Final Project Submission

The biggest problem that I was encountering was trying to get my android studio to work. With this problem at hand, I ended up learning how to either fix the problem that I was having, or how to work my way around it when it. One of the biggest problem that I was having was to run the flutter code through android studio because it was never detecting any devices to run the code on, whether it being the web, through the emulators that I created, or through my phone that I would plug in after setting it to developer mode. Because of this issue, I learned how to run files on the web through the terminal, how to create an emulator through the terminal, how to launch the emulators that I created through the terminal, and how to run the files on the emulator through the terminal. Another problem that I was having was not being able to get the dependencies through android studio. Whenever I would include new lines to the pubsec.yaml file, it would never get the dependencies whenever the ‘get dependencies’ or ‘upgrade dependencies’ buttons were clicked. Because of this, I ended learning how to get the new dependencies through the terminal. There was also a point where android studio stopped notifying me of errors that I had, and also stopped doing any autocorrect/autofill for me, but that ended up being resolved on its own after I uninstalled and reinstalled android studio. While testing my code on the emulator, there was a point where the widgets wouldn’t show up onto the emulator screen at all. Whenever I would try to run the file, it would only give me a white screen with nothing on it. For this problem, I ended up restarting android studio and that did end up resolving this problem that I was facing.

I did also end up learning some aspects as well in terms of coding. I learned how useful and API is when it comes to getting information that you are trying to use for you file. I learned how to find the API’s that I’m looking for, how to navigate through the API’s in a way that I would get only the information that would be useful to me, how to access this API into my file, and how to navigate through the API through code. I also learned how to display images through the API, and how to alter the image to display it how I would like for it to display. I did try to play around with a widget that would automatically scroll text from left to right like a scrolling banner but whenever I would try to use this widget, it would always give me an error that I was never able to figure out how to fix. I did learn how to convert the rating that I was given, which was in double format into icons in order to be able to make seem more appealing to the user. I ended up using a widget for this as well. I also learned how to transfer a user from the app that they are using to the browser that they have with a specific link that I would like for them to go to. I also ended up learning how to create my own widget. For the widget that I created, it was specific to displaying the movies information onto the screen, but I was able to learn the general idea of how to create my own widget and how to use it as a widget.

The purpose of this app is to show the users what is currently playing in the movie theaters and gives the user a general idea of how the movie is so the user can decide whether or not they would like to watch that movie, and once and if they do decide to watch the movie, then the app would take the user to the movie’s homepage, which then they could get more information about the movie, find the map of how to get to the movie theaters that are playing that movie, and also be able to pay for the movie online through if they decide that they would like to pay for the movie online.

Originally, I had planned to have the app display the movie’s title, the movie’s poster, the synopsis of the movie, the overall rating of the movie, and the map that shows all available movie theaters that are screening that movie, and to be able to cycle through the list of movies that are there. The way that I have the app set up right now, I did end up displaying the movie’s title, the movie’s poster, the synopsis of the movie, the overall rating of the movie, and being able to cycle through the movies available from the API, but I did end up changing the way that the user would be able to find the movie theaters that are displaying that movie. Instead of showing the map of where the available movie theaters that are screening that movie are, I decided to have it so that if the user were to click a button, then that button would open up a browser, and take the user to the movie’s homepage, which then they would be able to find the available movie theaters screening that movie, the price, the address, and even being able to pay for the movie through the movie’s homepage.

As for the user interface aspect of the app, most of what I had originally planned ended up being the same. The homepage came out looking the way that I had originally planned, and there were only a couple of changes to the second page. In the second page, instead of having a map that would display the locations, I ended up creating a button that would take the user to the movie’s homepage. Also, instead of having the buttons on the side of the movie’s poster to cycle through the available movies, I ended up putting the buttons that would allow the user to cycle through the movies at the bottom of the page. Other than that, everything ended up coming out as planned.

As for interesting design choices, I ended up finding myself using many interesting design choices. One of the interesting design choices that I ended up using was that I was originally going to have a file dedicated to getting the movies and splitting the information like we did in one of the homeworks, but after some research, I found out that there was an easier way to do accomplish this, which I ended up doing that way. I did end up also implementing part of the counter app to keep track of which movie should be displayed. I also altered the \_increment method a bit so that it would loop the starting movie with the ending movie of that list. Another interesting thing that I ended up doing was creating my own widget to be able to display the information of the movie one by one. I was also going to originally create a new file for the web client like we did in the previous homeworks, but I didn’t end up doing that, but instead put it in the file that was using this method. The reason for this was because there was only one instance in which this was being called and didn’t think it was necessary to create a new file for it to be called just once. For the sign in and log in page, I did end up doing something like what we first did when we were introduced this. I was also hoping to implement the database section for this like the other people who presented did but didn’t end up having time for this.

For the project complexity, I ended up with a total of 7 classes and 454 lines of code. As for features I used, I tried using marquee but was getting weird errors while trying to use it, I did end up successfully using URL launcher, and flutter rating bar.

Arrow

Description automatically generated with medium confidence

This is the homepage of the app.

Text, arrow

Description automatically generated

This is the homepage if nothing was inputted for the log in.

A picture containing text

Description automatically generated

This is the second page showing the title, movie poster, and the synopsis.

Text

Description automatically generated

This is the second page showing the rating, the button to launch the homepage, and two buttons, one to go to the previous movie, and one to go to the next movie.

Diagram

Description automatically generated