BBM104 ASSIGNMENT-4 REPORT

BARAN KILIÇ 2210356108

Demo link:

https://www.youtube.com/watch?v=dP-iDbNxbIY&ab_channel=BaranKılıç

THE PROBLEM

In this assignment, we were expected to make a ticket buying algorithm in different halls of a movie theater (This algorithm is expected to work in a GUI based application.). This ticket purchase process is done with users and requires registration and login screens for this (There are 3 types of users: club members only, admin only, and both club members and admins)). Club members can get discounted tickets and admins have the power to manage almost everything in this app. After logging in, new tabs are opened depending on admin and non-admin users. Depending on the window we are on and the type of user we are, different operations are performed. Processes on these different windows are of course not only for buying and selling tickets. Some of these are features such as changing the type of users, adding movies, deleting movies, adding a hall, deleting a hall, watching the trailer of the movie (this is valid for non-admin users).

THE SOLUTION

In this assignment, I used JavaFX to make a GUI based cinema application. For this, I started the program in the main class and then created a new class for each window (these classes implement an interface and I created objects of these classes in an array of that interface). In addition to these window classes, I created a processor class and most of the processing is done there. File reading operations, file printing operations, adding movies, adding halls, promote / demote processes, etc. many processes are some of these important processes. But that doesn't mean it's all done in this class. None of the GUI-related parts of the operations were done in the processes class, they were all done in the window classes. For example, transitions between windows are provided with buttons and the index in the interface array is used for each window transition. And since these button-like things are part of the GUI, they were normally inside the window classes.

Problems I faced and the solutions

The main problem was that I had no idea how to make GUI based applications. To get around this, I looked at a lot of JavaFX tutorials and practiced what I learned on the internet (JavaFX is the main topic of the assignment). After learning how to make a GUI, it was like a normal program algorithm design process. But the biggest problem with this algorithm was switching between windows. For this, I had to define an interface array and create a transition function between them. Another major problem I faced was keeping track of processes (like setting current user). I solved difficult operations like this in another class I created (process). Another problem is that I had to make duplicates in some places. Because (I think) there are some things that I do not understand and they forced me to do it. Other than these, I haven't had any major problems.

Benefits of this system

Its current state may not be very good, but when it is developed a little more, this system can be used by cinema companies as it is a cinema application. It currently includes many features such as ticket buying-selling operations, adding-removing halls, adding-removing movies. I added bot control as an extra feature. The main reason for this is to prevent machines or anything non-human from entering the system and doing harmful actions instead of real people. Of course, this is not a very advanced bot control system,

but I think a random 6-digit number is enough, at least for now. I only added this to the login section because transactions are processed after login and the security of these transactions is more important than registration. In addition, the fact that the structure of this system and all these processes is not very difficult increases the usability of this system.

Benefits of GUI

In order for people to better understand when making a process, the process needs to be easy. And users who perform this action do not need to know what is going on behind these processes. Here is a good example of GUI based applications providing this. A graphical user interface (GUI) is a type of user interface through which users interact with electronic devices via visual indicator representations. As it was said before in this assignment, we used JavaFX to make the GUI based application. JavaFX is a very useful software platform that can be used in the java programming language. The reason why especially JavaFX is used in this assignment (in my opinion) is that it is not very difficult and is suitable for beginners. In addition to being easy, its usefulness increases its importance. But of course we can make much more advanced GUI based applications using much more advanced platforms. There are many advanced frameworks to develop a GUI application (such as Swing, SWT, AWT, and the like). The platform we will choose according to its functionality may be different.

INDEX