

HACETTEPE UNIVERSITY COMPUTER SCIENCE

Project Assignment 4: HUCS Cinema Reservation System

Name : Nurullah

Surname : BAŞER

NO : 2200356077

**Advisors : Görkem AKYILDIZ, Merve ÖZDEŞ, Nebi YILMAZ and Assoc.
Prof. Ali Seydi KEÇELİ**

Due Date : 13.05.2022 - Friday (23:59:59)

Index:

Problem and Solution Approach

The Functioning of the System, Problems and Solutions

Benefits of JavaFX and GUI

Benefits of This System

Extra Features

- > Show Button

- > Settings

- > Change Username

- > Change Password

- > Time Slider & Mute Button

Resources

Problem and Solution Approach

In this assignment, I developed a cinema reservation application with a Graphical User Interface (GUI) using Java programming language. There are many frameworks to develop a Graphical User Interface (GUI) application such as Swing, SWT, AWT but I use JavaFX for this assignment. This system contains many scenes, such as the login and signup scene, where the user will login or signup, choosing a movie, watching a trailer for the movie their choice, choosing a movie hall and that they can take their seats in the hall or return the seat they received. In each scene, there are buttons belonging to this scene and texts that provide information about the scene. For example, after selecting the movie, there is a scene where we can watch the trailer of the movie and choose what we want from the halls of the movie. There are a lot of functional buttons from this scene such as, where we can rewind the trailer forward or backward for 5 seconds, start the trailer at the beginning, change the sound of the trailer, stop and start the trailer, select one of the movie halls, or go back to the movie selection screen. At the same time, there are texts that give the name and duration of the film, show the total time and current time of the movie trailer. I am using a file called backup.dat to store the data that is in the system. When the system starts working, it reads this file and writes the data that changes when the system is closed back there.

First I needed to learn what the JavaFX was to design the system. Because I had no previous experience with JavaFX. In order to learn something about this, I initially studied slides of Görkem AKYILDIZ teacher. The slides helped me a lot to create the system in my head. In general, I have learned the general applications that I will use when creating the system, such as buttons, texts, and scenes. Then I gradually started building the system. First, I created the login screen. While doing this, I learned the structures and functions of things such as buttons and scenes. In later places, thanks to this information I progressed faster. In some parts, the information I learned from the slide was not enough and I did some research on the Internet. For example, I made the table I used in the "edit users" scene where we can change the administrator and member information of users by learning it from the Internet.

The Functioning of the System, Problems and Solutions

I have defined objects such as the button, hbox, vbox, text, textfield, gridpane that I will use to make the login screen. This both made it easier to access later functions, and seeing objects at the very beginning helped in designing the code. Then I defined a function called createPane that creates objects that are on the screen, sets their positions and functions. I didn't know how to hide the password here. Doing research on the Internet, I found such a thing as PasswordField. This made the text written in it as secret as a password. After the design of the login screen was finished, I ran the code to try it out, and the screen worked the way I wanted it to. But in order to try my code, I needed user information. So I opened a class called DataManager, where I read the backup file and threw them into the necessary hashmaps. Now I could test my login screen. My code was working fine, but I couldn't switch to other screens, and I didn't know how to do it either. I learned how to switch screens by doing research on the Internet. I have created a ScreenManager class to provide these transitions, and I have called methods that are static in this class to change the screen. I also created an interface called scenes because I realized that I would write the createPane method to each scene.

After the login was finished, I also finished the signup screen in the same way. Then I started making the screen we chose for the movie. Here I learned something called ChoiceBox. It was like a button that allowed me to click and select the things I threw in. I was throwing the movies I took from the backup file in there. When I finished this screen, it occurred to me to make a settings section that would allow the user to change their name and password. And I made this settings scene on this screen. I was going to the settings scene when I pressed the settings button on the movie selection screen. I will explain this scene in detail later in the extras section. In this scene, there were three separate buttons that went to the scenes of adding a movie, deleting a movie, and editing a user. I finished the movie deletion scene without much difficulty. But I had a little difficulty in the other two scenes. At first I couldn't check if the trailer existed in the movie insert scene. Then it occurred to me to check it with the try-catch block and I got the problem sorted out. I didn't know how to make a table in user editing, and I had to search and find it on the Internet. After I got all this done, I moved on to making a screen where I could watch the trailer of the movie I chose and choose a hall.

In order to play a trailer for a movie scene, I first had to learn media. Then I started doing the scene. The buttons on this scene were not so difficult. With what I have learned in media, I have done the functions of forward, rewind, stop, start. I made the scenes of adding a hall and deleting a hall, as I found out earlier in the section of adding and deleting a movie. I also have my own additions to this scene, which I will explain later.

After doing all this, I came to the part of choosing a seat that has been scaring me since I first saw the assignment. When I was doing this scene, I couldn't make it up in my head at first. Later, when I started writing slowly, I started to understand. I couldn't quite get it working at first. I didn't realize what my mistake was. After a few different attempts, he started working. But there were a few minor mistakes. I couldn't print the price information correctly. After reviewing it, I realized that I had written the price calculation incorrectly. After I fixed it, this scene is also over.

After finishing the code, I wrote a method that records changes made to the DataManager class to save changes made to the backup file that I made inside the system.

Benefits of JavaFX and GUI

JavaFX is designed to provide applications with such sophisticated Graphical User Interface (GUI) features as smooth animation, web views, audio and video playback, and styles based on Cascading Style Sheets (CSS). Among many advantages of JavaFX, JavaFX is basically a standard compatible class library. JavaFX also allows developers to define, preview, and test user interfaces for web, mobile, and desktop. Furthermore, JavaFX supports a wide range of graphics, input, and compositing controls.

GUI is the acronym for the graphical user interface. GUI allows users to interact with electronic devices, such as computers, laptops, smartphones and tablets, through graphical elements. The GUI has many elements. Input controls such as buttons, text boxes, checkboxes, list boxes, dropdown list, toggle; Navigational components such as icons, slider, tags, search field, image carousel; Information components such as message box, progress bar. The GUI has a lot of benefits. It is easy to use and understand. It is attractive because it is visually rich. It does a lot of things in a short way.

Benefits of This System

Thanks to this program, the user can log in to the system without any code knowledge, watch the trailer of the desired movie and buy seats. It is the benefit of this system that it can only use a button when doing this, and perform the operations he\she wants without any code or background information.

EXTRA FEATURES

SHOW BUTTON:

Welcome to the HUCS Cinema Reservation System!
Please enter your credentials below and click LOGIN.
You can create a new account by clicking SIGN UP button.

Username:

Password: ☐ Show

I have added a show button that showing the password I entered. This button is available in every scene where I need to type a password.

SETTINGS:

Welcome admin(Admin - Club Member)!
You can either select film below or do edits.

I have added a settings scene that shows the user's information, he can change his name and password

Welcome iamnotadmin!

Admin ✗

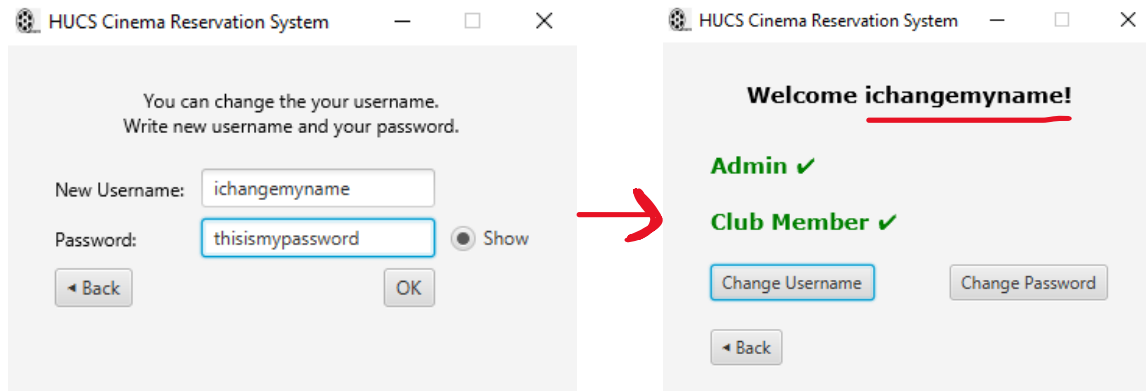
Club Member ✗

Welcome admin!

Admin ✓

Club Member ✓

CHANGE USERNAME:



HUCS Cinema Reservation System

You can change the your username.
Write new username and your password.

New Username:

Password: ☒ Show

HUCS Cinema Reservation System

Welcome ichangemyname!

Admin ✓

Club Member ✓

Errors:

ERROR: Username cannot be empty!

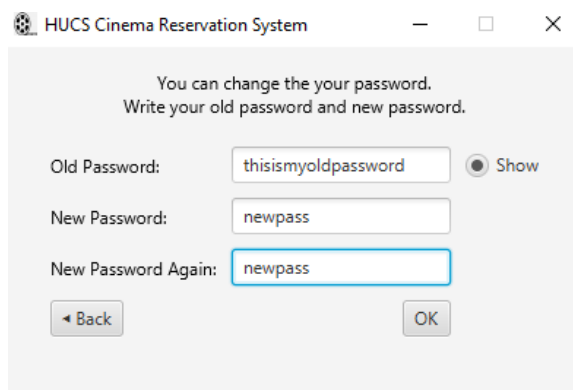
ERROR: Password cannot be empty!

ERROR: Password is wrong!

ERROR: This username already exists!

SUCCESS: You have successfully change your username!

CHANGE PASSWORD:



HUCS Cinema Reservation System

You can change the your password.
Write your old password and new password.

Old Password: ☒ Show

New Password:

New Password Again:

Errors:

ERROR: Old Password cannot be empty!

ERROR: New Password cannot be empty!

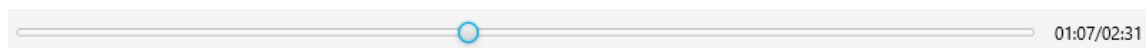
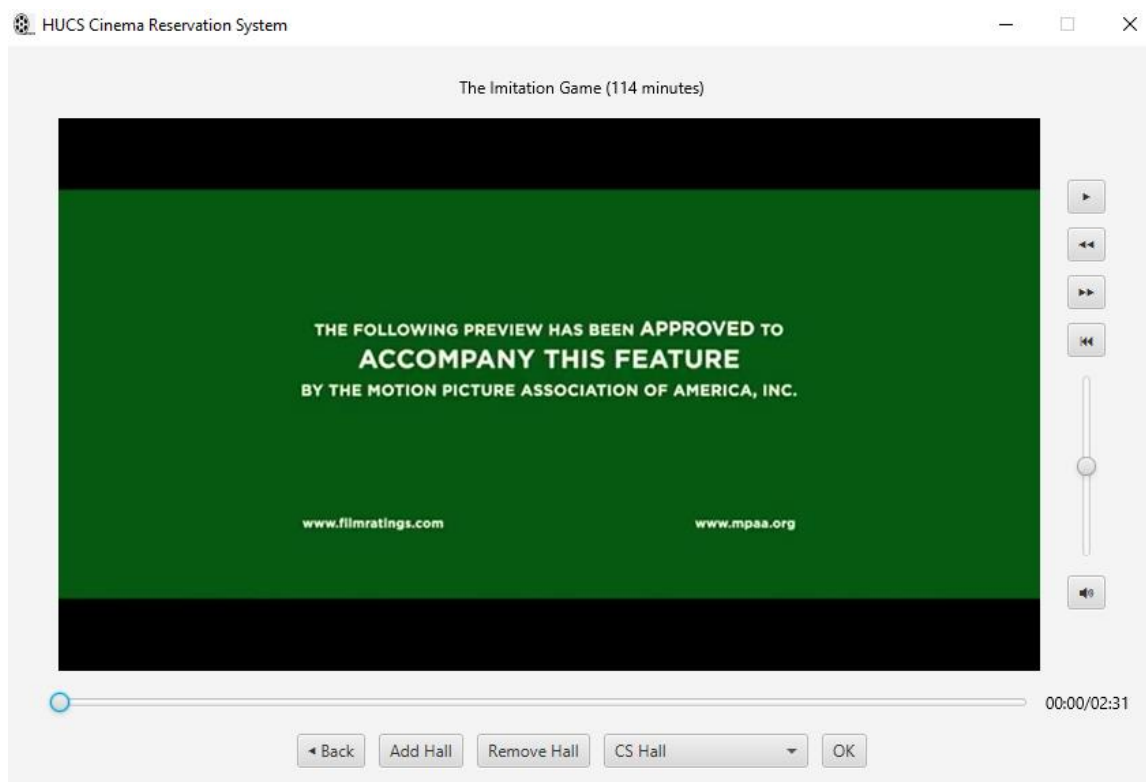
ERROR: New Password Again cannot be empty!

ERROR: Password is wrong!

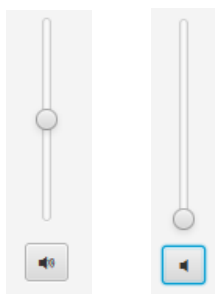
ERROR: Passwords do not match!

SUCCESS: You have successfully change your password!

TIME SLIDER & MUTE BUTTON:



I have added a time-varying slider. You can change the duration of the trailer by holding and sliding it from the Slider. There is also a text that shows the total duration of the trailer and the instantaneous duration.



You can turn off the sound completely or turn it back on by clicking the Mute button. The volume slider goes down automatically when you turn off the sound. At the same time, you can also change the sound by sliding with the volume slider.

RESOURCES:

<https://piazza.com/hacettepe.edu.tr/spring2022/bbm104/resources>

<https://zetcode.com/gui/javafx/>

<https://docs.oracle.com/javafx/2/media/playercontrol.htm>

https://docs.oracle.com/javafx/2/ui_controls/table-view.htm

<https://stackoverflow.com/questions/4044726/how-to-set-a-timer-in-java>

<https://docs.oracle.com/javafx/2/text/jfxpub-text.htm>

<https://jenkov.com/tutorials/javafx/passwordfield.html>

<https://jenkov.com/tutorials/javafx/choicebox.html>

<https://docs.oracle.com/javafx/2/swing/overview.htm>

https://devdojo.com/anvesh_kumar/the-battle-of-javafx-vs-java-swing-which-one-is-best-for-you#:~:text=Among%20many%20advantages%20of%20JavaFX,%2C%20input%2C%20and%20compositing%20controls

<https://www.indeed.com/career-advice/career-development/gui-meaning>

<https://www.youtube.com/watch?v=vego72w5kPU>

<https://unicode.org/emoji/charts/full-emoji-list.html#25b6>

<https://tr.piliapp.com/symbol/>