

Mobile Apps 2024

Assignment 2

Due: 11 April 2024 11:59 PM

Lucile Pelou

Student n°74526

GitHub: <https://github.com/Clarissee78/MobileAppCA2>

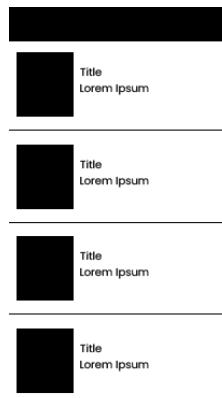
Video: <https://www.youtube.com/watch?v=HZXwKH7pLtE>

For this assignment we had to make an application allowing us to display movies and reserve (fictional) cinema seats. I have used [Google Translate](#) to translate from French to English.

Requirements Checklist:

1. Movie class created with structure from movie.json ✓
2. Data filled for at least 4 movies, including relevant movie data obtained from [imbd](#) ✓
3. Random number generated between 0 and 15 for seats_remaining for each movie ✓
4. Initial default seats_selected value set to 0 for all movies ✓
5. Show number of selected seats and hide remaining seats when seats are selected ✓
6. Clicking any item on the movie screen opens a new MovieActivity ✓
7. Plus and minus icons added, seats_selected displayed in the middle ✓
8. Seats_selected and seats_remaining updated on plus/minus click ✓
9. Validation added to disable minus when 0 seats selected and disable plus when 0 seats remaining ✓
10. Retain selected seats and reflect them on the screen when back is pressed ✓
11. Added "filling fast" badge if less than 3 seats remaining ✓
12. Used "Roboto Condensed" font to replicate same style ✓
13. Used original movie images from [impawards](#) ✓
14. README.md file included with name and student ID ✓
15. Git repository set to public and README.md with Name, Student ID and requirements checklist (with ticks) ✓
16. Video screencast link provided ✓
17. Added release with app-debug.apk file (located in the root) in the repository ✓

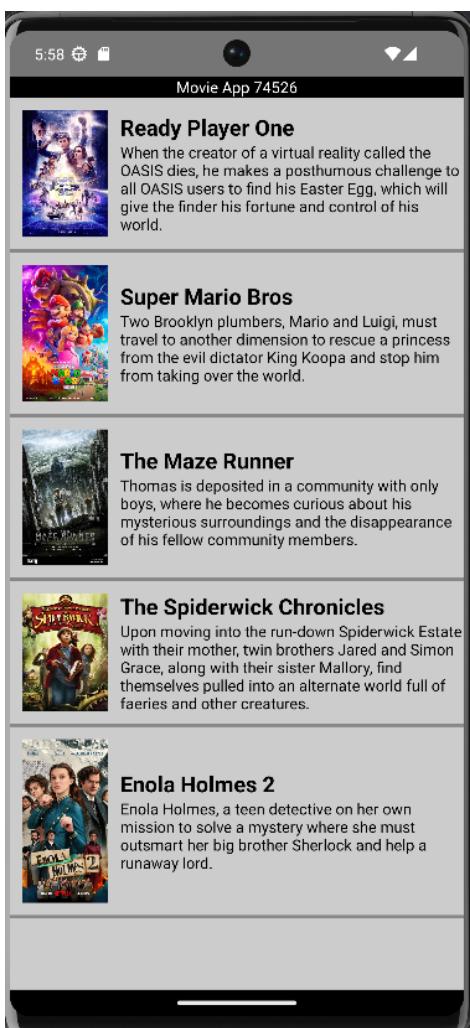
Before starting to code the assignment, I have made a little wireframe for the home page of the app. Here it is:



We weren't asked for anything in particular on this page, so I kept it as simple as the most important thing is the code and not the design.

1) Home page

After making the wireframe I started to code this first page. I first created a Movie class from the json file (ask the teacher here for the int and if so, talk about it). I found the site <http://www.impawards.com/> which provides movie posters. I also found information about the movies at <https://www.imdb.com/>. So, I was able to use these data to start my application. At the very beginning I did everything in the same class, MainActivity, I created 5 movies with my data and looped over them to display a card of each movie. The Home Page is made up of a main column which takes up all the space, this contains a box with a text for the title of the application as well as a column containing a loop iterating over all the movies. Each movie card is a button containing a row which itself contains an image and a column. This column allows you to display all the information regarding the movie. Initially I only put the title and description. So here is the result of the home page at that time:



2) Movie page

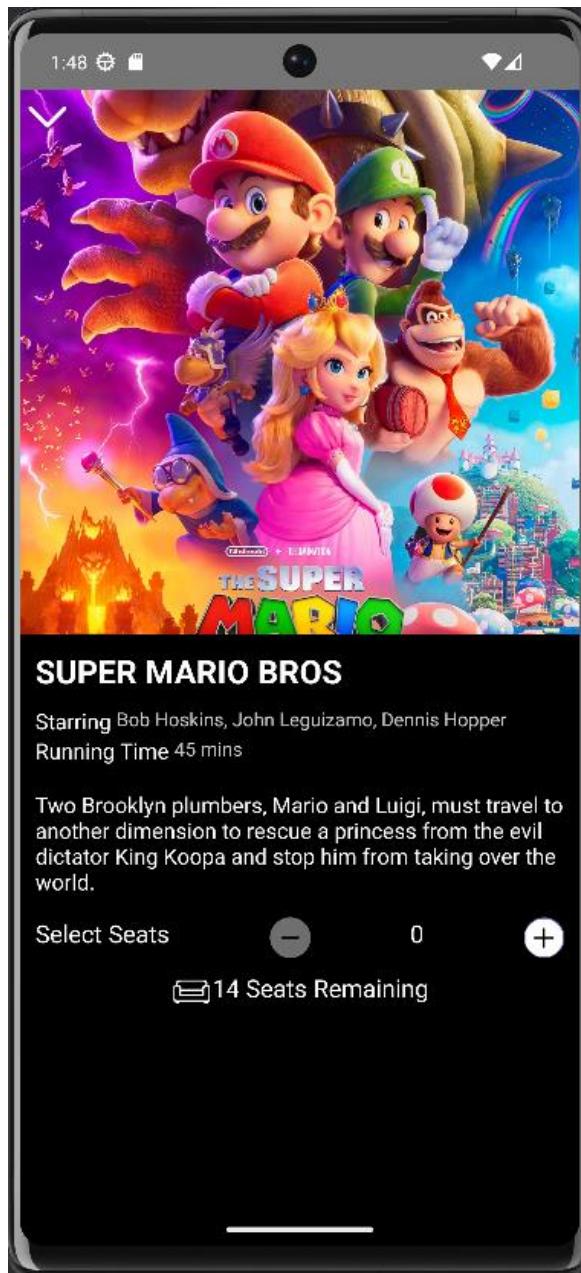
My home page being almost finished, I now had to do the Movie page. To do this I put a little order in the code and stored my Home page in a Kotlin file simply named Home. I also created a Navigation file, a MovieActivity file and a sealed class Routes. I set up the navigation logic from the home page to the MovieActivity page with the help of the videos here:

<https://www.youtube.com/watch?v=bluuWWq86AY&list=PLYDtNocY5-TeARDjhRs5flfUIR2mRVuap>. After that I also implemented passing the Movie id to the MovieActivity page using the documentation:

<https://developer.android.com/jetpack/compose/navigation?hl=en>. So, I now had access to the movie that the user had clicked on on the MovieActivity page. Before doing anything with it I have order the code a bit and created a new initMovie file which allows the data to be initialized. I also created an enum class for certifications.

The MovieActivity page contains a main column taking up all the space and centered containing a column with a box with an image and a button which will allow the user to return to the home page and containing also the rest of the body of the page. This body is a column that contains a row with text for the title and an image for the certification, a column with two rows containing two texts each for starring and the duration of the film. And the column of this bodysuit also contains the details about the seats from the movie. Depending on whether there are still places to be selected or if there are already selected, something different will be displayed in accordance with the instructions. Finally, the main column of the page also contains a column containing a box with text for the credits, which I also added to the home page later.

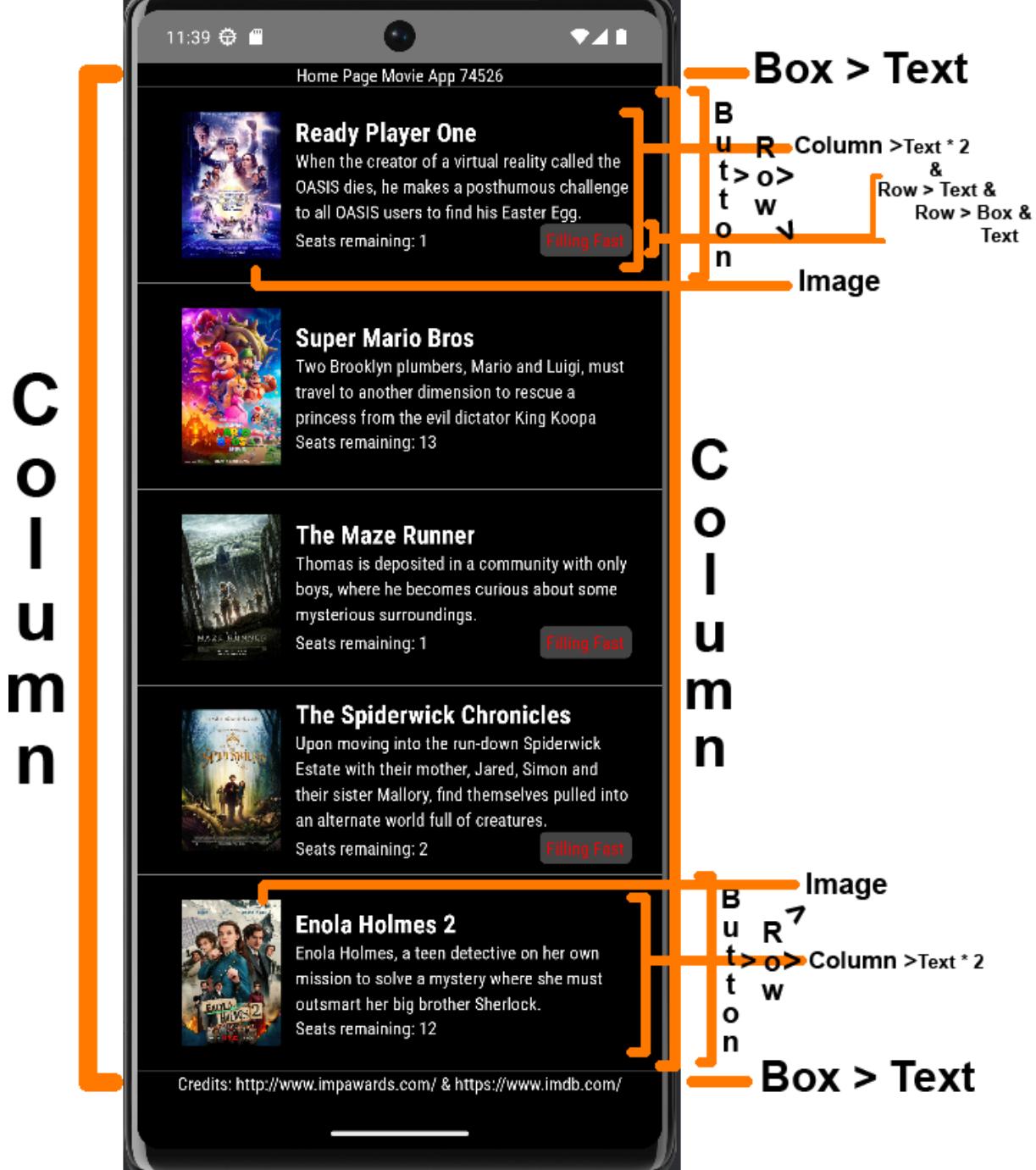
During production this is what a movie's MovieActivity page might have looked like:



I followed the instructions and added a minus and plus button which allows you to select a certain number of seats and the buttons are disabled, if necessary, for example in the image above the minus button is disabled because it cannot deselect a seat since he has selected 0 for the moment.

In terms of layout, here are graph that will speak more than words:

For the home page :



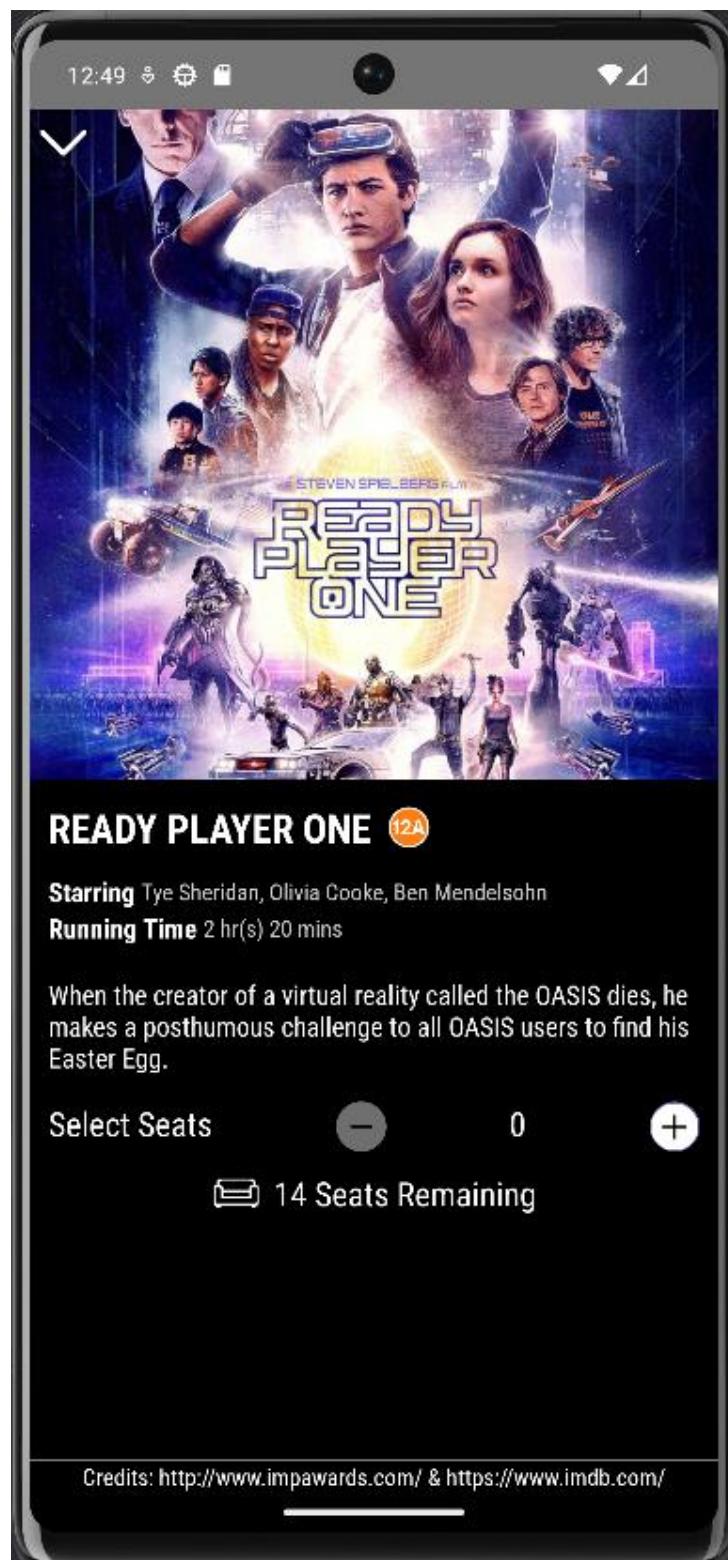
For the movie activity page:



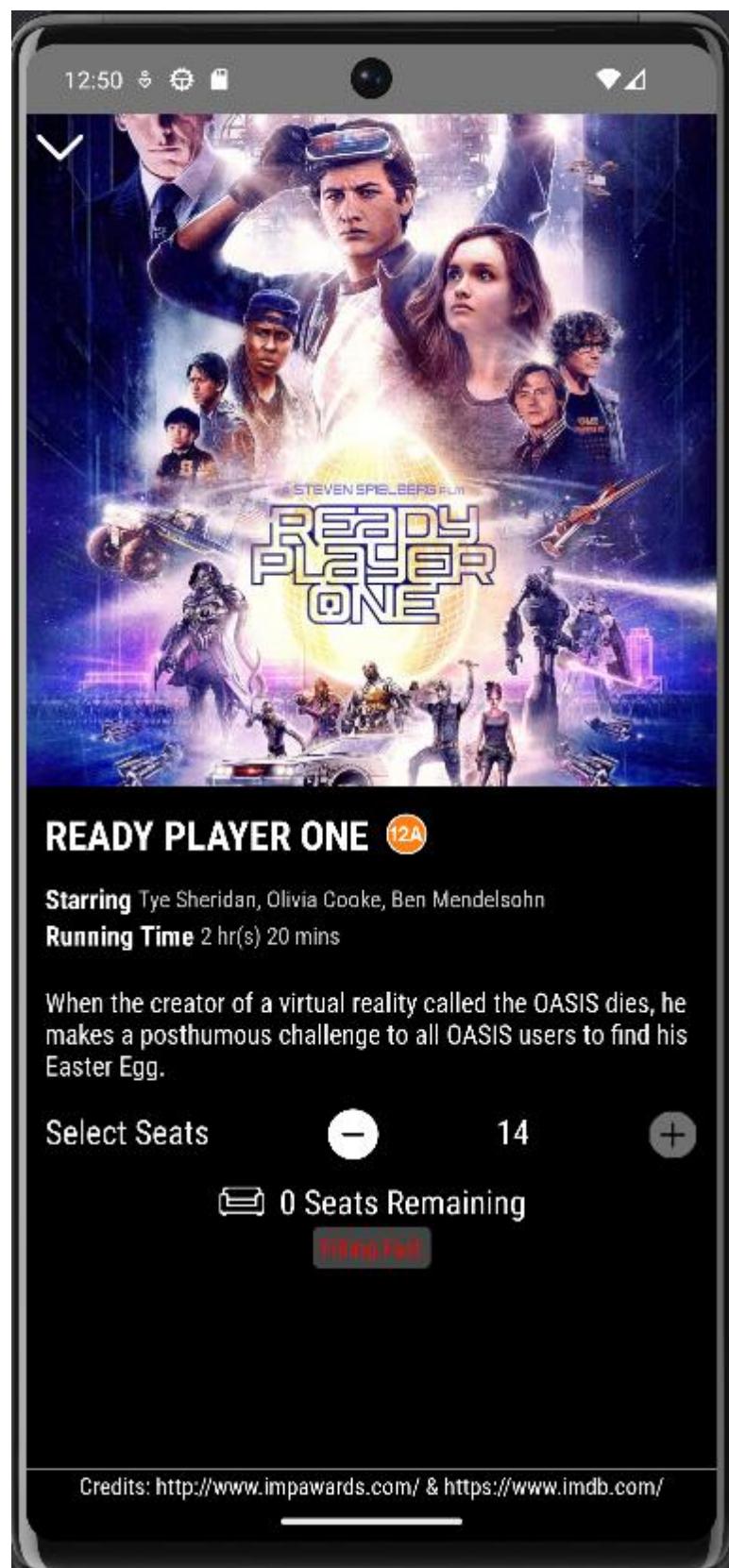
In term of arrangement, the main columns fill all the size with a colored background and for the movie page it has the vertical arrangement fixed to space between and the horizontal alignment fixed to center horizontally. Most of the boxes fill the maximum width and when there is an image in a button, the image fills all the size. Some texts like the title in the home page have bold aspect and others normal. Most of the texts have the white color and sometimes gray. Some elements have certain alignment or/and arrangement or/and padding if necessary. I didn't encounter any particular problems for this assignment.

Here are the final pages with the different aspects (depending on seat selection, etc.):

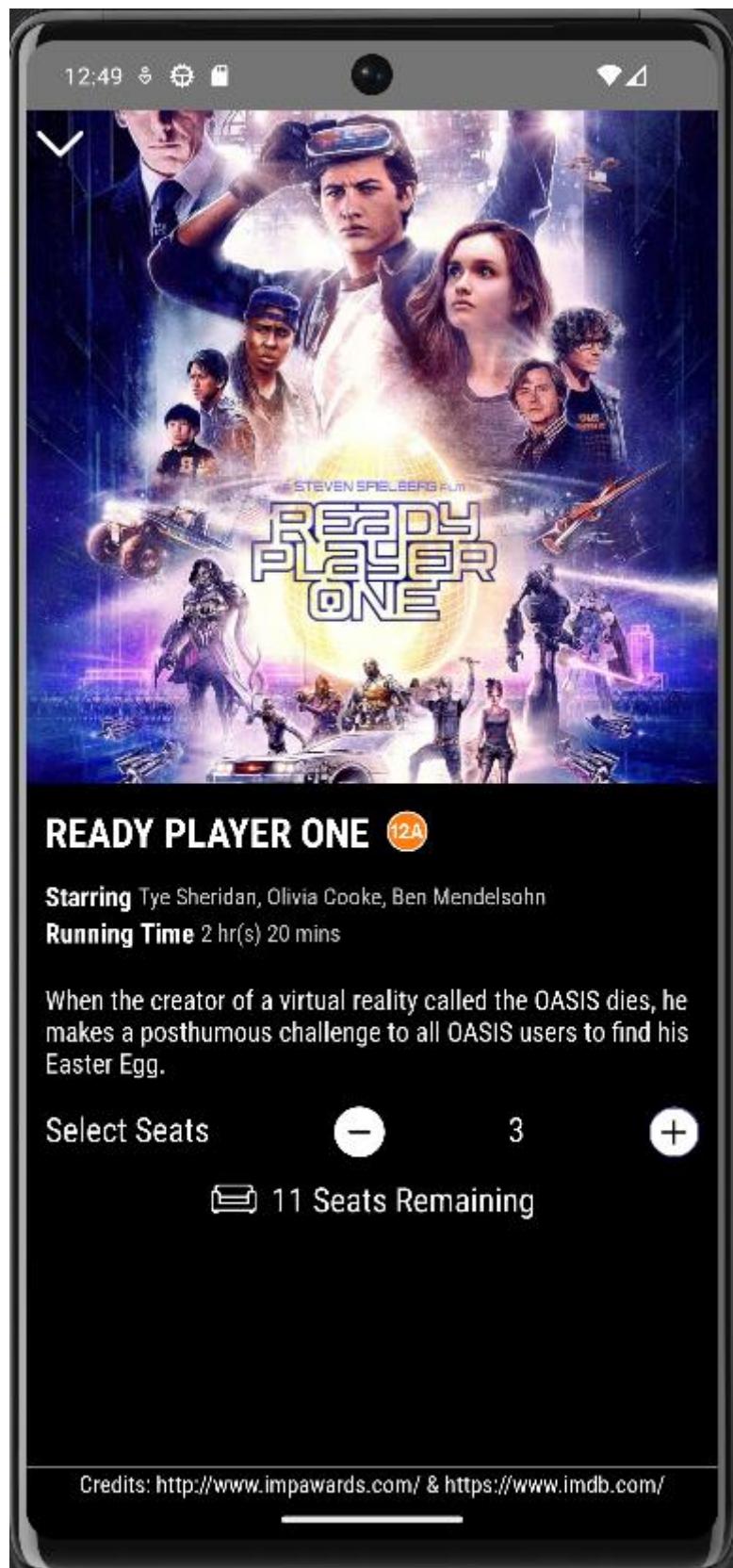
- 1) The movie page without selection:



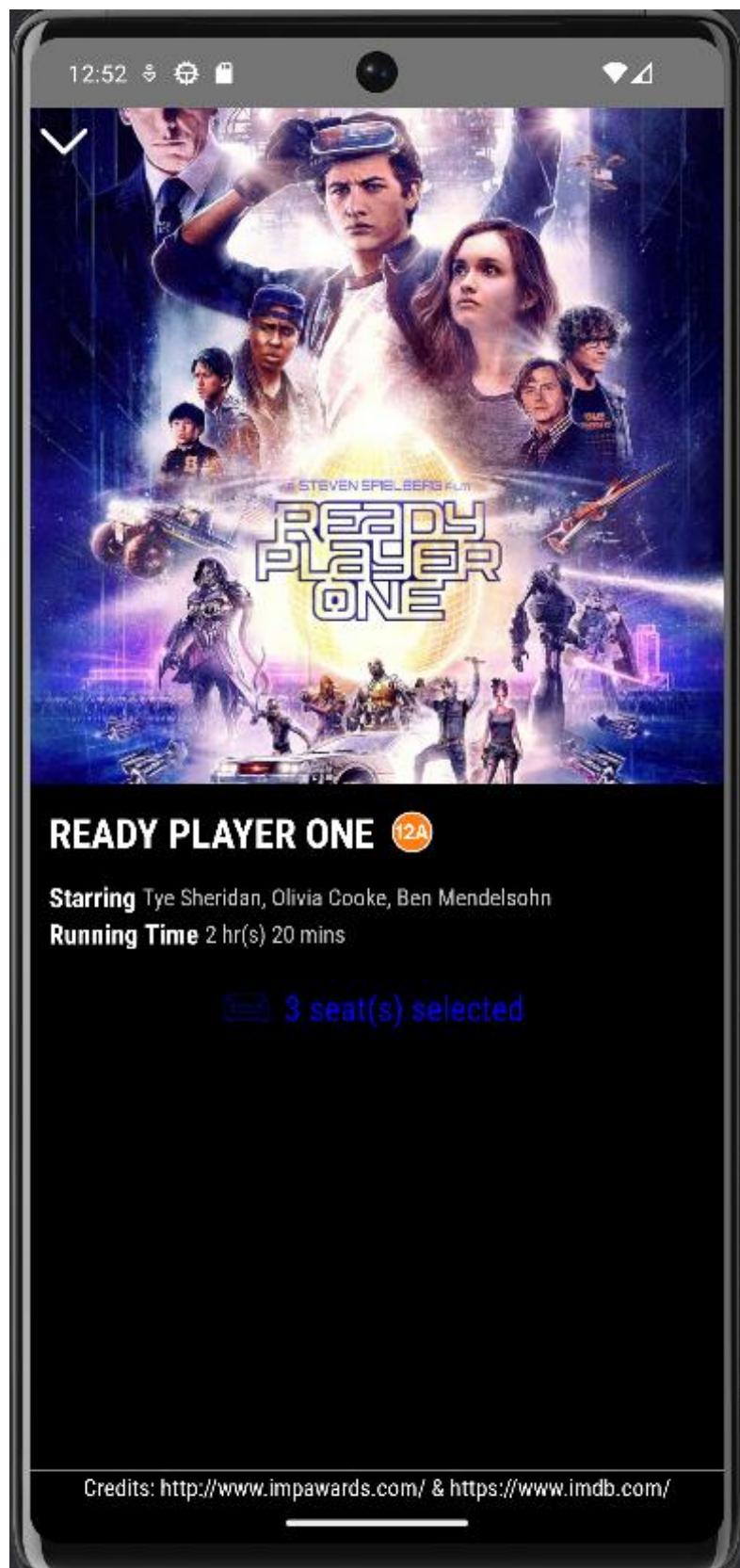
2) The movie page by selecting all available seats:



3) The movie page with selection:



4) The movie page after selection:



- 5) The home page with all selection possible (normal without selection, with filling fast badge and with selection)

