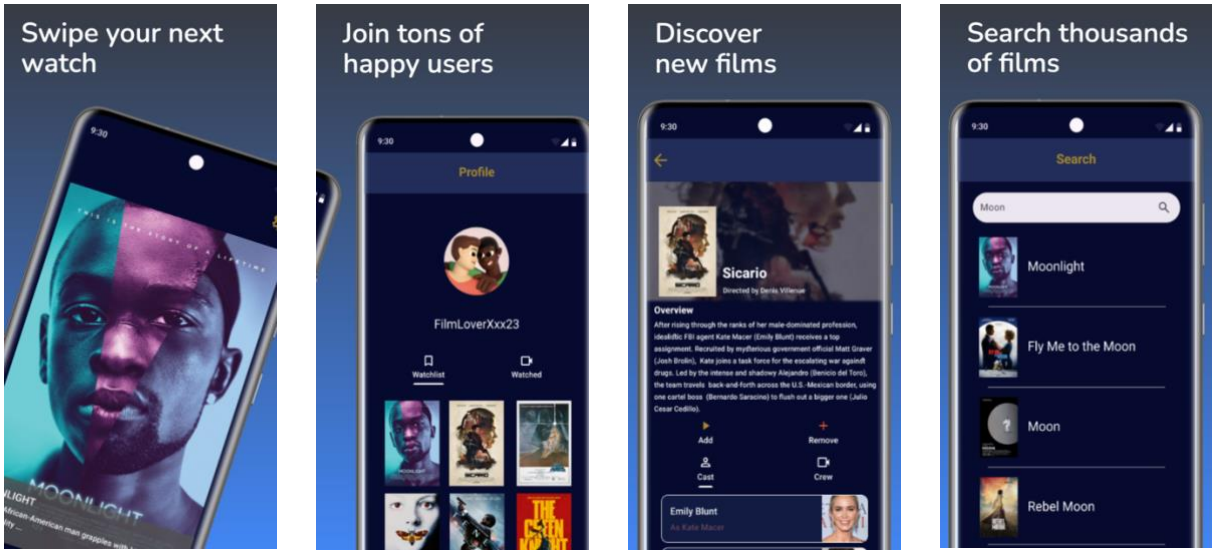


Mobile App Design Document (UFCF7H-15-3)

FilmSwipe

Student Name (Student Number):

Arthur Milner (21035478)

Rohaan Aslam (21017718)

Table of Contents

Requirements

High Concept:

Film Swipe is a movie based mobile application that combines the sociability, film exploration, and film logging of Letterboxd **(Letterboxd, 2024)** with the ease of the swipe interface of Tinder **(Tinder, 2024)**. Users can inspect and filter randomly generated movies by swiping right to add films to a watchlist, left to discard them and up to mark them as watched. There is quick access to detailed information on each film, including synopsis and genre, allowing users can make well informed choices. Users will have a profile section which houses their watchlisted/watched films and profile information. A search feature allows users to find and manage specific films and browse the profiles of other users. The aim of the app is to simplify film discovery into an intuitive interface accessible for all types of users.

Target Users:

The Film Swipe App is designed for everyone who wishes to broaden their film taste and discover great new films, but primarily an 18+ audience as the films generated can be any age rating. Consequently, our

audience is very broad, including both casual viewers and avid movie fans looking for an interactive way to discover new films. The apps intuitive interface will ensure it can easily be used by both the younger and older members of our demographic. Features such as viewing profiles helps appeal to avid movie fans, as they might wish to display their watched films to showcase their love for the medium. This also entices casual movie fans, they can browse their friend's profiles to see what they're watching, promoting the sociability of the app.

User Stories:

- As a film fan subscribed to Netflix, Amazon Prime or Disney+, I wish to interact with randomly generated films which are available on any of the streaming platforms and add the interesting ones to a watchlist, I wish to refer to these movies for my next film night.
- As a user I want to be able to customize my profile picture to add a personal touch, either by uploading an image or using the device camera. This, alongside my username, should allow my friends to recognize my profile and see what I am watching/plan-to-watch.
- As a user wanting to know the details on a particular film, I want to be able to search for this film, select it from a list of results and view the details. From here I want to be able to decide whether I add it to the watchlist/watched films on my profile.
- As a user wanting to see what their friend has watched, I want to be able to search for my friend's username, select their profile and filter to only show the films they have watched as a list.
- As a user swiping movies, I want to be able to be able to see the audience score at-a-glance, this is so I have a better understanding of the quality of the movie to inform my decision to swipe right.
- As a user who has previously logged-in, I want to be able to remain logged-in after I close/open the app, this is so I do not have to repeatedly log-in every time I wish to use the app.

Initial Research:

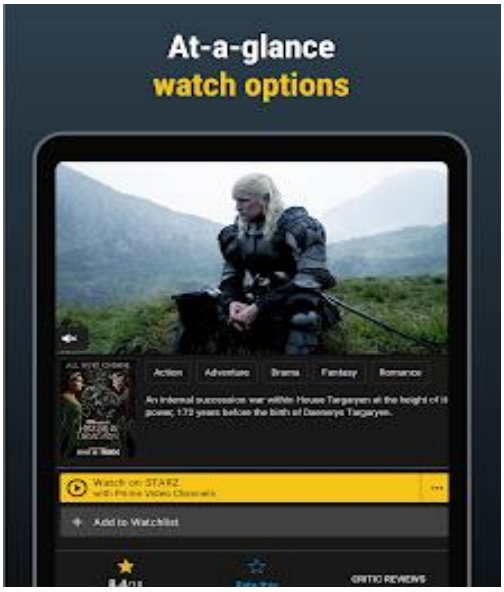


Figure 1 Image from IMDb's GooglePlay store listing, showing details on a film/series.

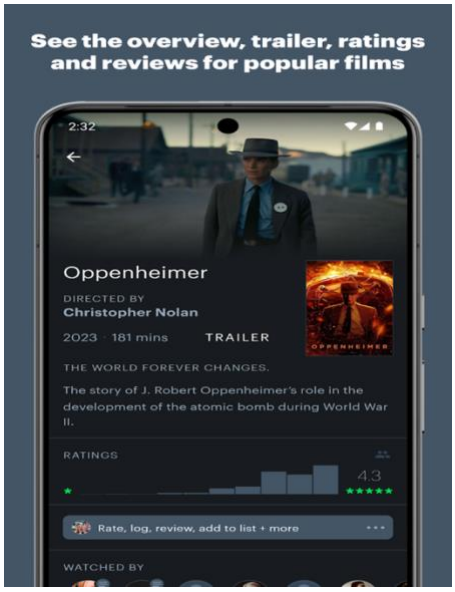


Figure 2 Image from Letterboxd's GooglePlay store listing, showing the expanded view of a movie's details.

Our app will take a lot of inspiration from pre-existing movie-based applications such as Letterboxd **(Letterboxd, 2024)** and IMDB **(IMDb, 2024)**, particularly regarding the layout of film information as both apps do a great job of presenting all the key information about the film without creating too much visual noise and maintaining a clean aesthetic. The apps also ensure good readability through appropriate spacing, the layouts promote the proper order in which you should read the contents through styling such as font size and colour.

Figure 3 Image from Tinder's GooglePlay store listing, showcasing the apps swipe functionality.

A huge inspiration for our app comes from the hugely successful dating app Tinder **(Tinder, 2024)**, the swipe functionality is appropriate for randomly selected films as it works when presenting users with a short overview, such as the overview presented in a tinder profile. The swipe gesture also feels more natural and intuitive compared to simply clicking a button a repeated number of times **(Medium, 2024)** and helps to maintain the user's attention by reducing UI clutter and keeping only the relevant details on display. Similar swipe cards are also tried and tested in multiple other popular apps, such as Hinge **(Hinge, 2024)** with 10+ million downloads and Bumble **(Bumble, 2024)**.

Development Requirements:

APIs and External Libraries

- **API**
  - **TMDb API**: Sourcing film data like titles, overviews, and posters.
- **External Libraries**
  - **Retrofit**: For API communication and network calls.
  - **Firebase**: For user authentication and database management.
  - **Canhub - CropImageActivity**: To enable profile picture & cropping functionality.
  - **Coil**: For asynchronous image loading.
  - **Gson**: For parsing JSON data from the API.

Error, Exception Handling and Loading

- **API Calls**:
  - Display user friendly error messages for failed API calls.
- **Database Operations**:
  - Use built in error handling (addOnSuccessListener, addOnFailedListener etc.) for failed operations and display error details.
- **Loading**:
  - Use loading indicator on API calls and database operations to provide visual feedback when waiting for a response.

Asset Collection

- Material Design icons to maintain a clean, modern user interface.
- Canva for creation of the app logo.
- Images from the web for placeholders where calls return no data, such as no movie poster or profile picture.

Technical Approach

- Use MVVM (Model-View-ViewModel) architectural pattern to organize the codebase and states.

Use of Data Models

- Models to appropriately store details within API responses from the TMDB API, such as Movie and MovieResponse models, where MovieResponse can store a list of Movie objects.
- Models to appropriately store details in database responses, such as a FilmswipeUser model.
- Use LiveData objects as containers for the models storing database and API responses, including live data objects for Loading & Error states. Live data used so the view dynamically updates when the contents have changed.

Local Storage/Database

- Use Shared Preferences for persistent login state variable to ensure users remain logged in when the app is reopened.
- Use firebase database to store user information such as watchlists, watched items, usernames, and profile pictures.

Authentication

- Use Firebase Authentication to manage login and registration with appropriate security.

Functional & Non-functional Requirements:

ID	Priority	Description	Rationale
FR-01	MUST	Users must be able to register and log in.	Basic functionality required for user personalisation for app features.
FR-02	MUST	Users must be able to swipe right to like a film, left to discard	This is a core feature of the app and enables users to

		a film and up to mark as watched on the home screen.	interact with films and create their list.
FR-03	MUST	The app must save liked and watched films to the user's profile.	Essential for creating a personal list of wishlist films.
FR-04	MUST	Users must be able to search for films on the Search page.	Allows users to find specific films and manually add them to wishlist.
FR-05	MUST	Users must be able to search for users on the Search page.	Allows users to interact with other users and view their watchlists and watched.
FR-06	MUST	Users must be able to view detailed information about a film on the film detail page, accessible from both the home page and search page.	Provides the necessary context and detail for whether a user chooses to add a film to their wishlist.
FR-07	MUST	The app must have a navigation bar with Home, Search, Profile and settings options available from all screens that are not authentication related.	Essential to improving User Experience (UX) across the app.
FR-08	SHOULD	Users should be able to edit their personal information.	Improves user control over their account information which improves personalization.

FR-09	SHOULD	Users should be able to filter films displayed on the home screen by streaming service.	Allows user filter appropriately dependent on their streaming provider available.
FR-10	COULD	Users could receive recommendations based on their liked films.	Increases user engagement by suggesting relevant content but is not critical.

ID	Priority	Type	Description	Rationale
NFR-01	MUST	Compatibility	The app must be compatible with Android SDK 34 or above.	Ensures compatibility with modern Android features and devices.
NFR-02	MUST	Security	The app must securely store user data and implement basic encryption for login information.	Protects user information and builds trust in the app's security and ensures GDPR compliance.
NFR-03	MUST	Performance	The app must load film data within 5 seconds on both home & search pages.	Fast responsive times improve user experience vastly.
NFR-04	MUST	Usability	The app must have an intuitive & easy-to-use modern UI adherent to ADA standards.	Improves the app's usability making it easier to use and will match competitor applications.

NFR-05	SHOULD	Compatibility	The app UI should adjust appropriately to different portrait screen sizes.	Ensures consistent user experience on different portrait size devices.
NFR-06	WON'T	Compatibility	The app won't support landscape screen orientation.	Landscape orientation is outside the initial scope of the application.
NFR-07	WON'T	Feature Scope	The app won't include social media integration for sharing liked films.	Outside of the initial feature scope, focusing the app on core functionality.

UI Requirements:

Requirement ID	Feature	Description	Rationale
UIR-01	Swipeable Cards	A card component that supports swipe gestures for actions like adding movies to watchlists or marking as watched.	Apps like Tinder use swipeable cards and gamify discovery interactions encouraging user engagement.
UIR-02	Profile Picture	An image representing the user's profile, with functionality to update the picture when clicked.	Modern applications use profile pictures to personalize and improve the user experience.
UIR-03	Navigation Bar	A component that enables users to navigate between key screens like Home, Profile, Search, and Settings.	Modern applications use navigation bars to provide quick access to key features.
UIR-04	Text Field	Input fields used for collecting user input, e.g. search queries, and login credentials.	Text fields are essential for user interaction and are necessary for a common authentication mechanism.
UIR-05	Buttons	Interactive elements for performing actions like logging in, signing up, confirming changes, and interacting with the app.	Found in every modern app (e.g., Netflix, Spotify etc.), as buttons provide clear and familiar way for interaction with the app.

UIR-06	Lazy Loading Lists/Grid	Performance efficient components that are used for rendering large datasets dependent on screen scroll location.	Lazy loading greatly improves performance of the app when displaying large sets of data (e.g. movies, cast or crew).
UIR-07	Checkbox	A toggle filtering component for enabling or disabling search options, such as switching between user and movie searches.	Checkboxes provide a familiar component seen in many modern applications as they provide an easy interactable filtering method.
UIR-08	Dropdown Menu	A menu component that allows users to filter or sort data, such as filtering by streaming provider.	Dropdown Menus are used by most modern applications for provide filtering as they provide an easy interactable filtering method.

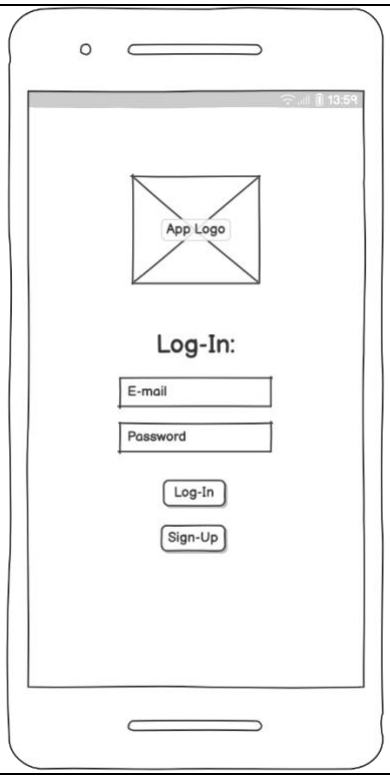
Wireframes

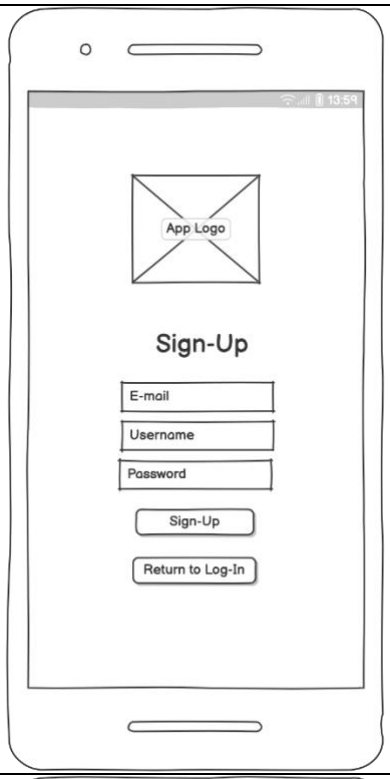
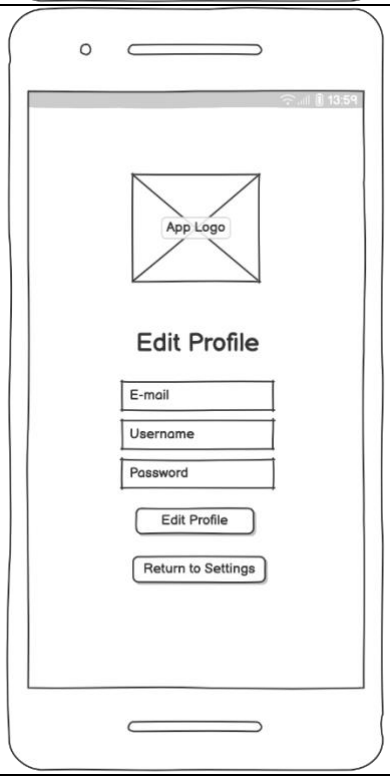
Technical Diagrams:

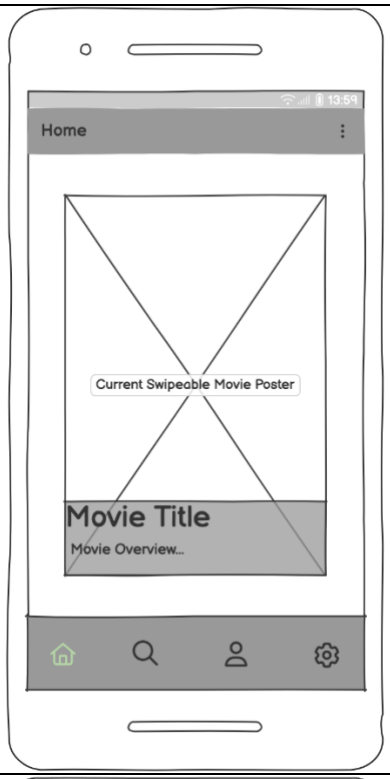
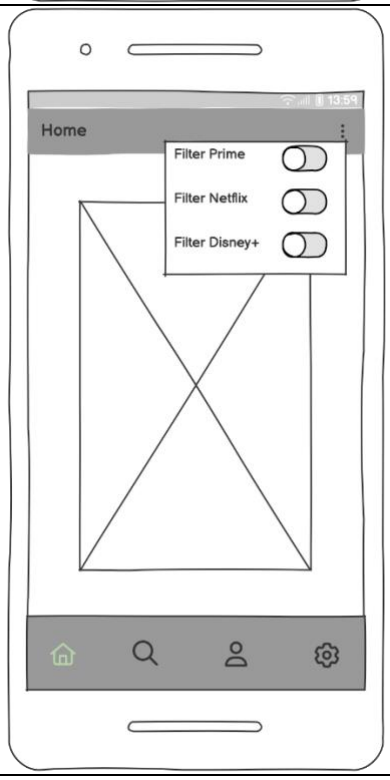
These requirements apply to all wireframes:

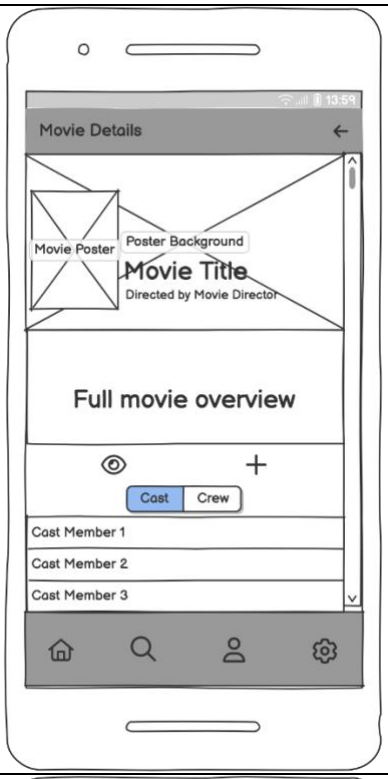
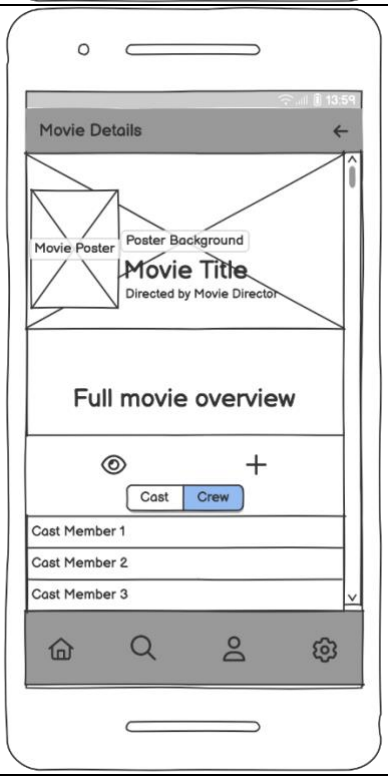
- **FR-07**: The app must have a navigation bar with Home, Search, Profile and settings options available from all screens that are not authentication related.
- **NFR-01**: The app must be compatible with Android SDK 34 or above.
- **NFR-04**: The app must have an intuitive & easy-to-use modern UI adherent to ADA standards.
- **NFR-05:** The app UI should adjust appropriately to different portrait screen sizes.
- **UIR-03 (Except Login/Signup):** A component that enables users to navigate between key screens like Home, Profile, Search, and Settings.

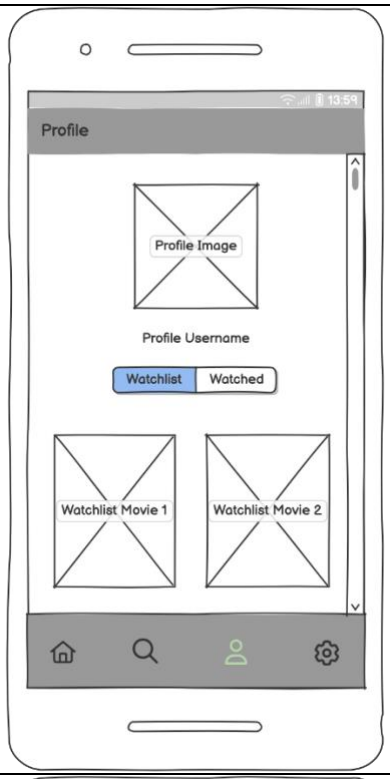
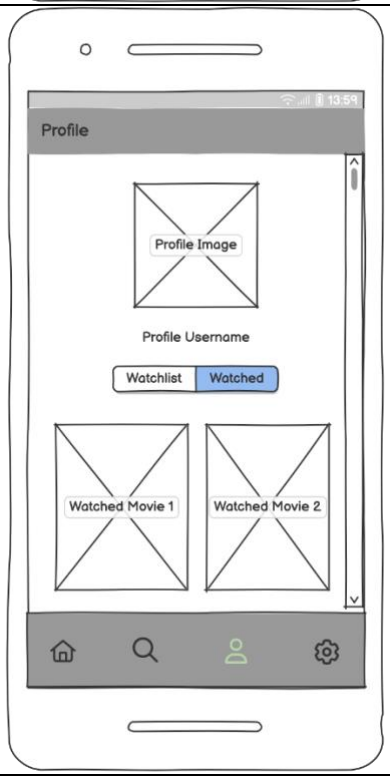
In our diagrams we have also considered the natural thumb zone for the frequently used aspects of our application to improve the UX and accessibility of the designs.

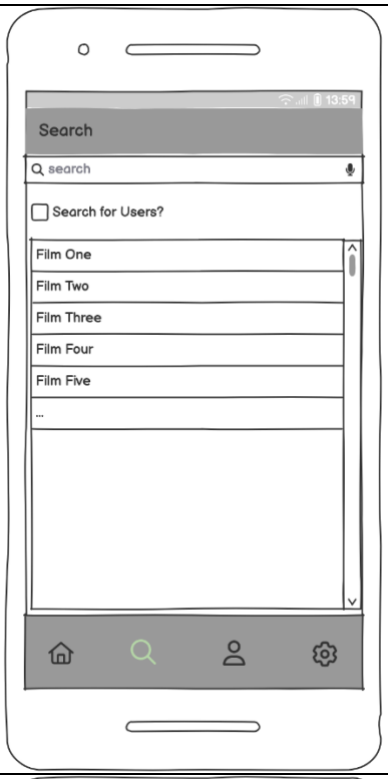
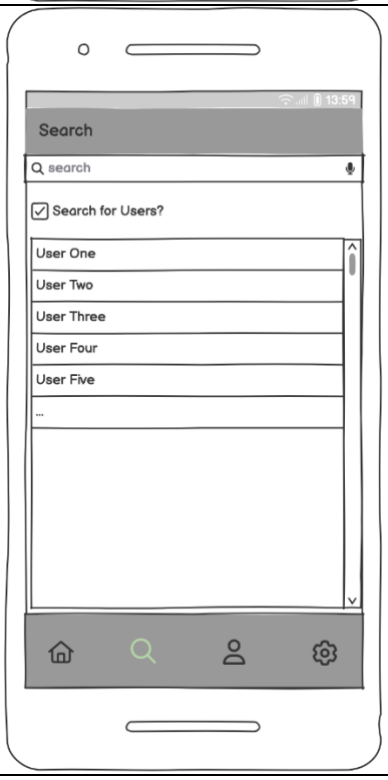
Wireframe	Description
	This page provides a login form where users can input their email and password to access the app. It includes a "Log In" button for authentication and a "Sign Up" button that navigates to the registration page for new users. **FR-01, NFR-02, UIR-04 and UIR-05.**

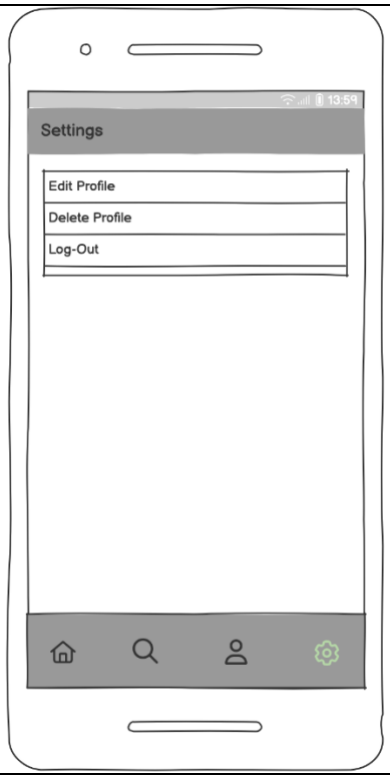
	This page allows users to create a new account by entering their email, username, and password. It has a "Sign-Up" button to confirm the registration process and a "Return to Log-In" button to redirect users back to the login screen if needed. **FR-01, NFR-02, UIR-04 and UIR-05.**
	This page allows users to edit their personal account information, including their email, username, and password. A button labeled "Edit Profile" confirms changes made, while a "Return to Settings" button takes users back to the settings menu. **FR-08, UIR-04 and UIR-05.**

	This page displays randomly generated movie posters that users can swipe to interact with: right to like, left to discard, and up to mark as watched. There is a filter menu that allows users to filter the movies displayed based on their preferred streaming service, such as Netflix, Amazon Prime or Disney+. There is a navigation bar at the bottom to access different pages. **FR-02, FR-09, NFR-03, UIR-01 and UIR-08.**
	This page displays randomly generated movie posters that users can swipe to interact with: right to like, left to discard, and up to mark as watched. There is a filter menu that allows users to filter the movies displayed based on their preferred streaming service, such as Netflix, Amazon Prime or Disney+. There is a navigation bar at the bottom to access different pages. **FR-02, FR-09, NFR-03, UIR-01 and UIR-08.**

	This page provides detailed information about a selected movie, including the title, poster, director, overview, and cast/crew details. Users can decide whether to add the movie to their watchlist or mark it as watched. It is accessible from both the home page and search results. **FR-06, NFR-03, UIR-05 and UIR-06.**
	This page provides detailed information about a selected movie, including the title, poster, director, overview, and cast/crew details. Users can decide whether to add the movie to their watchlist or mark it as watched. It is accessible from both the home page and search results. **FR-06, NFR-03, UIR-05 and UIR-06.**

	This page includes the user's profile image and username, along with tabs to view their "Watchlist" or "Watched" films. Each film is displayed as a thumbnail and once clicked redirects user to the movie details page. Clicking on the profile picture will prompt the user to select a photo from the camera roll or gallery for customisation. **FR-03, FR-08, UIR-02, UIR-05 and UIR-06.**
	This page includes the user's profile image and username, along with tabs to view their "Watchlist" or "Watched" films. Each film is displayed as a thumbnail and once clicked redirects user to the movie details page. Clicking on the profile picture will prompt the user to select a photo from the camera roll or gallery for customisation. **FR-03, FR-08, UIR-02, UIR-05 and UIR-06.**

	This page includes a search bar at the top, where users can look for specific films or other users. Users can toggle between searching for films or profiles using the checkbox. Search results are displayed in a list format, and selecting an item navigates to the respective details or profile page. **FR-04, FR-05, NFR-03, UIR-04, UIR-05, UIR-06 and UIR-07.**
	This page includes a search bar at the top, where users can look for specific films or other users. Users can toggle between searching for films or profiles using the checkbox. Search results are displayed in a list format, and selecting an item navigates to the respective details or profile page. **FR-04, FR-05, NFR-03, UIR-04, UIR-05, UIR-06 and UIR-07.**

	This page provides account management options, including editing the user's profile, deleting their account, and logging out. It ensures users can customize their account and access essential functions in one screen. **FR-08 and UIR-05.**

User Flow/Navigation:

Most of the navigation can be done through the bottom navigation bar which will take users to the 4 main screens of the app (Home, Search, Profile and Settings), the other navigation is detailed in the wireframe flow diagram along with functionalities of any buttons. It also includes the requirements to access certain screens, such as requiring the user to be logged in.

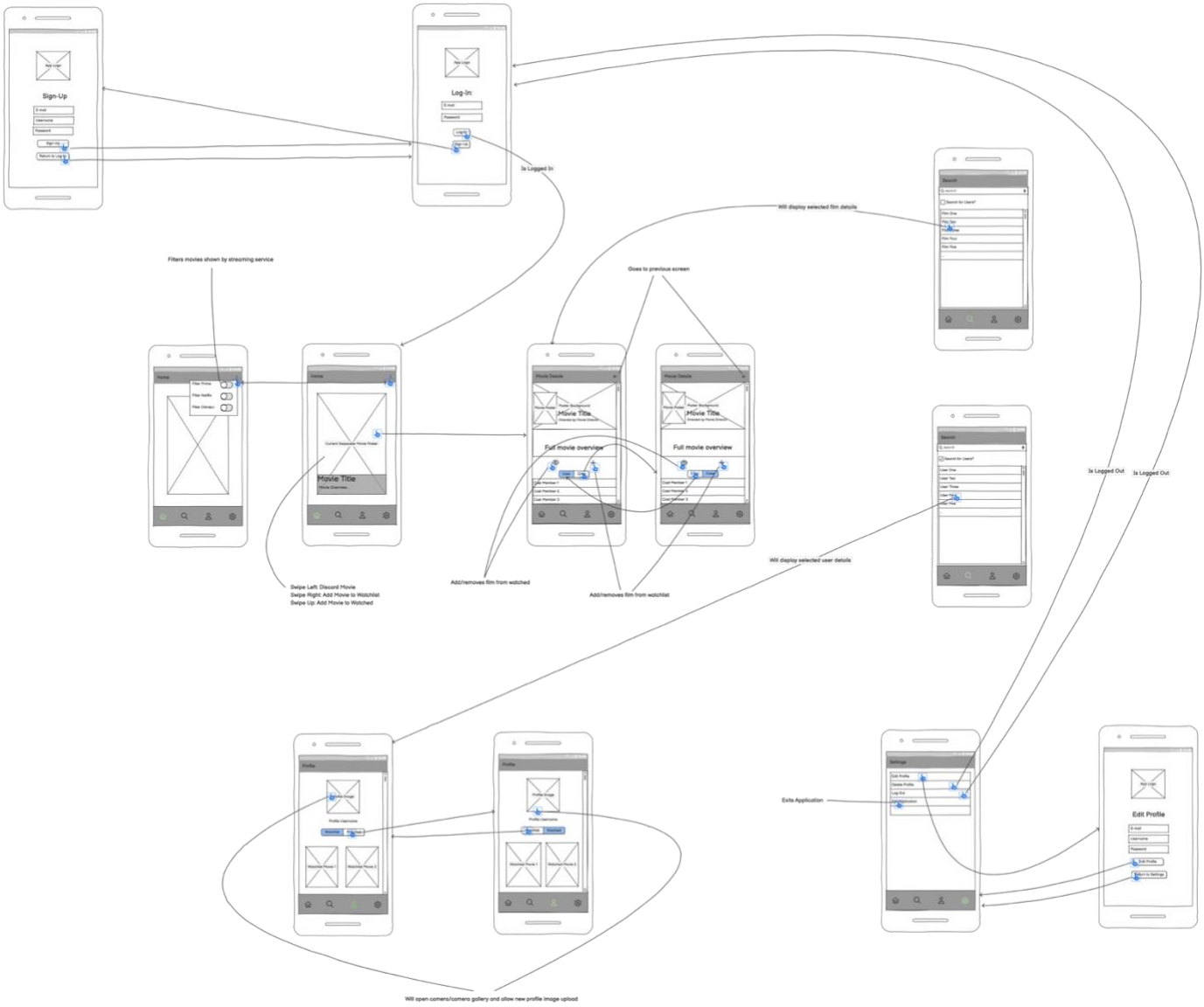


Figure 4 Wireflow diagram showing the possible user flow and navigation of the app, including required states to access certain screens

App Lifecycle:

- The user will log-in, this log-in session will be remembered across opening/closing the app, until the user logs-out.
- The application will not display in landscape mode if the phone is rotated horizontally.
- The built-in android back button and back swipe gesture will not work unless the user is on certain screens, such as movie details and viewing another user's profile.
- The account details data will be saved using Firebase Auth for security purposes as this will store hashed passwords.

- When a user marks a movie as watched or adds it to their watchlist, a reference will be stored for that movie in Firebase under their account, this reference will be used for API calls to return the bulk of the movie data as needed. For example, when a user visits their profile page the Firebase data will make an API call to display the movie information of their watched list.

Scale/Orientation:

The app will not support landscape mode as this is rarely supported for apps using movie-based swipe functionality. Perhaps eventually landscape mode could be considered with the swipe functionality transforming to separate buttons for watched, watchlist and discard. From our current designs however, the app can provide a cleaner and more intuitive user experience when operating in portrait mode. Below is a mock-up to show what a landscape mode could look like for our application.

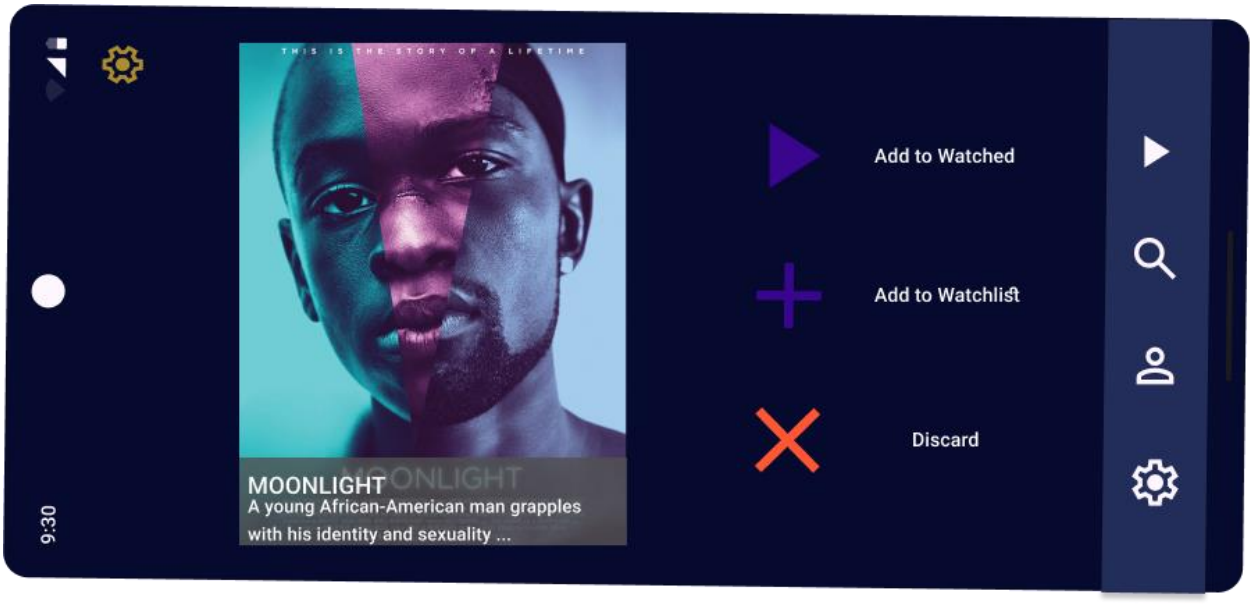


Figure 5 Mock-up showcasing a possible landscape mode for the application

When creating the initial layouts for our application, we considered different scales in the designs. Consequently, different scales require little changes, as shown in the wireframes below.

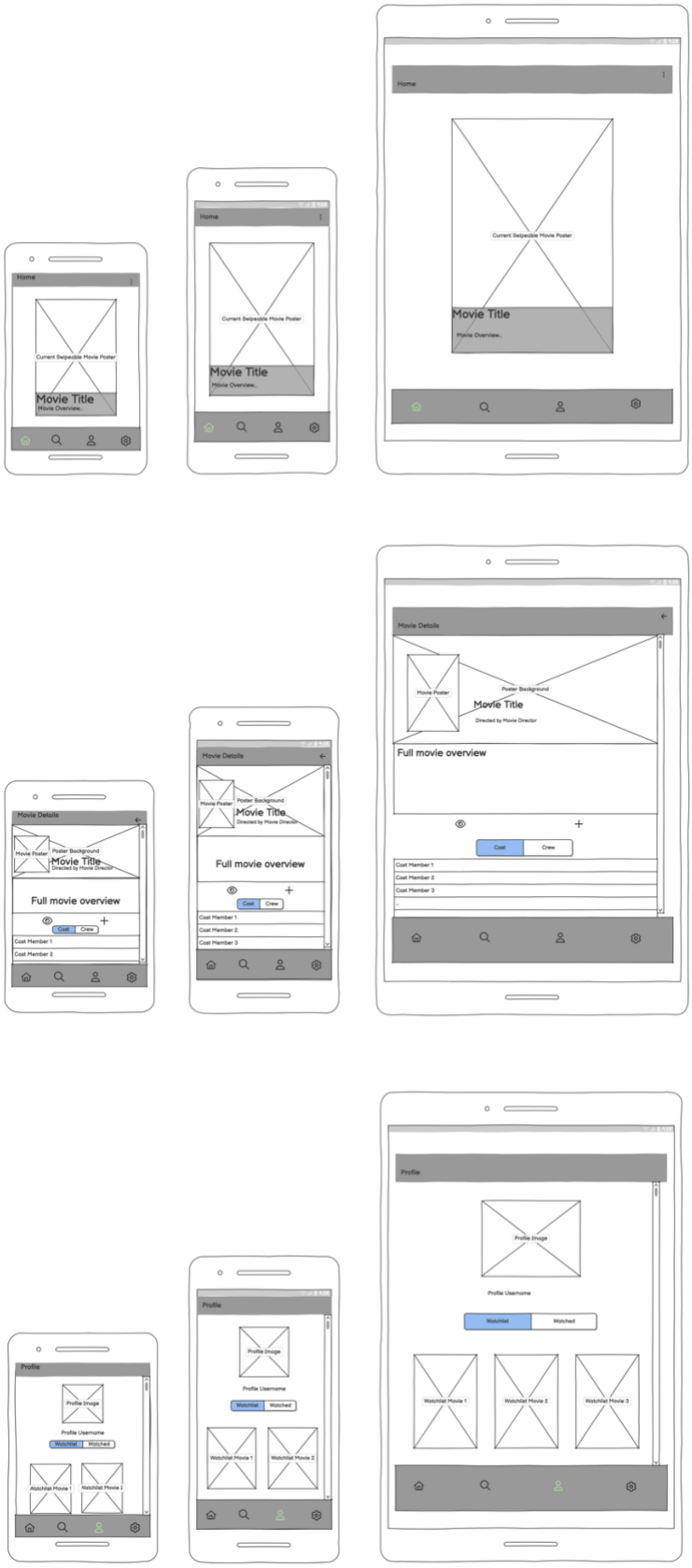


Figure 6 Wireframes showing how the app should look on different screen sizes/devices

Composites

Mock-ups:

For the creation of our composites, we used Figma **(Figma, 2024)**. This was because the free plan suited our needs, and its cloud-based collaborative features made it easy to work together on our designs. We created composites for both light and dark modes, alongside composites which considered smaller screen sizes, and composites which display the apps features in action, such as swiping and searching. Using our mockups with the web application **Previewed (2024)** allowed us to create app store images which can showcase the app to potential users.

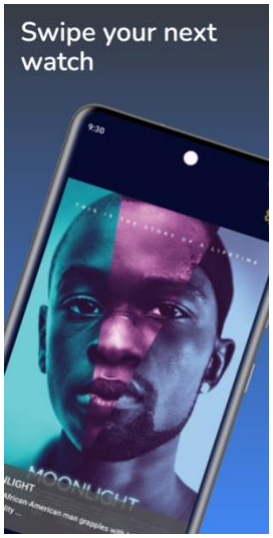


Figure 7 App store image 1, generated with Previewed

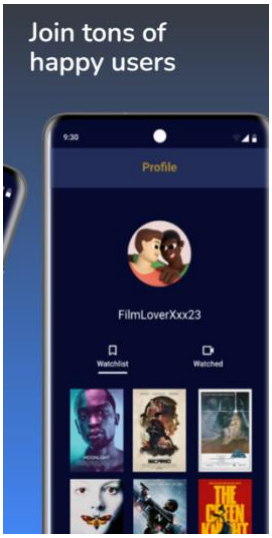


Figure 8 App store image 2, generated with Previewed

Figure 9 App store image 3, generated with Previewed

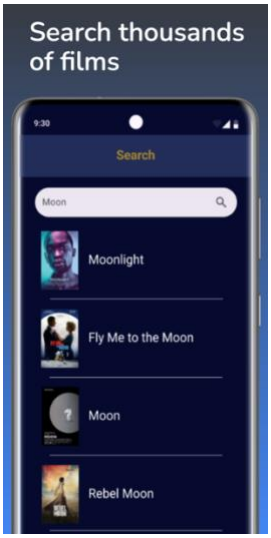


Figure 10 App store image 4, generated with Previewed

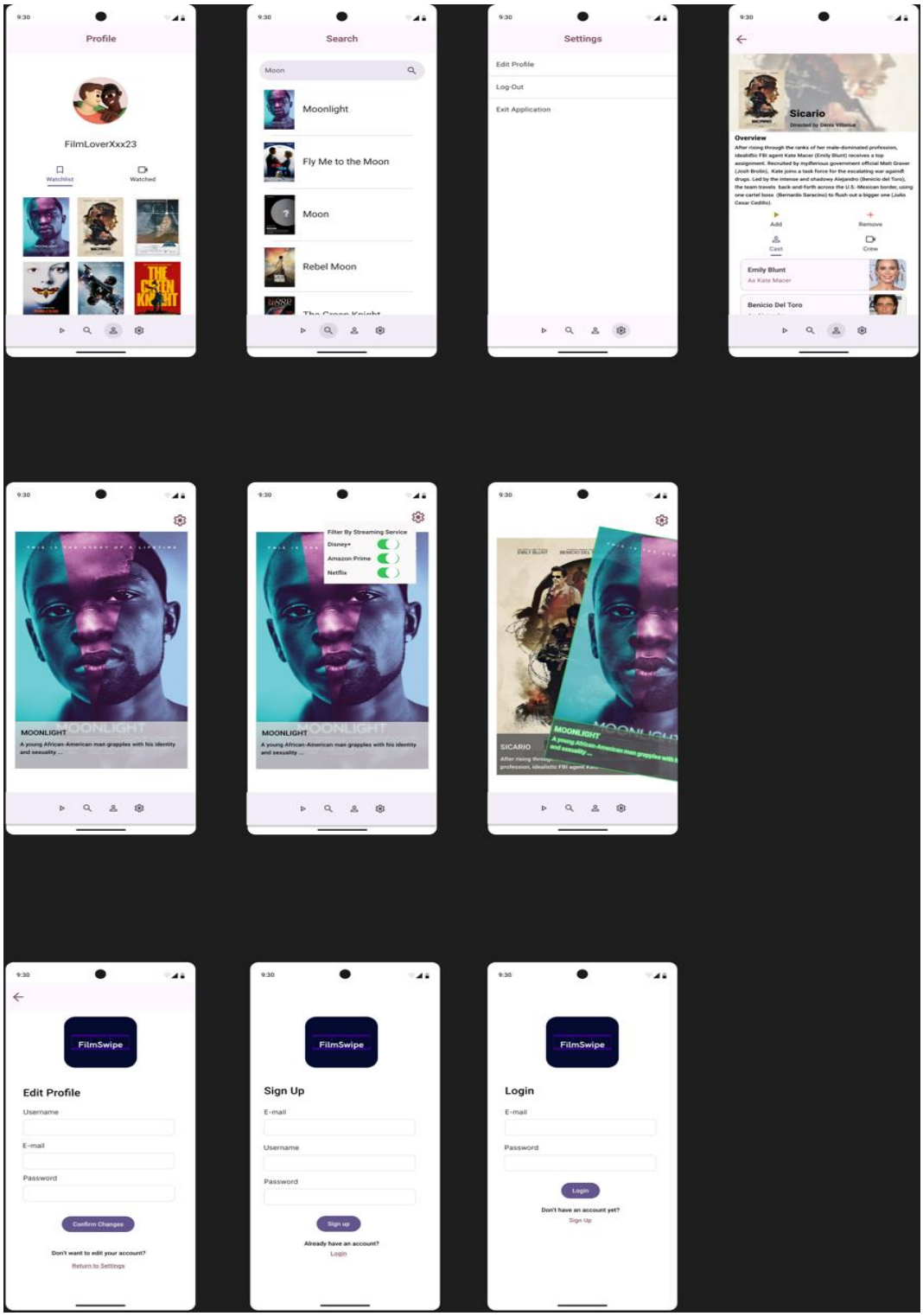


Figure 11 Composites showcasing the app in light mode

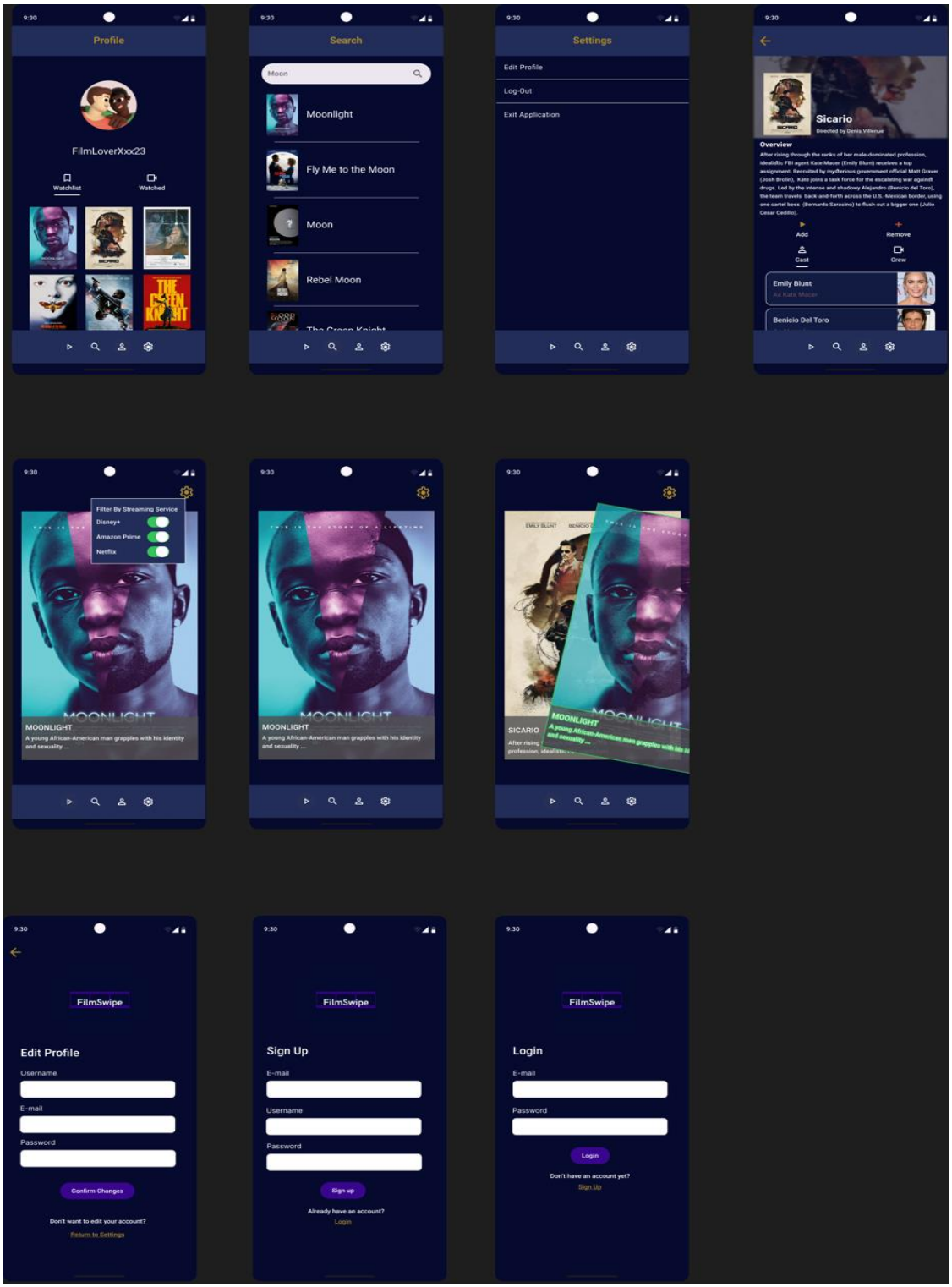


Figure 12 Composites showcasing the app in dark mode

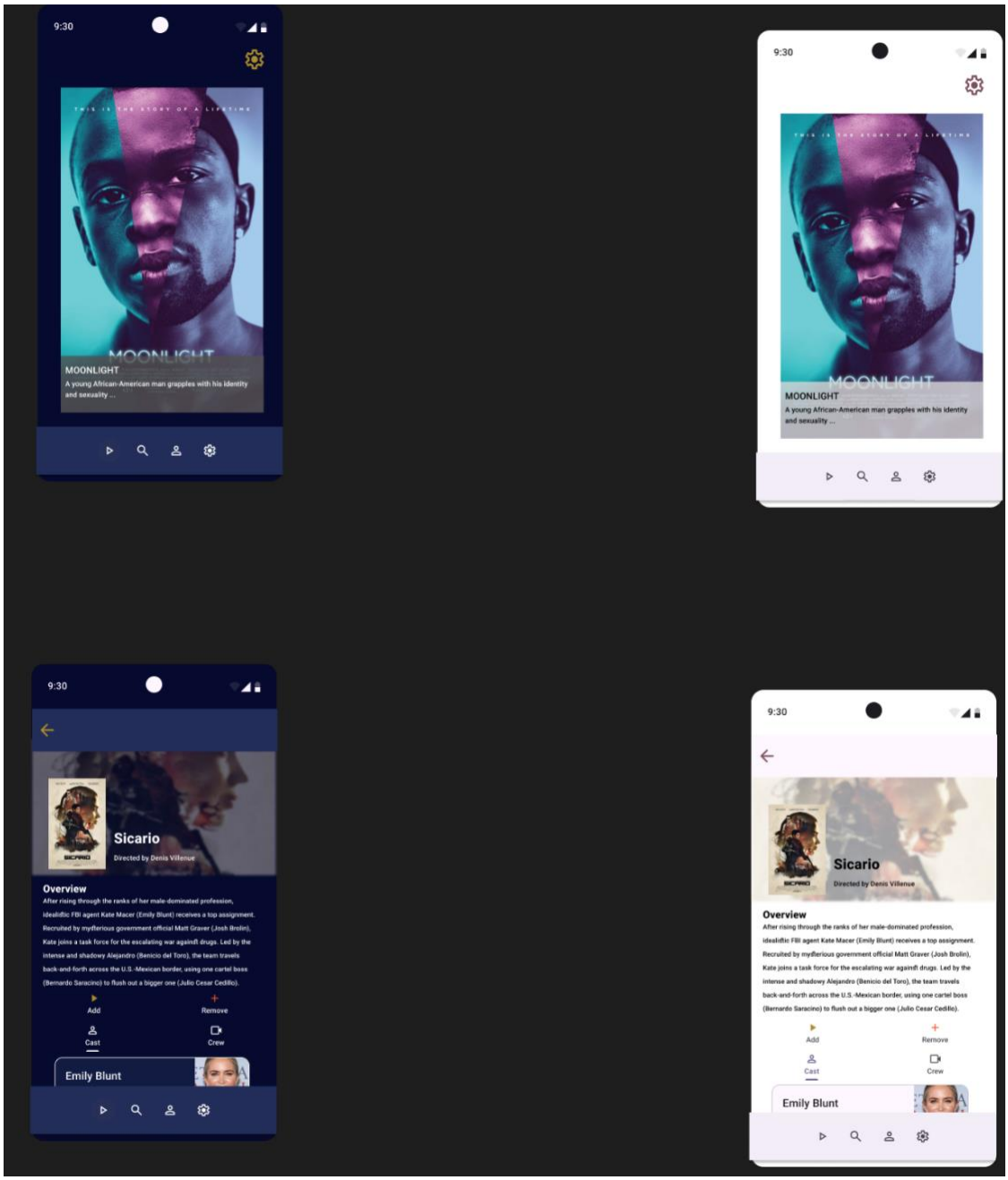


Figure 13 Composites showcasing the app on smaller screen sizes

App Icon:

To create our logo, we used the online tool **Canva (2024)**. Canva had an existing app logo template, and this ensured that our design met manufacturer requirements for resolution and scaling. When designing the icon, we wanted it to reflect the colours of our application, we also wanted it to represent the application's clean minimalistic UI. To achieve this, we kept the design simple and used the boldest

colours within our application. The font takes inspiration from the icon of **Letterboxd (2024)** to provoke user familiarity and we used the film reel to present the link our application has to movies.

Figure 14 The app icon in the different formats it will be used within Android OS

Colour Schemes:

Using **coolors (2024)** we were able to create a visual representation of our colour scheme including the hex values we will use within Android Studio. We also edited these representations to include where each colour should be used based on MaterialTheme palette values such as primary, secondary and surface, so we have a reference for where each colour is expected to be used.

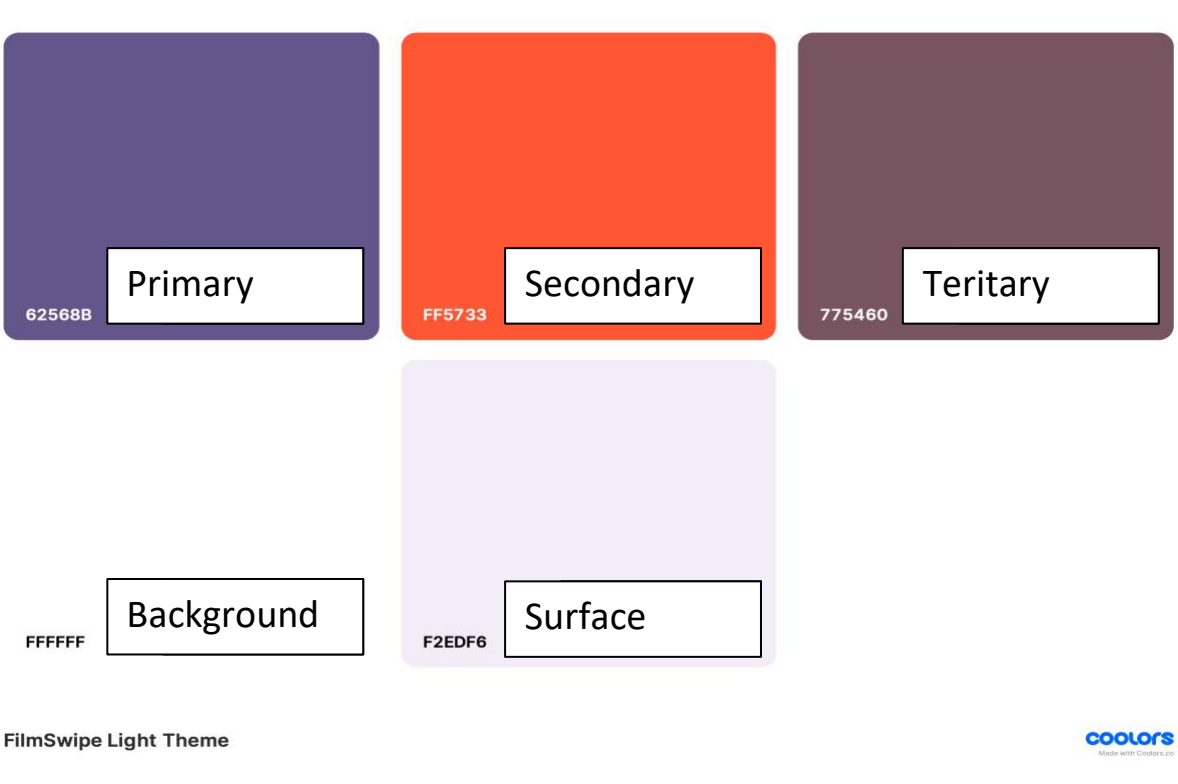


Figure 15 Colour scheme for the light mode

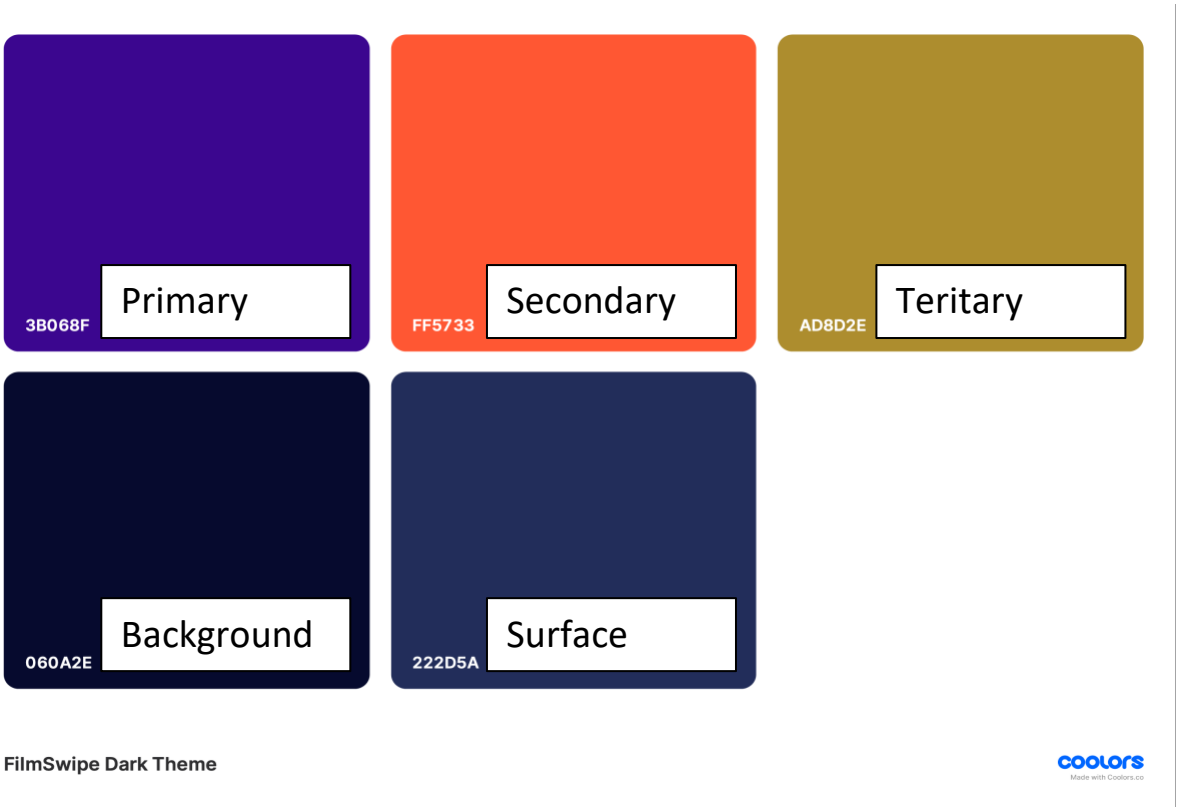


Figure 16 Colour scheme for the dark mode

Bibliography

- Bumble (2024) ***Bumble*** [mobile app] Available from: https://play.google.com/store/apps/details?id=com.bumble.app&hl=en_GB [Accessed 27 November 2024]
- Canva (2024) ***Canva*** [web app] Available from: https://www.canva.com [Accessed 20 November 2024]
- Coolors (2024) ***Coolors*** [web app] Available from: https://www.coolors.co [Accessed 26 November 2024]
- Figma (2024) ***Figma*** [web app] Available from: https://www.figma.com
- Hinge (2024) ***Hinge*** [mobile app] Available from: https://play.google.com/store/apps/details?id=co.hinge.app&hl=en_GB [Accessed 01 December 2024]
- IMDb (2024) ***IMDb*** [mobile app] Available from: https://play.google.com/store/apps/details?id=com.imdb.mobile&hl=en [Accessed 26 November 2024]
- Letterboxd (2024) ***Letterboxd*** (Version 2.19.2) [mobile app] Available from: https://play.google.com/store/apps/details?id=com.letterboxd.letterboxd&hl=en_GB [Accessed 25 November 2024]
- Medium (2024) ***The Importance of Mobile Gestures in Boosting App User Experience***. Available from: https://medium.com/@marketing_96275/the-importance-of-mobile-gestures-in-boosting-app-user-experience-dae53cd77d30 [Accessed 27 November 2024]
- Previewed (2024) ***Previewed*** [web app] Available from: https://previewed.app/ [Accessed 20 November 2024]
- Tinder (2024) ***Tinder*** [mobile app] Available from: https://play.google.com/store/apps/details?id=com.tinder&hl=en_GB [Accessed 26 November 2024]