
Java Fundamentals End Assignment

Hogeschool Inholland
Teachers: Mark de Haan & Wim Wiltenburg
Submission deadline: 25-10-2021 11:59 PM

Introduction

In order to complete the course of Java Fundamentals students need to complete two tasks:

1. An end assignment
2. A classroom exam session, extending the functionality of the end assignment with three new features.

The document before you details the requirements for this end assignment.

General requirements:

- The application must be written in JavaFX. You can use the [mavenfx](#) starter repo to setup your project. SceneBuilder and FXML are NOT to be used
- The application must be submitted as a zip (not .rar, .7z, or other formats) file containing the following:
 - /src – directory
 - pom.xml
 - Readme.md
- The name of your zip file must be: [your-name]-[your-student-number]-[end-assignment].zip, e.g. **wim-wiltenburg-123456-end-assignment.zip**
- Both code quality and style will be reviewed:
 - **Style:**
 - Java Naming Convention, appropriate naming of classes, variables and methods.
 - Use proper formatting
 - Profanity will lose you points
 - **Code Quality:**
 1. Clear and concise code. Should someone other than you read the code, they should be able to understand it. Don't assume they would. Teachers are people too.
 2. Methods must not be too long. A general rule of thumb is that 30 lines for a method is too long.
 3. Use Object Oriented Principles.
 4. Think about maintainability.
- Your application must be able to start up in the IDE of the reviewer. TEST THIS!
- The screenshots in this screens are indicative. Feel free to arrange and design it to your own taste, as long as the required functionality is there.
- If you have specific usernames/password that are needed to log into the application, please share them, e.g. in a README.md file in the root of the project.
- Before creating your zip file, do a code reformatting on all your files, and optimize your imports!

Objectives

Your task is to build an application for a movie theater. This application will have two types of users: a normal USER, and an ADMIN. Each type of user has access to a specific set of functionalities, which can overlap. You need to implement at minimum:

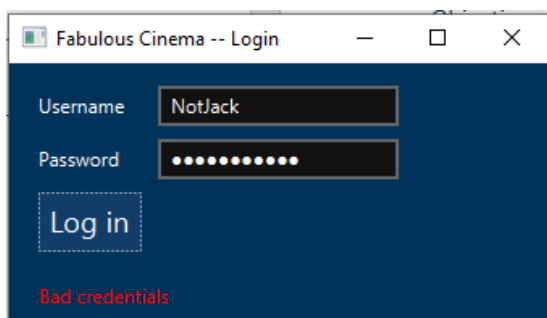
- A login screen
- A personalized menu according to type of user
- Different views:
 - A screen for selling tickets
 - A form for managing showings – accessible to the ADMIN
 - A screen for managing movies – accessible to the ADMIN
- Each view must have a title detailing the name of the cinema and what the purpose of that view is, e.g. purchase tickets
- Each view (except the login screen) will have the personalized menu on top
- You don't have to worry about pagination. Make sure any list does not overflow any page

Database object

The data in the application is maintained by a database object (not a real database) and is initialized at startup. **Important:** the data is not allowed to be static, and therefore the Singleton Pattern must **NOT** be used. Populate the database object with initial data and provide usernames & passwords in your README.md.

Login Screen

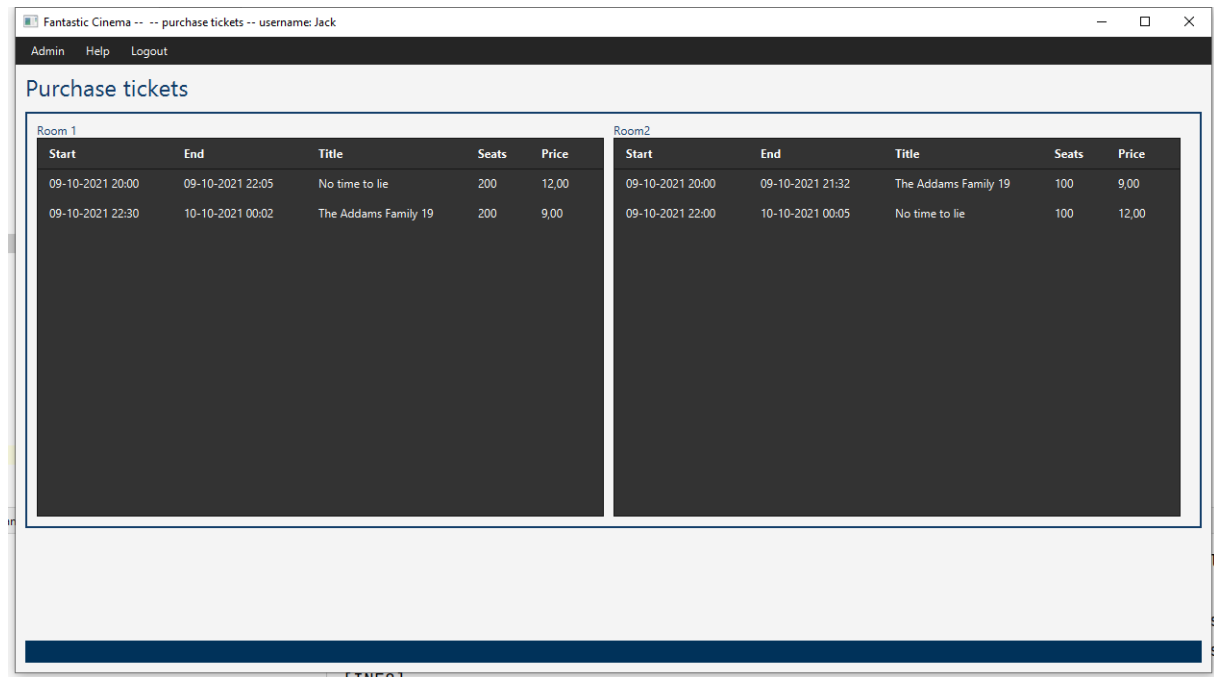
The login screen is a basic screen with username and password. You need to have a proper mechanism in place to validate the input and grant the user access to the application with the proper role.



The user must get a proper message if something goes wrong during the login process.

Main window

Once the user is logged in the main window is shown:



The cinema has two movie rooms: Room 1 and Room 2. They are shown side by side in the main window. Each room can have showings of movies.

Menu

On top of the screen is a menu. The ADMIN has all the menus and menu-items of the USER but some additional menus:

Role	Menu	Menu item
USER	Help	-
USER	Help	About (not implemented)
USER	Logout	-
USER	Logout	Logout...
ADMIN	Admin	-
ADMIN	Admin	Manage showings
ADMIN	Admin	Manage movies

Selling tickets

When the user clicks on a movie showing, a form is shown that allows the user to sell a ticket:

The screenshot shows a web application window titled "Fantastic Cinema -- -- purchase tickets -- username: Jack". The interface has a dark blue header with "Admin", "Help", and "Logout" links. Below the header, the title "Purchase tickets" is displayed. The main content area is divided into two columns, "Room 1" and "Room2", each containing a table of movie showings. Below these tables is a form with fields for "Room", "Room 1", "Movie title", "No. of seats", and "Name", along with "Purchase" and "Clear" buttons. At the bottom, there is a dark blue bar for messages.

Start	End	Title	Seats	Price
09-10-2021 20:00	09-10-2021 22:05	No time to lie	200	12,00
09-10-2021 22:30	10-10-2021 00:02	The Addams Family 19	200	9,00

Start	End	Title	Seats	Price
09-10-2021 20:00	09-10-2021 21:32	The Addams Family 19	100	9,00
09-10-2021 22:00	10-10-2021 00:05	No time to lie	100	12,00

Room	Room 1	Movie title	No time to lie
Start	09-10-2021 20:00	No. of seats	0
End	09-10-2021 22:05	Name	

Purchase

Clear

On the bottom is a message box that allows for showing error/information messages.

If the "Clear" button is clicked, the form disappears, and all the fields are cleared.

If the "Purchase" button is clicked and the number of tickets is not zero, and a name is present a confirmation pop-up window is shown, otherwise an error message must be shown.

The user can then confirm payment or cancel the purchase. If the purchase is done, then the number of tickets available in the room must decrease by the number of tickets purchased and the purchase form disappears again.

Managing showings

The ADMIN user can add showings to the system. When the 'Manage showings' menu option is clicked, a form to add showings should display. Here the user can select a movie, room and start date & time. The number of seats of the room is displayed, as well as the end time (the start time + movie duration) and the price.

Note: It should not be possible to add a showing that overlaps another showing in the same room. If the user tries to add a showing that is in the same room and overlaps with another showing, the system should notify the user that this is not possible.

Take into account that breaks between showings are necessary to allow people to leave and enter the room. There should not be any overlap with the start time minus 15 minutes, and the end time + 15 minutes.

The screenshot displays the 'Manage showings' interface for 'Fantastic Cinema'. The top navigation bar includes 'Admin', 'Help', and 'Logout'. The main content area is titled 'Manage showings' and contains two tables for Room 1 and Room 2.

Room 1				
Start	End	Title	Seats	Price
09-10-2021 20:00	09-10-2021 22:05	No time to lie	200	12,00
09-10-2021 22:30	10-10-2021 00:02	The Addams Family 19	200	9,00

Room2				
Start	End	Title	Seats	Price
09-10-2021 20:00	09-10-2021 21:32	The Addams Family 19	100	9,00
09-10-2021 22:00	10-10-2021 00:05	No time to lie	100	12,00

At the bottom, there is a form to add a new showing:

Movie title	<input type="text" value="No time to lie"/>	Start	<input type="text" value="25-10-2021"/>	<input type="text" value="20:00"/>	<input type="button" value="Add showing"/>
Room	<input type="text" value="Room 1"/>	End	<input type="text" value="25-10-2021 22:00"/>		
No. of seats	<input type="text" value="200"/>	Price	<input type="text" value="12.50"/>		<input type="button" value="Clear"/>

Managing movies

The ADMIN user can add movies to the system. When the 'Manage movies' menu option is clicked, a list of movies should display, as well as a form to add a movie to the list.

For movies, it is necessary to store at least the name, ticket price and running time.

Logging out

In the top menu there's an option to log out of the application. That allows for switching of the user. Using this option will close the main window and opens the login window. *This must not result in a loss of data or re-initialization of the database object.*