



**BINGE**  
**BUDDY**

Binge Buddy:  
Smarter Streaming

# Table of Contents

3	<a href="#"><u>Meet the Team</u></a>	15	<a href="#"><u>Development Tools</u></a>	26-28	<a href="#"><u>App Mockups</u></a>
4	<a href="#"><u>Problem Statement</u></a>	16	<a href="#"><u>Sprint Plan</u></a>	29-30	<a href="#"><u>User Stories</u></a>
5	<a href="#"><u>Identification of Users</u></a>	17	<a href="#"><u>MFCD</u></a>	31	<a href="#"><u>References</u></a>
6-8	<a href="#"><u>Problem Characteristics</u></a>	18	<a href="#"><u>Work Breakdown Structure</u></a>	32-33	<a href="#"><u>End</u></a>
9	<a href="#"><u>Current Process Flow</u></a>	19-20	<a href="#"><u>Algorithm</u></a>		
10	<a href="#"><u>Solution Statement</u></a>	21	<a href="#"><u>Data Schema</u></a>		
11	<a href="#"><u>Solution Process Flow</u></a>	22	<a href="#"><u>Real World Product vs Prototype</u></a>		
12	<a href="#"><u>What it will do</u></a>	23	<a href="#"><u>Required Libraries, Tools, and Technologies</u></a>		
13	<a href="#"><u>What it will not do</u></a>	24	<a href="#"><u>Monetization</u></a>		
14	<a href="#"><u>Competition Matrix</u></a>	25	<a href="#"><u>Risks</u></a>		

# Meet the Team



Hannah  
Dietrich

Computer Science  
Undergraduate student



Todd  
Beatty

Computer Science  
Undergraduate student



Joanna  
Graphman

Computer Science  
Undergraduate student



William  
Ward

Computer Science  
Undergraduate student

# Problem Statement

Viewers struggle to keep track of new episodes and content releases across a multitude of streaming platforms, leading to frustration and inefficiencies in managing their entertainment preferences.



# Identification of Users and StakeHolders:

- Our primary **users** are individuals who actively subscribe to multiple streaming services.
- These users are typically overwhelmed by the number of platforms, fragmented watchlists, and release schedules.
- They want a better way to manage their entertainment without jumping between apps.



# Problem Characteristics

- Navigating through multiple platforms is time consuming
- Users waste time switching between platforms to track their favorite shows
- Inconsistent notifications can cause viewers to miss new episodes
- There is a monetary strain of maintaining multiple subscriptions



r/movies • 9 mo. ago

filmstack

How do you keep track of what you want to watch/have watched?

Discussion

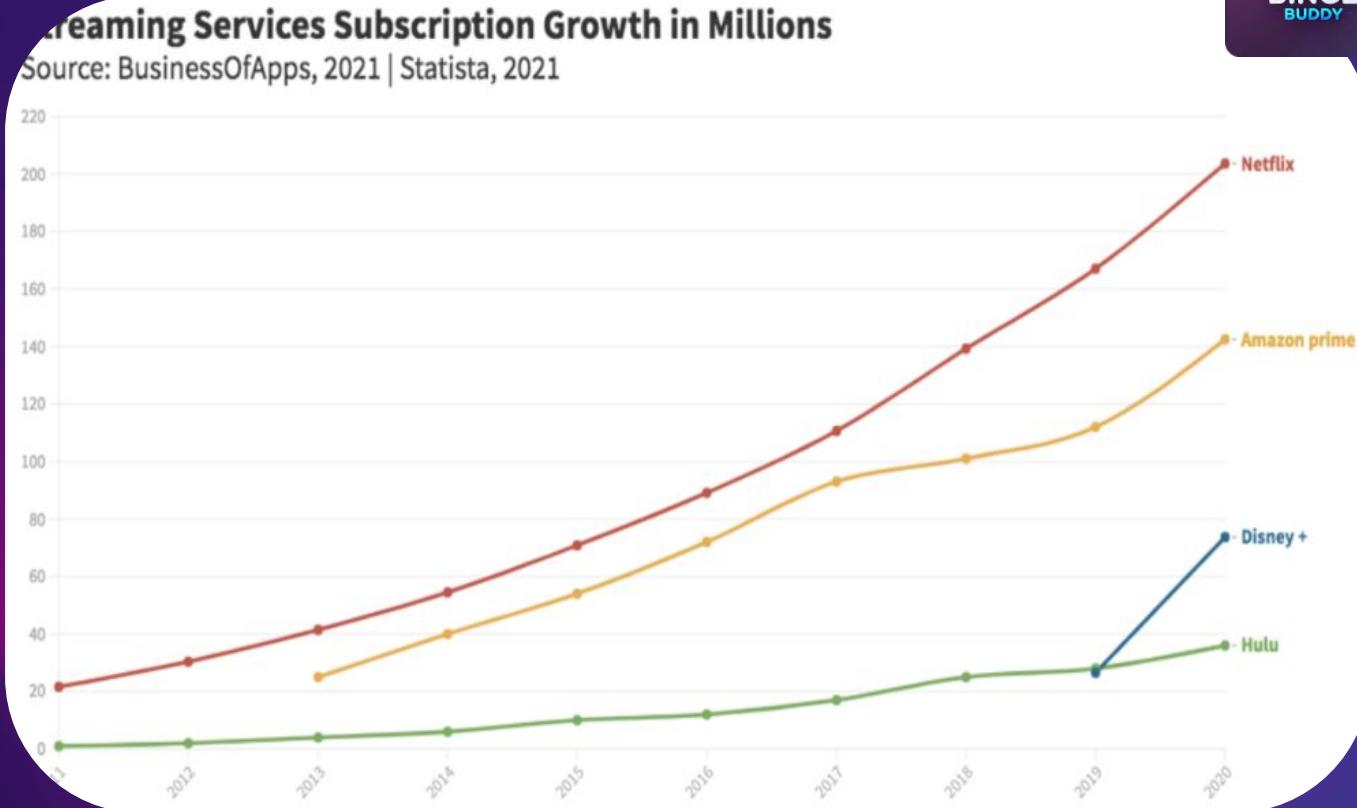
As the title says really! There's options out there like letterboxed and similar websites/apps.

I've personally never found something that ticks enough boxes on its own that I'd like

- Log what you've seen/want to see
- Added functionality to keep TV shows in the same place
- Be able to see where the film is streaming and get alerts when it is (sub vs rent)
- Some kind of calendar view to see when films will be released/come to streaming would be huge
- A way to follow news for upcoming films you're interested in easily
- Something to do with feel, mood, genre that makes picking from the stack much easier than spending an age deciding

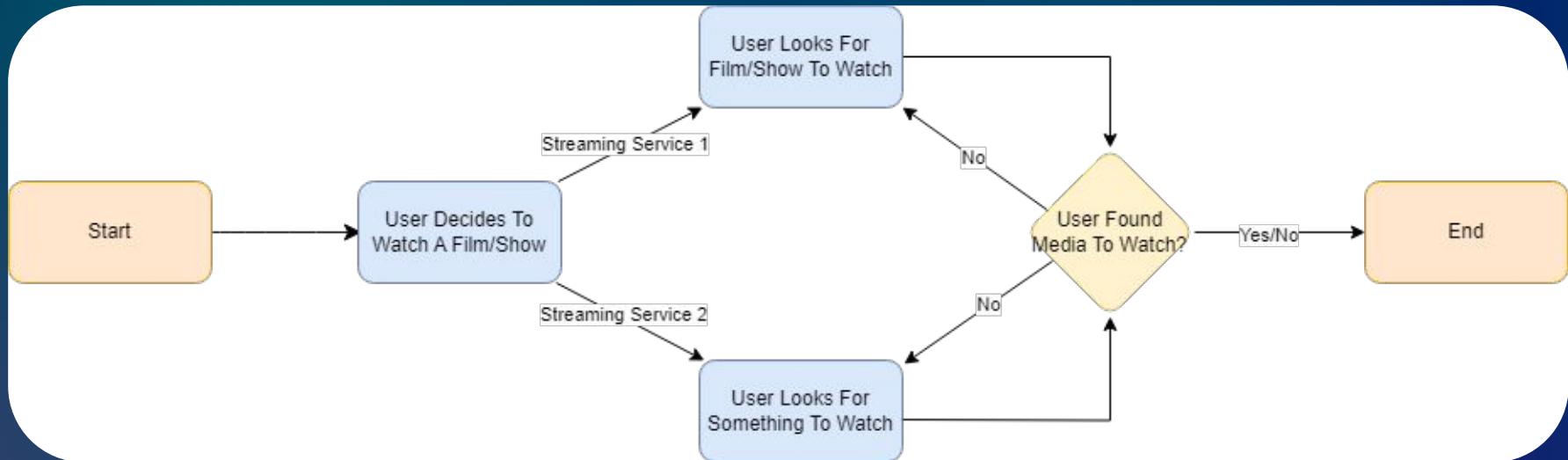
[https://www.reddit.com/r/movies/comments/1cv0gyu/how\\_d\\_o\\_you\\_keep\\_track\\_of\\_what\\_you\\_want\\_to/](https://www.reddit.com/r/movies/comments/1cv0gyu/how_d_o_you_keep_track_of_what_you_want_to/)

- Streaming service subscriptions have grown rapidly and will likely keep rising as platforms keep adding exclusive content.
- Clear need to help viewers track and bookmark across platforms.



<https://amt-lab.org/blog/2021/11/streaming-service-algorithms-are-biased-and-directly-affect-content-development>

# Current Process Flow



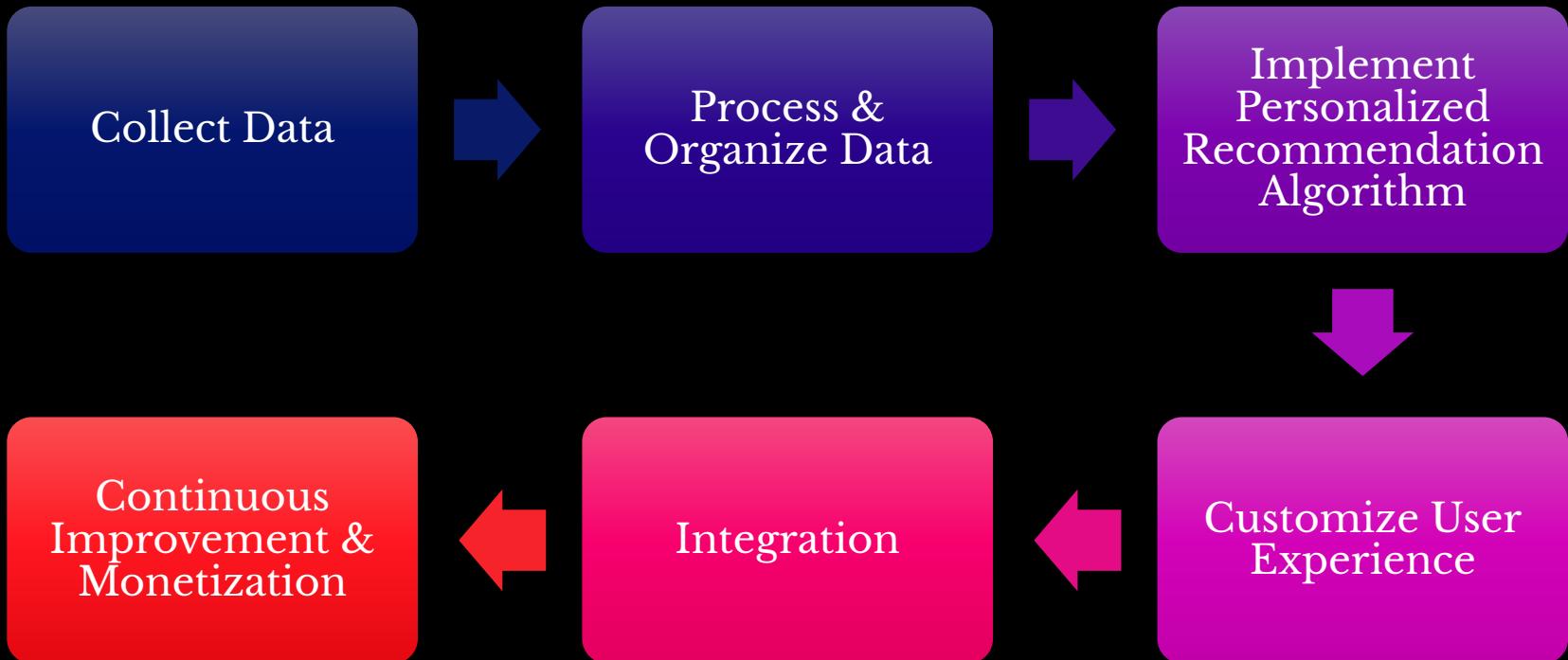
# Solution Statement

Use collected data from available apis and users to create a one-stop-shop for everything related to streaming ( i.e. watchlist, finding shows, etc.).



[me6ck9Ef2SZQrZEFqVxgzc.jpg \(4246×2388\)](#)

# Solution Process Flow



# What BingeBuddy Will Do



## DISPLAY SUBSCRIPTIONS

Display which streaming service(s) each showing is on (greyed out if not logged in/owned)



## BOOKMARKING

Allow bookmarking to track shows



## SEARCH AND FILTER

Allow filtering and searching



## INTEGRATE USER DATA

Recommend TV shows/movies based on user data across streaming services



## USER PREFERENCE

Have a similar interface to most streaming services (top rated, recommended, etc.) that is tailored to user preference.



## AI CHATBOT

Interactive AI chatbot that will find shows and navigate the app

# What BingeBuddy Won't Do

**NO PIRATED OR  
ILLEGAL  
CONTENT**

Won't allow user to watch  
films on the app  
  
Provides link to watch on  
streaming platform

**NO LIVE TV  
TRACKING**

No live sports or news  
tracking

**NO MANAGING  
SUBSCRIPTIONS**

Cannot cancel, renew or  
manage – only view

**NO GAMING  
CONTENT**



**NO PODCAST OR  
MUSIC  
STREAMING**



**NO OFFLINE  
WATCHLIST OR  
DOWNLOADS**



**DOES NOT  
COLLECT  
SENSITIVE DATA**

Such as sensitive  
emotional health data

# Competition Matrix

Feature	BingeBuddy	JustWatch	Reelgood	TV Time
Bookmark Movies & Shows across platforms	✓	✓	✓	✓
Specific new episode notifications	✓	✗	✗	✓
Aggregates multiple streaming services	✓	✓	✓	✗
Personalized recommendations	✓	✗	✗ (Limited)	✗
Syncs with user watch history	✓	✓	✓	✓
Custom user-curate lists	✓	✗	✓	✓
Cross-platform syncing	✓	✗	✓	✗
Price tracking for rentals/subscriptions	✓	✓	✓	✗
AI Chat Bot	✓	✗	✗	✗

# Development Tools

## Integrated Development Environment (IDE)

**VS: FOR ITS EASE OF USE AS WELL AS ITS INTEGRATION OF ADDONS TO AID IN PROGRAMMING**

**NOTEPAD++:**

**ECLIPSE: ITS FREE, GITHUB INTEGRATION.**

**INTELLIJ: CLEANER AND FEELS MORE MODERN IN UI, HAS A CLEAN AUTOCOMPLETE. HAS GITHUB INTEGRATION.**

**GITHUB: FOR BRANCH VERSION CONTROL BETWEEN THE MANY DIFFERENT ITERATIONS OF THE APP AND ITS ABILITY TO ROLL BACK AND KEEP BRANCHES SEPARATE BETWEEN EACH PERSON WORKING ON THE APP**

**GITHUB: FOR FURTHER DEVELOPMENTS THAT CAN BE CONTROLLED AND ROLLED OUT, GITHUB WILL BE USED TO PUSH UPDATES AND ADDED FEATURES TO THE APP AS NECESSARY.**



# Software Development Sprint Plan

## Sprint 0 Goals:

- Define Project
- Create GitHub
- Sprint schedule
- Backlog format
- Research tech
  - ex.React Native
  - ex.NoSQL
  - ex.JustWatch API

## Sprint 1 Goals:

- Begin designing program
- Login/Registration
- Main UI layout and navigation
- Set up basic watchlist creation and display
- Initial streaming data using JustWatch API
- Establish schema and backend structure

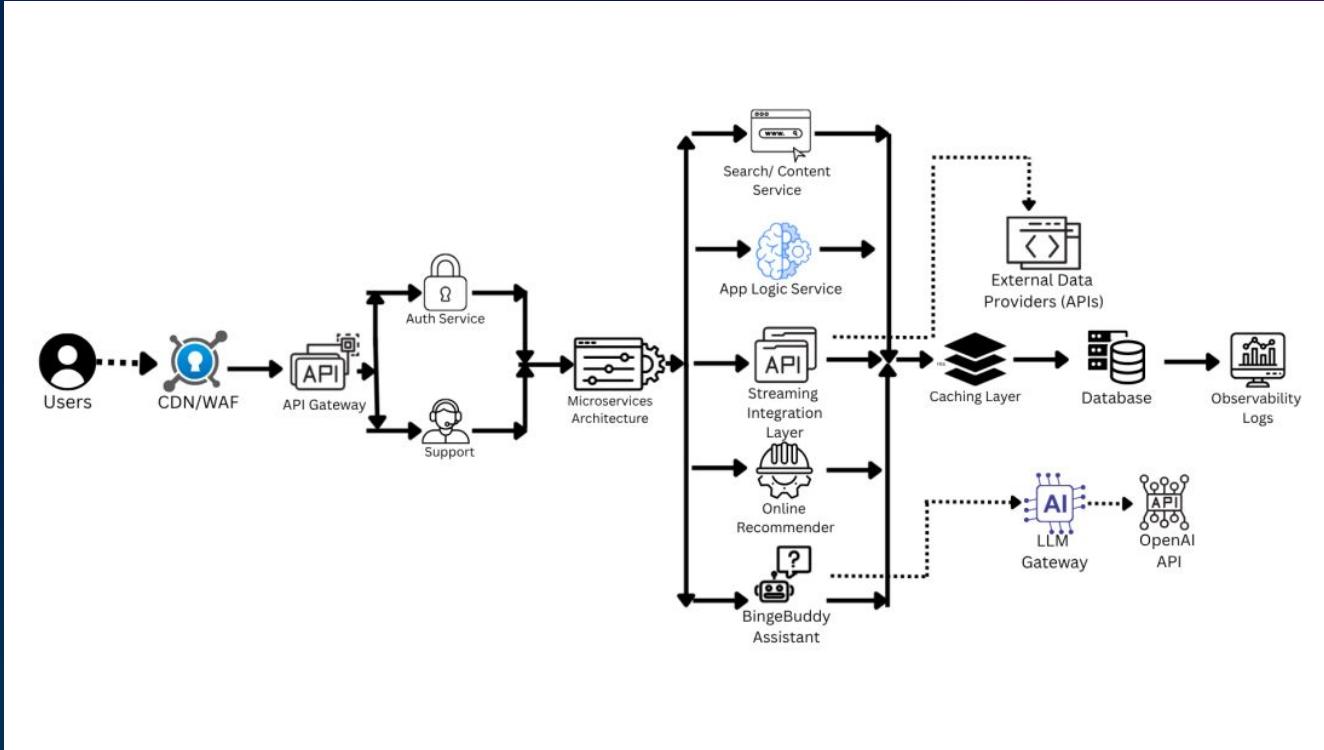
## Sprint 2 Goals:

- Add MoodBot to recommendation engine
- Add Notification scheduler
- Filtering and search functionality
- Begin unit testing

## Sprint 3 Goals:

- Complete beta testing & bug fixes
- Improve recommendation engine
- Add data validation
- Add privacy safeguards

# Major Functional Components Diagram



# Work Breakdown Structure

## Front End:

- GUI Design
- Navigation Logic
- User Authentication UI
- Watchlist UI
- MoodBot Chat UI
- Mood Input Component

## Backend:

- API Design
- User Management
- Notification System
- MoodBot Logic & Response Mapping
- Mood Tag Routing to Recommendation Engine

## Database:

- Schema Setup
- Watchlist Storage
- User Preferences

## DevOps & Testing:

- CI/CD
- Unit Testing
- Beta Testing
- Deployment Setup

# Algorithms

- Search & Filtering
- Recommendation Engine (based on watch history)
- Reminder Notification Scheduler
- Streaming Availability Checker



# Algorithms

- User Behavior Analysis (for analytics, if relevant)
- Duplicate Detection
- User Onboarding Personalization
- Recommendation Algorithm
- Content Matching
- Mood Response Flow

# Data Schema

## USERS

- user\_id
- email
- password\_hash
- platforms
- mood\_history
- created\_at

## MEDIA ITEMS

- media\_id
- title
- genre
- platforms
- mood\_tags
- release\_date
- rating

## WATCHLIST

- user\_id
- media\_id
- status
- added\_at

## NOTIFICATIONS

- notification\_id
- user\_id
- media\_id
- notify\_date
- type

# Real World Product vs Prototype

Feature	Prototype	Real World Product
Account setup and login	Present	Present
Cross-platform data syncing	Present	Present
Search for and filter shows	Present	Present
Display current shows and subscriptions	Present	Present
Watchlists	Present	Present
Personalized recommendations and price tracking	Not available	Present
New episode notifications	Not available	Present
AI Chat Bot	Present	Present

# Required Libraries, Tools, and Technologies



## Programming Languages

- Java
- JavaScript

## Database Technologies

- MySQL

## Frameworks

- React Native
- Spring Boot

## Build Tools

- Maven (For building and development with java)

## Development Tools

- Node.js
- VS code
- IntelliJ
- GitHub

# Monetization

## Ad based monetization:

- Allow companies to buy ad space
- Allow streaming services to give users recommendations based on liked or commonly watched tags or content.



# Risks

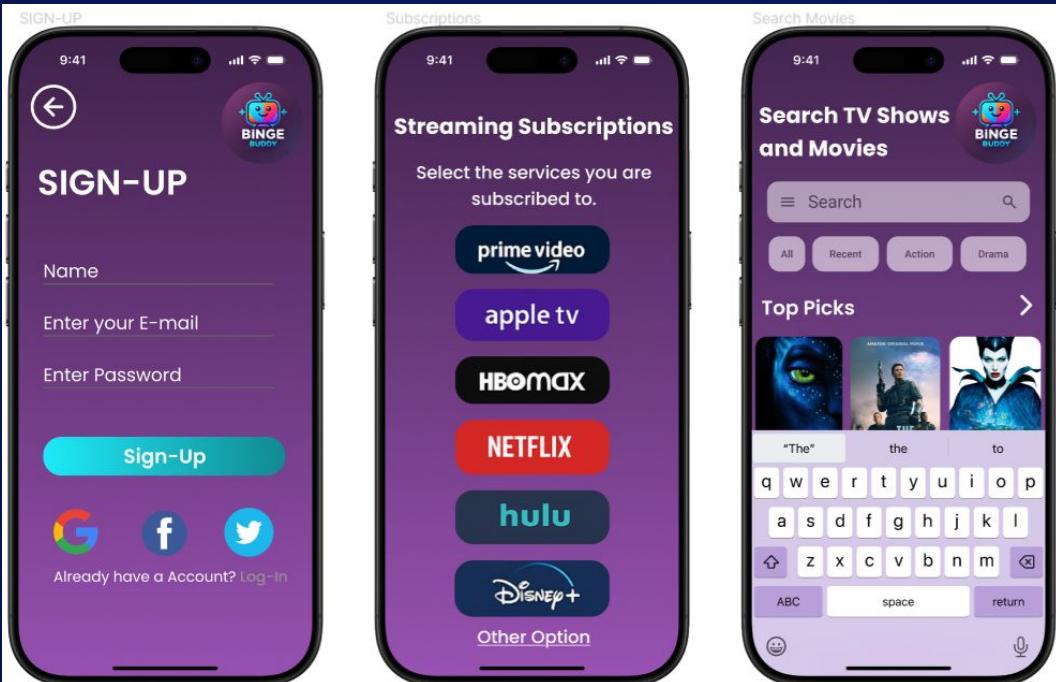
- Cybersecurity risks: internal hosting and storing of both personal data as well as Other PII
- Security risks: man in the middle attacks caused by improper encryption
- Streaming services not allowing access to API



# App Mockups



# App Mockups





# App Mockups



# USER STORIES

A photograph of two hands reaching out from opposite sides of the frame towards each other, set against a solid black background. The hands are positioned horizontally, with fingers slightly spread, suggesting a sense of connection or interaction.

# As a Registered User:

- As a registered user, I need to be able to login to my account.
- As a registered user, I need my streaming subscriptions to be integrated and displayed in one platform.
- As a registered user, I need to be able to search and filter for shows using a variety of filtering options.
- As a registered user, I want to create watchlists and bookmark shows.
- As a registered user, I want to get personalized recommendations.
- As a registered user, I want to get notified when a new episode, season, or movie releases/becomes available.
- As a registered user, I want to be able to use AI to filter shows.

# References

Pangarkar, Tajammul. "Streaming Services Has Experienced Remarkable Growth." *Market.Us Scoop*, Asia-Pacific market news, 3 June 2024, [scoop.market.us](https://scoop.market.us).

Martinez, Sandra. "Streaming Service Algorithms Are Biased, Directly Affecting Content Development." *AMT Lab @ CMU*, AMT Lab @ CMU, 16 Aug. 2022, [amt-lab.org/blog/2021/11/streaming-service-algorithms-are-biased-and-directly-affect-content-development](https://amt-lab.org/blog/2021/11/streaming-service-algorithms-are-biased-and-directly-affect-content-development).

"R/Movies on Reddit: How Do You Keep Track of What You Want to Watch/Have Watched?" *Reddit*, Reddit, 2024, [www.reddit.com/r/movies/comments/1cv0gyu/how\\_do\\_you\\_keep\\_track\\_of\\_what\\_you\\_want\\_to/](https://www.reddit.com/r/movies/comments/1cv0gyu/how_do_you_keep_track_of_what_you_want_to/).

Zhao, Yan, and Shoujin Wang. "MbSRS: A Multi-Behavior Streaming Recommender System." *Information Sciences*, Elsevier, 31 Jan. 2023, [sciencedirect.com](https://www.sciencedirect.com).



# Thank You

<https://strillakilla.github.io/BingeBuddy/>



BINGE  
BUDDY