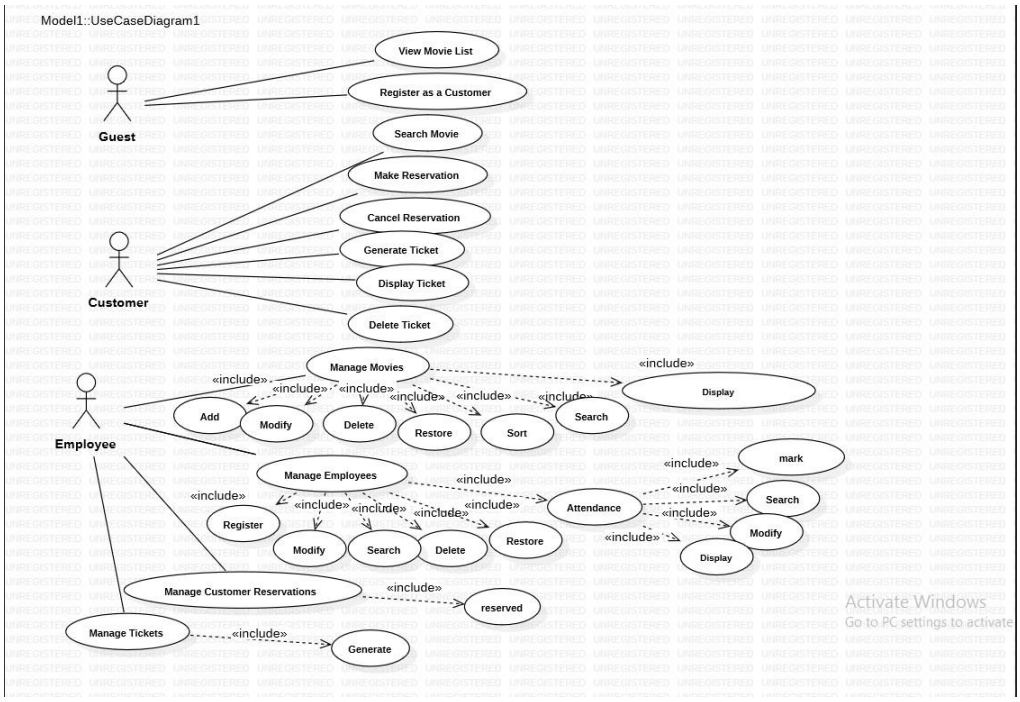
   - *Generate customer tickets with details*
   - *Display all generated tickets*
   - *Delete customer tickets by ticket ID*

2.3 User Classes and Characteristics

Class Diagram



Use Case Diagram



2.4 Operating Environment

The system will be compatible with Windows operating systems.

2.5 Design and Implementation Constraints

- *The system will be developed using C++ Programming Language.*
- *The database will be simple txt files.*
- *Compatibility with latest version of C++.*

2.6 User Documentation

Comprehensive user documentation will be provided, including user manuals and guides.

o Concepts

Object-Oriented Programming (OOP) Principles:

- **Class**

  Classes Used:

  - *Core entities within the system, such as CinemaManager, MovieManager, EmployeeManager, Reservation, TicketGenerator, and others, adhering to object-oriented programming principles.*
  - **Explanation***: A class is a user-defined data type that we can use in our program, and it works as an object constructor, or a "blueprint" for creating objects*

- **Encapsulation**

  - **Private Members:** *Various data members in classes are marked as private, encapsulating the internal details of each class***.**
  - **Explanation:** *Encapsulation is the bundling of data and the methods (functions) that operate on the data into a single unit, i.e., a class. This helps in controlling access to the data by providing access specifiers (public, private, protected).*

- **Abstraction**

  - *Employment of abstract classes like MovieManager for essential functionalities, exposing key details for external use.*
  - ***Explanation:*** *Abstraction is the concept of simplifying complex systems by modeling classes based on the essential properties and behaviors.*

- **Inheritance**

  - *Establishment of "is-a" relationships, exemplified by classes like EmployeeManager inheriting from MovieManager and TicketGenerator inheriting from Reservation.*
  - ***Explanation:*** *Inheritance is the mechanism by which one class can inherit properties and behavior from another class promoting code reusability.*

- **Composition**
    - *Integration of objects from different classes, for instance, MovieManager within CinemaManager, highlighting the composition design principle.*
    - **Explanation:** *Composition refers to the design principle where a class is composed of or contains objects of other classes, enabling the creation of more complex and reusable structures.*
- **Aggregation**
    - *Representation of "has-a" relationships in various classes, such as EmployeeManager and MovieManager,containing objects of Employee and Movie, respectively.*
    - **Explanation:** *Aggregation in represents a "has-a" relationship between classes, where one class contains another, but the contained class maintains independence.*

Software Engineering Principles:

- **Development Approach**
    - *Embracing an iterative development model for adaptability and ongoing refinement.*
- **DRY (Don't Repeat Yourself)**
    - *Avoiding duplicating code by encapsulating common functionalities within classes and methods.*
- **Modularity**
    - **Separation of Concerns:** *exemplified by different classes focusing on specific functionalities like movie management and employee management.*
- **Code Reusability**
    - **Inheritance for Reusability:** *Inheritance is used to reuse the functionality.*
- **Maintainability**
    - **Structured Code:** *The code is organized into classes, functions, and headers, enhancing readability and maintainability.*
- **Scalability**
    - **Dynamic Memory Usage:** *Implementation of dynamic memory usage, especially with vectors, for scalability in managing various system components.*
- **Readability**
    - **Descriptive Identifiers:** *Variable and function names are chosen descriptively, contributing to code readability.*

- **File Organization**
    - **Usage**: *Header files (MovieManager.h, EmployeeManager.h, Reservation.h, TickectGenerator.h, CinemaManager.h), Separation of Concerns.*
    - **Explanation:** *The code is organized into header files and source files, following good practices for file organization and separation of concerns.*
- **Documentation**
    - **Usage:** *Comments, descriptive variable names.*
    - **Explanation:** *The code includes comments that provide information about the purpose of the code blocks. Descriptive variable and function names also contribute to self-documenting code.*

Other Concepts

- **Vectors**
    - *Vectors are used for storing and managing dynamic arrays of data. Vectors are used in several places for offering flexibility in handling diverse system functionalities.*
    - **Explanation***: Vectors provide dynamic arrays with automatic resizing.*
- **Algorithm**
    - *The algorithm header is used for sorting movies based on the movies name, price, genre.*
    - **Explanation:** *The algorithm header provides a collection of functions (algorithms) to perform operations on sequences of elements. In this software, sort is used to sort the playerScores vector based on the scores, using the custom comparison function compares cores.*
- **File Handling**
    - *Incorporation of file handling mechanisms for persistent data storage, maintaining Separation of concerns.*
- **Exception Handling**
    - *To guarantee code robustness, we used error-handling techniques such as file existence checks and appropriate user input validation. For increased dependability, confirm the existence of the file before opening it, respond properly to successful actions, and manage invalid user input.*

2.7 Assumptions and Dependencies

- *Users are assumed to have access to a computer with a Windows operating system.*
- *The system relies on standard input and output for user interactions.*
- *The console-based application assumes that users are familiar with basic command-line interactions.*
- *Dependencies include the availability of the C++ runtime environment on users' machines.*
- *Any changes in the Windows environment or command-line interface may affect the application's functionality.*

3. System Features

3.1.1 Functional Requirements

REQ-1: Movie Management

- *Display Movies*

  The system shall display a list of current movies.

- *Register Movie*

  Users can register new movies with details such as name, genre, and price.

- *Modify Movie*

  The system allows modification of existing movie information.

- *Delete/Restore Movie*

  Users can delete or restore movies.

- *Sort/Search Movies*
  Movies can be sorted and searched for better organization.

REQ-2: Employee Management

- *Register Employee*

  Users can register new employees with relevant details.

- *Display Employee List*

  The system shall maintain a list of current employees.

- *Modify Employee*

  Users can modify or update employee information.

- *Delete/Restore Employee*

  Employees can be deleted or restored.

  Search Employee:

  Users can search for employees by name.

- *Employee Attendance*

The system shall track and manage employee attendance.

REQ-3: Customer Reservation System

- *Register Reservation*

  Users can register customer reservations for movie slots, including seat selection.

- *Display Reservations*

  The system shall display a list of current customer reservations.

- *Search Customer*

  Users can search for customers by name.

- *Delete/Cancel Reservation*

  Reservations can be deleted or canceled.

REQ-4: Ticket Management

- *Generate Ticket*

  The system generates tickets for confirmed customer reservations.

- *Display Tickets*

  A list of all generated tickets is displayed.

- *Delete Ticket*

  Users can delete tickets based on ticket ID.

REQ-5: User Authentication

- *Login Mechanism*

  The system shall require users to log in with a valid username and password. Successful authentication grants access to system functionalities

REQ-6: Main Menu Navigation

- *Menu Options*

  The main menu provides options for movie management, employee management, customer reservations, ticket management, and exit.

  Users can navigate between different modules seamlessly.

REQ-7: Loading Bar

- *Post-Login Loading:*

  After successful login, a loading bar simulates system loading for a better user experience.

4. External Interface Requirements

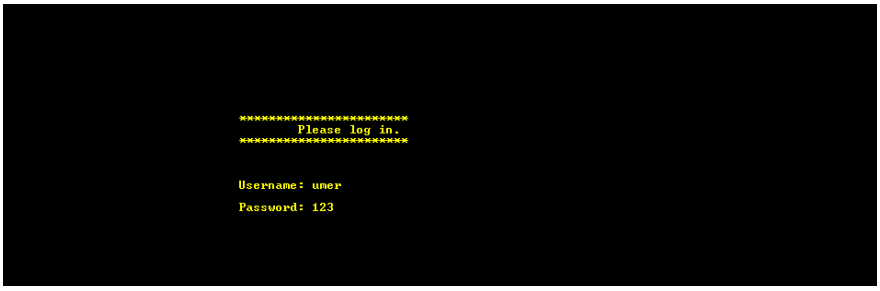
4.1 User Interfaces

The system will have a console-based interface suitable for interaction within a command-line environment on Windows.

1. Cinema Interface

Log In Bar

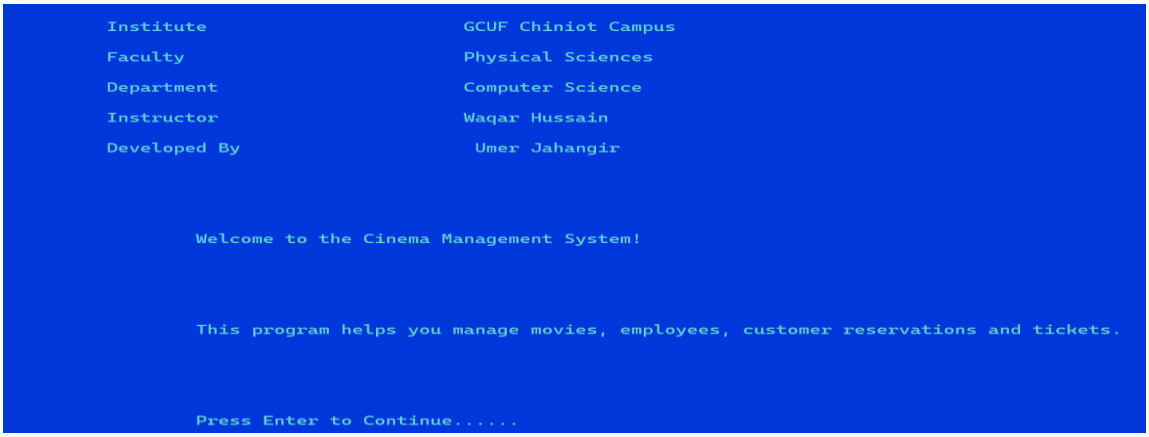
First of all, when you use the program, you will asked to enter username and password.



- ***Username/Email Field:*** *Users enter their registered username or email.*
- ***Password Field:*** *Secure entry for user passwords.*

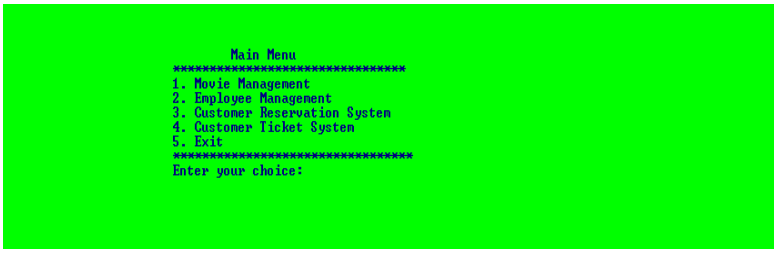
Getting Started

In this you will be greeted with a welcome screen displaying the Project title and a brief description. Press "Enter" to proceed to the main menu.



Main Menu

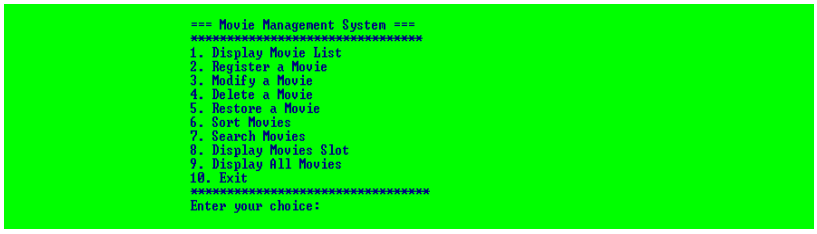
In the main menu, you are able to see the following options:



1. **Movie Management:** *Enter 1 go to the movie menu.*
2. **Employee Management:** *Enter 2 go to the employee menu.*
3. **Reservation system:** *Enter 3 to go customer reservation menu.*
4. **Ticket system:** *Enter 4 to go customer ticket menu.*
5. **Log Out:** Enter 5 to exit cinema complex system program.

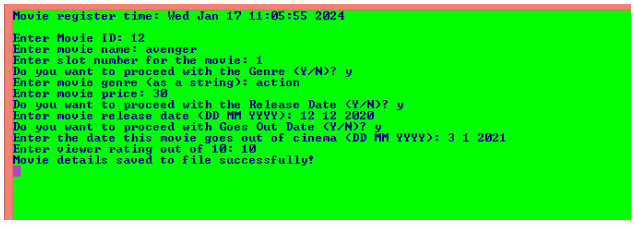
1. Movie Management Menu

Upon entering '1' into the main menu, you are able to see the following options for movie management.

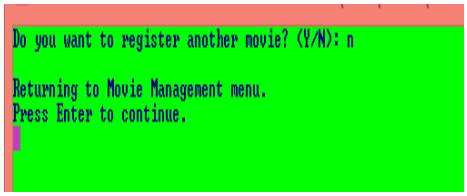


- **Register Movie**

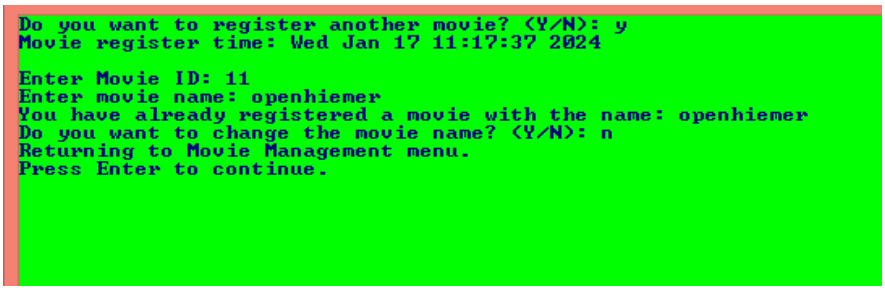
  Upon entering '2' into the movie management menu, you are able to register a movie.



After register a movie it asked to register other movie.

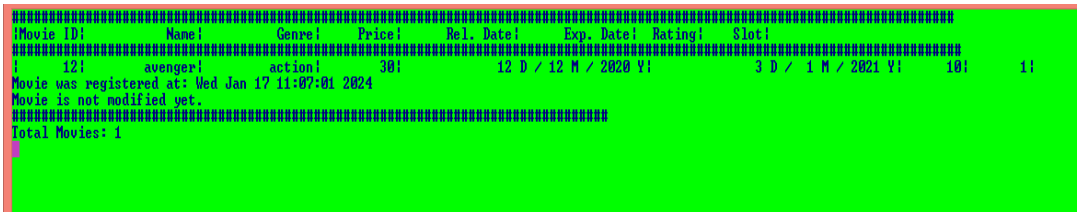


If you want to register another movie press 'y' and if another movie name match to previous register movie then it display.



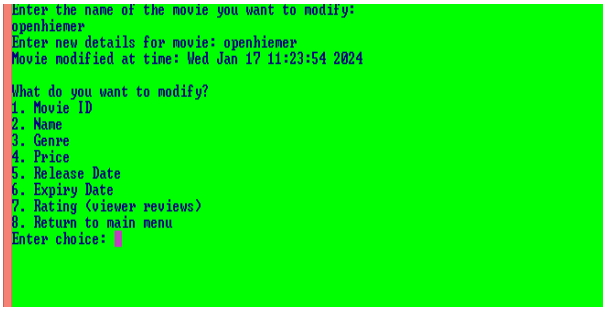
- **Display Movie**

Upon entering '2' into the movie management menu, you are able to see all registered movie.



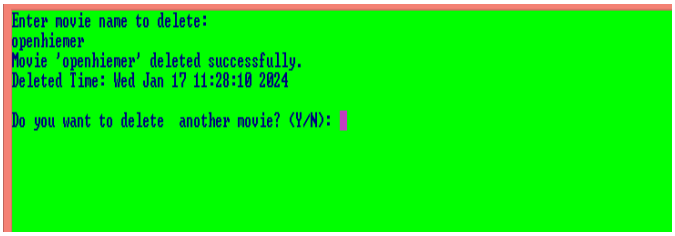
- **Modify Movie**

Upon entering '3' into the movie management menu, you are able to modify a movie.



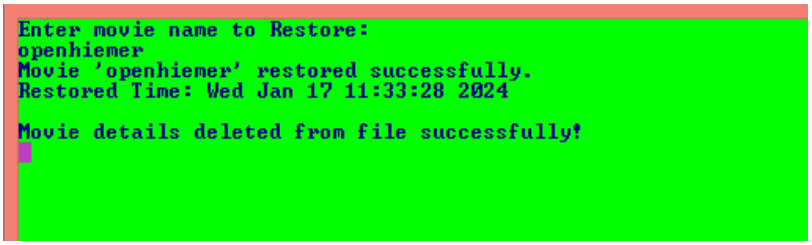
- **Delete Movie**

Upon entering '4' into the movie management menu, you are able to delete a movie.



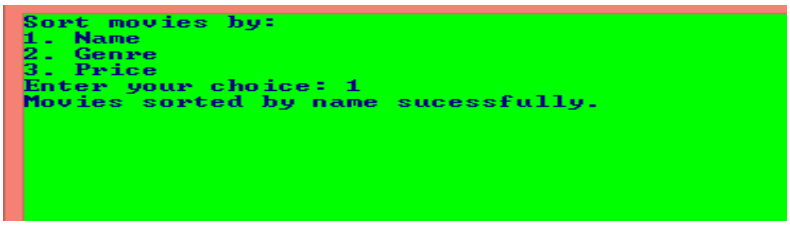
- **Restore Movie**

  Upon entering '5' into the movie management menu, you are able to restore a deleted movie.



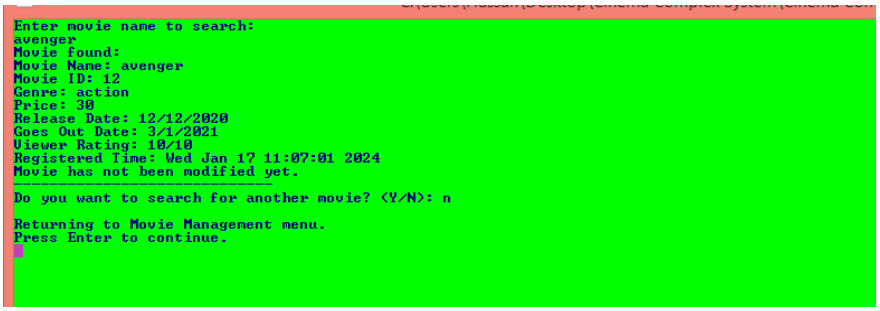
- **Sort Movie**

  Upon entering '6' into the movie management menu, you are able to sort registered movies in display movie menu.



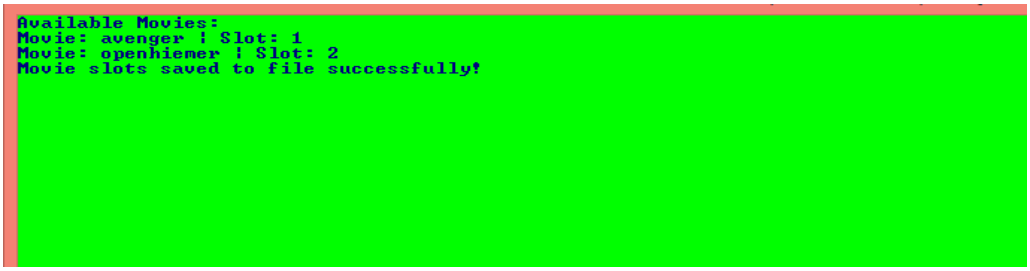
- **Search Movie**

  Upon entering '7' into the movie management menu, you are able to search registered movies in display movie menu.



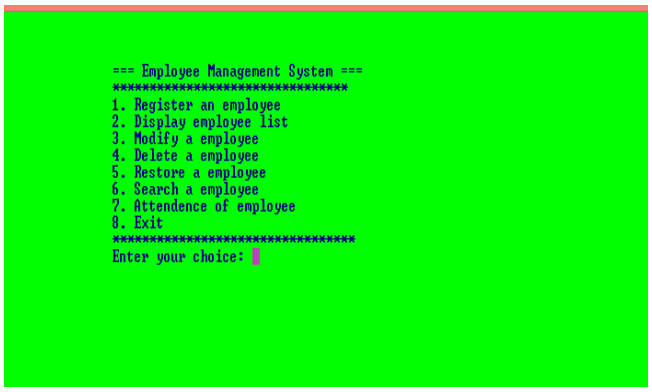
- **Display Movie Slot**

  Upon entering '8' into the movie management menu, you are able to see registered movies slot number.



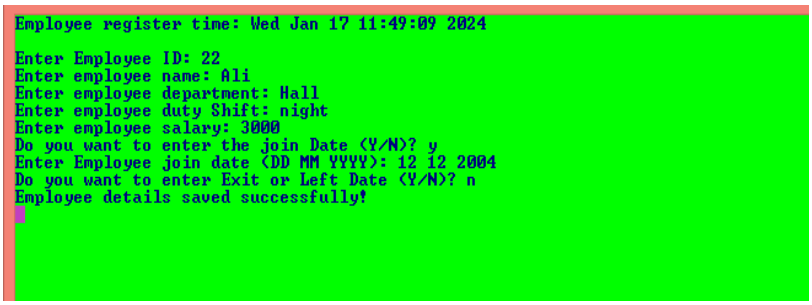
2. Employee Management Menu

Upon entering '2' into the main menu, you are able to see the following options for employee management.



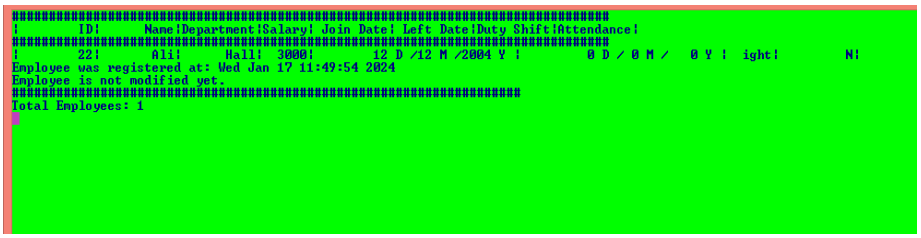
- **Register Employee**

Upon entering '1' into the employee menu, you are able to register a employee.



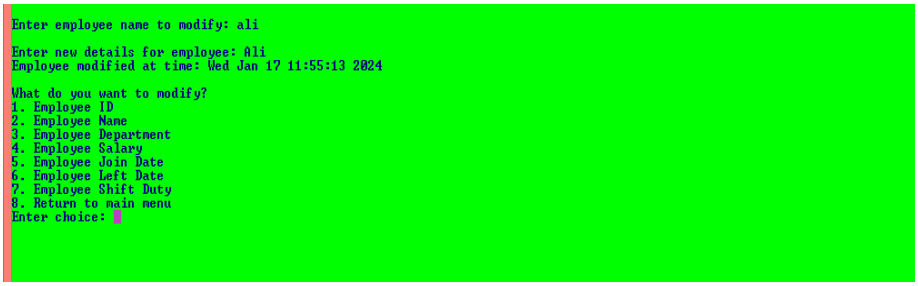
- **Display Employee list**

Upon entering '2' into the employee menu, you are able to see register a employee.



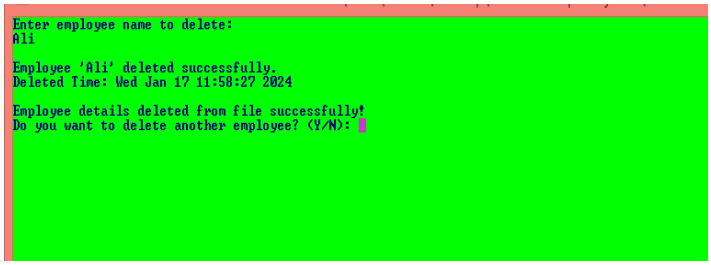
- **Modify Employee**

Upon entering '3' into the employee menu, you are able to modify register employee.



- **Delete Employee**

  Upon entering '4' into the employee menu, you are able to delete a register employee.

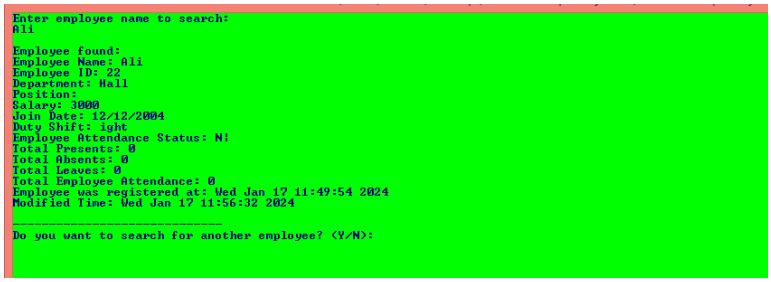


- **Restore Employee**

  Upon entering '5' into the employee menu, you are able to restore a delete employee.

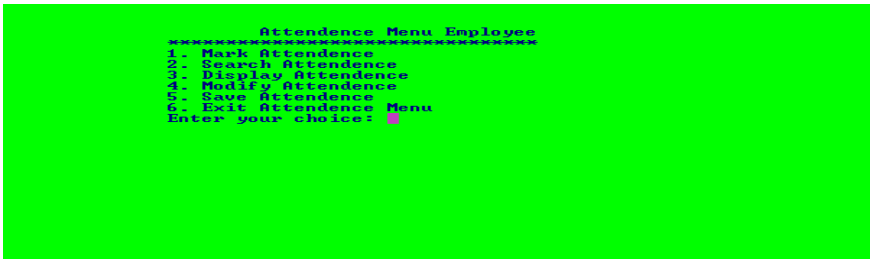
- **Search Employee**

  Upon entering '6' into the employee menu, you are able to search employee.



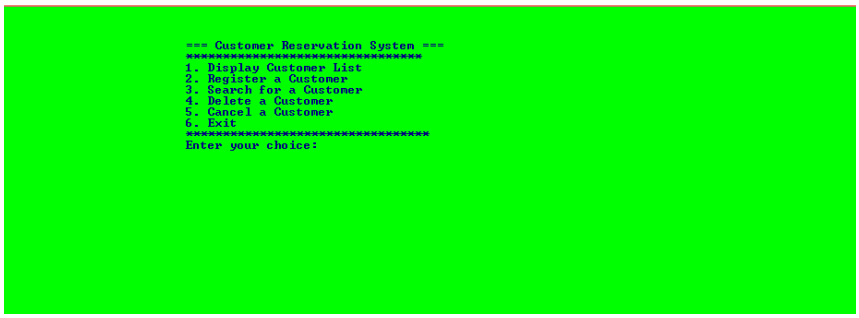
- **Attendence Employee**

  Upon entering '7' into the employee menu, you are able to see following option.



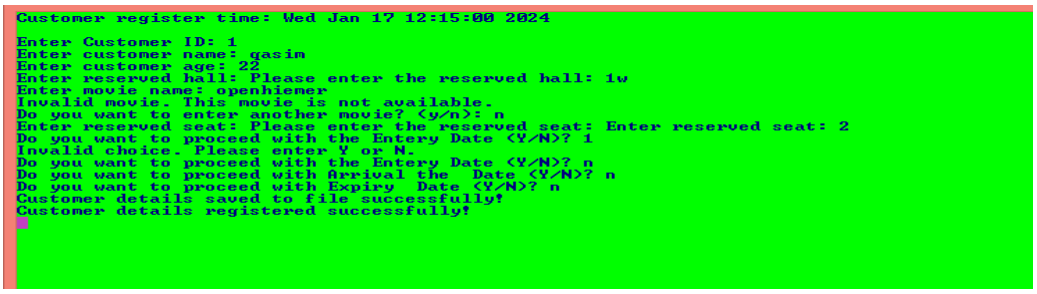
3. Customer Reservation System Menu

Upon entering '3' into the main menu, you are able to see the following options for Customer Reservation System.



- **Register Customer**

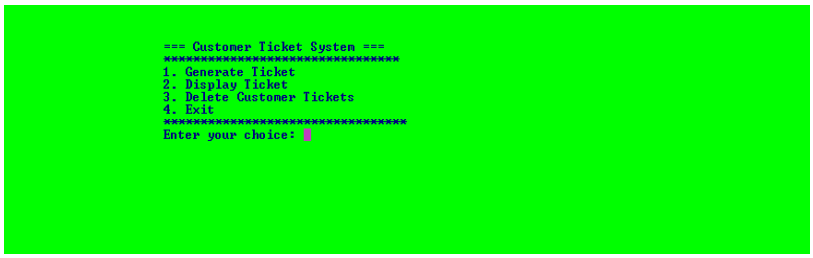
Upon entering '2' into the reservation menu, you are able to register a reserved customer.



Note: *Search, delete and cancel same as other menu fuctions. But these function used in there for reservation.*

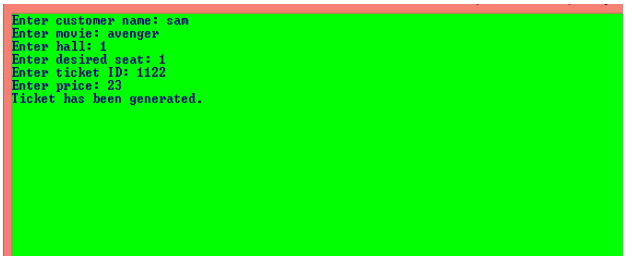
4. Customer Ticket System Menu

Upon entering '4' into the main menu, you are able to see the following options for Customer ticket System.



- ## Generate Ticket

Upon entering '1' into the ticket menu, you are able to generate a ticket for customer.



Notes: *Delete and display same as other menu fuctions. But these function used in there for n ticket system.*

4.2 Hardware Interfaces

The system will require computing hardware compatible with the Windows operating system.

4.3 Software Interfaces

The application relies on the Windows command-line interface for user interactions.it will use txt files as database.

4.4 Communications Interfaces

The application does not require external communication interfaces, as it operates within a standalone console environment.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

The system shall respond to user actions promptly.

Loading times for data display and operations shall be optimized.

5.2 Safety Requirements

The system should ensure the security of user data stored locally on the machine.

5.3 Security Requirements

User authentication shall be secure, requiring a valid username and password.
Access control mechanisms shall ensure that only authorized users can perform specific actions.

5.4 Software Quality Attributes

Reliability: *The system should function without errors, providing a stable experience.*

Usability: *The console-based interface should be designed for ease of use, with clear prompts and instructions for users interacting via the command line.*