



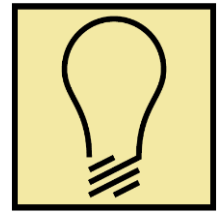
**Mobile Software
Design and Development
2022/2023**

**MoviInf
Design documentation**

Name	Student Number	E-mail address
<i>Max Sandberg Hansen</i>	<i>494247</i>	<i>maxha20@student.sdu.dk</i>



Strategy



Product Objectives

Business goals

The business goal is to make it easier for users to find information about movies and discover movies to watch, by making clear divisions between the movies, and highlight information that could be important for the users. We want to make the information easily accessible hence the business is building the app MovieInf. To prevent users from having annoying ads to earn money, the app is not aiming to earn money. To gain users the app aims to be spread between users by them referring their friends and family to use the app.

Product Overview

MovieInf is an app where users can find information about movies. The movies are divided into groups to make it easier to find a certain movie. The goal is to make basic information easy to see and find, without the need to scroll through a lot of unnecessary information.

Competitors

The possible competitors for the app could be sites such as IMDb and The movie db.

IMDb is an app that allows users to search for movies, add the movie to a watch list, rate the movie with stars from 1-10 and has all the information about the movie such as the cast etc. The app shows a lot of information that seems unnecessary for the normal user that only wants basic information about the movies here MovieInf has a competitive advantage as the app only shows the most necessary information of the movie.

The movie db is a website that provides information about movies. This is only a website and not an app for the mobile. The website shows the user ratings for the movie as well as the different information of the movie. This is not necessarily a huge competitor as this is only a website.

User Needs

Needs & goals

The goal of the app is to make uses able to:

- Easily find top rated movies
- Easily distinguish if a movie has a good rating or not
- Easily see which movies are currently trending
- See which movies are coming and is currently running in the theaters

User Research

Moviefn aims to help users in all ages with finding the next movie to watch. There are a huge number of movies out in the world, and nothing is worse than watching a bad movie. Moviefn aims to help find the best movies, by sorting the movies in three generic categories, to find the most popular, top-rated, and upcoming movies.

The duration of each user session in Moviefn is going to be short, as the user quickly can find a great movie to enjoy in the late evening or night.

Personas

Dorthe



Age: 45

Occupation: Office worker

Family: Single, 2 cats

Internet use: Facebook, Internet Banking, Streaming apps

Technical profile: Has an iPhone 6, is uncomfortable with a lot of information. Feels strong in apps after learning for a while.

Dorthe is a working full-time in an office with a lot of different tasks. She is single after being in a relationship for 15 years. Dorthe likes to make time pass by watching movies in the evenings. Dorthe sometimes get stressed when there is a lot of information, where she loses the overview of the information. She likes to check movies out before watching them, to make sure the movie is a hit or miss.

Ronald



Age: 35

Occupation: Teacher

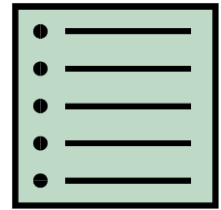
Family: Married, 1 child, 1 dog

Internet use: Facebook, Twitter, Instagram, and Streaming apps

Technical profile: Has an OnePlus 7T Pro. Is technically very strong.

Ronald is a full-time teacher. He has a wife with who he has one child and one dog. Ronald loves to see movies that is currently trending, so he can keep up with the movie talks in the institution where he teaches.

Scope



Features

The features of MovieInf are shown as functional and non-functional requirements, both being prioritized with the use of MoSCoW (Must have, Should have, Could have, Wont have).

Functional requirements

ID	Requirement	Prioritization (MoSCoW)
F01	View list of movies	M
F01.1	View list of currently popular movies	
F01.2	View list of currently top-rated movies, highest rating in the top	
F01.3	View list of upcoming movies	
F02	See details of movie	M
F03	Change color of star representing rating (Green, Orange, and Red)	S

Non-functional requirements

ID	Requirement	Prioritization (MoSCoW)
NF01	Performance	M
NF01.1	The app should run on all platforms (IOS, Android)	
NF01.2	The app should have an uptime of at least 99.7%	
NF01.3	The app should have a fast response time	
NF02	Useability	M
NF02.1	The app should be easy to navigate	

Scenarios

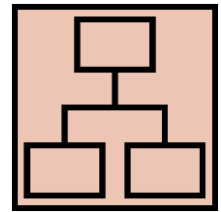
Dorthe:

Dorthe comes home after a long working day. She makes dinner and goes to the couch in the livingroom to eat. She wants to find a movie with a fairly good rating, where the description of the movie makes the movie sound good. She asks her friends on a group chat, if they can recommend any movies to watch. Dorthe has already seen the movies the friends recommend. One friend did not recommend any movies, but recommended the app MovieInf and gave her a brief overview of what the app was about. Dorthe decides to download the app and finds a movie in the top-rated section.

Ronald:

Ronald was working at the instution where he overheard a lot of kids talking about this special movie, Ronald wants to join the talk, but he has not seen any movies that are popular recently. He saw an ad of the app MovieInf, which he in the break-time decides to download. Ronald sees the popular section, and right away saw the movie the kids talked about. He read the description and decided to watch that movie with his family as soon as he gets home.

Structure



Navigation model

The navigation model for MovieInf is showed in Figure 1. The app is made of 4 different views as the app is small, to keep the app simple and quick to use:

HomePage: This is the landing page when the user opens MovieInf, this page shows the most popular movies.

Top Rated Movies: This is a page where the top-rated movies are listed.

Upcoming Movies: This is the page where the new movies and movies that are running in the theaters are listed.

MovieDetails: This page shows information about a movie.

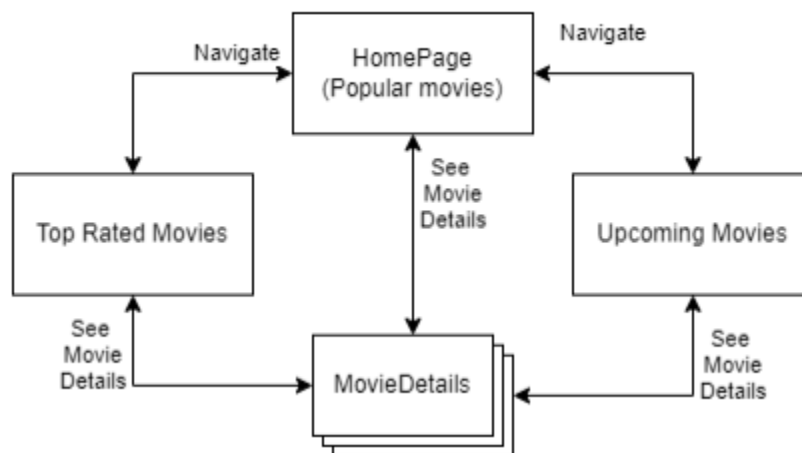


Figure 1 - Navigation flow in MovieInf

Data model

The main idea of MovieInf is to use APIs to collect movies into the different categories, the movies will therefor always be up to date, and the app will have a low cost as there is no data that needs to be stored. The data model of MovieInf is showed in Figure 2.

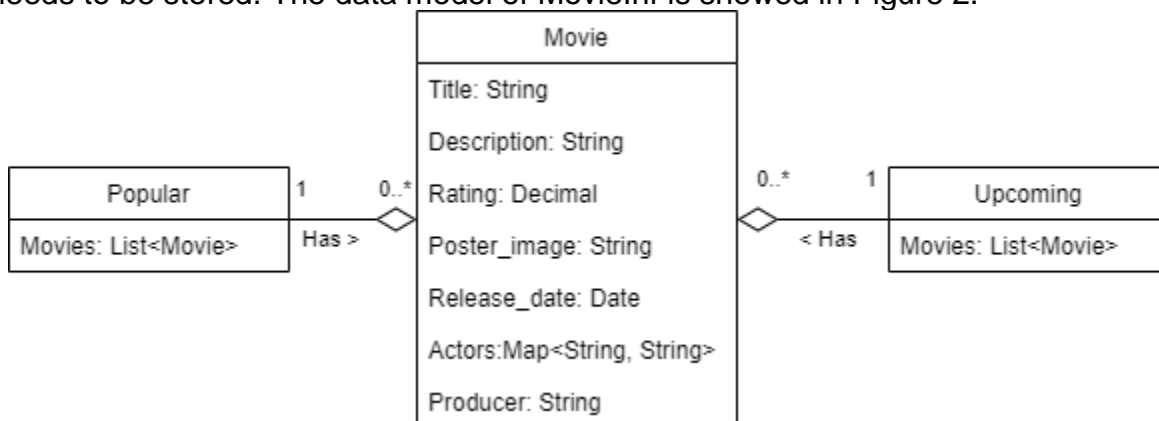


Figure 2 - Data model of MovieInf

Movie: This entity represents the movies in the app. This entity has all the information about the movie, including the actors in the movie.

Popular & Upcoming: These are entities, which has the movies in their respective area. So, the Popular entity contains the movies which currently are popular. The movies are saved in a list. A movie can be in both entities at the same time.

Skeleton

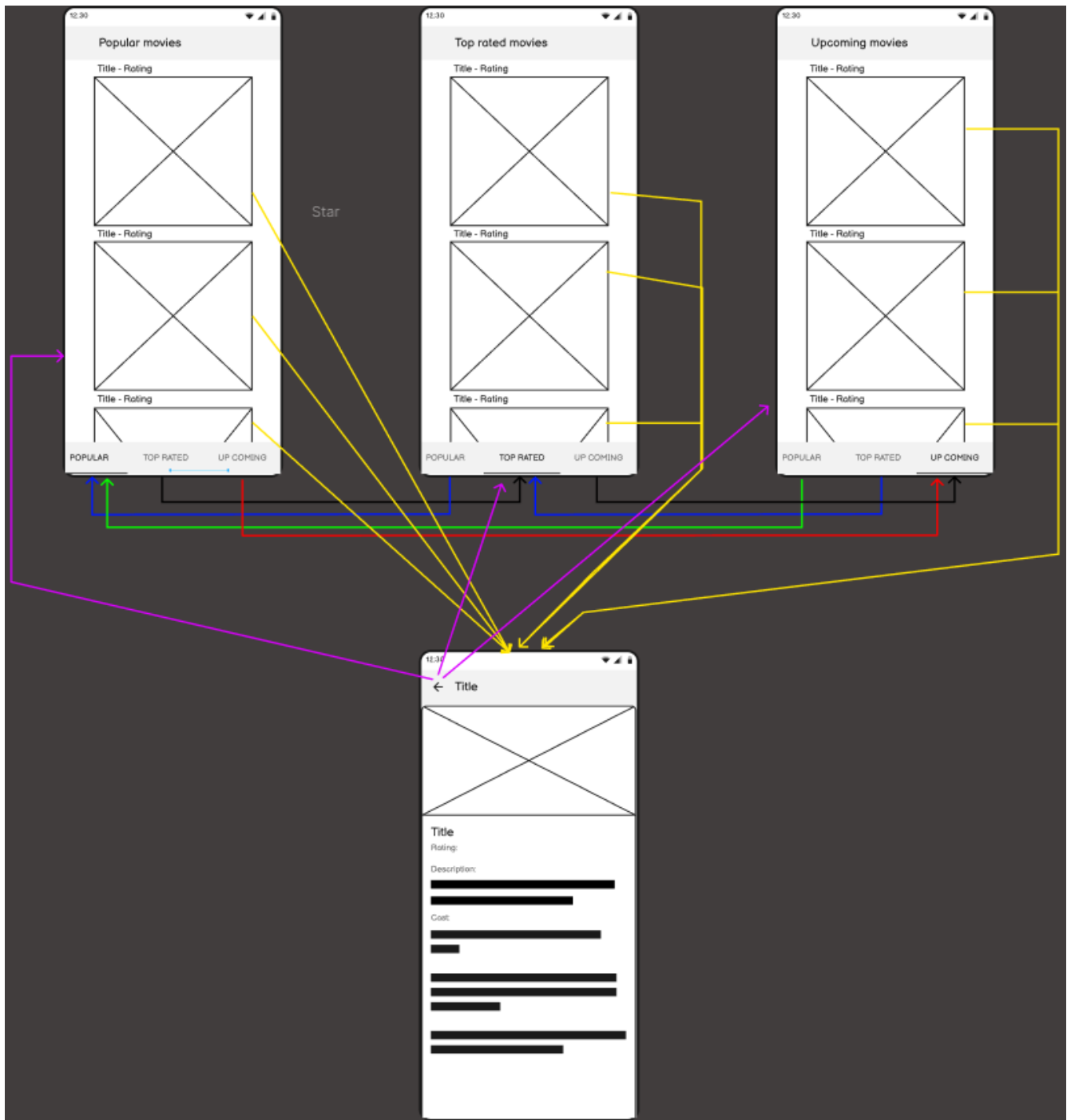
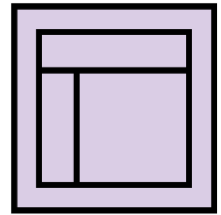


Figure 3 - LOFI wireframe of MovieInf

Popular, Top Rated, and Upcoming:

I chose to collect these 3 views under 1 description as the views are the same and has the same functionality. These views are the main views, where the “Popular movies” view is the view, the user will see when opening the app, see the view in Figure 4. These views contain a list of movies which can be scrolled through. The movies will be shown as their poster pictures, with their title and rating above the poster picture. In the bottom of the view, it is possible to switch between the three views with a tab navigation. When the user clicks on a movie picture the app will take the user to the “Movie details” view, where the movie information is shown.

Movie details:

The movie details are a view where the selected movie from the user is shown, see Figure 5. The view is built up by the movie poster in the top of the view, with the movie’s information below the picture. To go back to where the user came from, the user should then click on the arrow back, in the top left of the view.

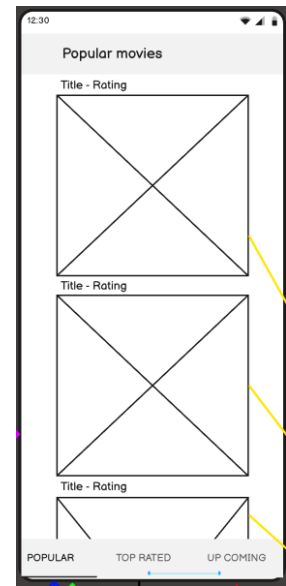
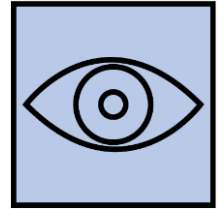


Figure 4 - Popular, Top Rated, and Upcoming example



Figure 5 - Movie details view

Surface



Theme and color palette

The app is going to be dark themed. Dark colors do not seem bright in the users' eyes at night, and it's often in the night or evening when the users are watching movies, hence protection of their eyes is important.

With the help of Adobe Colors, a color palette for the dark theme has been chosen as shown in figure 6. The red and green color seems at first a bit aggressive, but as shown in figure 8 the color fits very well with the dark background and does not seem aggressive in that matter.

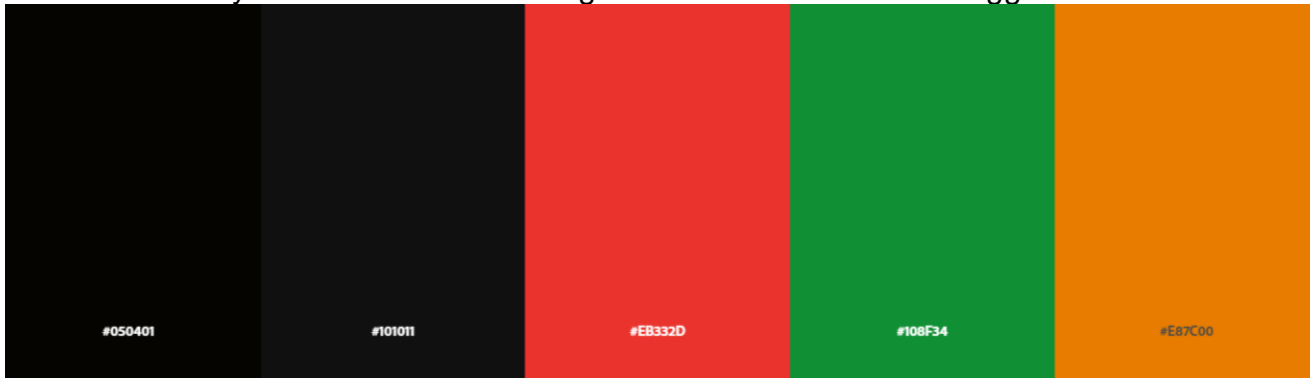


Figure 6 - Color palette

Font

The font used in the app is Roboto. Roboto has friendly and open curves. Roboto allows each letter to be in its natural with giving a more natural reading rhythm¹.

The color of the font is going to be a light font that gives contrast from the dark background. To ensure that the contrast is alright the Color Contrast Analyzer tool from Adobe is being used. The contrast is important to help the users read the text shown in the app, a low contrast can make it hard to differentiate between the background and the text, which is tiresome on the user.

Icon

The app uses one icon to represent the rating of a movie, the icon is a bootstrap star icon shown in figure 7. A star in a movie context is an indication of a rating of the given movie. The star is hollow which avoids it taking too much of the users' attention.

The color of the rating changes between Green, Orange, and Red depending on the rating of the movie. This helps the user to differentiate between a good, middle, and bad movie easily by looking at the color.



Figure 7 - Bootstrap star icon

¹ (Robertson, n.d.)

HI-FI

The combination of the pictures above together with the LO-FI wireframes, generates these HI-FI wireframes of the app. The HI-FI wireframes shown are the main features of the app, the ability to scroll through movies, and to see more details about a movie. The wireframes can be seen in figure 8.

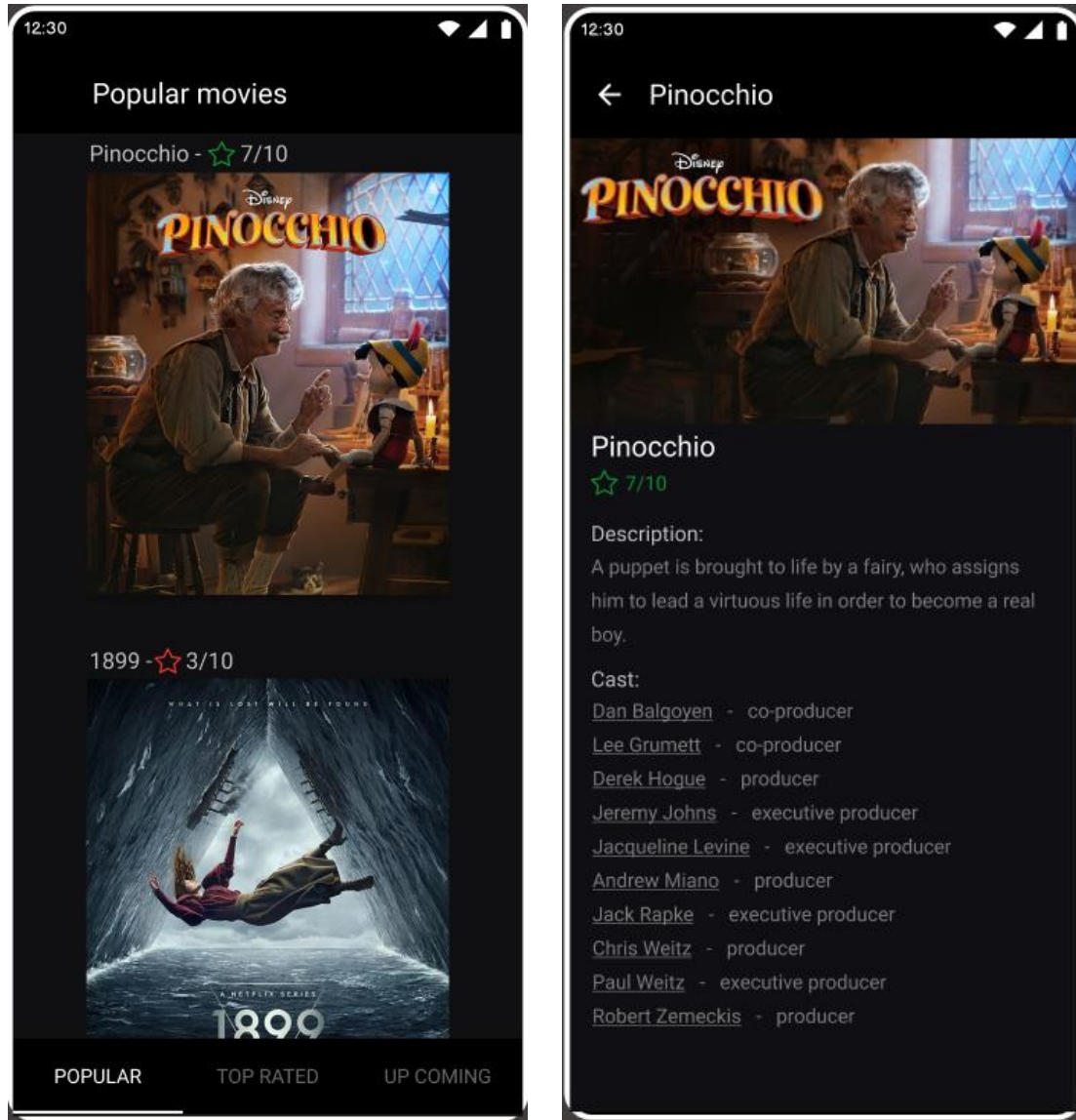


Figure 8 - HI-FI wireframes

References

Robertson, C. (n.d.). *Roboto*. (Google Fonts) Retrieved 12 12, 2022, from <https://fonts.google.com/specimen/Roboto/about>