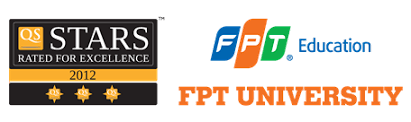
****

**MINISTRY OF EDUCATION AND TRAINING FPT UNIVERSITY CAN THO SCIENCE AND INFORMATION TECHNOLOGY**

**Topic**

**Online Ticketing Application**

**Sinh viên : Diệp Trí Thành**

**Nguyễn Văn Khương**

**Nguyễn Đăng Khoa**

**Võ Anh Nhiều**

**Vũ Tuấn Minh**

**Thank you**

With the permission of subject teacher Luong Hoang Huong, our team implemented the topic "Online booking application". In order to successfully complete the topic, my first thank you to Mr. Luong Hoang Huong for orienting and supporting the group in the past.

Although tried to implement the topic well. However, due to technical problems as well as knowledge limitations in RMI programming, errors are inevitable. I look forward to receiving your suggestions, teachers and friends to make the subject more complete, and create a better premise for future research.

  Thank you sincerely!

**Table of contents**

**Introduce** .............................................................................................................**4**

1. Question………………………………………………………………...4
2. Subject title……………………………………………………………..4
3. Content of the subject…………………………………………………..4
4. Thesis layout……………………………...............................................5

CONTENT……………………………………………………………………….6

1. CHAPTER 1 - DESCRIPTION OF THE PROBLEM…………………6

I.1 THEORETICAL BASIS ……………………………………………6

I.2 DESCRIPTION OF THE MATH……………………………………9

I.3 MATERIAL ANALYSIS……………………………………………9

I.4 Recommended Configurations*………………………………………………12*

II. CHAPTER 2 - DESIGN AND SETTING THE SOLUTION…..12

II.1 SYSTEM DESIGN………………………………………….12

II.2 INSTALLATION OF SOLUTIONS ……………………….17

1. CHAPTER 3 - TESTING AND ASSESSMENT……...30

II.1 TESTING TESTS…………………………………………...30

II.2 TEST SCORE……………………………………………….30

II.3 EVALUATION……………………………………………..31

CONCLUSION………………………………………………………..35

I. Results set……………………………………………………..35

II. LIMIT………………………………………………………...35

III. DEVELOPMENT……………………………………………35

**Introduce**

1. **QUESTION**

Vietnam is a country on the way of economic, political development, ... Accompanied with that development is the entertainment needs of every citizen. One of the entertainment needs that are most interested in is watching movies.

In return for increasing demand, movie theaters have problems selling tickets (the main form of sales is offline), as well as film management and seat numbers.

Contrary to the situation of the cinema, such areas as bill payment, online shopping, ... are very convenient because you know and understand how to apply IT into it. In this situation, in order to reduce the time for buying tickets and create favorable conditions to meet the entertainment needs of the people, the application of IT in this field is an essential requirement.

1. **Subject title**

The objective of the project is to research and write programs with NetBeans IDE 8.2 with java programming language. The software allows users to book tickets online. Using MySQL to easily manage the ticket number and seats of a movie.

The product of the topic is the software "managing a movie theater". Is an IT application in the field of entertainment.

1. **Content of the subject**

The software will be divided into 2 different operating directions and interact with each other regularly over time: 1 is admin access and 2 is customer access.

With admin access, this software helps a lot with theatrical management. Through the software, managers can refine and review information updated regularly by theater staff with the ability to view certain customer information (name, date of birth, number telephone, email) and staff information, information of each cinema room (empty seats, reserved seats, staff in charge), promotion events of the theater and time to take place. Through the above functions, the management of the cinema becomes much easier and more convenient and although the utilities are many, the operation will be very simple, easy to get used and friendly.

With the client's access, this software will be of great help in booking movie tickets, instead of having to go to the ticket booth, the customer can book tickets anywhere with this software (internet required) and can avoid running out of movie tickets or desired seats. Through the software, customers can view information and book tickets for current and upcoming theaters (name, author, showtimes, extended duration of the film), furthermore, customers If you have a seat that is available, checking your account balance or recent transactions, checking your account's personal information is also important in some cases. In the meantime, interesting events are happening in theaters.

1. **thesis layout**

The composition of the thesis consists of three parts:

  - Introduction introduces the problem, the goal of the topic, the contributions of the topic achieved.

  - The content consists of 3 chapters, chapter 1 describes the problem, chapter 2 designs and installs the solution, chapter 3 is testing and evaluation.

  - The conclusions section presents the achieved results, limitations and development directions of the topic.

- Finally, there are references and appendices.

CONTENTI. CHAPTER 1 - DESCRIPTION OF THE PROBLEMI.1 THEORETICAL BASIS

**I.1.1 Java (programming language)**

*a. Introduce*

Java is a general-purpose programming language that is class-based, object-oriented, and designed to have as few implementation dependencies as possible. It is intended to let application developers write once, run anywhere (WORA),meaning that compiled Java code can run on all platforms that support Java without the need for recompilation. Java applications are typically compiled to bytecode that can run on any Java virtual machine (JVM) regardless of the underlying computer architecture. The syntax of Java is similar to C and C++, but it has fewer low-level facilities than either of them. As of 2019, Java was one of the most popular programming languages in use according to GitHub, particularly for client-server web applications, with a reported 9 million developers.

Java was originally developed by James Gosling at Sun Microsystems (which has since been acquired by Oracle) and released in 1995 as a core component of Sun Microsystems' Java platform. The original and reference implementation Java compilers, virtual machines, and class libraries were originally released by Sun under proprietary licenses

It is used for:

* Mobile applications (specially Android apps)
* Desktop applications
* Web applications
* Web servers and application servers
* Games
* Database connection
* And much, much more!

Why Use Java?

* Java works on different platforms (Windows, Mac, Linux, Raspberry Pi, etc.)
* It is one of the most popular programming language in the world
* It is easy to learn and simple to use
* It is open-source and free
* It is secure, fast and powerful
* It has a huge community support (tens of millions of developers)
* Java is an object oriented language which gives a clear structure to programs and allows code to be reused, lowering development costs
* As Java is close to C++ and C#, it makes it easy for programmers to switch to Java or vice versa

*b. Java Syntax*

**Hello world example**

**public** **class** **HelloWorldApp** {

**public** **static** void main(String[] args) {

System.out.println("Hello World!"); *// Prints the string to the console.*

}

}

**Java Comments**

Single-line comments start with two forward slashes (//).

Multi-line comments start with /\* and ends with \*/.

**Java Variables**

In Java, there are different types of variables, for example:

* String - stores text, such as "Hello". String values are surrounded by double quotes
* int - stores integers (whole numbers), without decimals, such as 123 or -123
* float - stores floating point numbers, with decimals, such as 19.99 or -19.99
* char - stores single characters, such as 'a' or 'B'. Char values are surrounded by single quotes
* boolean - stores values with two states: true or false

**Java Data Types**

Data types are divided into two groups:

-Primitive data types - includes byte, short, int, long, float, double, boolean and char

-Non-primitive data types - such as String, Arrays and Classes (you will learn more about these in a later chapter)

**Java Methods**

*How to use method*

Example:

**Create a method** inside MyClass:

public class MyClass {

static void myMethod() {

// code to be executed

}

}

* **myMethod()** is the name of the method
* **static** means that the method belongs to the MyClass class and not an object of the MyClass class. You will learn more about objects and how to access methods through objects later in this tutorial.
* **void** means that this method does not have a return value. You will learn more about return values later in this chapter

**Call a Method**

To call a method in Java, write the method's name followed by two parentheses **()** and a semicolon**;**

public class MyClass {

static void myMethod() {

System.out.println("I just got executed!");

}

public static void main(String[] args) {

myMethod();

}

}

**I.1.2 MySQL**

*a. Introduce*

MySQL is an Oracle-backed open source relational database management system (RDBMS) based on Structured Query Language (SQL). MySQL runs on virtually all platforms, including Linux, UNIX and Windows. Although it can be used in a wide range of applications, MySQL is most often associated with web applications and online publishing.

Today, MySQL is the RDBMS behind many of the top websites in the world and countless corporate and consumer-facing web-based applications, including Facebook, Twitter and YouTube.

*b. How MySQL works*

MySQL is based on a client-server model. The core of MySQL is MySQL server, which handles all of the database instructions (or commands). MySQL server is available as a separate program for use in a client-server networked environment and as a library that can be embedded (or linked) into seperate applications.

MySQL operates along with several utility programs which support the administration of MySQL databases. Commands are sent to MySQLServer via the MySQL client, which is installed on a computer.

MySQL was originally developed to handle large databases quickly. Although MySQL is typically installed on only one machine, it is able to send the database to multiple locations, as users are able to access it via different MySQL client interfaces. These interfaces send SQL statements to the server and then display the results.

**I.2 DESCRIPTION OF THE MATH**

**I.2.1 Introduction**

*a. Target*

The objective of this section is to provide an overview to readers about the system based on the functional descriptions of the product and the specification of each function to show the relationship between the components in the product.

As a basis for analysis and collation for later stages of product design, development and maintenance.

This part of the document is aimed at a group of readers: designers, programmers, testers, maintainers and all those who want to learn in depth how the product works.

*b. The product*

The product includes a desktop software

Allows users to log in to the software and perform ticket purchase, seat reservation and report export statistics in excel, pdf, and text formats.

**I.2.2 Product description (Not yet)**

**I.3 MATERIAL ANALYSIS**

**I.3.1 Product functions**

Prioritizing product functions include: high, medium, low:

- High: High priority functions are functions that play a very important role, directly affecting the operation of the product.

- Medium: Medium priority functions play an important role in product operation, if these functions are not stable, the product may not function correctly even. cause malfunctions.

- Low: these functions are less important to the product, usually the extensions on the product. Lacking these functions, the product is still functioning normally.

*a. Portfolio management function*

i. Description and priority

- Description: This function is used to update the list of movies to be played so that users can choose. Includes release dates and showtimes.  
- Priority level: high

ii. Login  
- Login: Admin login to the system  
- After logging in, the admin system will have the right to update new movies to the list

iii. Functional requirements

|  |  |
| --- | --- |
| Request code | R01 |
| Name required | Film management |
| Purpose | New movie updates |
| Subjects used | Admin |
| Prerequisites | Login successfully to the system with rights  Admin |
| How it works | Not yet |
| Result | Catalog updated successfully or failed |
| Note | Must enter all required fields in each category of director such as director name, genre. |

*b. Account management function*

i. Description and priority

- Description: this function is used to update user accounts

- Priority level: high.

ii. Login

- Login: User login to the system with username and password

- After successfully logging into the system with an account user, select manager, select Manage account and conduct account management, the system will notify success or failure.

iii. Functional requirements

|  |  |
| --- | --- |
| Request code | R02 |
| Name required | Account management function |
| Purpose | Change information account |
| Subjects used | User |
| Prerequisites | Login successfully to the system |
| How it works | Not yet |
| Result | Catalog updated successfully or failed |
| Note | Need to fill out account information |

*c. Buy tickets online*

i. Description and priority

- Description: this function is used to buy tickets online

- Priority level: high.

ii. Login

- Login: User login to the system with username and password

- After successfully logging into the system with an account user, select buy tickets, choose the film and choose the seat, the system will notify success or failure.

iii. Functional requirements

|  |  |
| --- | --- |
| Request code | R03 |
| Name required | Buy tickets |
| Purpose | Buy tichkets online, no need to go to movie theater |
| Subjects used | User |
| Prerequisites | Login successfully to the system |
| How it works | Not yet |
| Result | Catalog updated successfully or failed |
| Note | Need to fill out account information |

**I.3.2 The non-functional requirements**

*a. Execution required*

If there is any error, the system must immediately notify. The notice must provide specific details to help users understand what the system is doing.

*b. Safety requirements*

After logging in for the first time, users will update personal information. Users are responsible for the information they update into the system.

*c. Security requirements*

The user password must be encrypted as MD5

*d. Software quality characteristics*

Software respond quickly: queries of users must respond promptly in a short time

**I.3.3 Other requirements**

* + The language used on the control application must be short, concise and easy to display.
  + Icons must be symbolic for the functions, so that the user can see and understand the functions behind each icon and must be arranged appropriately.

**I.4 Recommended ConfigurationsI.4.1 Hardware**

Microsoft Windows 7 Professional/Windows 8/Windows 8.1:

Processor: Intel Core i5 or equivalent

Memory: 2 GB (32-bit), 4 GB (64-bit)

Disk space: 1.5 GB of free disk space

**I.4.2 Software**

The most famous free Java IDE like NetBeans, Eclipse or JDeveloper

**II. CHAPTER 2 - DESIGN AND SETTING THE SOLUTION**

II.1 SYSTEM DESIGN

**II.1.1 Introduction**

a*.* *Purpose*

The content of chapter two describes the overview of the system, system architecture, and data.

The target audience for this chapter is movie theater owners and movie ticket customers

b. Limit

The scope of the system is to manage movie data, staff and customers and for those who want to open a movie theater

c. Overview of chapter two

This chapter consists of 5 parts:

- Part 1: Introduction

- Part 2: System overview.

- Part 3: System architecture

- Part 4: Data design

- Part 5: Install solutions.

**II.1.2 System overview**

- System functions:

• Ensuring the accuracy, consistency, ease of use and maintenance requirements.

• Functions that respond quickly to user requests.

- Context:

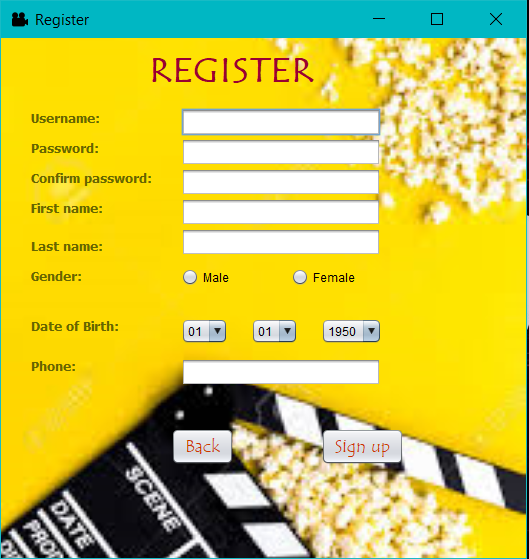
Meet the needs of managing cinemas and customer information of users

- Design:

* Friendly interface easy to use layout of logical buttons.
* Icons are suggestive, and the messages are easy to understand.
* The program is easy to understand and support the upgrade phase

**II.1.3 System architecture**

1. *Architectural design*

**

Yêu cầu hệ thống:

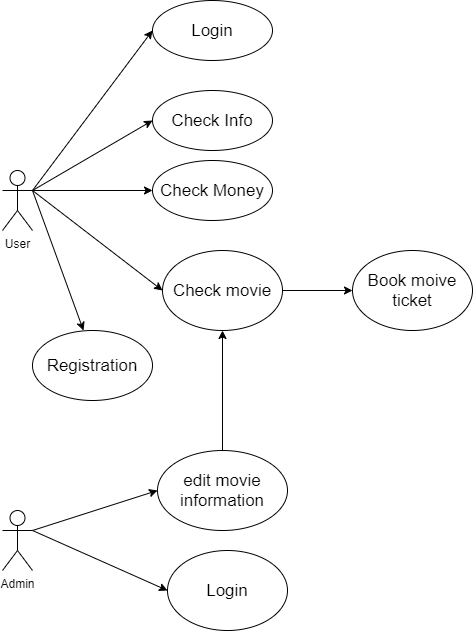
Operating system: Microsoft Windows 7 Professional/Windows 8/Windows 8.1:

Processor: Intel Core i5 or equivalent

Memory: 2 GB (32-bit), 4 GB (64-bit)

Disk space: 1.5 GB of free disk space

*b. Diagram Use case*

**

*c. Facility design*

- Design the system architecture according to each module separately and according to the order of dependencies among modules, modules dependent on other modules will be designed after the module on which it depends. This type of design is stratified, easily checks for components when errors occur in a bottom-up check, and due to the simple system, it does not take long to check from the bottom up. on.

- Design functions divided by functional groups, each group may have smaller functions

**II.1.4 Data design**

1. Data description

Product does not require magic storage. Includes customer information and money. The data will be saved to the database and if there are errors. Customers can change information easily on the application to make it more convenient for users

1. Data model

User entity

Explain: show user information. Where username is the primary key.

|  |  |  |
| --- | --- | --- |
| Attribute name | Datatypes | Explain |
| ID | nvarchar(10) | User code |
| Username | nvarchar(32) | Username of the user |
| Password | nvarchar(32) | User password |
| Firstname | nvarchar(20) | First name of user |
| Lastname | nvarchar(20) | Last name of user |
| Gender | nvarchar(6) | user gender |
| DOB | nvarchar(15) | Date of birth of user |
| Phone | nvarchar(13) | Number phone of user |
| Ticket | int | The number of movie tickets that users bought |
| Money | int | Amount that the user currently has |
| Totalmoney | int | Total amount that users have deposited |

Movie entity

Explain: show movie information. Where IDfilm is the primary key.

|  |  |  |
| --- | --- | --- |
| Attribute name | Datatypes | Explain |
| IDFilm | nvarchar(50) | Movie code |
| Name | nvarchar(50) | Name movie |
| Seat | Int | Seat number already booked |
| Booked | Int | Number of ticket user buy |
| Time | Int | Movie duration |
| Room | int | Room number |

Admin entity

Explain: show admin account information. Where username is the primary key.

|  |  |  |
| --- | --- | --- |
| Attribute name | Datatypes | Explain |
| Username | nvarchar(32) | Admin username |
| Password | nvarchar(32) | Admin password |

II.2 INSTALLATION OF SOLUTIONS

**II.2.1 Function list**

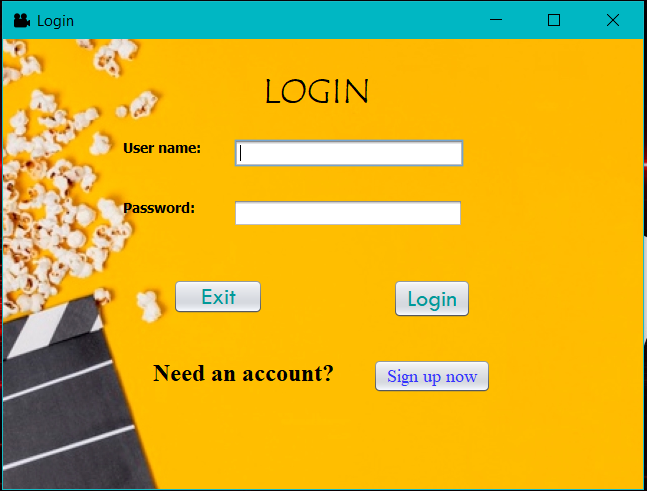
|  |  |  |  |
| --- | --- | --- | --- |
| No. | Function name | | User groups |
| 1 | Film information management | Insert new film | Admin |
| Delete flim |
| Update film information |
| 2 | Edit personal information | | User |
| 3 | Wallet | | User |
| 4 | Book and buy movie tickets | | User |

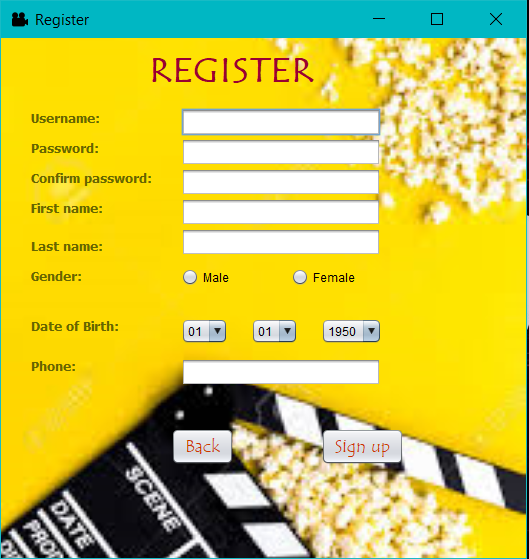
**II.2.2 The main function of the system**

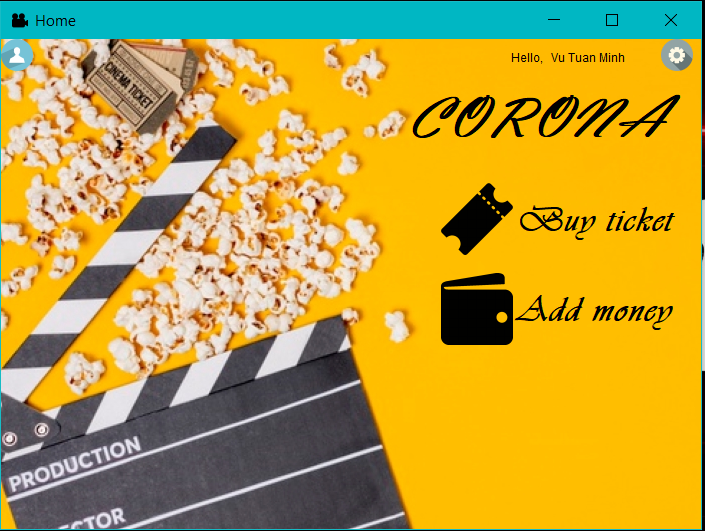
1. *Login & Register function*

Purpose: In order to protect users' personal information, we will have a login function and if users do not have an account, they can register an account easily. The user can then enter main page

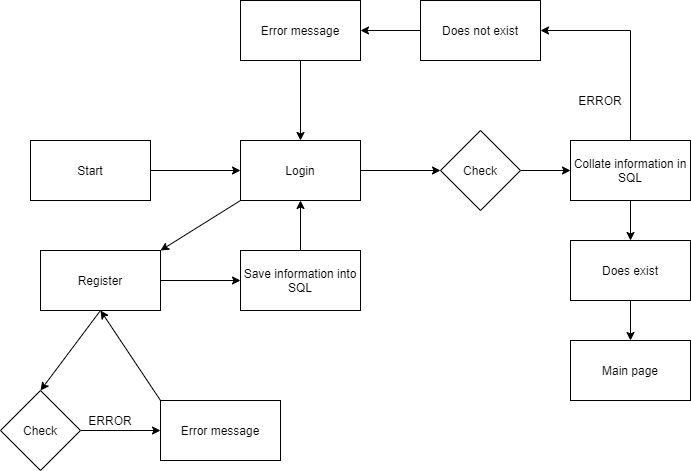
Display:

**

**

**

How it works ?



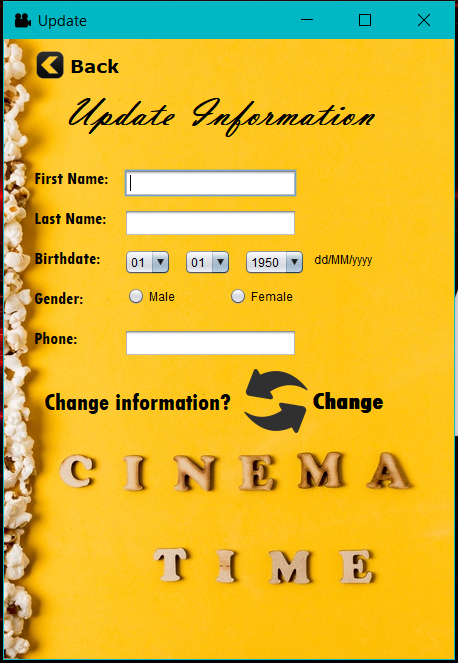
*b Edit personal information*

Purpose: In order for users to check information and provide false information, they can quickly update personal information.

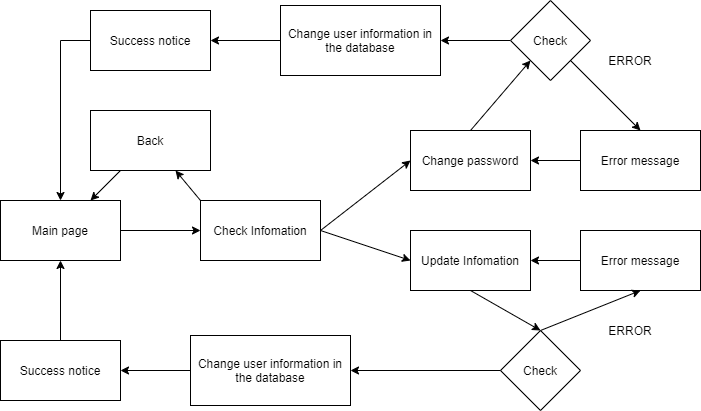
Display:







How it works ?



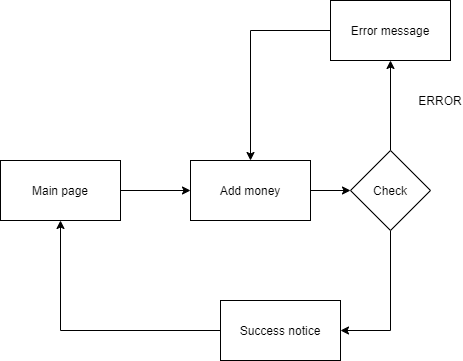
1. *Add money funtion*

Purpose: For users to add money to their accounts easily. The future can be a potential customer

Display:



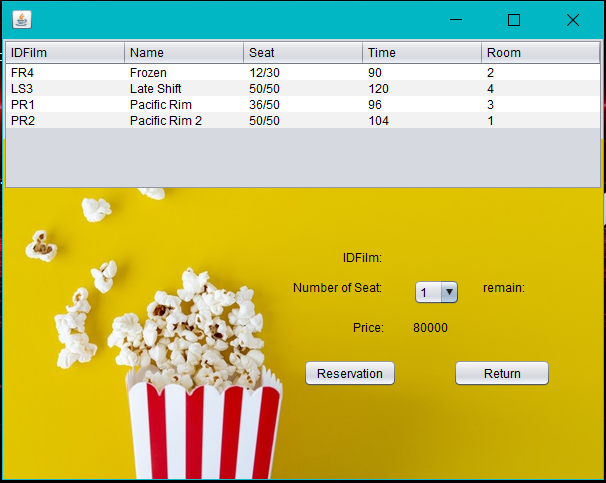
How it works ?



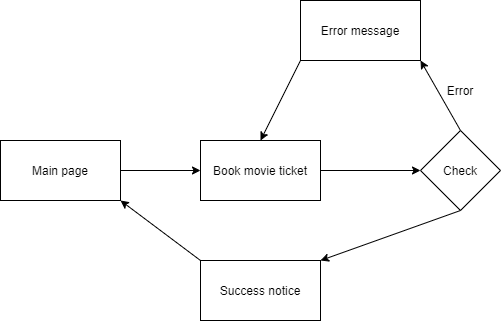
1. *Book movie ticket funtion*

Purpose: Aims to help users easily book movie tickets as well as the option to buy tickets easily without spending a lot of time

Display:



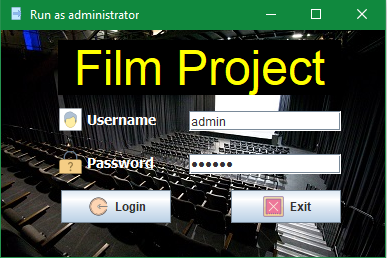
How it work ?

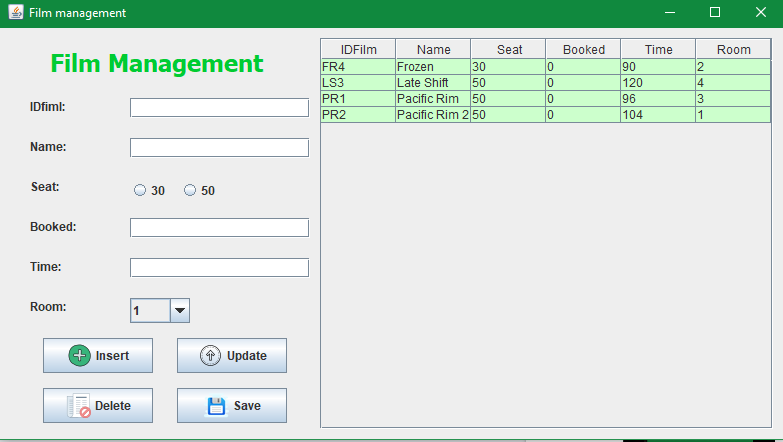


1. *Login admin funtion*

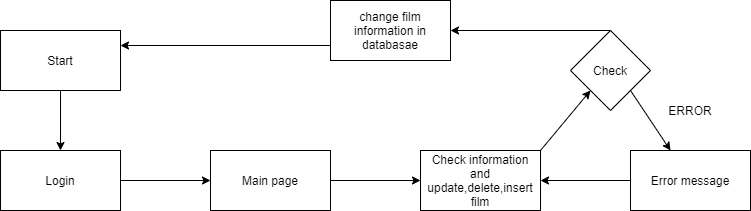
Purpose: Aims to manage film information confidentially and strictly. Provide information as quickly as possible

Display:





How it work ?



**II. CHAPTER 3 - TESTING AND ASSESSMENT**

**II.1 TESTING TESTS**

**II.1.1 Objectives**

In order to verify that the product is functioning as described, the functions function as expected. Describe test methods and find solutions for errors when they occur.

**II.1.2 Login**

Create and test test cases corresponding to functions in specification documents and design documents

**II.2 TEST SCORE**

**II.2.1 Test case**

1: Log in

* + 1. Target

In order to find errors during system log-in, see if there are errors in the case of wrong account or password input.

1. Import

User :

|  |  |  |  |
| --- | --- | --- | --- |
| Case | Account | Password | Value |
| 1 | minhvt | 184265minh | True |
| 2 | minhvt |  | Fasle |
| 3 |  | 182465minh | Fasle |
| 4 |  |  | Fasle |
| 5 | aa | 51s356 | Fasle |

Admin

|  |  |  |  |
| --- | --- | --- | --- |
| Case | Account | Password | Value |
| 1 | admin | 123456 | True |
| 2 | admin |  | False |
| 3 |  | 123456 | False |
| 4 |  |  | False |

1. Export

|  |  |  |  |
| --- | --- | --- | --- |
| Case | Desired results | Achieved | Response time T (seconds) |
| 1 | Correct account and password(user and admin) | ✓ | T<1s |
| 2 | Wrong account or password(user) | ✓ | T<1s |
| 3 | Wrong account or password(admin) | ✓ | T<1s |

1. Environmental requirements

|  |  |
| --- | --- |
|  | Minimum computer configuration |
| Operating system | Microsoft Windows 7 Professional/Windows 8/Windows 8.1 |
| CPU | Intel Core i5 or equivalent |
| Ram | 2 GB (32-bit), 4 GB (64-bit) |
| Disk space | 1.5 GB of free disk space |

**II.2.2 Test case 2: add customer account(user) and movie settings(admin)**

1. Target

To find errors during the account creation process

1. Import & Export

User

|  |  |  |  |
| --- | --- | --- | --- |
| Case |  | Achieved | Response time T (seconds) |
| 1 | Leaving blank will be error | ✓ | T<1s |
| 2 | Username already exists will be error | ✓ | T<1s |
| 3 | Check password confim | ✓ | T<1s |
| 4 | Enter phone numbers without text | ✓ | T<1s |
| 5 | Check the date of the month to be valid | ✓ | T<1s |

Admin

|  |  |  |  |
| --- | --- | --- | --- |
| Case |  | Achieved | Response time T (seconds) |
| 1 | Leaving blank will be error. Want to delete or add movies must fill out the information | ✓ | T<1s |

1. Environmental requirements

|  |  |
| --- | --- |
|  | Minimum computer configuration |
| Operating system | Microsoft Windows 7 Professional/Windows 8/Windows 8.1 |
| CPU | Intel Core i5 or equivalent |
| Ram | 2 GB (32-bit), 4 GB (64-bit) |
| Disk space | 1.5 GB of free disk space |

**II.2.3 Test case 3: Modify information (user)**

1. Target

Update user information such as name, password or check money

1. Import & Export

|  |  |  |  |
| --- | --- | --- | --- |
| Case |  | Achieved | Response time T (seconds) |
| 1 | Leaving blank will be error | ✓ | T<1s |
| 2 | Username already exists will be error | ✓ | T<1s |
| 3 | Check password confim | ✓ | T<1s |
| 4 | Enter phone numbers without text | ✓ | T<1s |
| 5 | Check the date of the month to be valid | ✓ | T<1s |

1. Environmental requirements

|  |  |
| --- | --- |
|  | Minimum computer configuration |
| Operating system | Microsoft Windows 7 Professional/Windows 8/Windows 8.1 |
| CPU | Intel Core i5 or equivalent |
| Ram | 2 GB (32-bit), 4 GB (64-bit) |
| Disk space | 1.5 GB of free disk space |

**II.2.4 Test case 4: Recharge account (User)**

1. Target

Users can add money to the account

1. Import & Export

|  |  |  |  |
| --- | --- | --- | --- |
| Case |  | Achieved | Response time T (seconds) |
| 1 | The total limit of deposit accounts is 50 million and future software upgrades they will become VIP customers | ✓ | T<1s |
| 2 | Typing text will fail | ✓ | T<1s |
| 3 | Display money information and notifications when entering less than 50k | ✓ | T<1s |

1. Environmental requirements

|  |  |
| --- | --- |
|  | Minimum computer configuration |
| Operating system | Microsoft Windows 7 Professional/Windows 8/Windows 8.1 |
| CPU | Intel Core i5 or equivalent |
| Ram | 2 GB (32-bit), 4 GB (64-bit) |
| Disk space | 1.5 GB of free disk space |

**II.2.5 Test case 5: Buy movie tickets (User)**

1. Target

Easy-to-see interface for users who want to easily select the movie they want

1. Import & Export

|  |  |  |  |
| --- | --- | --- | --- |
| Case |  | Achieved | Response time T (seconds) |
| 1 | Insufficient money can not buy movie ticket | ✓ | T<1s |
| 2 | Click the movie on the screen to buy tickets | ✓ | T<1s |
| 3 | If the seat is full, it cannot be booked | ✓ | T<1s |

1. Environmental requirements

|  |  |
| --- | --- |
|  | Minimum computer configuration |
| Operating system | Microsoft Windows 7 Professional/Windows 8/Windows 8.1 |
| CPU | Intel Core i5 or equivalent |
| Ram | 2 GB (32-bit), 4 GB (64-bit) |
| Disk space | 1.5 GB of free disk space |

II.3 EVALUATION

* Easy-to-see interface makes it easy for users to use besides displaying information that helps users do not need to remember much.
* About the function to help report errors when users make a mistake. Besides creating the function to help upgrade for later.

CONCLUSION

**I. Results set**

- Solve the requirements of the problem, combining well with the database management system, SQL Server,

  - Using the optimal hardware solution

  - Successfully researched and applied the theoretical basis of java programming, SQL Server database management system

- Preliminary completion of online booking application

  - Complete the user manual website.

**II. LIMIT**

- Sever is still null when combined with SQL

- Still having trouble getting RMI

**III. DEVELOPMENT**

- Continue developing and completing applications to make them easier to use

- Improve database and code optimization

- Further improved functions as well as easy to communicate for users