

BBM104

Programming Assignment 4 Report

HUCS Cinema Reservation System

By: Batuhan Efe Alkış
2210765035
13th May 2022

Index

1. The Problem	1
2. The Solution Approach	1
3. The Problems and Fixing Them	1

1. The Problem

In this assignment, creating a basic cinema booking system and a graphical user interface for the application were wanted. Using only JavaFX code was wanted, SceneBuilder and FXML were prohibited to use. Basic things that were wanted are to create Login and Sign Up pages, a welcome page different for admins and non-admins, the ability to add and remove films, editing users, showing the trailer, modifying time and volume data, the ability to add and remove halls, buying and refunding seats and more.

2. The Solution Approach

The learning step has started since proper JavaFX files and info were already given. More than enough of the codes that needed to use were already in the files, so learning didn't take too much. After, building the GUI step has started. Firstly, the login screen has been built using a primary GridPane and a scene. It wasn't enough though – proper alignment has to be done. After getting familiar with setAlignment and setPadding methods, the login screen was completed. The same steps were used to build almost all pages as well. Then to make buttons active, EventHandler -or more simply, lambda expressions- were needed. After some slight learning curve, the buttons and their actions were completed. The third step of the solution was to use media and files correctly. To do that a test class was created. Playing around with media for three to five hours, media and files were added to the base code. The last step before debugging was to create the seat page. After some trials and help from StackOverflow, the page was completed. There was only one step left and it was debugging. Most parts of the code needed to be changed, so they were changed and fixed.

3. The Problems and Fixing Them

Buttons disappearing when changing scenes was the first problem. After a little research, it was because the buttons were supposed to stay in one pane at a time (or something like that), and removing and adding buttons back seemed to fix the problem, so that was used.

The second problem was the media player itself. The media player was not playing the media and was not showing any errors either. Later, learned that it did not take the path correctly, switching to URI and then to String was the way to go (Thanks Teacher Gökem, really appreciated.)

The third problem came from setting up the block time on the wrong error count times. Setting up a timer seemed to be complicated and Mr. Gökem has told that threads are usable, so used threads when blocking users. Thread switches the button to unclickable, making it a (kind-of cheap) way of banning users.

The fourth problem was changing seats' status (bought, refunded, price, owner, etc.) Clicking was changing the seats one time only and making them stay stuck too. Getting images and setting the seat according to that has fixed the problem.

The last problem came because of some bugs that could've been skipped if the checklist didn't exist.

There were a lot of problems that the app had but were foreshadowed because of other errors caused by the system or compiler. This issue was fixed anyways by modifying the codes slightly.

There are some little gimmick features in the app as well (e.g. when backup doesn't exist, the app terminates itself, and some minor glitches (trailer video not stopping when switching scenes, when removing a film the choice box can't be centered correctly, when there are no halls and/or films, clicking OK will do nothing) but most of the test cases are done correctly.

4. Resources & Appendix

- [Piazza.com JavaFX slides](#)
- [Oracle.com JavaFX documents](#)
- [thenewboston youtube channel](#)
- Appendix part of the introduction PDF for hash password method