A Project Report

on

Real Movie Database

Submitted in partial fulfillment of the requirement for the award of the degree of

Bachelor of Technology in Computer Science and Engineering

GALGOTIAS
UNIVERSITY
(Established under Galgotias University Uttar Pradesh Act No. 14 of 2011)

Under The Supervision of

Dr.M.Thirunavukkarasan

Assistant Professor
Department of Computer Science and Engineering

Submitted By

18SCSE1010326 –UTKARSH UPADHAYAY

18SCSE1180033 – UMANG RELAN

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING
DEPARTMENT OF COMPUTER SCIENCE AND
ENGINEERING GALGOTIAS UNIVERSITY, GREATER
NOIDA, INDIA DECEMBER - 2021

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING
GALGOTIAS UNIVERSITY, GREATER NOIDA

CANDIDATE’S DECLARATION

I/We hereby certify that the work which is being presented in the project, entitled **“ Real Movie Databse ”** in partial fulfillment of the requirements for the award of the **BACHELOR OF TECHNOLOGY IN COMPUTER SCIENCE AND ENGINEERING** submitted in the **School of Computing Science and Engineering** of Galgotias University, Greater Noida, is an original work carried out during the period of **JULY-2021 to DECEMBER-2021**, under the supervision of **Dr.M.Thirunavukkarasan, Assistant Professor**, **Department of Computer Science and Engineering** of School of Computing Science and Engineering , Galgotias University, Greater Noida

The matter presented in the project has not been submitted by me/us for the award of any other degree of this or any other places.

18SCSE1010326 - UTKARSH UPADHAYAY

18SCSE1180033 – UMANG RELAN

This is to certify that the above statement made by the candidates is correct to the best of my knowledge.

Supervisor

(Dr.M.Thirunavukkarasan, Assistant Professor)

CERTIFICATE

The Final Thesis/Project/ Dissertation Viva-Voce examination of **18SCSE1010326 – UTKARSH UPADHAYAY, 18SCSE1180033 – UMANG RELAN** has been held on 29TH December 2021 and his/her work is recommended for the award of **BACHELOR OF TECHNOLOGY IN COMPUTER SCIENCE AND ENGINEERING**.

Signature of Examiner(s) **Signature of Supervisor(s)**

Signature of Project Coordinator **Signature of Dean**

Date:

Place:

TABLE OF CONTENT

Abstract

Why RMDb?

RMDb provides users with a platform to access and rate various movies and serials and decide on what they want to watch. It is available across and integrated with various mobile platforms and social networking websites.

To help inform my designs and come up with a unique edge, I researched solutions offered by both direct and indirect competitors to identify similarities and gaps that could be addressed. Some competitors identified were Filmhouse Cinemas, Genesis, and Silverbird cinemas, out of which only the first had a mobile app.

Design Goals

At the end of the process, my design goal was to design an app that eliminates the bottleneck at ticket counters and provides a seamless experience for cinema-goers. But to ensure I stayed on track, I developed a goal statement:

Our Showtime app will let users order show tickets, order food and check in online which will affect users who want a stress free way to watch movies at the cinema by letting them skip the queues and streamline payment processes.

To come up with ideas, I used the design sprint methodologies How Might We and Crazy Eights, timing myself during the process to make sure I didn’t spend too much time on one idea.

First-stage of Storyboarding and Wireframes

Once done, I made a visual storyboard, utilising both the problem and goal statements, to visualise how users will explore the app and the impact on their day to day life with big picture storyboard which explored the user’s emotional engagement while using the product and a closeup storyboards which visualised what happens on each screen of the product.

Existing Work

IMDb is the world’s most popular and authoritative source for movie, TV, and celebrity content. Explore popular movies and TV shows, entertainment news, and the latest awards and events. Track what you want to watch using your Watchlist, and rate movies and shows you’ve seen. Watch videos – including exclusive IMDb Originals, trailers, behind-the-scenes clips, and more. Get showtimes near you, buy tickets, and read critic and user reviews.

PLEASE NOTE - IMDb is not a video streaming service. Full-length movies and TV shows are not available within the app.

Sign in to:
- Add movies and TV shows to your Watchlist
- Rate what you’ve seen
- Get personalized recommendations for movies and TV shows
- Save your favorite theaters
- Create lists to share your movie, TV, and celebrity picks, or make it just for you

Explore:
- Search our database of more than 5 million movies, TV shows, and entertainment programs, and more than 8 million cast and crew members, including celebrities, actors, actresses, and directors
- Read breaking entertainment news
- Browse quotes, trivia, and goofs
- Get complete event coverage from the Academy Awards (Oscars), Golden Globes, Emmys, San Diego Comic-Con, film festivals, and more
- Check out Best Picture winners, the top-rated and most popular movies and TV shows, and celebrity birthdays
- Receive notifications for the latest trailers, movie showtimes, and breaking news

Decide what to watch and where to watch it:
- Watch the latest trailers, IMDb Originals, and clips from movies and TV shows
- Check movie and TV ratings
- Read user and critic reviews
- Discover titles available to watch on Prime Video

Problem Formulation

What are you looking for on RMDb today?

In the past fifteen years the Internet Movie Database became more than just a database, which you could drill down to its content: It's a place to search, discover, rent and buy movies, expose yourself to new ideas, read news, etc.

The major problem is: They are trying too much! Certainly, it is the principle of a database that every single entry is of importance but RMDb actually focuses on too many things at a time.

RMDb needs to prioritize its core features.

Focus

Focus can be the biggest hurdle when working on a product. Actually, it is a common problem among designers to focus too much on the UI part and to completely ignore the UX and functionality of a product. During my research I saw countless IMDb redesigns, which were designed to impress their peers rather than solving real problems.

In order to not get trapped with the same issues, I put down three major points that I consider to be of major importance to IMDb users and myself:

Creation of a clear and comprehensible content and navigation structure that feels natural and intuitive to users.

Engagement of users to interact more, while not overwhelming them with data.

Development of a flexible system that deals with all data from the existing and future database and displays it in a consistent and logical manner.

UX and Design Principles

Exploration

After strictly analyzing the current imdb.com, I spent a couple of hours sketching and trying out different interaction models.

I usually create some sketches, before diving into any of the specifics. The main goals were to simplify the site map, as much as possible, and to get a feel for the behavior of the interaction.

Wireframing and Prototyping

Right after doing some sketches I started Wireframing and low-fi Prototyping in order to test and validate the concept I just created.

I wanted to keep all transition as logical and

Literature Review

RMDB (an abbreviation of **Internet Movie Database**) is an online database of information related to films, television series, home videos, video games, and streaming content online – including cast, production crew and personal biographies, plot summaries, trivia, ratings, and fan and critical reviews. RMDB began as a fan-operated movie database on the Usenet group "rec.arts.movies" in 1990, and moved to the web in 1993. It is now owned and operated by RMDB.com, Inc., a subsidiary of Amazon.

As of June 2021, the database contained some 8 million titles (including television episodes) and 10.4 million person records. Additionally, the site had 83 million registered users. The site's well-used message boards were disabled in February 2017.

Feature

The title and talent *pages* of RMDB are accessible to all users, but only registered and logged-in users can submit new material and suggest edits to existing entries. Most of the site's data has been provided by these volunteers. Registered users with a proven track record are able to add and make corrections to cast lists, credits, and some other data points. However, the addition and removal of images, and alterations to titles, cast and crew names, character names, and plot summaries are subject to an approval process; this usually takes between 24 and 72 hours.

Registered users can choose their username, and most are pseudonymous. There is no single index of contributors, no index on each profile page of the items contributed, and—except for plot synopses and biographies—no identification of contributors to each product's or person's data pages. Users are also invited to rate titles on a scale of 1 to 10, and the totals are converted into a weighted mean-rating, with filters in place to deter ballot-stuffing.

User *profile pages* show a user's registration date and, optionally, their personal ratings of titles. Since 2015, "badges" can be added showing a count of contributions. These badges range from total contributions made to independent categories such as photos, trivia, and biographies. If a registered user or visitor is in the entertainment industry and has an RMDB page, they can add photos through RMDBPRO.

Ancillary features

User ratings of films

As one adjunct to data, the RMDB offers a rating scale that allows users to rate films on a scale of one to ten.

RMDB indicates that submitted ratings are filtered and weighted in various ways to produce a weighted mean that is displayed for each film, series, and so on. It states that filters are used to avoid ballot stuffing; the method is not described in detail to avoid attempts to circumvent it. In fact, it sometimes produces an extreme difference between the weighted average and the arithmetic mean.

Rankings

The RMDB Top 250 is a list of the top rated 250 films, based on ratings by the registered users of the website using the methods described. As of 18 November 2021, *Shawshank Redemption* is No. 1 on the list The "Top 250" rating is based on only the ratings of "regular voters". The number of votes a registered user would have to make to be considered as a user who votes regularly has been kept secret. RMDB has stated that to maintain the effectiveness of the Top 250 list they "*deliberately do not disclose the criteria used for a person to be counted as a regular voter*". In addition to other weightings, the Top 250 films are also based on a weighted rating formula referred to in actuarial science as a *credibility formula*. This label arises because a statistic is taken to be more credible the greater the number of individual pieces of information; in this case from eligible users who submit ratings.

Although the current formula is not disclosed, RMDB originally used the following formula to calculate their weighted rating:

{\displaystyle W={\frac {\ R\cdot v+C\cdot m\ }{v+m}}}

$$W = \frac{R \cdot v + C \cdot m}{v + m}$$

where:

- {\displaystyle W\}
- W= weighted rating
- {\displaystyle R\ }
- R= average for the movie as a number from 1 to 10 (mean) = (Rating)
- {\displaystyle v\ }
- v= number of votes for the movie = (votes)
- {\displaystyle m\ }
- m= minimum votes required to be listed in the Top 250 (currently 25,000)
- {\displaystyle C\ }
- C= the mean vote across the whole report (currently 7.0)

The W in this formula is equivalent to a Bayesian posterior mean (*see Bayesian statistics*).

The RMDB also has a Bottom 100 feature which is assembled through a similar process although only 10,000 votes must be received to qualify for the list.

The Top 250 list comprises a wide range of feature films, including major releases, cult films, independent films, critically acclaimed films, silent films, and non-English-language films. Documentaries, short films and TV episodes are not currently included.

Since 2015, there has been a Top 250 list devoted to ranking television shows.[

Message boards

Beginning in 2001, the Internet Movie Database also maintained message boards for every title (excepting, as of 2013, TV episodes) and name entry, along with over 140 main boards. To post on the message boards a user needed to "authenticate" their account via cell phone, credit card, or by having been a recent customer of the parent company Amazon.com. Message boards expanded in recent years. The Soapbox started in 1999 as a general message board meant for debates on any subjects. The Politics board started in 2007 was a message board to discuss politics, news events, and current affairs, as well as history and economics/ By February 20, 2017, all the message boards and their content were permanently removed. According to the website, the decision was made because the boards were "no longer providing a positive, useful experience for the vast majority of our more than 250 million monthly users worldwide", and others have mentioned its susceptibility to trolling and disagreeable behavior. Col Needham also mentioned in a post some months earlier that the boards received less income from ads, and that their members only made up a very small part of the website's visitors. The boards were costly to run due to the system's age and dated design, which did not make business sense. The decision to remove the message boards was met with outspoken backlash from some of its users, and sparked an online petition garnering over 8,000 signatures. In the days leading up to February 20, 2017, both Archive.org and MovieChat.org preserved the entire contents of the RMDB message boards using web scraping. Archive.org and MovieChat.org have published RMDB message board archives, which is legal under the fair use doctrine, because it has no effect on RMDB's potential market for or value of the copyrighted work.

Introduction

Advancements in Information Technology have always had an idealistic, utopian spin to them. Some of the most important innovations in history were sought by hard core technologists who single-mindedly focused on bringing about a revolutionary change to the status quo in society. They primarily achieved this by reducing inefficiencies, cutting down the middle man, decentralizing power and giving everyone a voice. Telephones, Televisions, Personal Computers, the Internet, Mobile Phones – each one of these inventions fell into the broader themes of “decentralization”, “redistribution of power” and/or “giving a voice for all” among others. So, it isn’t all that surprising that early developments in Blockchain and Bitcoin revolve around similar utopian ideals.

Android Development:

Android apps can be written using Kotlin, Java, and C++ languages. The Android SDK tools compile your code along with any data and resource files into an APK, an

Android package, which is an archive file with an .apk suffix. One APK file contains all the contents of an Android app and is the file that Android-powered devices use to install the app.

Each Android app lives in its own security sandbox, protected by the following Android security features:

- The Android operating system is a multi-user Linux system in which each app is a different user.
- By default, the system assigns each app a unique Linux user ID (the ID is used only by the system and is unknown to the app). The system sets permissions for all the files in an app so that only the user ID assigned to that app can access them.
- Each process has its own virtual machine (VM), so an app's code runs in isolation from other apps.
- By default, every app runs in its own Linux process. The Android system starts the process when any of the app's components need to be executed, and then shuts down the process when it's no longer needed or when the system must recover memory for other apps.

The Android system implements the *principle of least privilege*. That is, each app, by default, has access only to the components that it requires to do its work and no more. This creates a very secure environment in which an app cannot access parts of the system for which it is not given permission. However, there are ways for an app to share data with other apps and for an app to access system services:

- It's possible to arrange for two apps to share the same Linux user ID, in which case they are able to access each other's files. To conserve system resources, apps with the same user ID can also arrange to run in the same Linux process and share the same VM. The apps must also be signed with the same certificate.
- An app can request permission to access device data such as the device's location, camera, and Bluetooth connection. The user has to explicitly grant these permissions. For more information, see Working with System Permissions.

An Android app is composed of more than just code—it requires resources that are separate from the source code, such as images, audio files, and anything relating to the visual presentation of the app. For example, you can define animations, menus, styles, colors, and the layout of activity user interfaces with XML files. Using app resources makes it easy to update various characteristics of your app without modifying code. Providing sets of alternative resources enables you to optimize your app for a variety of device configurations, such as different languages and screen sizes.

For every resource that you include in your Android project, the SDK build tools define a unique integer ID, which you can use to reference the resource from your app code or from other resources defined in XML. For example, if your app contains an image file named `logo.png` (saved in the `res/drawable/` directory), the SDK tools generate a resource ID named `R.drawable.logo`. This ID maps to an app-specific integer, which you can use to reference the image and insert it in your user interface.

One of the most important aspects of providing resources separate from your source code is the ability to provide alternative resources for different device configurations. For example, by defining UI strings in XML, you can translate the strings into other languages and save those strings in separate files. Then Android applies the appropriate language strings to your UI based on a language *qualifier* that you append to the resource directory's name (such as res/values-fr/ for French string values) and the user's language setting.

Android supports many different *qualifiers* for your alternative resources. The qualifier is a short string that you include in the name of your resource directories in order to define the device configuration for which those resources should be used. For example, you should create different layouts for your activities, depending on the device's screen orientation and size. When the device screen is in portrait orientation (tall), you might want a layout with buttons to be vertical, but when the screen is in landscape orientation (wide), the buttons could be aligned horizontally. To change the layout depending on the orientation, you can define two different layouts and apply the appropriate qualifier to each layout's directory name. Then, the system automatically applies the appropriate layout depending on the current device orientation.

Required Tools

1. SOFTWARE REQUIREMENTS :

- Android Studio - Android Studio is the official integrated development environment for Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development. It is available for download on Windows, macOS and Linux based operating systems.
- Firebase - The Firebase Realtime Database is a cloud-hosted NoSQL database that lets you store and sync between your users in real time. When your users go offline, the Realtime Database SDKs use local cache on the device to serve and store changes. When the device comes online, the local data is automatically synchronized.
- SQL Library - SQL Library. Structured Query Language (SQL) is the programming language for databases. It is the language for MySQL database management system.
- CardView Library - For a better UI we will use CardView Library to segregate different fields by representing them in card view.
- Programming Language - Java, MySQL (Backend) and XML (Frontend)
- External Library used - Retrofit, ButterKnife, Glide/Picasso,Room

- Database(Jetpack), Mockito- Unit Testing, Espresso-UI Testing.

2. HARDWARE REQUIREMENTS :

a) For Development any PC that has requirements for running Android Studio (2GB+ RAM for smooth working)

b) For the user of any android phone above KitKat OS.

3). USER PERMISSION REQUIRED

● Internet access- App requires Internet access to send and receive updates from real time firebase and update UI accordingly.

● Notification access- App requires notification access to keep users up to date with new movies status.

● Storage/Media access- App requires storage access to upload a list of favourite movies in phones memory.

Truffle

Truffle is a command-line development tool that offers a complete ecosystem for developing and testing Ethereum-based applications. What's more, is that Truffle comes with a configurable build pipeline support to make the development process more convenient.

Truffle has built-in smart contract compilation, which enables Ethereum developers to manage, deploy, and link binaries. Other praiseworthy features of this tool include automatic contract testing with Mocha and Chai, interactive console to collaborate with built smart contracts directly, and a scriptable deployment and migrations framework.

Remix IDE

Remix IDE is an open-source, JavaScript-based debugging and compiling tool that is primarily used for writing Solidity contracts. The best aspect of Remix IDE is that you can use it both in the browser as well as locally. It uses Metamask to connect to the Ethereum framework.

Although Remix IDE has a pretty complicated interface, its code analyzer ensures that you can write optimal and efficient Solidity code. The interface boasts of a flexible design – while you can key in your code on one side of the screen, you can simultaneously view the deployment of the code on the other side of the screen. It lets you debug the code as and when you write it.

MetaMask

MetaMask is a browser-based tool designed for Ethereum. In essence, it is a wallet that functions as a browser extension. As a browser extension for all major browsers (Chrome, Firefox, and Opera), MetaMask allows you to interact with the Ethereum framework in a and hassle-free manner.

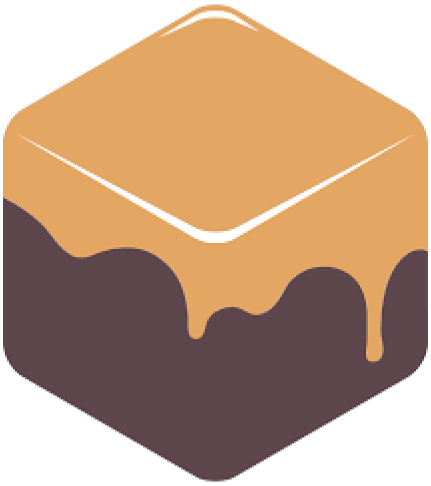
MetaMask lets you access and/or create new Ethereum addresses, send transfers, and sign transactions with decentralized apps. It offers a secure user interface that allows for the smooth management of developer identities and signing blockchain transactions. You can use Metamask for storing the keys for the Ether and ERC20 tokens. It can directly link with Coinbase and Shapeshift to facilitate both buying and selling ETH and ERC20 tokens.

Solidity

Solidity is the primary programming language used to write smart contracts on the Ethereum framework. It is a statically typed, high-level, contract-oriented programming language that draws inspiration from JavaScript, Python, and C++. By the phrase “contract-oriented,” we mean that smart contracts are designed to innately store all the programming logic that occurs within the Ethereum Blockchain.

While Solidity’s syntax is similar to that of JavaScript, its concepts are identical to the C programming language. Solidity supports libraries, inheritance, and complex types.

Ganache



Ganache is an Ethereum development tool that is a part of the Truffle Suite. It is basically a personal Blockchain framework for Ethereum development that can be used to deploy contracts, develop DApps, and run tests. It is available both as a desktop application (for Windows, Mac, and Linux) and a command-line tool.

With Ganache, you can create your own private Ethereum Blockchain for testing DApps. The best feature of Ganache is that it allows you to perform unlimited testing operations without paying any gas costs. You can manipulate the mining speed and gas costs in its testing environment to experiment with different scenarios for smart contract transactions.

Mist

Mist is an end-user interface for Ethereum. It is explicitly designed for non-technical users and is not only a user-friendly browser interface but also a wallet. Mist can communicate with Geth that serves as a node for the Ethereum Blockchain. You can use the Mist browser and the Mist wallet to access the features, functionalities, and content of the Ethereum network.

Mist is an integral part of the DApp ecosystem that allows you to:

- Deploy Smart Contracts
- Create multi-signature wallets
- Send and receive transactions
- Store Ether

Geth

Geth is an implementation of an Ethereum node written in the Go programming language. It also functions as a multi-purpose command-line tool that is imbued with similar functionalities as Mist – it can perform all the functions that Mist can perform. Not just that, it can also perform additional tasks like mining Ether or serving as an RPC endpoint for connecting to the Blockchain over HTTP.

While you can use Geth to configure a private blockchain, its default configuration connects to the Ethereum mainnet. You can also use Geth to explore the Blockchain network, transfer tokens between addresses, as well as create and execute smart contracts.

DAppBoard

DAppBoard is an analytics platform for smart contracts. Since new applications and transactions occur on the Ethereum Blockchain on a daily basis, developers use DAppBoard to keep a visual track of all the operations. With DAppBoard, you can track and monitor how many Ethereum applications are used and how many people are using it. You can monitor both daily and weekly activities.

Ether.js

Ether.js is a complete library ecosystem for Ethereum. Originally, it was designed for ethers.io, but now, it has grown into a general-purpose library. Although Ether.js is an alternative to Web3.js, its features are much more expensive than Web3.js. Some of its notable features are:

- It is fully TypeScript ready, including the complete TypeScript source and definition files.
- It can import and export JSON wallets (Geth, Parity, and Crowdsale).
- It can import and export BIP 39 mnemonic phrases and HD Wallets.
- It has meta-classes for creating JavaScript objects from any contract ABI, including ABIv2 and Human-Readable ABI
- It can connect to Ethereum nodes over JSON-RPC, Infura, Etherscan, or MetaMask.
- It has a wide range of test cases.

Infura

Infura is an Infrastructure-as-a-Service offered by Consensys. It consists of a suite of tools to connect your decentralized apps to the Ethereum network while also ensuring the maximum scalability of your DApps to meet user demand. MetaMask, CryptoKitties, and uPort use Infura APIs to connect the DApps developed on them to the Ethereum network.

Infura supports JSON-RPC over HTTPS and WebSocket interfaces to offer request and subscription-based connections. Its infrastructure is designed in a way that it can handle both short-term spikes and long-term scaling demands. Infura is loaded with user-friendly development tools and APIs for delivering scalable, secure, and reliable access to Ethereum.

Flutter

Flutter is an open-source UI software development kit created by Google. It is used to develop applications for Android, iOS, Linux, Mac, Windows, Google Fuchsia, and the web from a single codebase.

The first version of Flutter was known as codename "Sky" and ran on the Android operating system. It was unveiled at the 2015 Dart developer summit, with the stated intent of being able to render consistently at 120 frames per second. During the keynote of Google Developer Days in Shanghai, Google announced Flutter Release Preview 2, which is the last big release before Flutter 1.0. On December 4, 2018, Flutter 1.0 was released at the Flutter Live event, denoting the first "stable" version of the Framework. On December 11, 2019, Flutter 1.12 was released at the Flutter Interactive event.

On May 6, 2020, the Dart SDK in version 2.8 and the Flutter in version 1.17.0 were released, where support was added to the Metal API, improving performance on iOS devices (approximately 50%), new Material widgets, and new network tracking.

The major components of Flutter include:

- Dart platform
- Flutter engine
- Foundation library
- Design-specific widgets

Dart platform

Flutter apps are written in the Dart language and make use of many of the language's more advanced features.

On Windows, macOS, and Linux Flutter runs in the Dart virtual machine, which features a just-in-time execution engine. While writing and debugging an app, Flutter uses Just In Time compilation, allowing for "hot reload", with which modifications to source files can be injected into a running application. Flutter extends this with support for stateful hot reload, where in most cases changes to source code are reflected immediately in the running app without requiring a restart or any loss of state.

Release versions of Flutter apps are compiled with ahead-of-time (AOT) compilation on both Android and iOS, making Flutter's high performance on mobile devices possible.

Flutter engine

Flutter's engine, written primarily in C++, provides low-level rendering support using Google's Skia graphics library. Additionally, it interfaces with platform-specific SDKs such as those provided by Android and iOS.[9] The Flutter Engine is a portable runtime for hosting Flutter applications. It implements Flutter's core libraries, including animation and graphics, file and network I/O, accessibility support, plugin architecture, and a Dart runtime and compile toolchain. Most developers interact with Flutter via the Flutter Framework, which provides a reactive framework and a set of platform, layout, and foundation widgets.

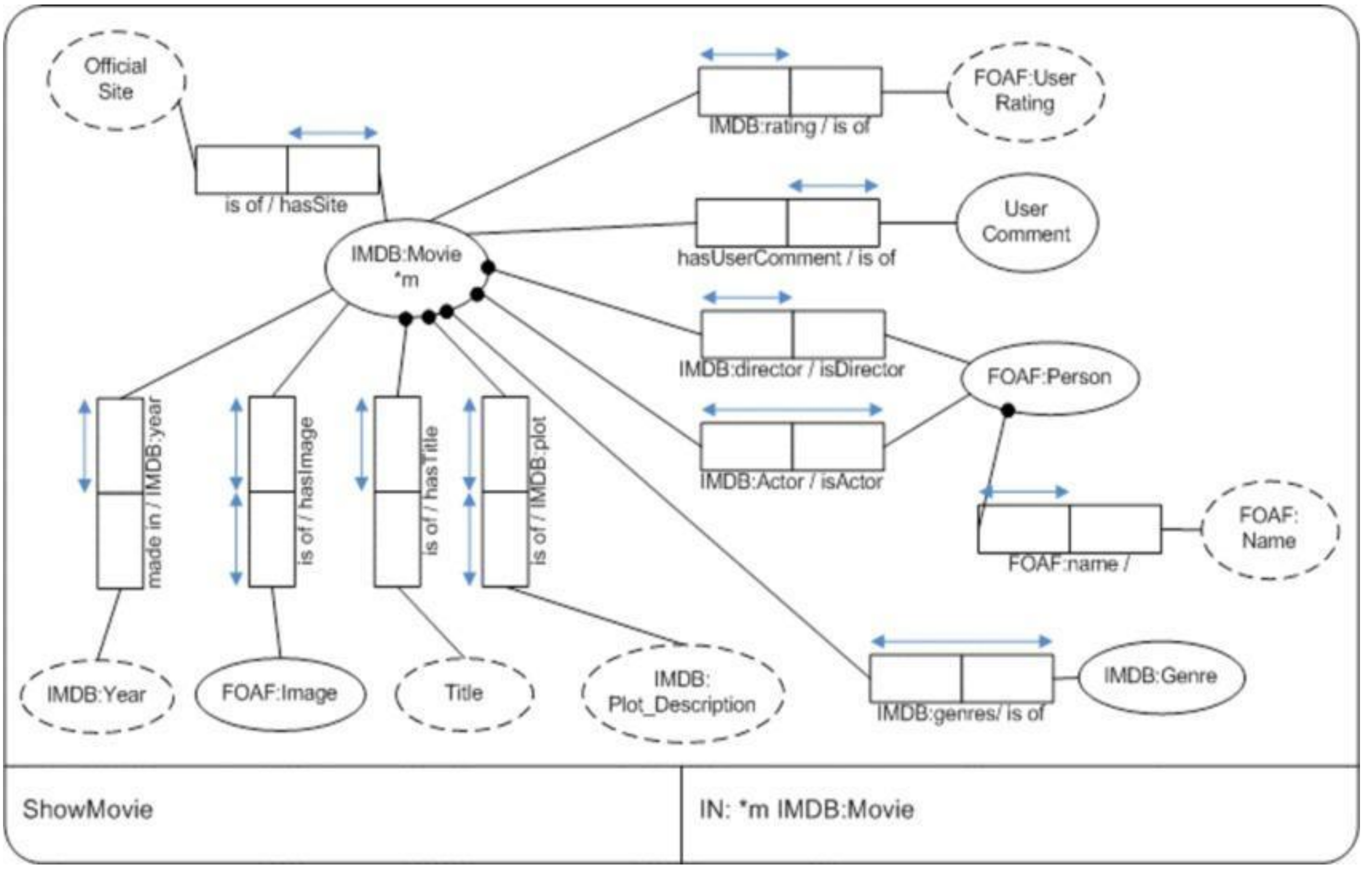
Foundation library

The Foundation library, written in Dart, provides basic classes and functions that are used to construct applications using Flutter, such as APIs to communicate with the engine.

Design-specific widgets

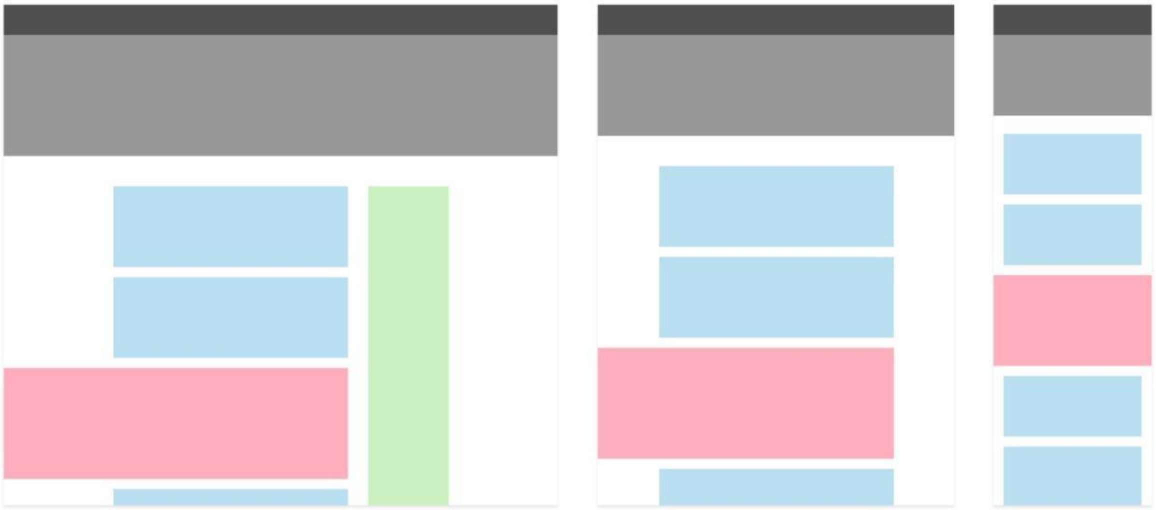
The Flutter framework contains two sets of widgets that conform to specific design languages: Material Design widgets implement Google's design language of the same name, and *Cupertino* widgets implement Apple's iOS Human interface guidelines.

Architecture

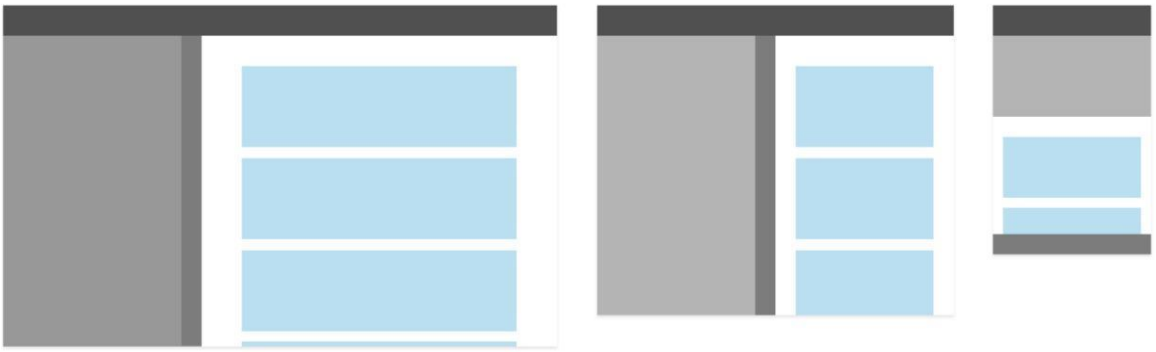


The Grids

In order to develop a system that is flexible enough to deal with all the existing and new data from IMDb´s database, I choose a modular layout.
As it is all about being responsive without limitations, I created three optimized grids for desktop, tablet and mobile.



Home Grids

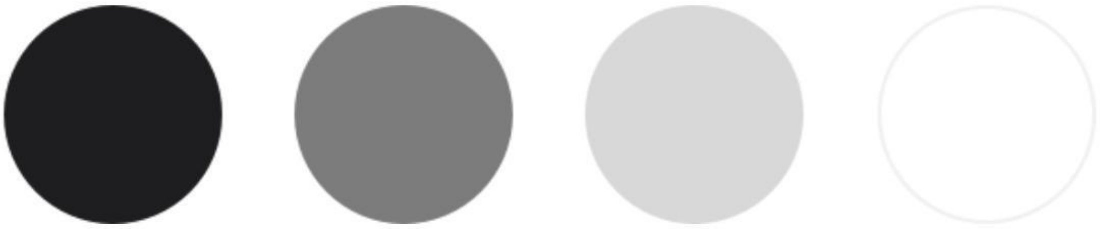


Detail View Grids

An endless scrolling wall of content becomes overwhelming, so I created some natural breaks (the red modules above) to help alleviate this.

Colors

To keep the iconic and well-known appearance, I just slightly adjusted the old color scheme in order to create a more modern and contemporary look.



Typeface

I spent a lot of time selecting the right typeface. I ended up using Open Sans, which is pretty ubiquitous, but causes a neutral and friendly appearance, while representing the simplicity of RMDb. In addition using Open Sans ensures a good readability across all screen densities and being available in most alphabets.

User Persona

Our target user is Steve who wants to find the best movie or show to watch next, Know more about a movie, show or its cast and crew after watching it, get updates on upcoming projects and share his entertainment knowledge and opinions by engaging in discussions with a community of people online.

Understanding the problems

We conducted a survey to find out about the preferences, problems and the scope of improvement in the overall experience of the website for the user. It included questions about usability, competitors and the UI/UX.

Competitor analysis

Metacritic -Not limited to movie ratings and is the only one to feature full user reviews right next to critic reviews. This makes it easy to compare what the general public thinks compared to the professionals.

Rotten Tomatoes -It has the advantage of sourcing its reviews from trusted critics. The Rotten Tomatoes criteria page explains that the site only takes reviews from trusted newspapers, podcasts, and websites. This means that only the opinions of the most-trusted movie critics influence the Rotten Tomatoes review.

Fandango -Includes Movie showtimes and has the feature of ticket booking.

Some of the popular competitors of IMDb are Rotten Tomatoes, Metacritic, Fandango, and Letterboxd. This survey showed us that some of the competitors have better usability than IMDb and are more popular among the users due to certain additional features and the clean and clutterless look and feel of the websites.
The rating of the movies is given less importance compared to other elements of information on the website, most of the people who visit the website are primarily there to check the rating and reviews of a movie before watching it.

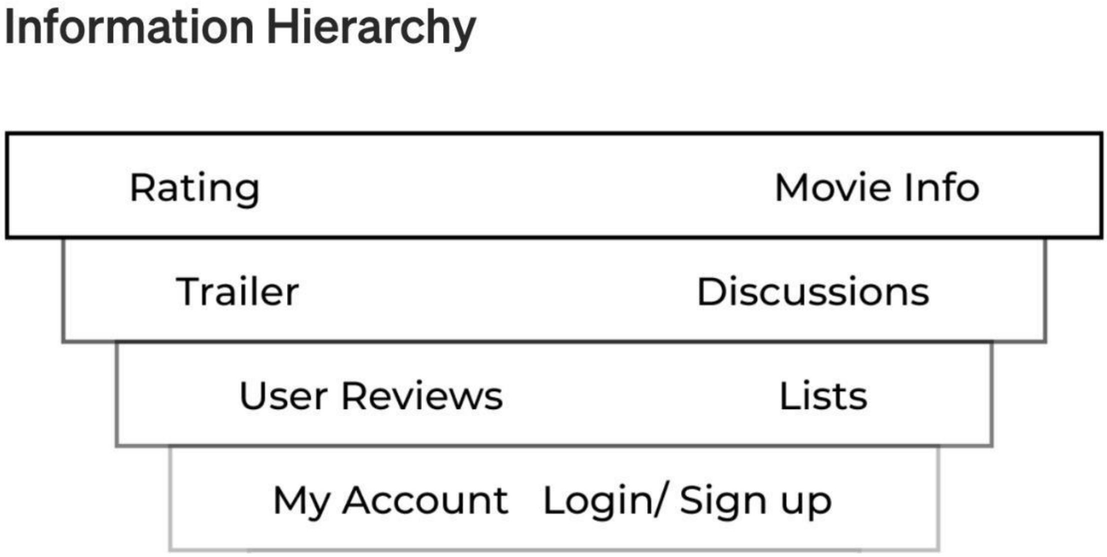
Less breathing space
The overall content of the website is very tightly packed which makes finding information complicated for the user. The content on the home page and what to watch page is repetitive.

Absence of a community
One of the problems was that with the absence of a community feature, People want a place where they can express their opinions, ideas and simply discuss the content that they watched. Discussing something with a group makes people feel connected. Finding people with a similar interest promotes social growth and a sense of belonging is achieved.

No personalization of content
It follows a one size fits all approach currently there isn’t much personalization in the content shown to the user, Personalization helps keep the users engaged. IMDb currently does not collect any specific user data to provide them with personalized content and recommendations. They could recommend viewer boards or movies that a registered user might like to watch according to their past interactions on the website. The homepage has too many sub-features pushing for the user's attention at the same time, they should focus on one core feature and build the experience around it by using the other features. The main focus should be on the movie card and ratings as that's the information most of the users want.

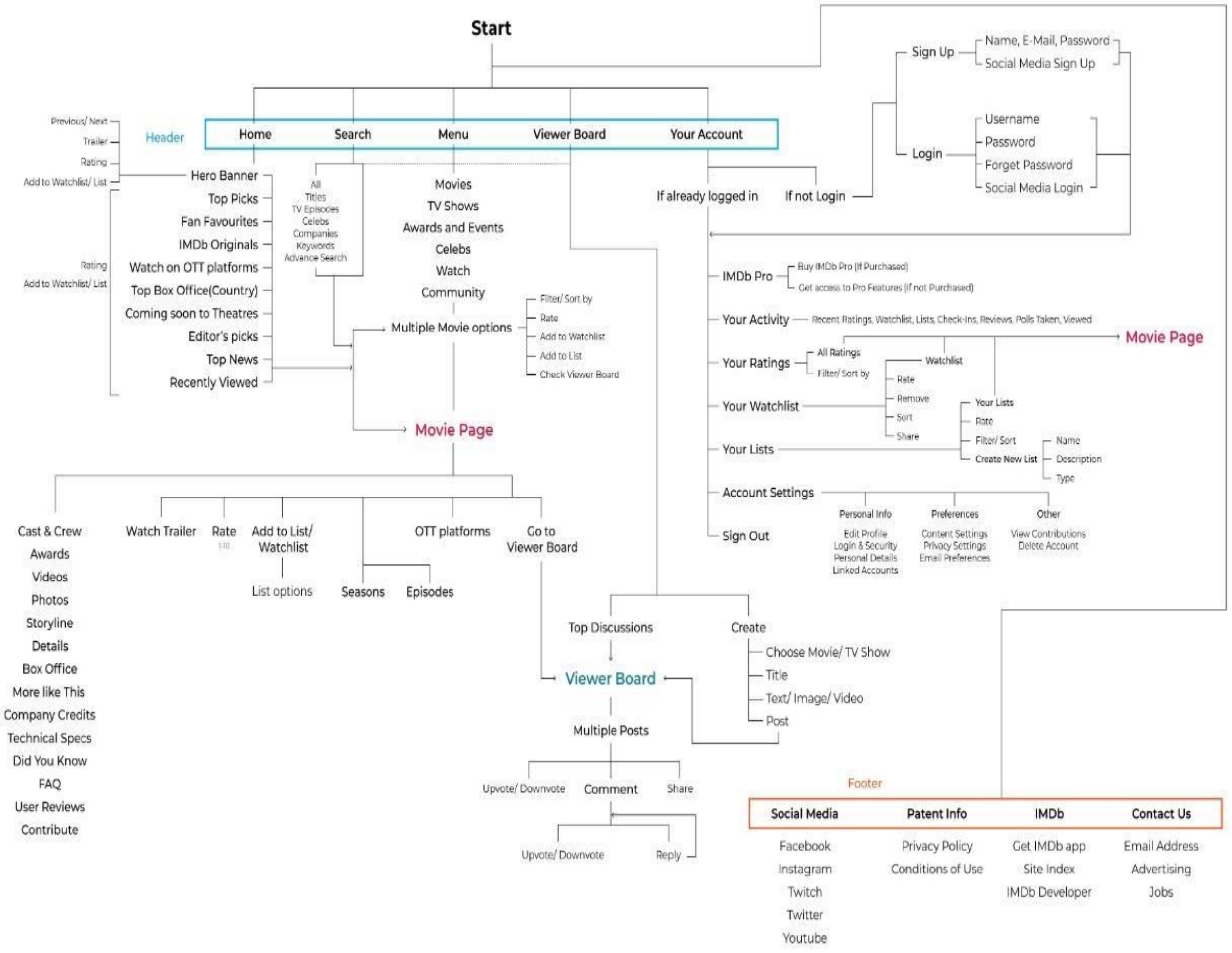
Competition from offline sources
It is not able to substitute offline modes of information regarding movies like magazines, TV shows. Many people still rely on offline alternatives for information due to no accessibility.
Critic reviews
No feature/page which draws a comparison between the reviews of the users and critic reviews. It would be interesting to see the contrasts or similarities between the opinions of the general public and professionals. If they feature reviews from trusted

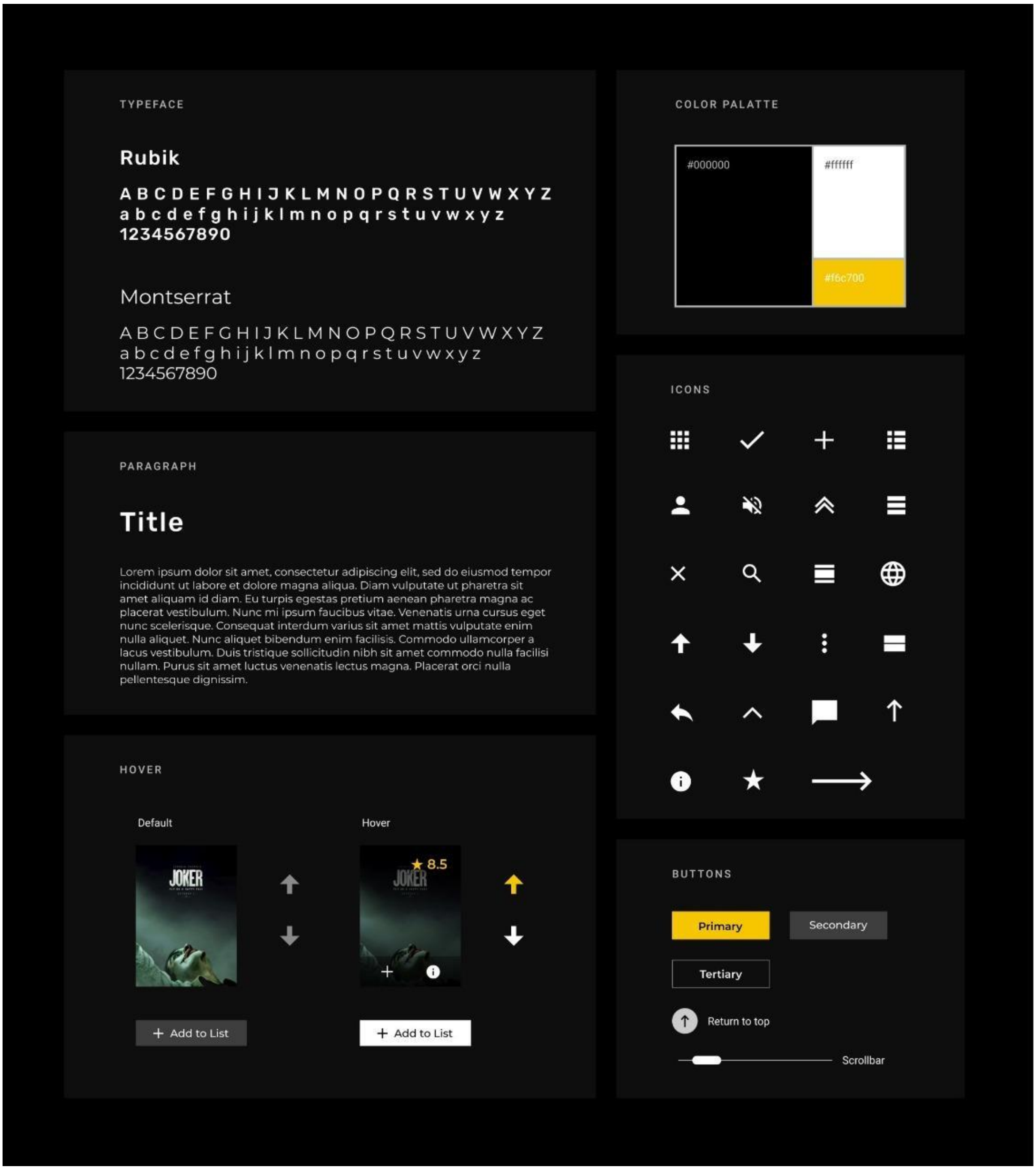
newspapers, podcasts, and websites it would boost their reliability and draw more people to the website.



The rating and movie info is the topmost priority as it is the reason most of the users visit the website, to get a quick look at the rating of the content before watching it and reading a small description of the content. Most people base what they watch over the impression they get off the website.

Users like to engage in discussions and watch snippets of upcoming projects which creates hype among the community. Many users create watchlists which help them keep a track of things that they have watched in the past or would like to watch in the future. There is a direct link provided to the different ott platforms from the website so that if something catches the attention of the user they can directly view it.


Start
Header
Home
Search
Menu
Viewer Board
Your Account
Previous/ Next
Trailer
Rating
Add to Watchlist/ List
Hero Banner
Top Picks
Fan Favourites
IMDb Originals
Rating
Add to Watchlist/ List
Watch on OTT platforms
Top Box Office(Country)
Coming soon to Theatres
Editor's picks
Top News
Recently Viewed
All
Titles
TV Episodes
Celebs
Companies
Keywords
Advance Search
Movies
TV Shows
Awards and Events
Celebs
Watch
Community
Multiple Movie options
Filter/ Sort by
Rate
Add to Watchlist
Add to List
Check Viewer Board
Movie Page
Cast & Crew
Awards
Videos
Photos
Storyline
Details
Box Office
More like This
Company Credits
Technical Specs
Did You Know
FAQ
User Reviews
Contribute
Watch Trailer
Rate
1-10
Add to List/ Watchlist
List options
Seasons
Episodes
OTT platforms
Go to Viewer Board
Top Discussions
Create
Choose Movie/ TV Show
Title
Text/ Image/ Video
Post
Viewer Board
Multiple Posts
Upvote/ Downvote
Comment
Share
Upvote/ Downvote
Reply
Sign Up
Name, E-Mail, Password
Social Media Sign Up
Login
Username
Password
Forget Password
Social Media Login
If already logged in
If not Login
IMDb Pro
Buy IMDb Pro (If Purchased)
Get access to Pro Features (If not Purchased)
Your Activity
Recent Ratings, Watchlist, Lists, Check-Ins, Reviews, Polls Taken, Viewed
Movie Page
Your Ratings
All Ratings
Filter/ Sort by
Watchlist
Rate
Remove
Sort
Share
Your Watchlist
Your Lists
Your Lists
Rate
Filter/ Sort
Create New List
Name
Description
Type
Account Settings
Personal Info
Edit Profile
Login & Security
Personal Details
Linked Accounts
Preferences
Content Settings
Privacy Settings
Email Preferences
Other
View Contributions
Delete Account
Sign Out
Footer
Social Media
Facebook
Instagram
Twitch
Twitter
Youtube
Patent Info
Privacy Policy
Conditions of Use
IMDb
Get IMDb app
Site Index
IMDb Developer
Contact Us
Email Address
Advertising
Jobs



Rubik

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
a b c d e f g h i j k l m n o p q r s t u v w x y z
1234567890

Montserrat

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
a b c d e f g h i j k l m n o p q r s t u v w x y z
1234567890

#000000 #ffffff #f6c700

Title

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Diam vulputate ut pharetra sit amet aliquam id diam. Eu turpis egestas pretium aenean pharetra magna ac placerat vestibulum. Nunc mi ipsum faucibus vitae. Venenatis urna cursus eget nunc scelerisque. Consequat interdum varius sit amet mattis vulputate enim nulla aliquet. Nunc aliquet bibendum enim facilisis. Commodo ullamcorper a lacus vestibulum. Duis tristique sollicitudin nibh sit amet commodo nulla facilisi nullam. Purus sit amet luctus venenatis lectus magna. Placerat orci nulla pellentesque dignissim.

Style guide

This is the style guide of our redesigned website. We used the same colours as the existing website. The icons we used are sharp as they fit in more with the theme, The existing website has soft curvy icons. We have used the Rubik and Montserrat typefaces as they provide a geometric and clean look.

Changes in the redesigned UI

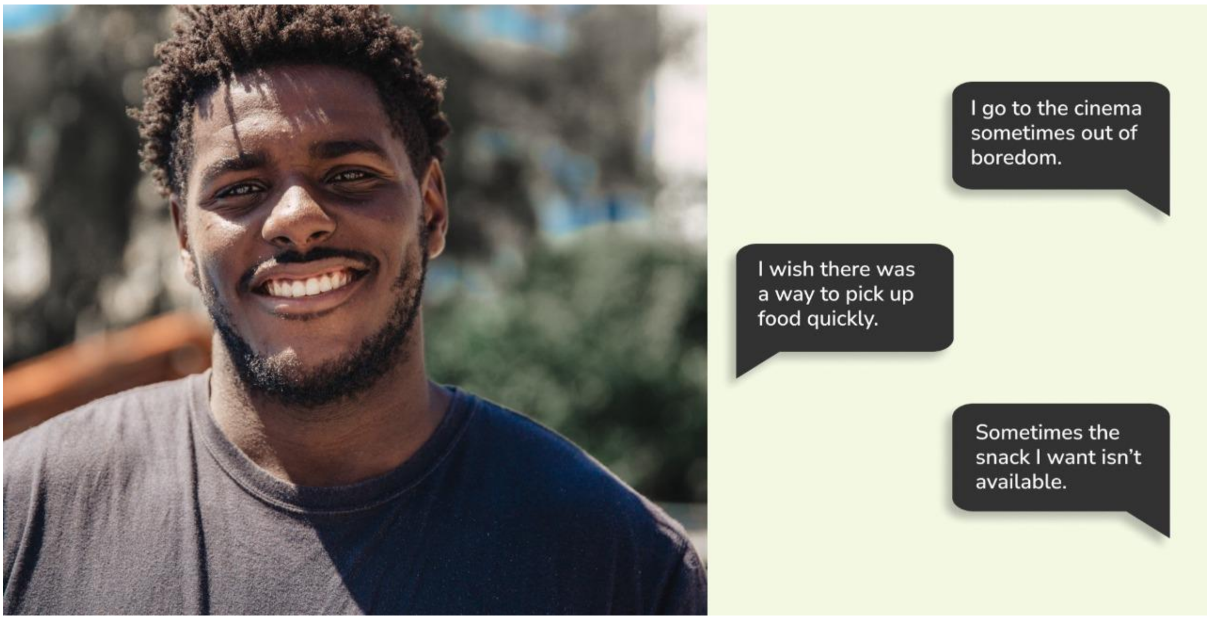
Watchlist option removed from the menu bar.
Hero banner showcases upcoming movie trailers.

Movie cards redesigned.
Viewer board feature added.
Return to the top button.
Rating added on the hero banner and recently viewed tab.
Content evenly distributed with clean spacing.

Personas
After the user research, I developed empathy maps for each interviewee.
Problem Statement:
David is a software engineer who needs a stress-free and dependable way to get tickets and food at the cinema because he doesn't want to wait in line.

Persona Summary:

David is a software engineer living in the metropolitan city of Lagos, Nigeria. Once a week, he and his girlfriend go on movie nights to see the latest movie. David would prefer to do everything online but the platform isn't dependable, so he often has to buy tickets and food over the counter which includes a long wait time especially when it's a blockbuster. David would specifically love for there to be a way to pick up popcorn and hotdogs without waiting so long in line as he hates overcrowded spaces.


I go to the cinema sometimes out of boredom.
I wish there was a way to pick up food quickly.
Sometimes the snack I want isn't available.

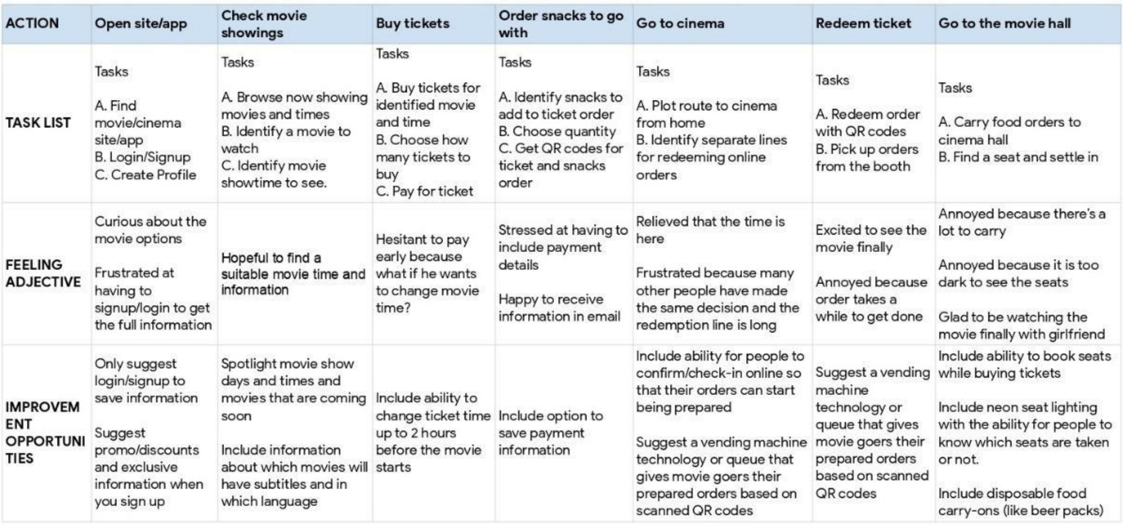
Goals:

To stay up-to-date on his favourite blockbuster movies
To have the opportunity to bond with his girlfriend during movie nights.
Frustrations:
"The payment platform doesn't work so I have to get my ticket from the counter."

"I wish there was a way to pick up food quickly especially when we are running late"

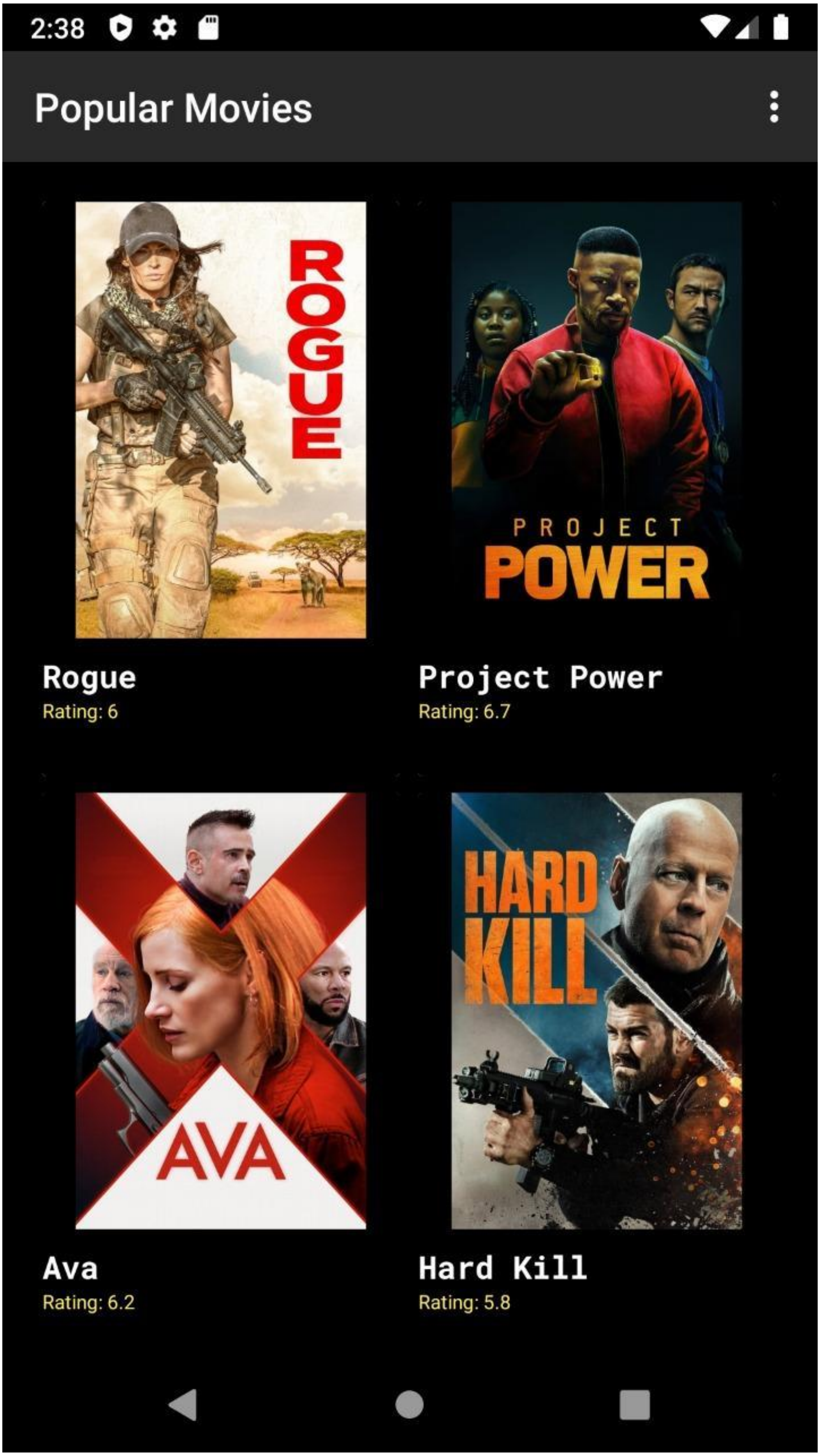
User Journey

Using the user stories and persona and problem statement, a user journey was developed for each stage of the process to watch a movie at the theatre which included the actions a user had to take, the tasks for each action and the way they felt while doing them, as well as opportunities for improvement. The basic actions identified that each user will have to go through included interactions with the site/app to check movie showings, buying movie tickets, buying snacks and completing the order, going to the cinema to redeem the ticket and finally, taking a seat in the theatre to watch the movie.



ACTION	Open site/app	Check movie showings	Buy tickets	Order snacks to go with	Go to cinema	Redeem ticket	Go to the movie hall
TASK LIST	Tasks A. Find movie/cinema site/app B. Login/Signup C. Create Profile	Tasks A. Browse now showing movies and times B. Identify a movie to watch C. Identify movie showtime to see.	Tasks A. Buy tickets for identified movie and time B. Choose how many tickets to buy C. Pay for ticket	Tasks A. Identify snacks to add to ticket order B. Choose quantity C. Get QR codes for ticket and snacks order	Tasks A. Plot route to cinema from home B. Identify separate lines for redeeming online orders	Tasks A. Redeem order with QR codes B. Pick up orders from the booth	Tasks A. Carry food orders to cinema hall B. Find a seat and settle in
FEELING ADJECTIVE	Curious about the movie options Frustrated at having to signup/login to get the full information	Hopeful to find a suitable movie time and information	Hesitant to pay early because what if he wants to change movie time?	Stressed at having to include payment details Happy to receive information in email	Relieved that the time is here Frustrated because many other people have made the same decision and the redemption line is long	Excited to see the movie finally Annoyed because order takes a while to get done	Annoyed because there's a lot to carry Annoyed because it is too dark to see the seats Glad to be watching the movie finally with girlfriend
IMPROVEMENT OPPORTUNITIES	Only suggest login/signup to save information Suggest promo/discounts and exclusive information when you sign up	Spotlight movie show days and times and movies that are coming soon Include information about which movies will have subtitles and in which language	Include ability to change ticket time up to 2 hours before the movie starts	Include option to save payment information	Include ability for people to confirm/check-in online so that their orders can start being prepared Suggest a vending machine technology or queue that gives movie goers their prepared orders based on scanned QR codes	Suggest a vending machine technology or queue that gives movie goers their prepared orders based on scanned QR codes	Include ability to book seats while buying tickets Include neon seat lighting with the ability for people to know which seats are taken or not. Include disposable food carry-ons (like beer packs)

Implementation & Testing



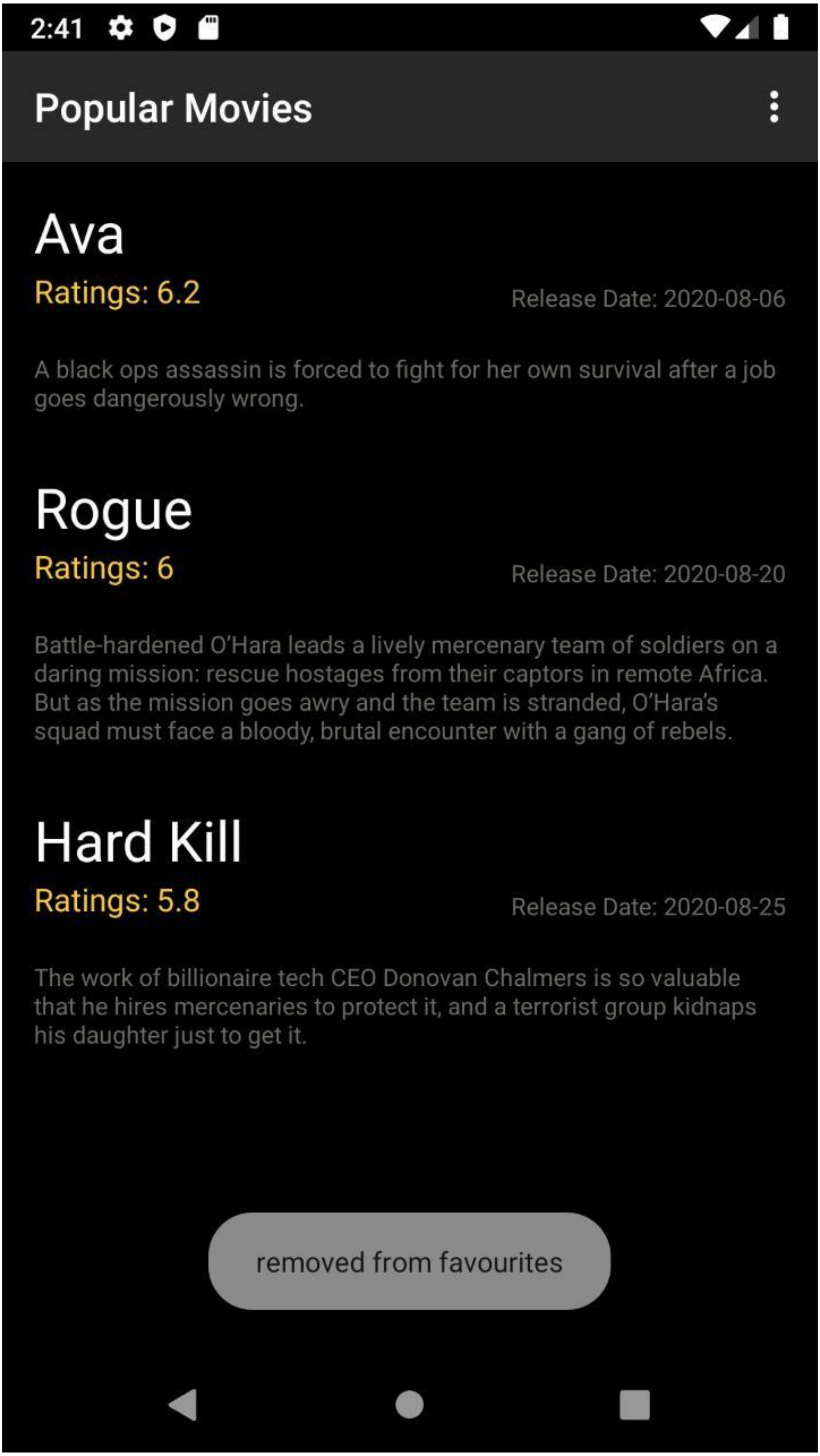
Popular Movies

Rogue
Rating: 6

Project Power
Rating: 6.7

Ava
Rating: 6.2

Hard Kill
Rating: 5.8



Popular Movies

Ava

Ratings: 6.2

Release Date: 2020-08-06

A black ops assassin is forced to fight for her own survival after a job goes dangerously wrong.

Rogue

Ratings: 6

Release Date: 2020-08-20

Battle-hardened O'Hara leads a lively mercenary team of soldiers on a daring mission: rescue hostages from their captors in remote Africa. But as the mission goes awry and the team is stranded, O'Hara's squad must face a bloody, brutal encounter with a gang of rebels.

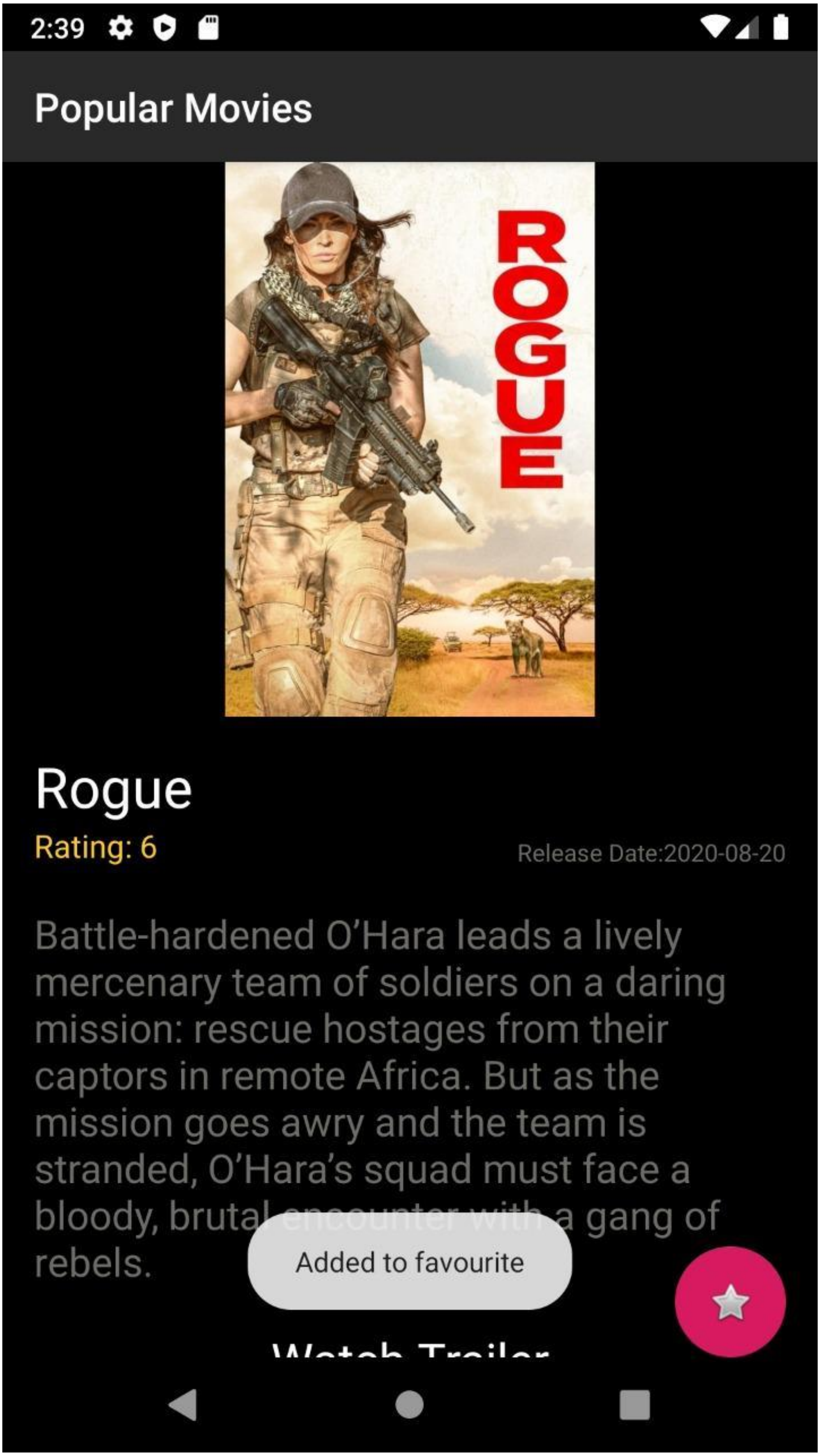
Hard Kill

Ratings: 5.8

Release Date: 2020-08-25

The work of billionaire tech CEO Donovan Chalmers is so valuable that he hires mercenaries to protect it, and a terrorist group kidnaps his daughter just to get it.

removed from favourites



Popular Movies

Rogue

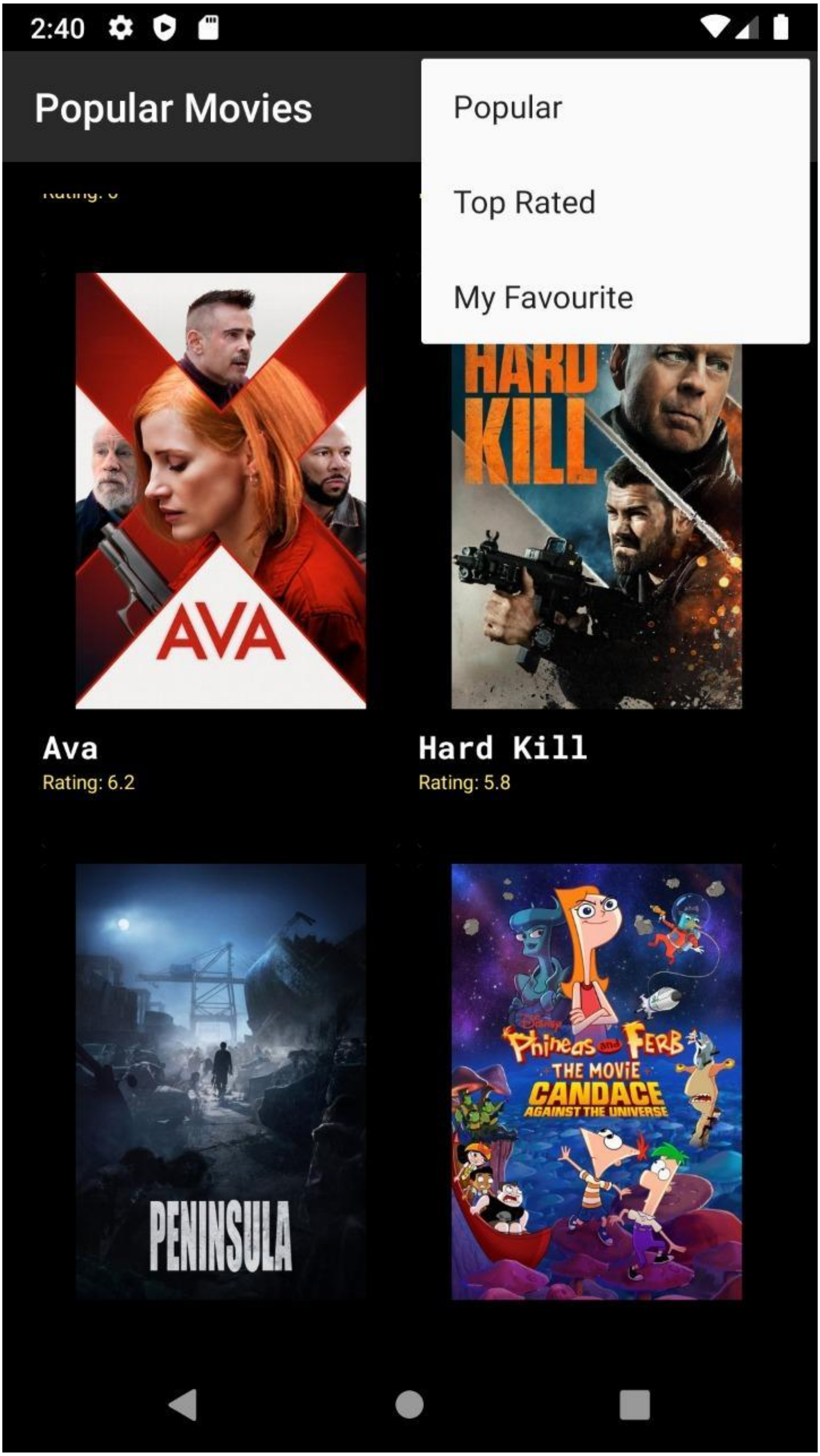
Rating: 6

Release Date:2020-08-20

Battle-hardened O'Hara leads a lively mercenary team of soldiers on a daring mission: rescue hostages from their captors in remote Africa. But as the mission goes awry and the team is stranded, O'Hara's squad must face a bloody, brutal encounter with a gang of rebels.

Added to favourite

Watch Trailer



Popular Movies

Ava
Rating: 6.2

Hard Kill
Rating: 5.8

Conclusion

At the end of the day, a consensus from users was that the app felt simple and straightforward.
One user gave the feedback: “This is a really convenient way to see a movie.”
What can I do better?
Accessibility: There was no participant with visual impairments included in the study. While I had the intention to do so, it was a bit difficult finding someone who had those disabilities and who went to the cinemas as well.
The UI aspect was a bit challenging to achieve. Selection of colours and fonts seem easy until you catch yourself spending one whole day on it!

Learnings

While designing this app, I came face to face with my designer bias and saw the impact of conducting interviews for the product you are designing for users to challenge pre existing notions. Embarking on this project has enhanced my understanding of the broad spectrum of design, while also thinking about the limitations.