Binge-o-Philia

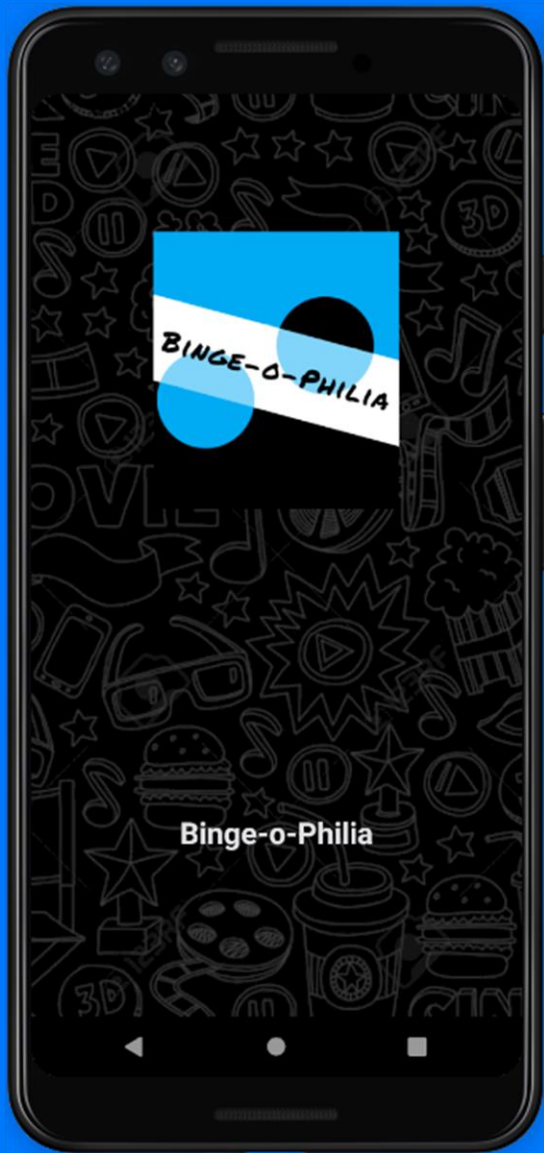## Social-Networking Platform for Movie Buffs

About

Binge-o-Philia is a social-networking platform for movie buffs.

A single place for:

- Connecting with people ("binge-philes")
- Managing leisure (show-watching) time
- Reducing the friction of finding a perfect time to watch a show together
- Keeping a record of shows you've watched
- Browsing all the information about new and happening going on in the fiction world

Not to mention, the proper reminders/notifications about upcoming watch or when a friend just posted a new Movie Review ... all sorted!

Download from Playstore

DOWNLOAD

Binge-o-Philia

GET IT ON
Google Play
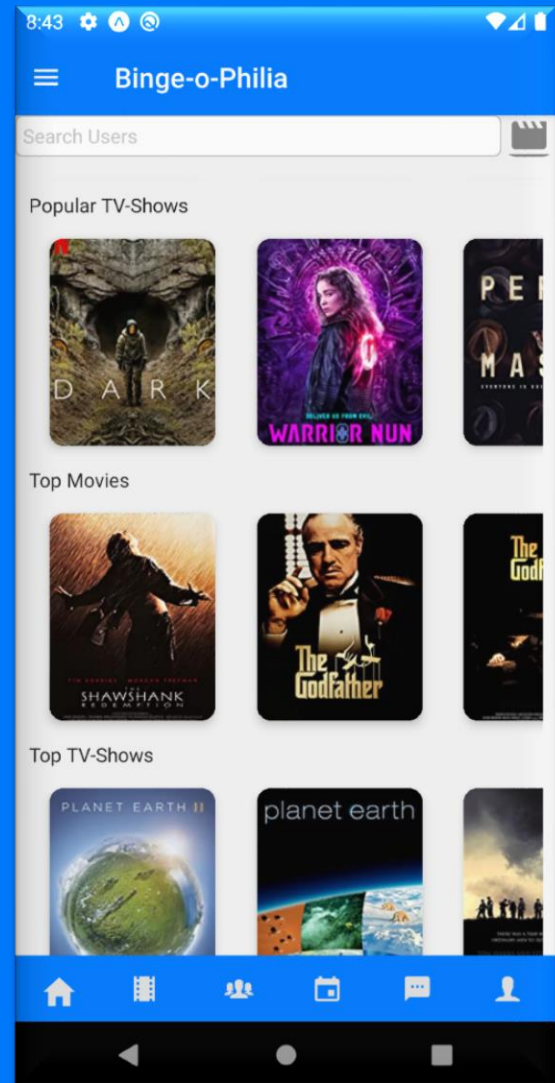
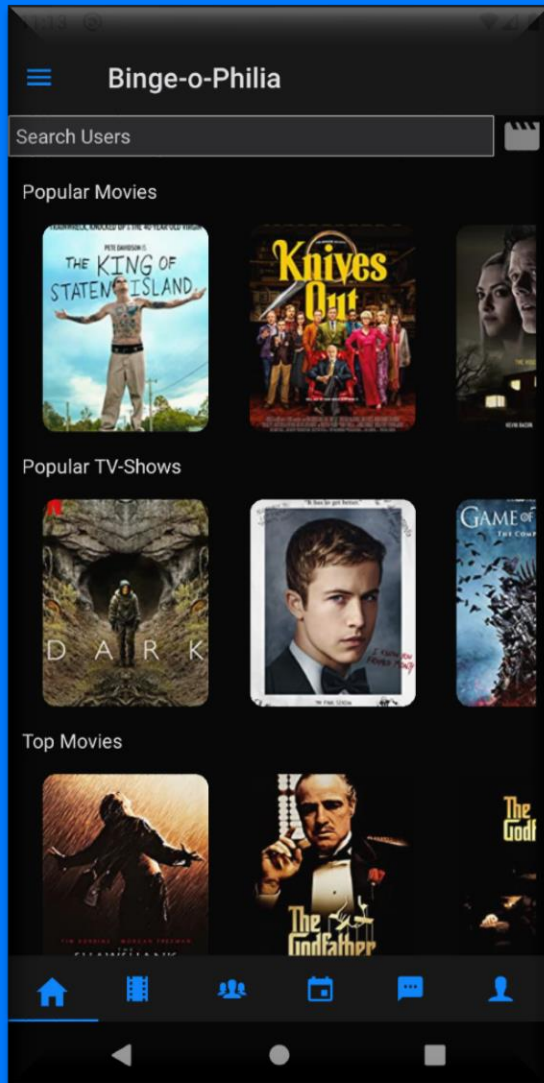# Contents

Technology-Stack

The App is primarily made using React Native Framework (JavaScript and Typescript) and is built around tools and libraries provided by Expo.

The application is backed by Django framework (Python Programming Language) based back-end which is hosted on Heroku Servers.

Database and Storage is completely decoupled from the back-end by using separate services for them in the form of Amazon's AWS S3 and PostgreSQL.
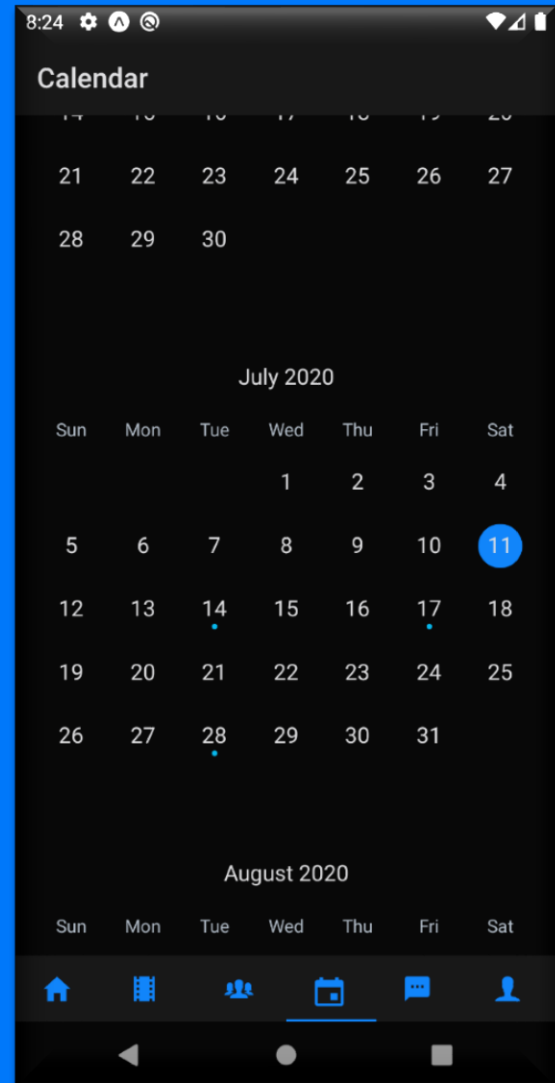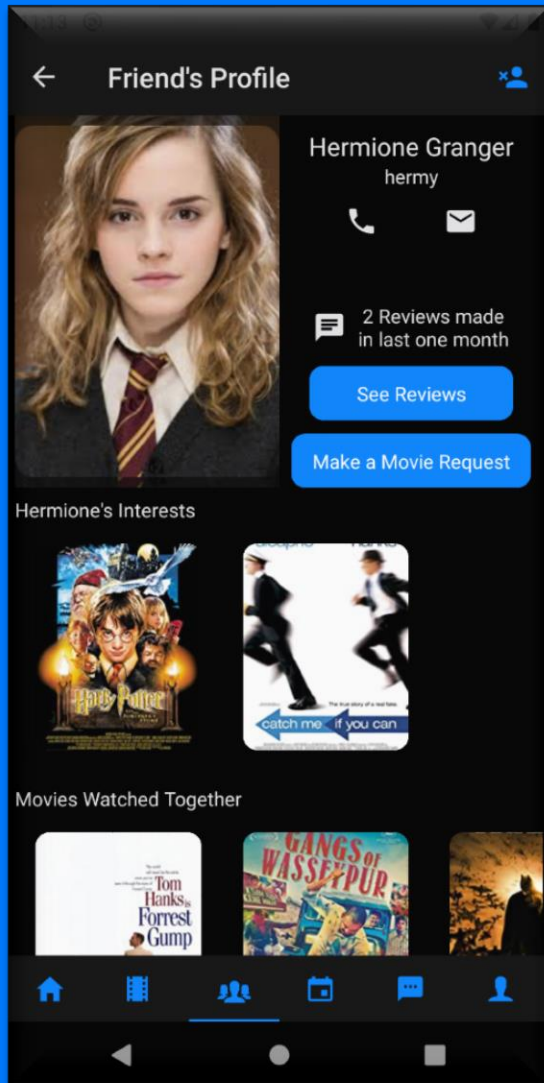
Themes

The application has two supported themes: Dark-Mode and Light-Mode.

Both themes have a sleek design and appropriate colour balance to help users easily spot different elements and navigate seamlessly.

UI and UX

The UI is composed of reusable components which are used all over the app. This provides a coherent experience and easy-to-use interface.

Simple elements like buttons, select-menus and transitions follow standard IXD patterns to provide an overall great User-Experience.
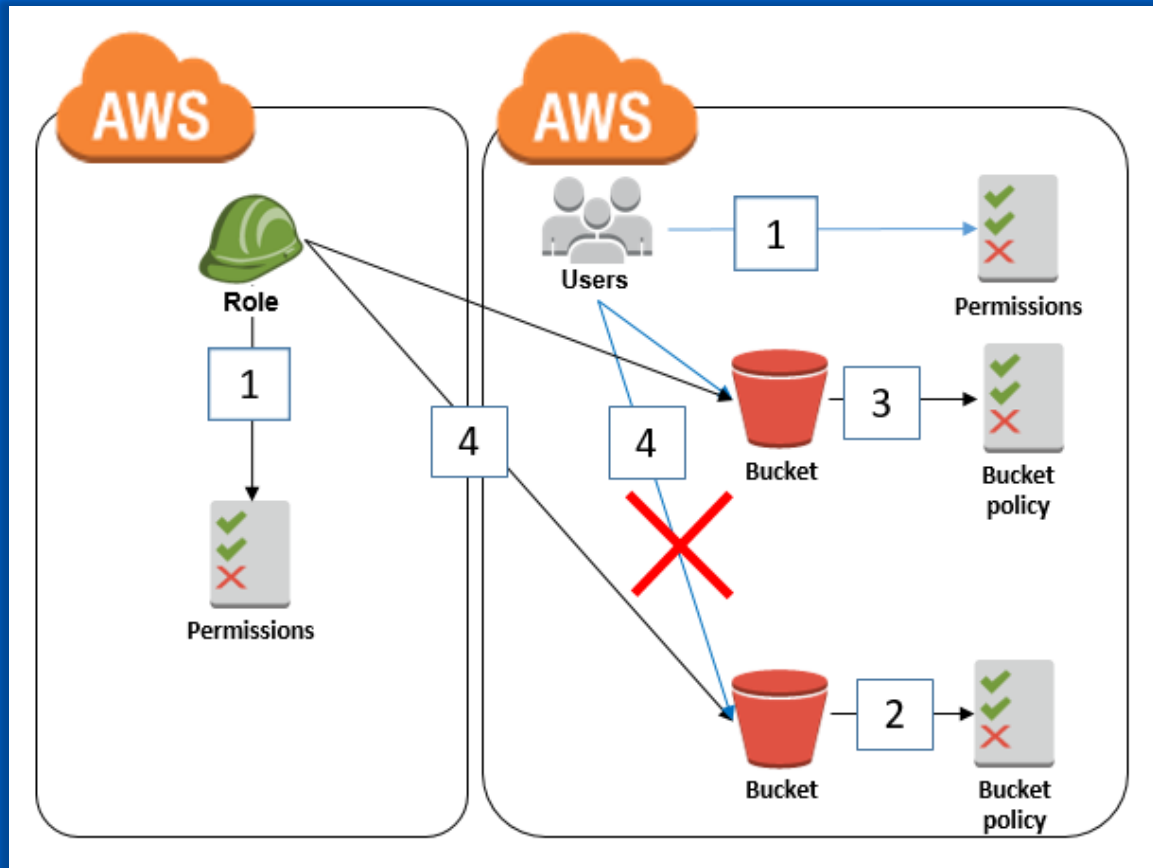
Back-end and Security

Using Django as a back-end framework provides protection against most common attacks such as SQL Injection and Host Header Injection out of the box.

Authorization and authentication are carried out using secure JSON Web Tokens signed by the server in middleware layers in authentication system.

In-app security comes in the form of privacy settings of users – they can choose from whom they want to receive Movie-Requests – *Friends* or *Anyone*.
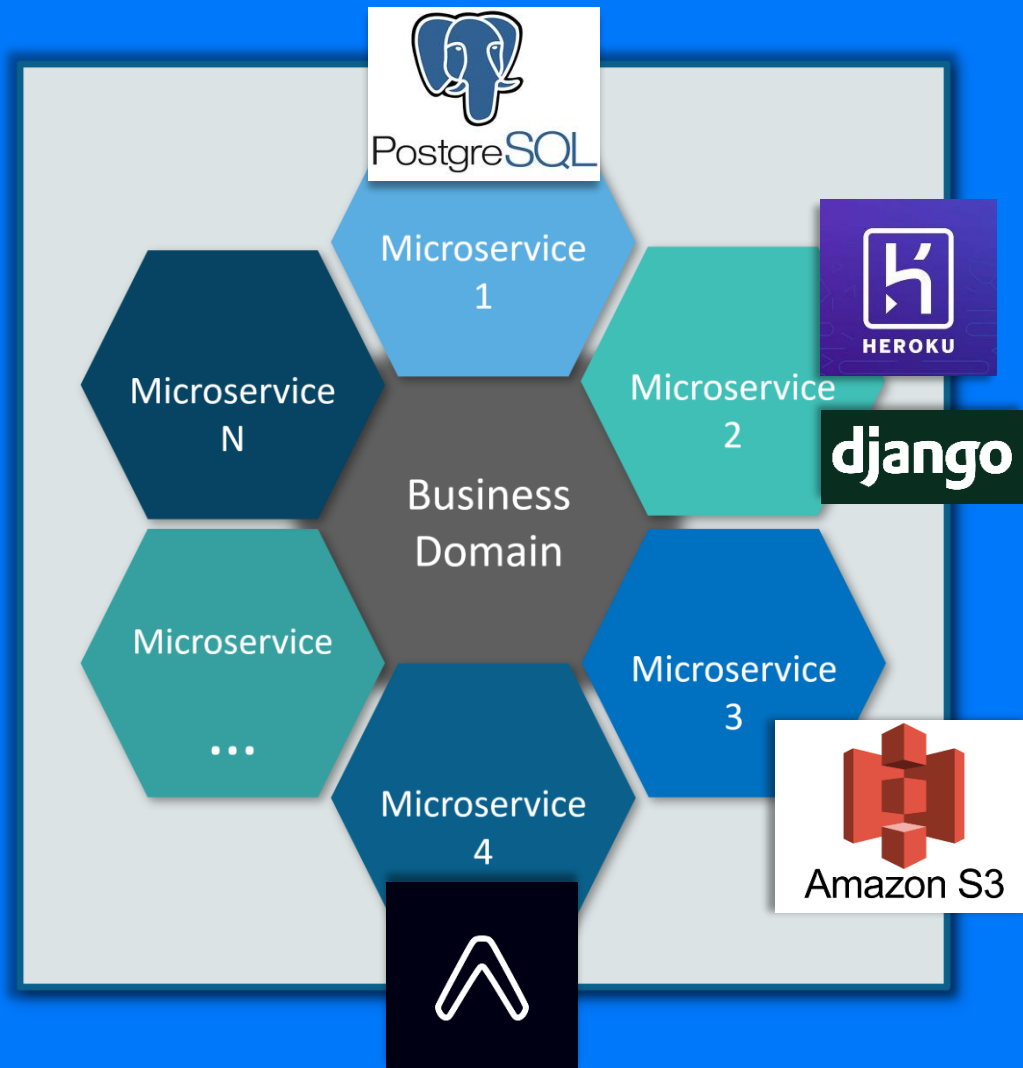
Back-end and Security

Storage security is being handled by the using AWS' IAM Users with bucket-specific policies which only give permissions to particular storage space.

The storage mostly contains the graphics such as profile pictures of the users which is stored in encrypted manner and can only be accessed through IAM' user credentials stored securely in the environment of back-end servers.
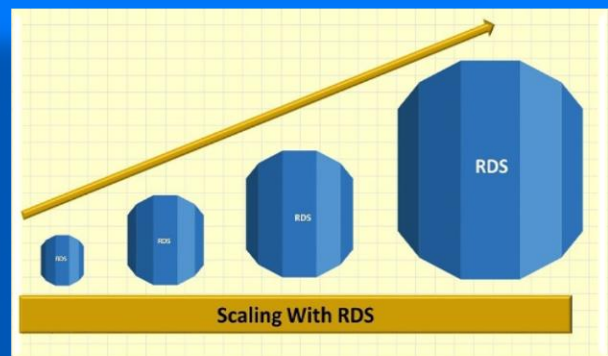
Database and Servers

The resources used: PostgreSQL, Back-End Server (Django + Heroku), Storage Server (AWS S3) and client-side service (Expo) are completely decoupled and interact through secure web network or through inter-process communication.

This decoupling enables easy migration from service to another by simply changing the configuration settings with associated services.
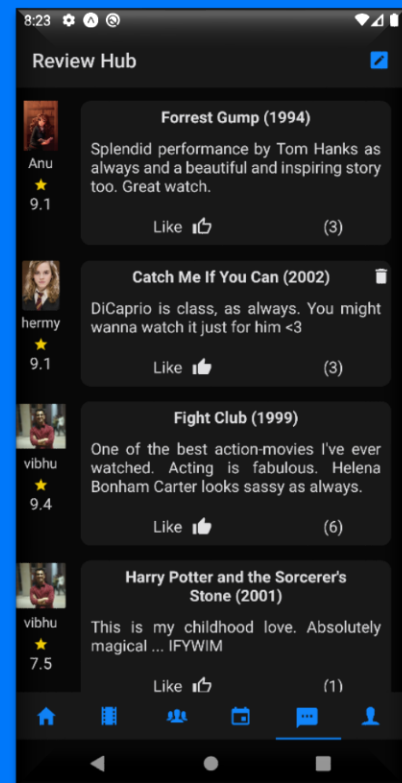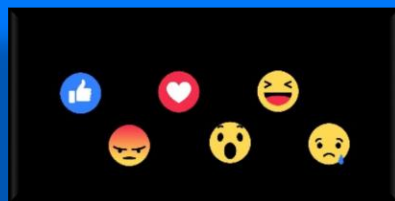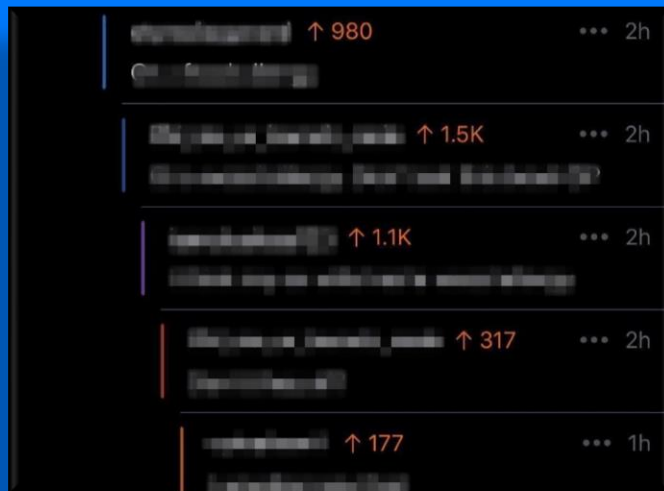
Scalability

PostgreSQL is currently hosted as a service on Heroku which is configured to scale up to 10 million rows if it flows over 10 thousand rows (current plan).

With in-flow of a large user interaction and upcoming budget plans, Binge-o-Philia intends to migrate to Amazon RDS due to its low pricing and awesome set of complementary services.

Auto-Scaling will be configured to adjust according to the fluctuating traffic and will also be able to handle peak usage over holidays and vacations.

Features in Development
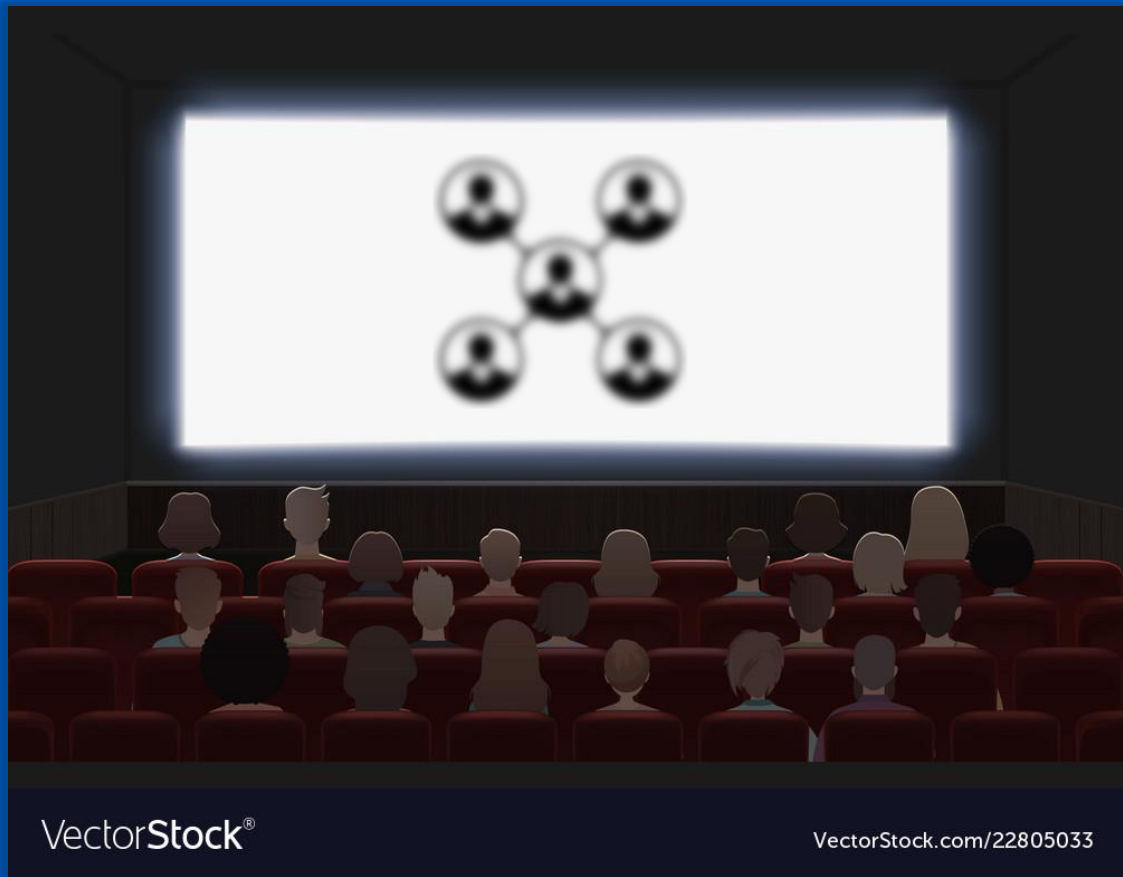
Chatting, Review Reacts and Comments

Communication is the Key

Ranging from a generic messenger to Facebook-like reactions on Movie-Reviews along with comments to further interact with movie-buffs.
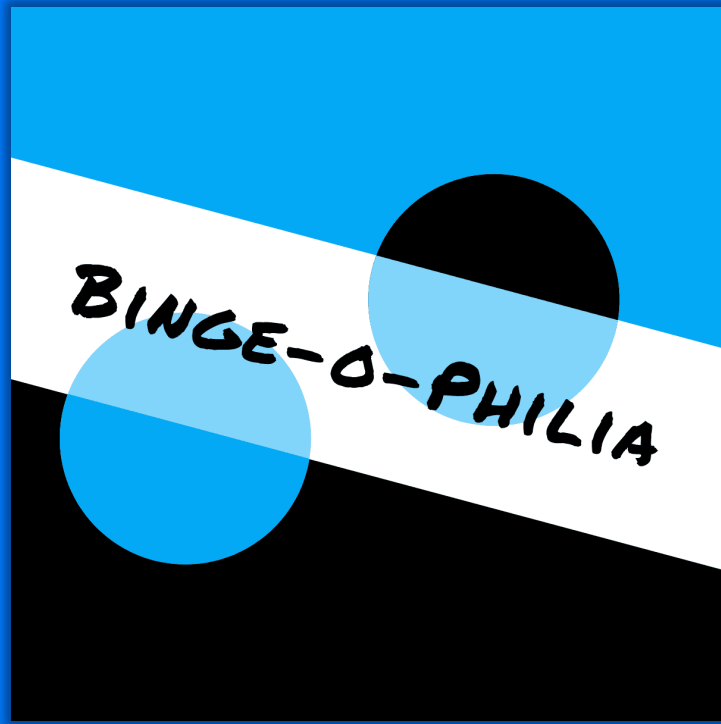
These are few of the major features currently in development for Binge-o-Philia which users will be able to access following the new release scheduled to roll out in a couple of months.

Goals

The goal for Binge-o-Philia is to serve as a single platform for social-networking for anyone who loves cinema, watching movies, tv-shows or fiction, overall.

To bridge the gap between entertainment enthusiasts and increase communications with each other is what the work at Binge-o-Philia strives for.

**Happy
Binge-Watching :)**