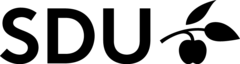
****

**Mobile Software**

**Design and Development**

**2022/2023**

**MDb-Go**

**Design documentation**

|  |  |  |
| --- | --- | --- |
| **Name** | **Student Number** | **E-mail address** |
| **Marc L. W. Bertelsen** |  | *berte20@student.sdu.dk* |

A drawing of a face

Description automatically generated with medium confidenceStrategy

# Product Objectives

# Business goals

The business goal is to make the application minimalistic while remaining informative, it can be used by everyone who want easy information about movie they might only know the name of. The application must be simple and easy to use and have few, but useful, features. The main source of income will be advertisements in the application, as to not scare uses of with a purchase in a store.

# Product Overview

MDb-Go is an application that help people find movies and details about these movies quick and easily. It will have a simple and minimal interface. MDb-Go aims to be the default choice for people when wanting to find information about a movie while on the go.

# Competitors

* IMDb: An online database that is most known for its rating system, they claim: “*IMDb is the world's most popular and authoritative source for movie, TV and celebrity content*.” They are known for ratings and reviews. Whereas MDb-Go is for getting unopinionated information about movies.
* Rotten Tomatoes: Another online database best known for a rating system, they claim: “*Rotten Tomatoes, home of the Tomatometer, is the most trusted measurement of quality for Movies & TV. The definitive site for Reviews, Trailers, Showtimes, ...*”  
  MDb-Go only aims to give information about movies.

**User Needs**

**Needs & goals**

The goal of the application is to allow uses to quick and easily find the information they want about a given movie. This needs to take as little time as possible. They want a way to find a specific movie. They want a simple way to find movies. They want to know about new movies.

# User Research

MDb-Go helps uses with finding movies and relevant information about these movies, as quickly as possible, is gives uses the ability to search for movies.

The application also gives users insight into what movies are new and trending so they can choose what movie to watch.

Using the application should take very little time.

MDb-Go only has on stakeholder, regular uses. They want an easily navigable application with a typical session having a short duration, as the application aims to be useful for people who might not have the time or need to go on more detailed review sites.

# Personas

**Cooper**

|  |  |
| --- | --- |
| Age: 56 | Occupation: Carpenter |
| Family: Married, 2 Children | Internet use: Facebook, banking and online carpentry ordering. |
| Technical profile: Not comfortable with technology, Samsung Galaxy S2 |  |

Cooper is a man who like to work with his hands, he does not have much time or need for technology, but he still like to spend time with his family, like going the cinema to watch movies. He is not always aware of what new movise are out.

**Hugo**

|  |  |
| --- | --- |
| Age: 23 | Occupation: Engineering Student |
| Family: Single | Internet use: Discord, reddit and school related |
| Technical profile: Very comfortable with technology, Google Pixel 6. |  |

Hugo is often busy with his studies, but still wants to watch movies with his friend, he like to know what a movie is about and how much time he needs to set aside, before he wants to go.

**Pauline**

|  |  |
| --- | --- |
| Age: 43 | Occupation: Accountant |
| Family: Engaged | Internet use: Facebook, banking, movie forums and work related |
| Technical profile: Somewhat comfortable with tecknology, iPhone 12 |  |

Pauline is an avid movie goer and likes to watch all kinds of movies, both old and new. She spends a lot of time at work, so she does not always have time to get up to date with the newset releases.

A picture containing text

Description automatically generatedScope

**Features**

|  |  |
| --- | --- |
| **Functional requirements** | **Description** |
| View trending movies. | The list of movies must initially show trending movies. |
| View a list of movies. | Movies must be in a scrollable list. |
| Genre selection | It should be possible to only view movies by a given genre. |
| Search to narrow the list of movies. | There must be a search field where uses can enter a search query that the movie list then shows. |
| View details of different movies. | When a user finds the movie, they are looking for they must be able to get details of that movie. |

**Other requirements**

|  |  |
| --- | --- |
| **Non-Functional requirements** | **Description** |
| No user data. | There should be no login or any other form of collection of users personal data. |
| Few view | The application should only consist of as few view as possible. |
| Minimal design. | The design of the application should simple while not looking boring, it should have few buttons and inputs. |
| Less than 3 steps. | It should take 3 or less steps to find the movie. |

**Scenarios**

**Scenario 1: Cooper**

Cooper just finished work late, be promised his family that they would have a movie night and that he would decide the movie, but it has been a long week and so he forgot to find a suitable movie for them to watch. Before he comes home, he wants to know want movie they should watch. His children told him to use MDb-Go. He opens MDb-Go and views the tending movies, to quickly find a movie he thinks they all might enjoy.

**Scenario 2: Hugo**

Hugo’s friends are all going to watch a new movie in the cinema and has asked him if he would like to join them. Hugo is current very busy with his schoolwork that need to be done soon. He has done most of the work already, he does not know the movie, and would like to know if it might be something he wants to watch. He checks the store to see if there is an app for quickly finding movies and details, and he finds MDb-Go. After quickly installing it he opens it and finds the movie on the trending list, where he can open the detailed page for the movie and find the summary of the movie.

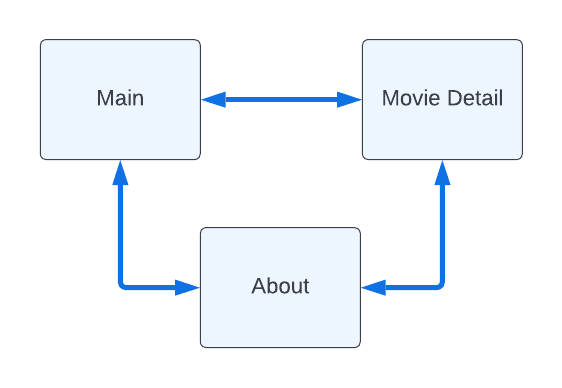
**Scenario 3: Pauline**

Pauline is planning a movie night for her friends, but she only has for lunch break to find the last movies for the event. Pauline saw someone mention MDb-Go on a movie forum, se is somewhat comfortable with technology and decides to find and install the app. She can quickly find appropriate movies for the event by selecting the genre of movie she wants.

A picture containing text, gallery, room

Description automatically generatedStructure

# Navigation model



**Main:**

This is the initial view of the app. It is here the searching and selecting of movies are done.

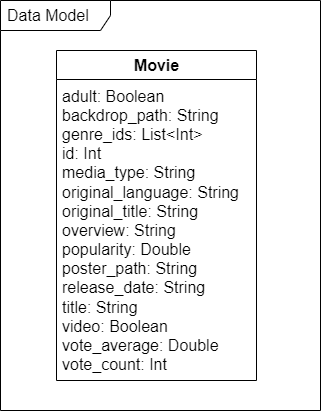
**Movie Detail:**

This view displays the detail of the selected movie from the Main view.

**About:**

This view contains information about the app. It is possible to go back to either Movie Detail of Main depending on what the previous view was.

# Data model

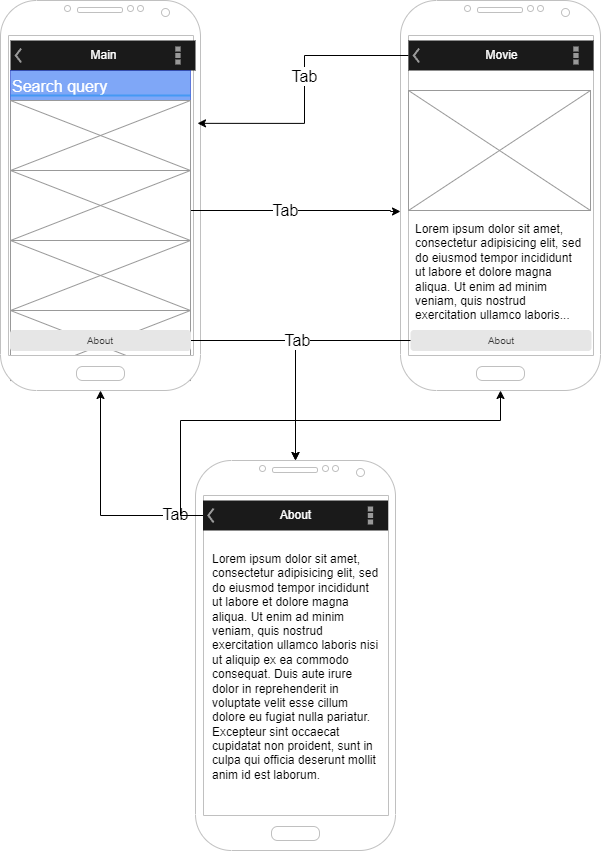


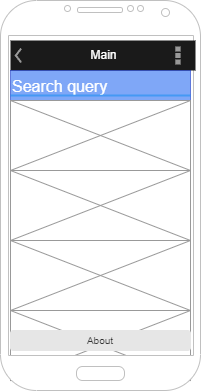
**Movie:**

The movie entity is based on the returned value from the API (TMDB), where the different attributes are used for the item in the searchable list in the Main view and to fill the Movie Detail view with the details of a given movie.

A picture containing window, table, console table, picture frame

Description automatically generatedSkeleton

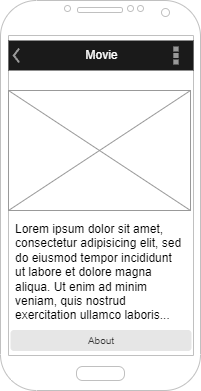
****



**Main:**

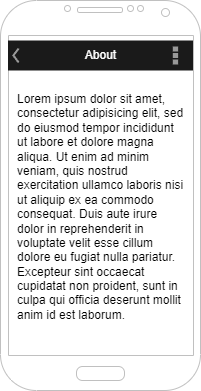
Each item in the list on movies are made up of a large image of the movie backdrop with a large font title of the movie on top.

The search field is always visible on the to

****

**Movie Detail:**

Here the details of the selected movie can be viewed. The title of movie will be at the very top, so the user can verify that they selected the correct movie. A large movie poster is show near the top. Information about the movie is shown to the right of the poster and a summary is located at the bottom.

****

**About:**

If the user wants to know information about the app itself, they can do so on this view. Ideally this view would contain app version, author/company contacts, licensing, etc.

Shape

Description automatically generatedSurface

REMOVE THE DESCRIPTION

Description of the relevant choices you made about the layout and color palette, fonts, icons, etc.

Provide a Hi-Fi Wireframes of a **single (or two) view** of your app. Please choose a representative view to show here.