



CineMatch: Movie Recommendation App

Version 1.0

Prepared By:

Baldivino, Reynante O.

Lead Developer, 09268100382

Gallardo, Jessa Y.

QA and Documentation Specialist, 09945286529

Lagare, Angel Mae L

Project Manager, 09979258119

May 9, 2025

Table of Contents

I. Document Overview	1
1.1 Scope	1
1.2 Audience	1
II. Project Overview	1
2.1 Executive Summary	1
2.1.1 Objectives	1
2.1.2 Main Features	2
2.2.1 User Needs	3
2.2.2 Market Analysis	4
III. Functional Specifications	5
3.1 Feature list	5
3.2 User Stories & Requirements	16
3.2.1 User Stories	16
3.2.2 Acceptance Criteria	17
3.2.3 Non-functional Requirements	18
IV. Technical Specifications	20
4.1 Architecture	20
4.1.1 Overview Diagram	20
4.1.2 Design Patterns	21
4.2 Platform-Specific Considerations	22
4.2.1 iOS and Android Guidelines	22
4.2.2 Hardware & OS Compatibility	22
4.3 Data Management	23
4.3.1 Data Flow Diagrams	23
4.3.2 Database Schemas	24
4.4 Security & Privacy	29
4.4.1 Authentication & Authorization	29
4.4.2 Data Encryption	29
4.4.3 Compliance Requirements	29
4.4.4 User Data & Privacy Policies	30
4.5 Third-Party Integration	31
4.5.1 SDKs & Libraries	31
4.5.2 Integration Details	32
V. UI/UX Design Specifications	33
5.1 Wireframes & Mockups	33
5.2 Navigation & Flow	38
5.2.1 User Flow Diagrams	38
5.2.2 Accessibility Guidelines	39

VI. Deployment & Maintenance	41
6.1 Deployment Plan	41
6.1.1 Release Process	41
6.1.2 Rollout Strategy	42
VII. Appendices	43
7.1 Glossary	43
7.2 References & Resources	44

Date and Revision History

Name	Date	Reason For Changes	Version

I. Document Overview

1.1 Scope

CineMatch's functionality centers on TMDb integration, offering personalized movie recommendations, seamless search capabilities, bookmarking movies and trailer playback. The Android application helps users discover films efficiently while keeping their favorites easily accessible through bookmarking. While the system includes Google account authentication for secure login, it currently does not support iOS devices due to development scope limitations. The project excludes social features, offline viewing modes, and subscription-based services to maintain its focus on streamlined movie discovery.

1.2 Audience

CineMatch is designed for movie lovers and developers. For users, the movie lovers, it helps casual viewers discover films, they can search movies according to their preferences, and also bookmarked their favorite movies and or to watch it later. On the technical side, the documentation supports Android developers learning TMDb API integration, designers following Material Design, and testers checking core features like search and trailers. Whether you're exploring movies or code, CineMatch keeps things simple and practical.

II. Project Overview

2.1 Executive Summary

2.1.1 Objectives

- Provide personalized movie recommendations based on user preferences, watch history, and bookmarks.
- Enable users to easily search, discover, and explore movies by title, genre, or keyword.

- Allow users to view detailed movie information, including trailers, cast, ratings, and summaries, all within the app.

2.1.2 Main Features

Table 1: Main Features of CineMatch

Feature	Description	API Used
Google Sign In	Allows users to securely log in using their Google account.	Google Sign-In API
Personalized Recommendation	Displays movies tailored to the user's viewing history and selected genres.	TMDb API
Movie Search	Enables users to search movies by title or category such as action or romance.	TMDb API
Movie Details	Provides summary, rating, genre, and cast for each selected movie.	TMDb API
Watch Trailers	Lets users watch trailers directly within the app using embedded video.	YouTube Data API
Bookmarked Movies	Allows users to save movies to a bookmark list.	Firebase Firestore
View Bookmarked List	Displays a collection of saved movies in a dedicated tab.	Firebase Firestore
Profile Customization	Lets users update their profile picture and username within the app.	Firebase Authentication

2.2 Problem Statement

Many users spend too much time trying to find the right movie to watch. Existing movie apps often show suggestions that do not match the user's taste or are not personalized. This leads to frustration and wasted time, especially when users are tired or unsure of what to pick. CineMatch solves this problem by giving recommendations that are based on the user's actual preferences and previous choices. This makes it easier to discover enjoyable movies without the usual hassle. In addition to the need for better recommendations, users also want simple and quick access to movie details and trailers. Some apps make users switch between different platforms just to see a trailer or learn more about a film.

2.2.1 User Needs

Three main user needs that result in the development of CineMatch.

1. **Personalized Recommendations** - Users need a movie app that understands their preferences and viewing habits. They want accurate suggestions that match their mood, favorite genres, and past watch history without spending too much time scrolling through unrelated content.
2. **All-in-One Movie Information** - Users prefer an app that offers everything in one place trailers, summaries, ratings, genres, and cast details. They want a smooth experience without switching between different platforms just to learn about a movie.
3. **Easy Watchlist Management** - Users expect a simple way to bookmark and organize movies they like or plan to watch later. They want to track their favorites and explore similar titles without starting from scratch each time they open the app.

2.2.2 Market Analysis

Table 2: Competitive Landscape of Cinematch

Solution	Target Audience	Strengths	Gaps Addressed by CineMatch
Netflix	General audiences, binge-watchers	Popular content, autoplay trailers	Lacks personalized suggestions across genres; no bookmark system for future viewing
IMDb	Movie researchers, film critics	Detailed movie info, ratings, user reviews	No built-in trailer viewing; lacks bookmark and recommendation features
Just watch	Streaming platform searchers	Multi-platform search engine	No personalized recs or in-app trailer viewing
Youtube	Casual viewers and trailer watchers	Massive trailer library, easy video access	Not focused on movie discovery; no personalized lists or summaries
CineMatch	Young adults, movie lovers	Personalized suggestions, all-in-one movie info hub	Combines login, smart suggestions, trailer viewing, and bookmarking in one lightweight Android app

III. Functional Specifications

3.1 Feature list

Feature Name: Login with Google

This feature lets users log in quickly and securely using their Google account, eliminating the need to create a new username or password. It streamlines access, personalized movie recommendations, and keeps user data safe within a trusted platform.

User Flow:

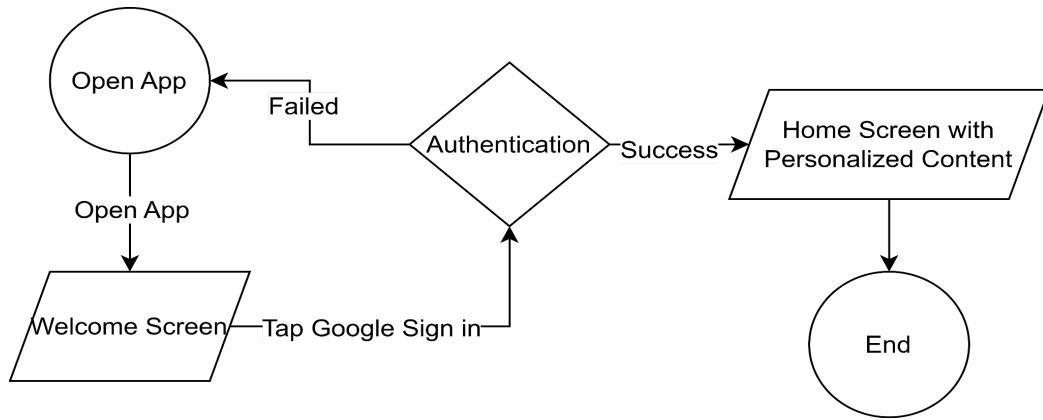


Figure 1: User Flow Login With Google

Use Case:

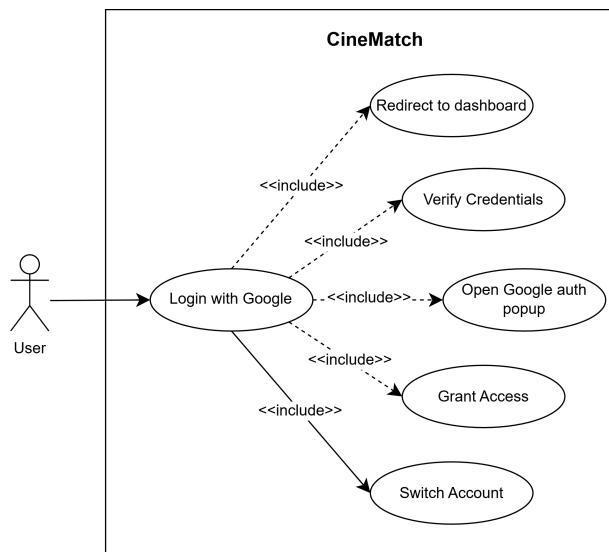


Figure 2: Primary Flow Login With Google

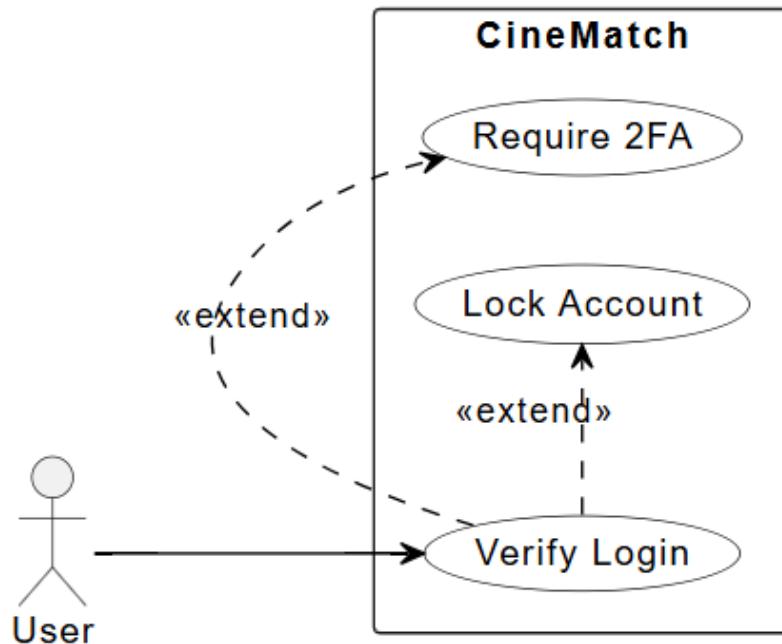


Figure 3: Alternative Flow Login With Google

Feature Name: Personalized Movie Recommendation/ Suggestion

After logging in, users are shown a list of movies specially selected based on what they like. These suggestions are not random; they're picked using the user's viewing history, favorite genres, and saved bookmarks. This feature helps users quickly find something they're more likely to enjoy, instead of scrolling endlessly through unrelated titles. The recommendations update regularly to give users new options each time they use the app. It is the heart of CineMatch, designed to save time and improve the overall movie-picking experience.

User Flow:

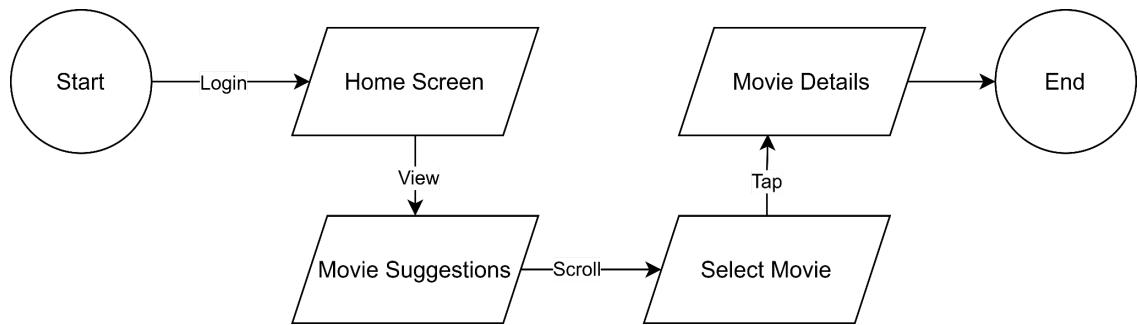


Figure 4: User Flow Personalized Movie Recommendation

Use Case:

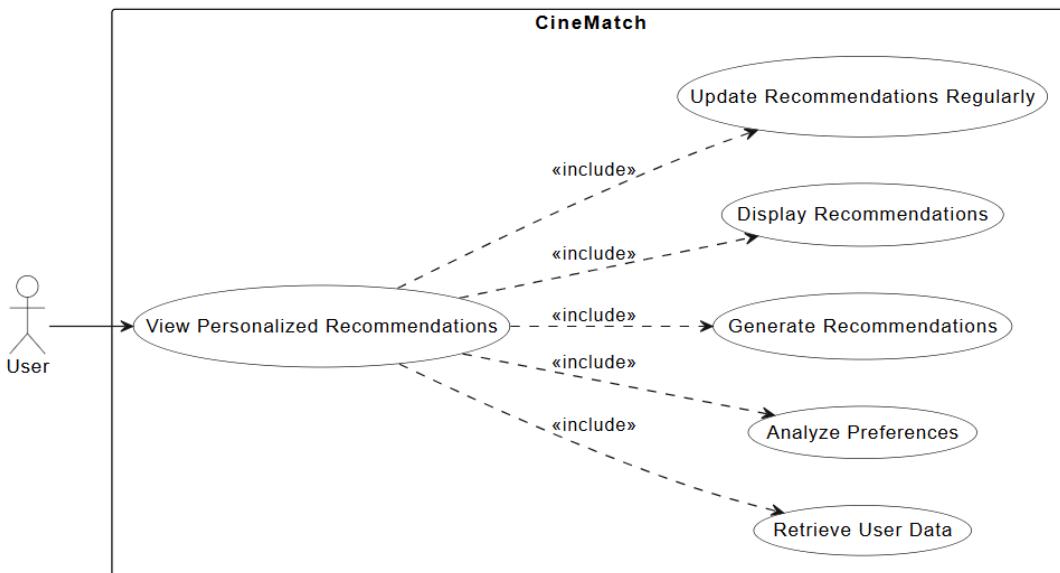


Figure 5: Primary Flow Movie Recommendation

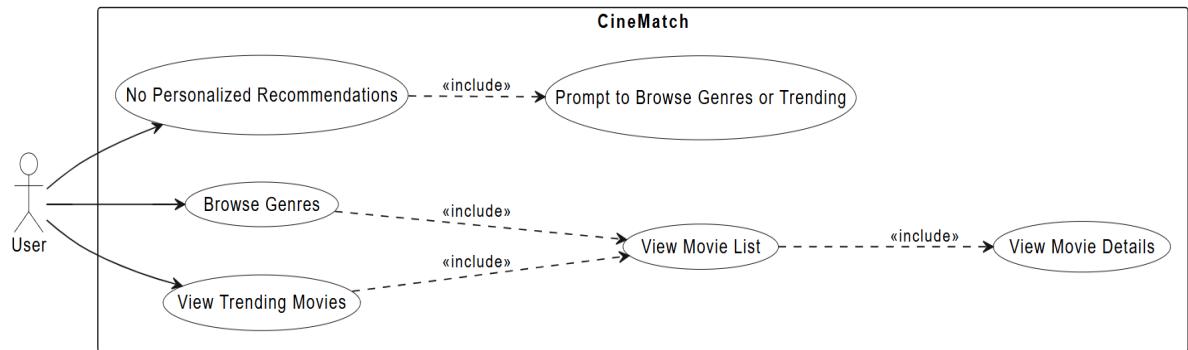


Figure 6: Alternative Flow Movie Recommendation

Feature Name: Movie Search

The search feature allows users to find specific movies by typing the title, a keyword, or even a genre. It is useful when the user already has a movie in mind or wants to explore something related to a certain theme or topic. The search results appear instantly as the user types, making it quick and responsive. This feature makes it easy to look up older films, new releases, or even movies based on mood. It helps users take control of what they want to watch instead of relying only on suggestions.

User Flow:

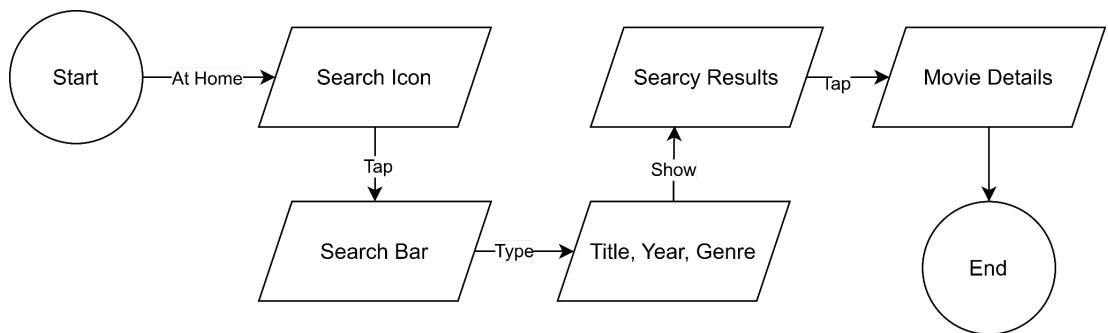


Figure 7: User Flow Movie Search

Use Case:

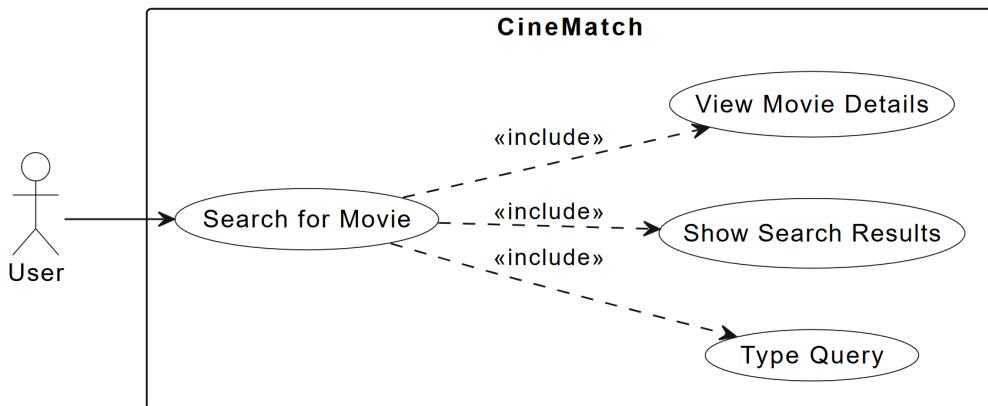


Figure 8: Primary Flow Movie Search

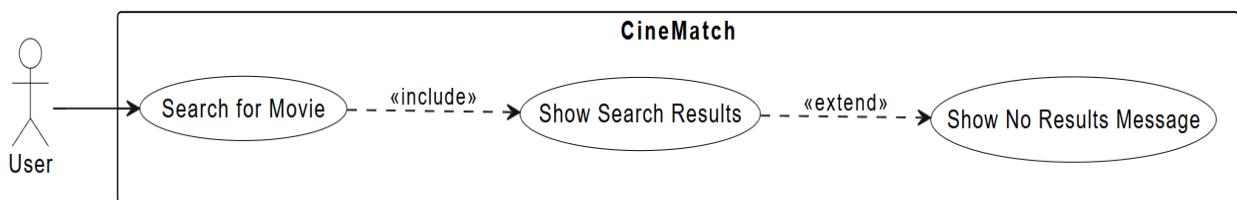


Figure 9: Alternative Flow Movie Search

Feature Name: Movie Details

Each movie listed in CineMatch includes a detailed page that shows everything the user needs to know before deciding to watch. This includes a short plot summary, the movie's rating, genre, cast, release year, and more. It helps the user understand the story, tone, and main characters without needing to search elsewhere. This saves time and helps with decision-making, especially for users who want to know more before committing to a film. It also gives quick access to related features like bookmarking and trailers.

User Flow:

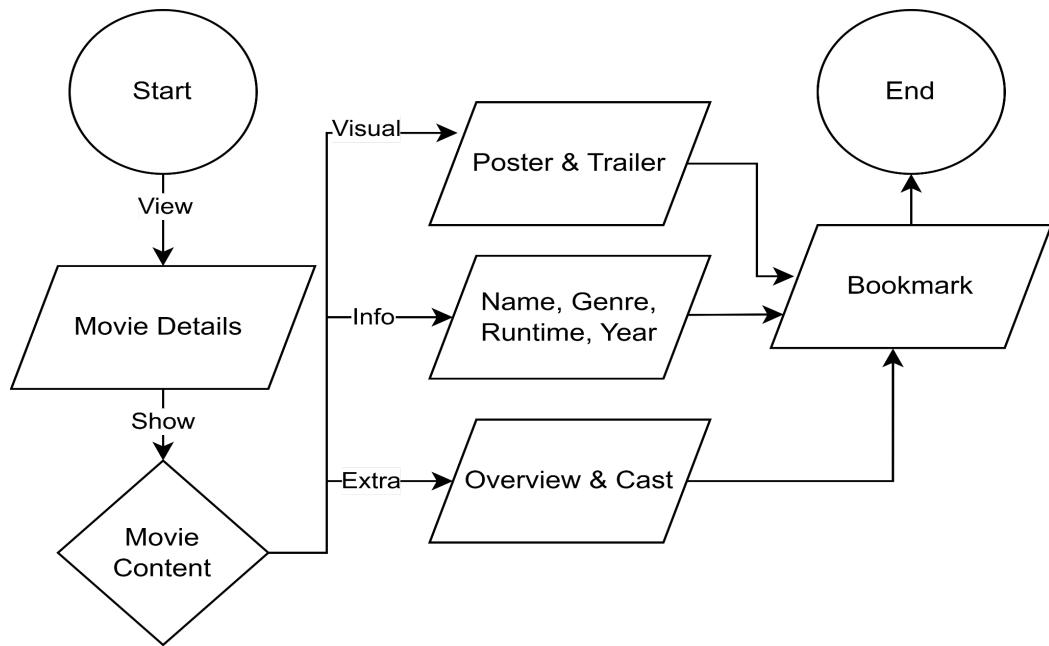


Figure 10: User Flow Movie Details

Use Case:

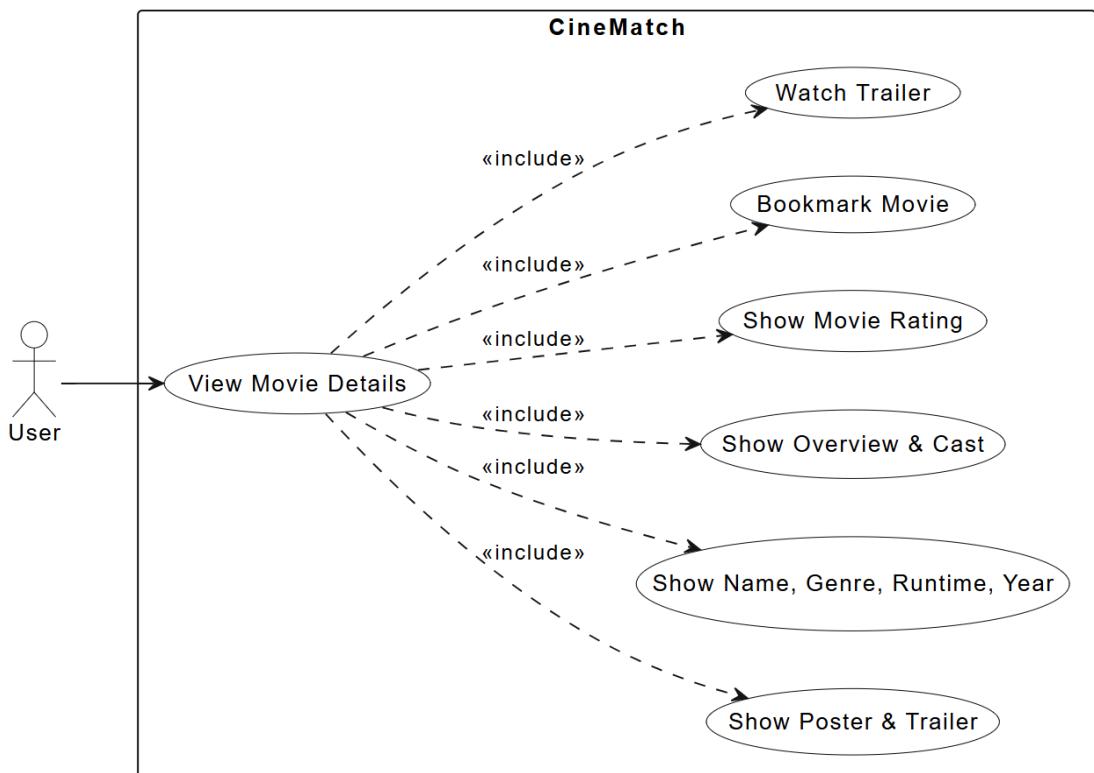


Figure 11: Primary Flow Movie Details

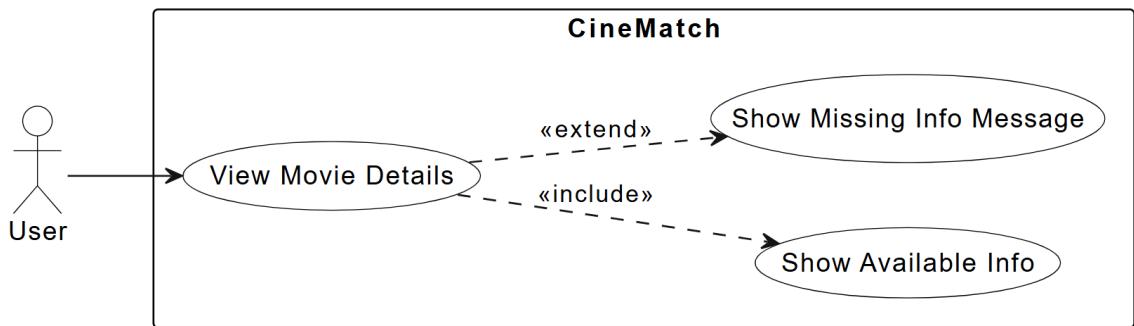


Figure 12: Alternative Flow Movie Details

Feature Name: Watch Trailers

CineMatch lets users watch movie trailers directly within the app, using a built-in YouTube player. This means there's no need to switch to another app or browser just to see what a movie looks like. The trailer feature gives users a better sense of the film's tone, visuals, and storyline before they watch the full movie. It also adds excitement and helps users decide whether to bookmark the film. It's quick, convenient, and works smoothly within the app.

User Flow:

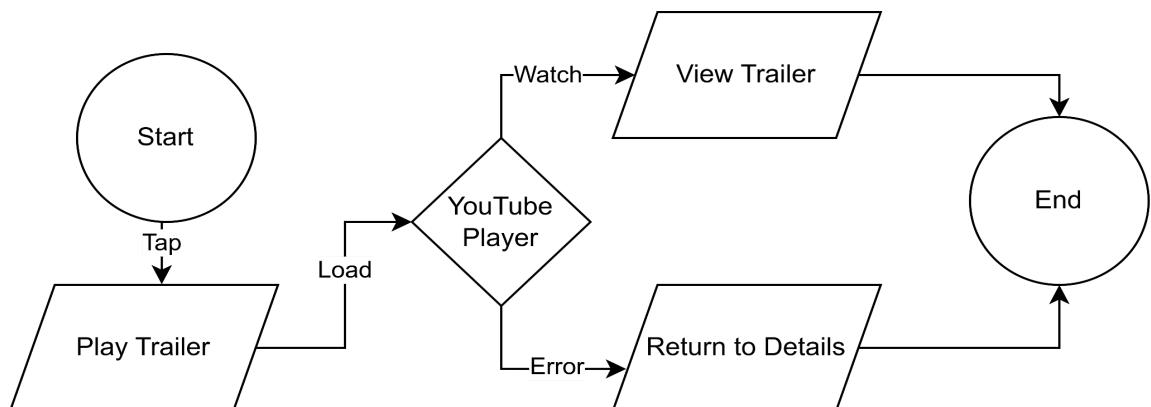


Figure 13: User Flow Watch Trailers

Use Case:

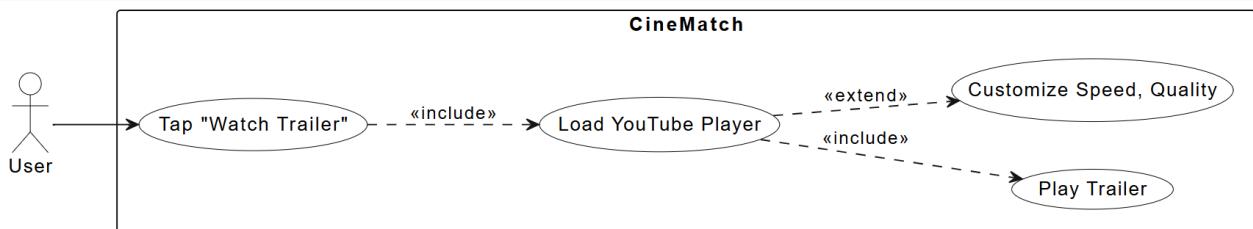


Figure 14: Primary Flow Watch Trailers

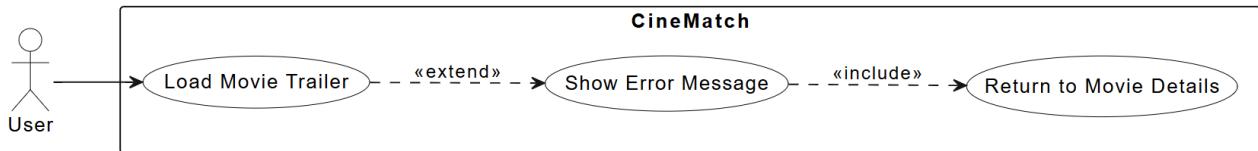


Figure 15: Primary Flow Watch Trailers

Feature Name: Bookmark Movies

This feature allows users to save the movies they like or plan to watch later by tapping a simple bookmark icon. It's helpful for users who want to keep track of interesting titles without having to search for them again. Bookmarked movies are stored in a separate list for easy access anytime. Whether users are planning a movie night or just collecting favorites, this feature keeps everything organized. Bookmarks are saved securely and stay available as long as the user is logged in.

User Flow:

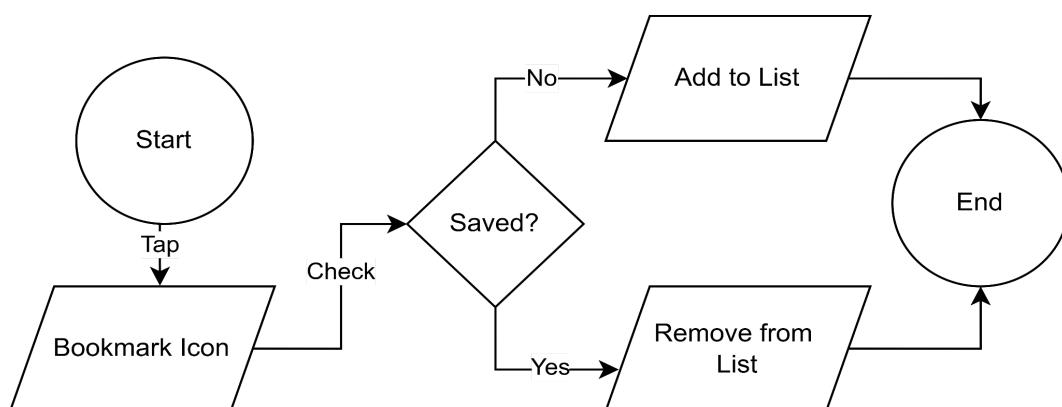


Figure 16: User Flow Bookmark Movies

Use Case

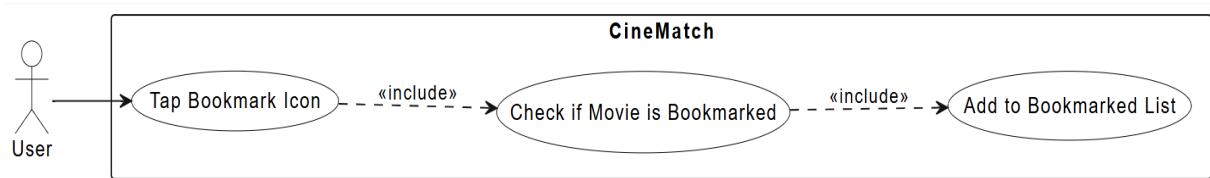


Figure 17: Primary Flow Bookmark Movies

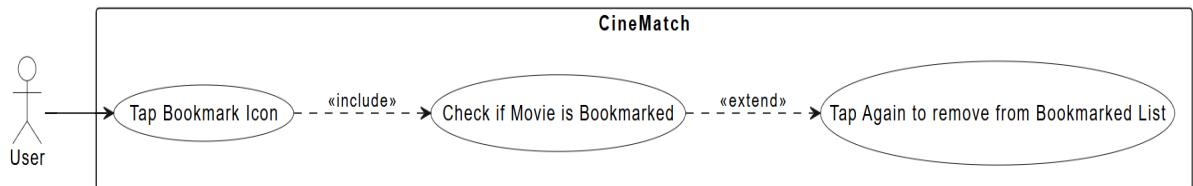


Figure 18: Alternative Flow Bookmark Movies

Feature Name: View Bookmarked List

Users can view all the movies they have bookmarked in one dedicated section of the app. This feature makes it easy to revisit saved movies without needing to remember their titles or search again. Each bookmarked movie can be tapped to see details, play the trailer, or decide to watch it. It acts like a personal movie shelf, helping users plan future watchlists or remember past favorites. The list can grow and update as users continue to explore the app.

User Flow:

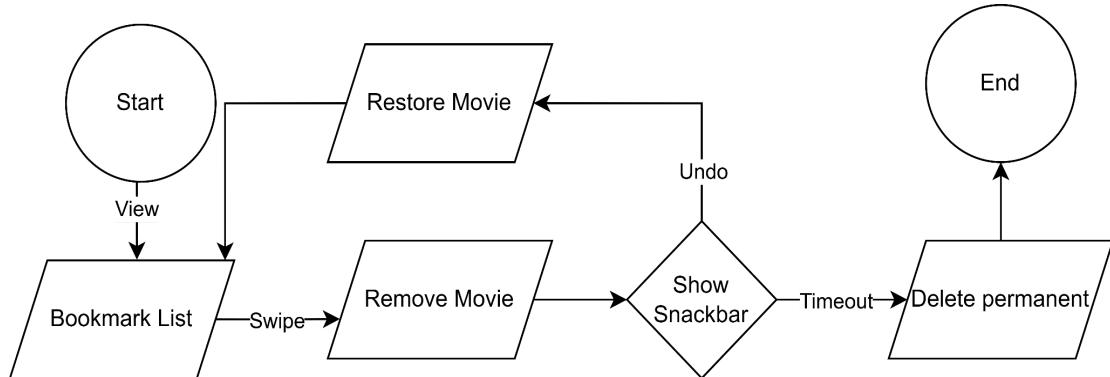


Figure 19: User Flow View Bookmarked List

Use Case

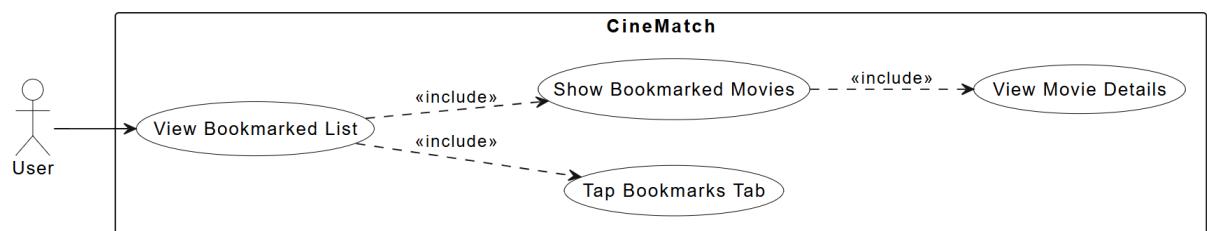


Figure 20: Primary Flow View Bookmarked List

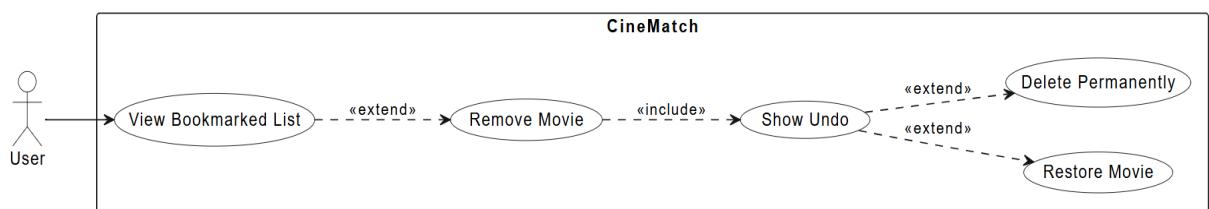


Figure 21: Alternative Flow View Bookmarked List

Feature Name: Profile Customization

Users can manage their personal profile information in a dedicated settings section of the app. This feature allows users to update their profile picture and username, while also displaying their email address for reference. The settings page serves as a personal hub where users can maintain their account information. Profile changes are saved automatically and reflected across the app, ensuring a consistent user experience. The interface is designed to be intuitive, making it easy for users to keep their profile information up to date.

User Flow:

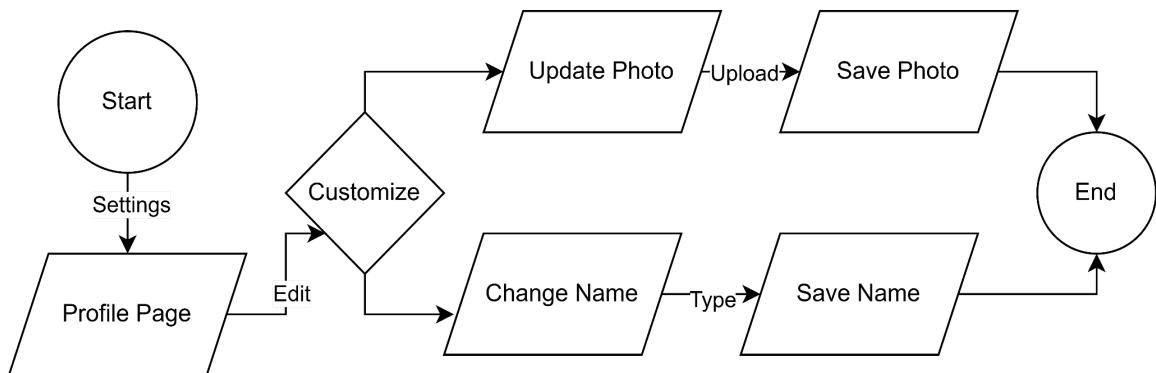


Figure 22: User Flow Profile Customization

Use Case:

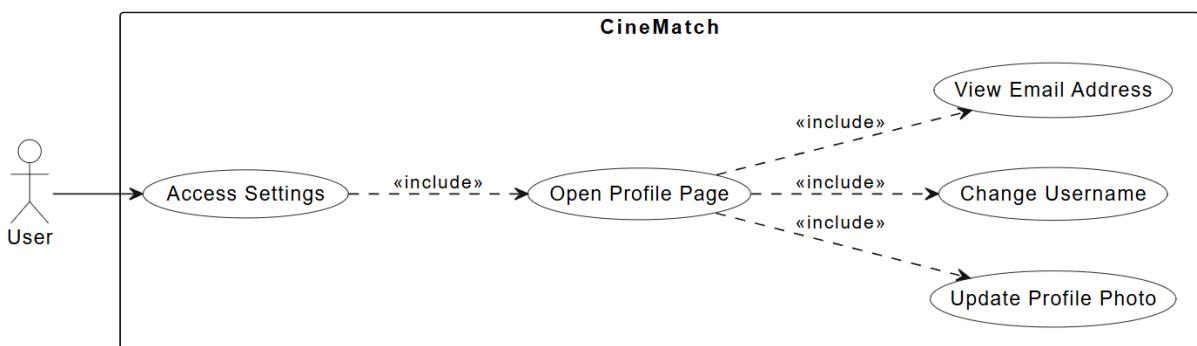


Figure 23: Primary Flow Profile Customization

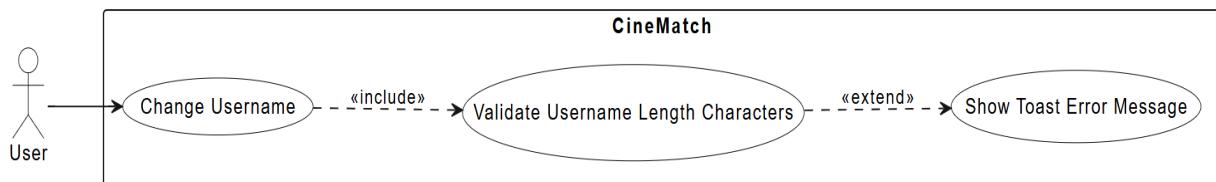


Figure 24: Alternative Flow Profile Customization

3.2 User Stories & Requirements

3.2.1 User Stories

Table 3. User Stories for CineMatch

Category	User Role	User Story
Authentication	Remote Worker	As a movie enthusiast, I want to log in via Google OAuth so I can securely access my account without creating a new one.
	Casual Viewer	As a casual viewer, I want session persistence so I don't have to log in every time I open the app.
Movie Discovery	Indecisive Watcher	As an indecisive watcher, I want personalized movie recommendations so I can quickly decide what to watch.
	Watcher	As a watcher, I want to filter movies by genre so that I can find movies matching my current preferences.

Movie Info Access	Film Critic	As a film critic, I want access to real-time movie details from trusted databases so I can explore movie reviews and metadata.
Bookmarked List	Returning Viewer	As a returning viewer, I want to access all my bookmarked movies in one place so I can easily continue watching or review my saved choices.
User Interface	First-Time User	As a first-time user, I want a clean and simple interface so I can use the app easily without needing instructions.

3.2.2 Acceptance Criteria

Table 4. Acceptance Criteria for CineMatch

Feature	Acceptance Criteria
Google Login	User logs in with Google account within 2 clicks; session token expires after 24h.
Personalized Recommendations	Recommended movies update based on user's watch history, ratings, and bookmarks; list refreshes on each login.
Movie Search	Search returns relevant results for title, genre, or year within 2 seconds.

Movie Details	Movie details page displays poster, trailer, summary, genre, cast, year, and rating.
Bookmarked Movies	Users can add or remove a movie from bookmarks with a single tap; changes reflect instantly in the list.
Watch Trailers	Trailers play in-app without redirecting to external apps; playback starts within 3 seconds.
Profile	Users can update profile photo and username; changes are saved and shown across the app immediately.
Customization	

3.2.3 Non-functional Requirements

Table 5. Non-functional Requirements for CineMatch

Category	Requirement	Metric/Target
Performance	Load movie recommendations	Less than 2 seconds to see the output
	Search response time	Less than 2 seconds to see the output
Security	Encrypt user data	HTTPS + Firestore encryption
	Secure OAuth token storage	Firebase Secure Storage
	Data compliance	GDPR-compliant deletion

		requests
Reliability	System uptime for core features	99.9% availability
Scalability	Database capacity	50,000+ monthly active users
	Offline functionality	Android calendar data caching
Usability	Movie search/bookmark efficiency	Less than to 3 taps for core actions (search/bookmark movie)
	Profile completion efficiency	Less than to 3 taps to update photo or username

IV. Technical Specifications

4.1 Architecture

4.1.1 Overview Diagram

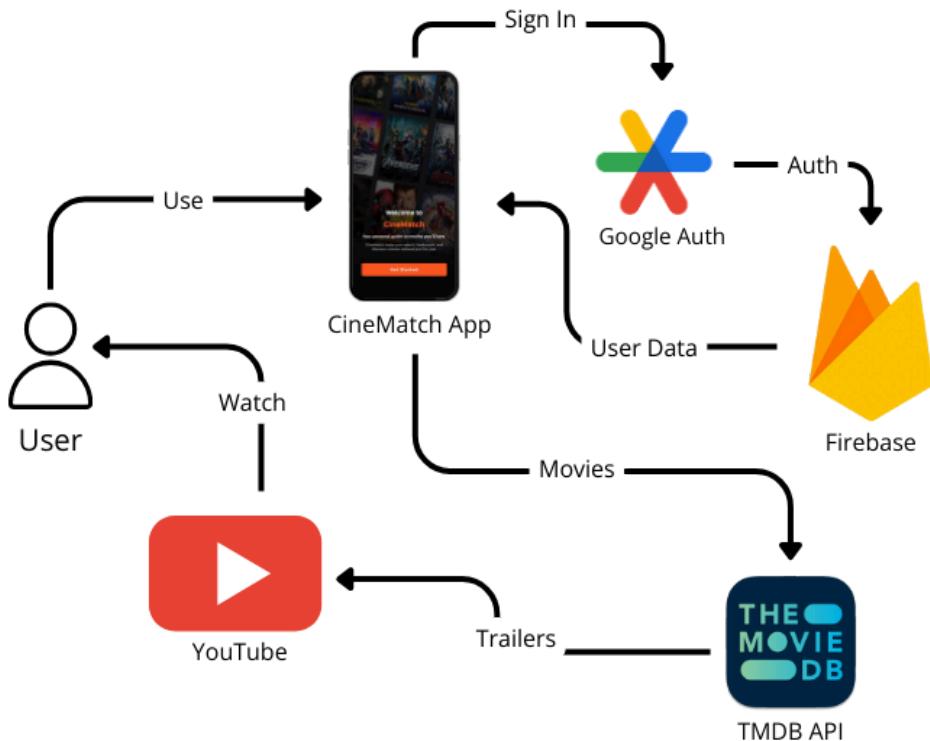


Figure 25. System Architecture of the CineMatch

CineMatch's architecture combines Google Authentication, Firebase, TMDB, and YouTube to create a seamless movie discovery experience. Users securely sign in with Google Auth, while their preferences and watchlists are stored and synced in Firebase. Movie details are fetched from TMDB, and trailers are played via YouTube integration. This setup ensures real-time data access, personalized recommendations, and smooth cross-device experiences by handling authentication, content delivery, and user data storage in a fully integrated system.

4.1.2 Design Patterns

Primary Pattern: MVVM (Model-View-ViewModel)

Role in CineMatch:

- Model - The model layer handles all data operations including TMDB API integration, local storage through Room database, and data structures for movies, cast, and videos.
- View - The view layer manages the user interface components including activities, adapters, and XML layouts that display movie information and handle user interactions.
- ViewModel - The ViewModel layer acts as a mediator between Model and View, processing business logic, managing data flow, and providing clean data to the UI while surviving configuration changes.

Why MVC?

- MVVM was used in CineMatch for its clear separation of UI and logic, making the app easier to maintain and test. ViewModels with LiveData support reactive updates and handle configuration changes, while Android Architecture Components enable features like caching and real-time updates.

4.2 Platform-Specific Considerations

4.2.1 iOS and Android Guidelines

Table 6. iOS and Android Guidelines for CineMatch

Platform	UX/UI Standards	TaskFlow Implementation
Android	Material Design 3	<ul style="list-style-type: none"> • Floating Action Button (FAB) for task creation • Bottom navigation bar
iOS	Human Interface Guidelines	<ul style="list-style-type: none"> • (Future: Flutter implementation) • Cupertino-style pickers • Navigation bars with back buttons

4.2.2 Hardware & OS Compatibility

Table 7. Hardware & OS Compatibility for CineMatch

Platform	UX/UI Standards	TaskFlow Implementation
Android	8.0 (Oreo)	<ul style="list-style-type: none"> • Requires Google Play Services • Not optimized for foldables
Web	Chrome 100+, Firefox 100+, Safari 14+	<ul style="list-style-type: none"> • No offline support for Drive files
iOS	Not supported	(Future: iOS 15+ with Flutter or React Native)

4.3 Data Management

4.3.1 Data Flow Diagrams

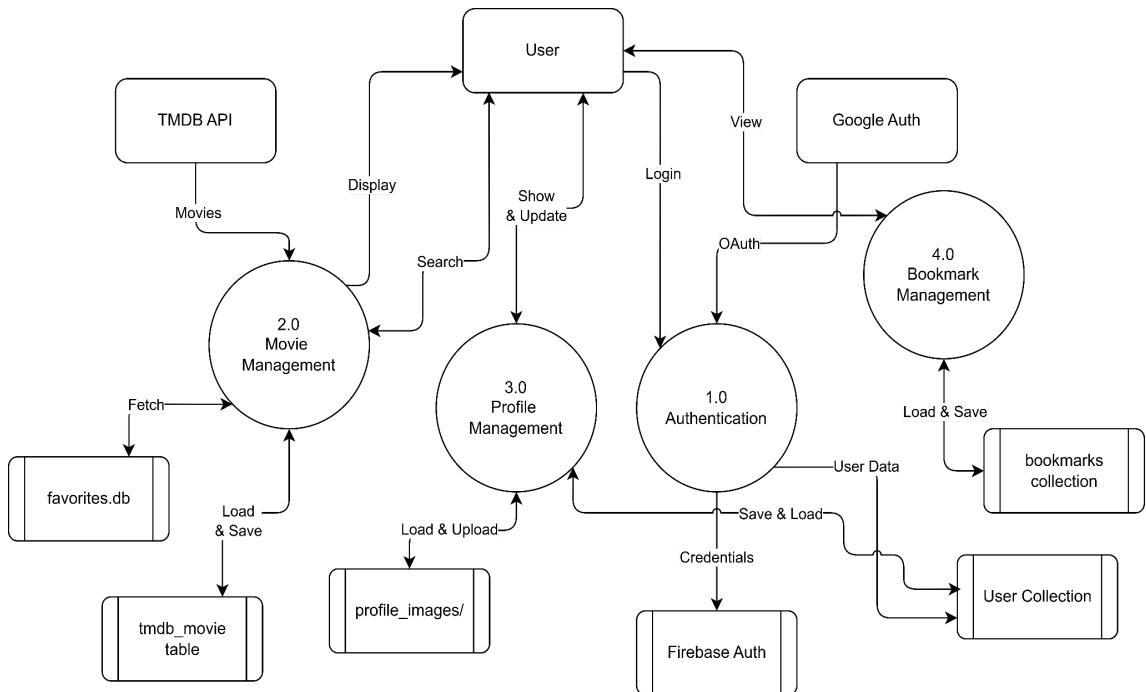


Figure 26: Data Flow Diagram for CineMatch

Figure 25 shows how data moves within CineMatch. Users log in through Google Auth, which connects to Firebase Auth for secure authentication. Once logged in, users can access Movie Management, where data from the TMDB API is used to display movie suggestions. Users can bookmark movies through the Bookmark Management module, which saves data in the bookmarks collection linked to their profile. The Profile Management module allows users to update their info and images, keeping the experience personal and user-focused.

4.3.2 Database Schemas

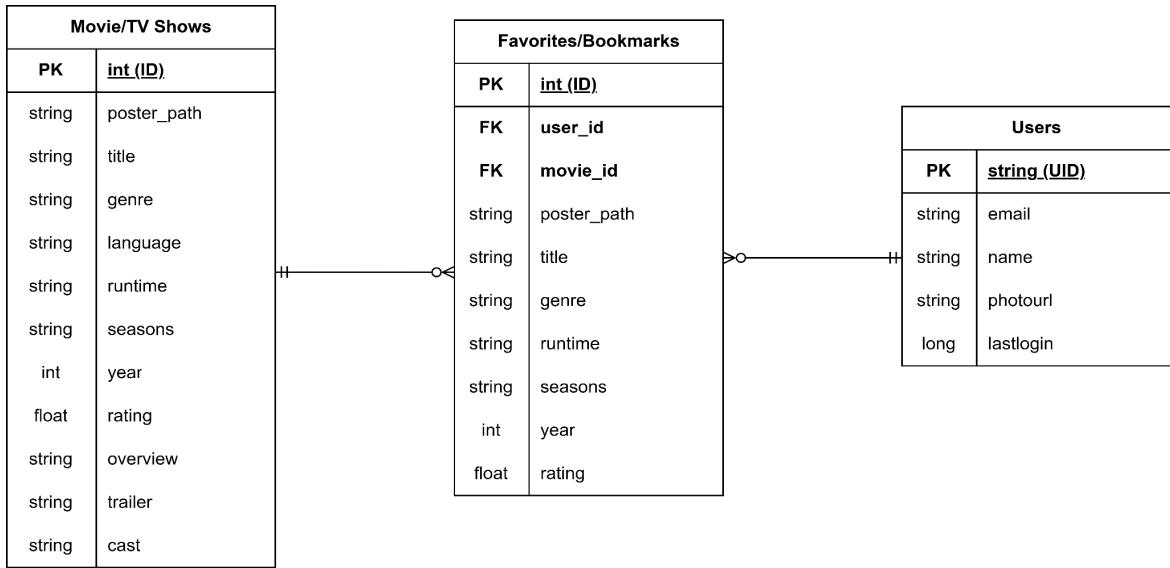


Figure 27: Database Schema for CineMatch

Figure 26 shows the database design of CineMatch is made up of three main tables: Movie/TV Shows, Favorites/Bookmarks, and Users. The Movie/TV Shows table contains detailed information about each movie or show, such as the title, genre, rating, year, and cast. This data is used to display movie details and power the search and recommendation features of the app. The Users table stores personal details like the user's unique ID (UID), name, email, profile photo, and last login time. These records help manage user sessions and personalize the experience.

The Favorites/Bookmarks table acts as a bridge between the user and the movies they bookmark. It uses foreign keys to connect each saved movie to the user who bookmarked it. This table also stores a copy of some movie data for quick access, such as the poster, title, and rating. This structure supports both local and server-side storage, ensuring that users can view their saved content smoothly across sessions. The overall

schema is designed to keep data organized, fast, and easy to manage for all CineMatch features.

4.3.3 API Endpoints

The following are CineMatch's detailed endpoints for TMDB API integration. They include HTTP methods, parameters, configurations, and expected responses.

MOVIE API

Module Description

Core movie data endpoints for fetching movies by categories, movie details, videos, and credits.

Endpoints:

GET /discover/movie

GET /movie/{movie_id}

GET /movie/{movie_id}/videos

GET /movie/{movie_id}/credits

GET /genre/movie/list

Discover Movies (Categories/Genre-based)

Version: 1.0

Description: Discover Movies Endpoint.

Table 8. Retrieve Popular Movies Endpoint for CineMatch

Endpoint	/discover/movie
Method	GET

Configuration	Requires TMDB API key
Parameters	<ul style="list-style-type: none"> - api_key (required): TMDB API key - with_genres (optional): Genre ID(s) - sort_by (optional): "popularity.desc", "vote_average.desc" - page (optional): Page number - language (optional): Response language
Response	Returns a list of movies based on specified genre and sorting criteria for category-based browsing.

FIREBASE AUTHENTICATION API

Module Description

User authentication endpoints using Firebase and Google Sign-In.

Endpoints:

POST

<https://identitytoolkit.googleapis.com/v1/accounts:signInWithIdp>

POST <https://identitytoolkit.googleapis.com/v1/accounts:signUp>

POST <https://securetoken.googleapis.com/v1/token>

Google Sign-In Authentication

Version: 1.0

Description: This endpoint enables user authentication using a Google account via Firebase Authentication.

Table 9. Google Sign-In Endpoint

Endpoint	/accounts:signInWithIdp
Method	POST
Configuration	Requires Firebase configuration

Parameters	- requestUri (required): OAuth redirect URI - postBody (required): ID token - returnSecureToken: boolean
Response	Returns Firebase user credentials including ID token and refresh token.

CLOUD FIRESTORE API

Module Description

Database endpoints for user data and preferences storage.

Endpoints:

- /users/{userId}
- /users/{userId}/favorites
- /users/{userId}/watchlist
- /users/{userId}/preferences

Save Movie to Favorites

Version: 1.0

Description: This endpoint allows an authenticated user to save a specific movie to their list of favorites using the TMDB movie ID.

Table 10. Save to Favorites Endpoint

Endpoint	/users/{userId}/favorites/{movieId}
Method	POST
Configuration	Requires Firebase Authentication
Parameters	- movieId (required): TMDB movie ID - title: string - genre: string - runtime: string

	<ul style="list-style-type: none"> - seasons: string - posterPath: string - rating: number
Response	Returns confirmation of the favorite movie being saved to the user's collection.

YOUTUBE PLAYER API

Module Description

Video playback endpoints for movie and TV show trailers.

Endpoints:

GET <https://www.googleapis.com/youtube/v3/videos>

Load Video Player

Version: 1.0

Description: This endpoint retrieves video metadata and configuration needed to load and play a YouTube trailer using the YouTube Data API.

Endpoint	/youtube/v3/videos
Method	GET
Configuration	Requires YouTube API key
Parameters	<ul style="list-style-type: none"> - id (required): YouTube video ID - part: string (typically "snippet") - key: API key
Response	Returns video player configuration and metadata for the requested trailer.

Table 12. YouTube Player Endpoint4.4 Security & Privacy

4.4 Security & Privacy

4.4.1 Authentication & Authorization

Table 13. Authentication & Authorization for CineMatch

Standard	Implementation
OAuth 2.0	Google Sign-In via Firebase Authentication. Only Google accounts are supported for login.
Email/Password	Firebase-generated JWTs (valid for 24h) stored in secure HTTP-only cookies.
Session Management	Firebase Auth state persistence and automatic token refresh.

4.4.2 Data Encryption

Table 14. Data Encryption for CineMatch

Type	Method
In-Transit	HTTPS (basic SSL) for web traffic + Basic API authentication.
At-Rest	Basic password hashing for user accounts + Database connection security

4.4.3 Compliance Requirements

Table 15. Compliance Requirements for CineMatch

Framework	Relevance to CineMatch	Implementation
GDPR (EU General Data Protection Regulation)	Applies to all EU user data. Governs data collection, consent, and deletion.	- General privacy and security measures described in UI. - Minimal data collection. - No explicit user

		consent management, data export, or deletion tools as required by GDPR.
CCPA/CPRA (California Consumer Privacy Act/California Privacy Rights Act)	Applies to California users. Requires opt-out of data sale and user data rights.	- No data is sold to third parties. - No user data export/deletion tools
PIPEDA (Canada's Privacy Law)	Applies to Canadian users' personal data.	- General privacy and security measures.

4.4.4 User Data & Privacy Policies

Table 16. User Data & Privacy Policies for CineMatch

Policy Aspect	Implementation	Technical Details
Collected Data	<ul style="list-style-type: none"> User email (via Google/Firebase Authentication) Profile photo Movie preferences, watchlists, and ratings 	No sensitive PII collected beyond what is necessary for account creation and personalized recommendations.
Data Storage	<ul style="list-style-type: none"> All user data is securely stored in Google Firebase Cloud Firestore and Firebase Authentication. 	Data is encrypted at rest and in transit using industry-standard protocols (TLS/SSL).
Data Management	<ul style="list-style-type: none"> Users can update their profile information (username, profile photo) at any time. Profile updates are immediately reflected in 	Account deletion triggers removal of all user-associated data from Firebase.

	the database.	
Data Sharing	• CineMatch does not sell or share user data with third parties for marketing.	Data sharing is governed by strict contractual agreements.
User Rights	• Users can access, update, or delete their data upon request	Support contact: support@cinematch.com
Security Measures	• Industry-standard security practices are implemented to protect user data.	Firebase Authentication for secure login.

4.5 Third-Party Integration

4.5.1 SDKs & Libraries

Table 17. SDKs & Libraries for CineMatch

Component	Purpose	Source
Firebase SDK	Authentication, user management, cloud database, and secure data storage	Google Firebase
Android Jetpack Libraries	Modern Android UI components, navigation, lifecycle, and local database	Android Developers
Retrofit2	Networking and REST API calls	Square/Retrofit
Glide	Efficient image loading and caching (profile photos)	Bumptech/Glide

4.5.2 Integration Details

- **Firebase Services**

Auth Flow - User signs in via:

Firebase → Grants scopes → Tokens passed to backend

Data Sync:

Firestore:

- Serves as the single source of truth for user-generated data (profiles, preferences, watchlists, ratings).
- All CRUD operations on user data are performed via Firestore's secure API.

Firebase Storage :

- Used for storing user-uploaded profile images and other media.
- Only file URLs are referenced in Firestore; actual files are securely stored and accessed via Firebase Storage.

• **Android Jetpack Libraries**

Data Sync:

UI & Navigation:

- AppCompat, ConstraintLayout, and Navigation components

V. UI/UX Design Specifications

5.1 Wireframes & Mockups

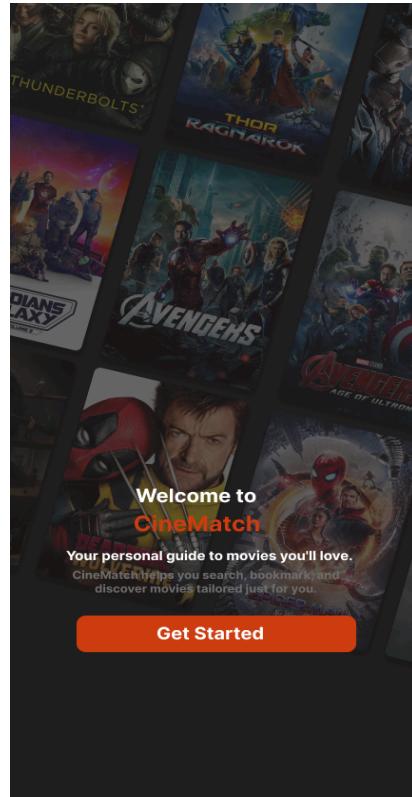


Figure 27: Database Schema of CineMatch

The CineMatch splash screen features a sleek, dark theme with Marvel movie posters in the background, creating an engaging first impression. It includes a welcoming message, a tagline—“Your guide to movies you’ll love”—and a brief description of the app’s purpose: helping users search, bookmark, and discover personalized movie recommendations. A bold orange “Get Started” button invites users to begin their journey. The design is clean and clearly communicates CineMatch’s focus on personalized movie discovery.

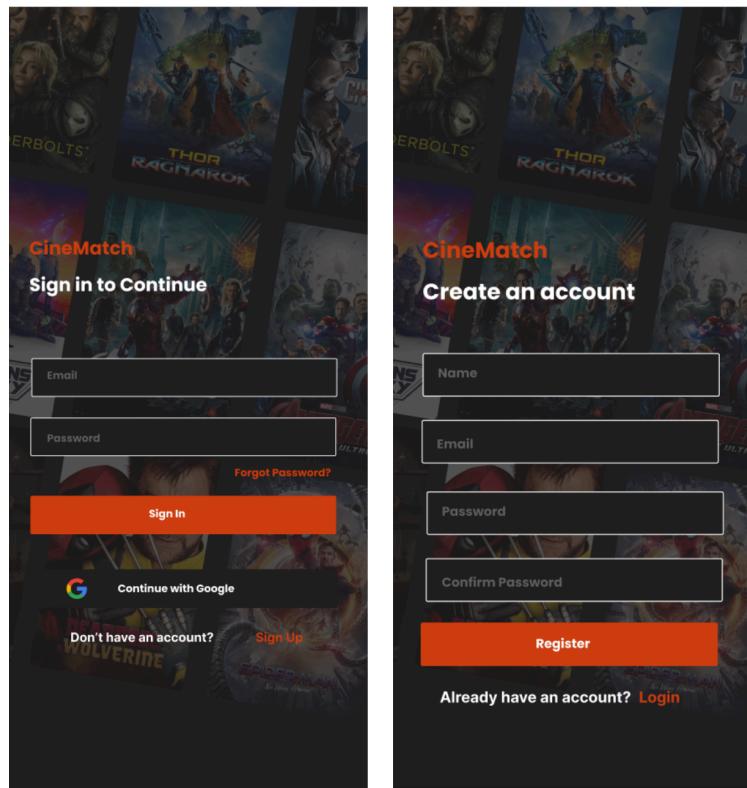


Figure 28: Database Schema of CineMatch

The CineMatch app features two essential authentication screens with a consistent dark theme and Marvel poster backdrop. The sign-in screen includes email and password fields, a “Forgot Password?” link, an orange “Sign In” button, and a “Continue with Google” option, along with a prompt to sign up. The registration screen provides fields for name, email, password, and confirmation, topped with a “Register” button. Both screens reflect CineMatch’s branding and offer smooth navigation between login and sign-up, ensuring a seamless and visually cohesive onboarding experience for movie fan

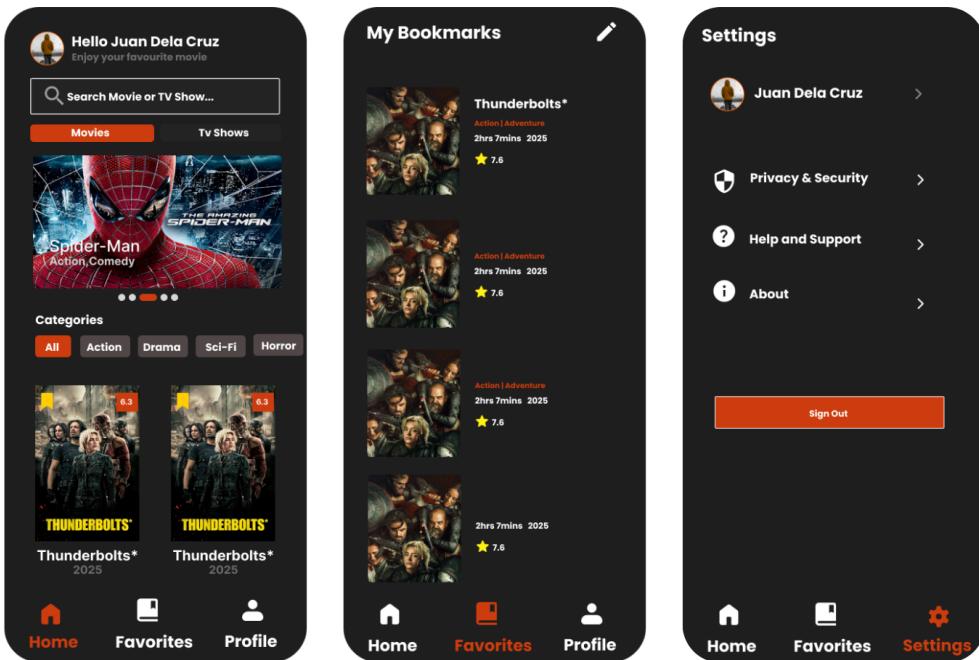


Figure 29: Database Schema of CineMatch

The CineMatch app's core functionality is showcased across three main screens, all unified by a sleek dark theme and intuitive layout. The Home screen greets users with a personalized message, a search bar, a toggle between Movies and TV Shows, a featured content carousel, genre filters, and movie thumbnails with ratings—supported by a bottom navigation bar linking to Home, Favorites, and Profile. The Bookmarks screen displays saved content with detailed movie info (e.g., Thunderbolts), along with an edit option and consistent navigation. The Settings screen offers quick access to Profile, Privacy & Security, Help and Support, About, and a Sign Out button. Together, these screens ensure smooth navigation, easy content discovery, and a cohesive user experience.

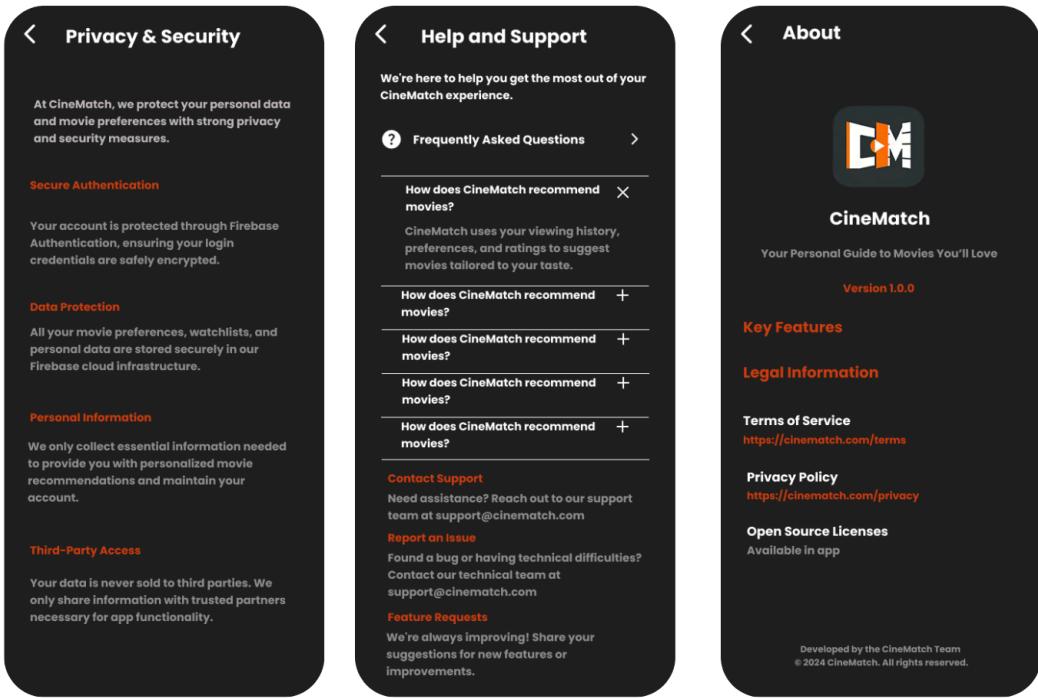


Figure 30: Database Schema of CineMatch

The CineMatch app includes three essential information screens, all styled with a dark theme for clarity and consistency. The Privacy & Security screen highlights the app's commitment to protecting user data through Firebase Authentication, secure cloud infrastructure, minimal data collection, and limited third-party access. The Help & Support screen offers a FAQ section, support contact via email, and options to report issues or request features—reflecting CineMatch's dedication to user feedback and improvement. The About screen showcases the CineMatch logo, tagline, app version, legal links, open source licenses, and team credits. Together, these screens deliver key information and support while maintaining a clean, professional design.

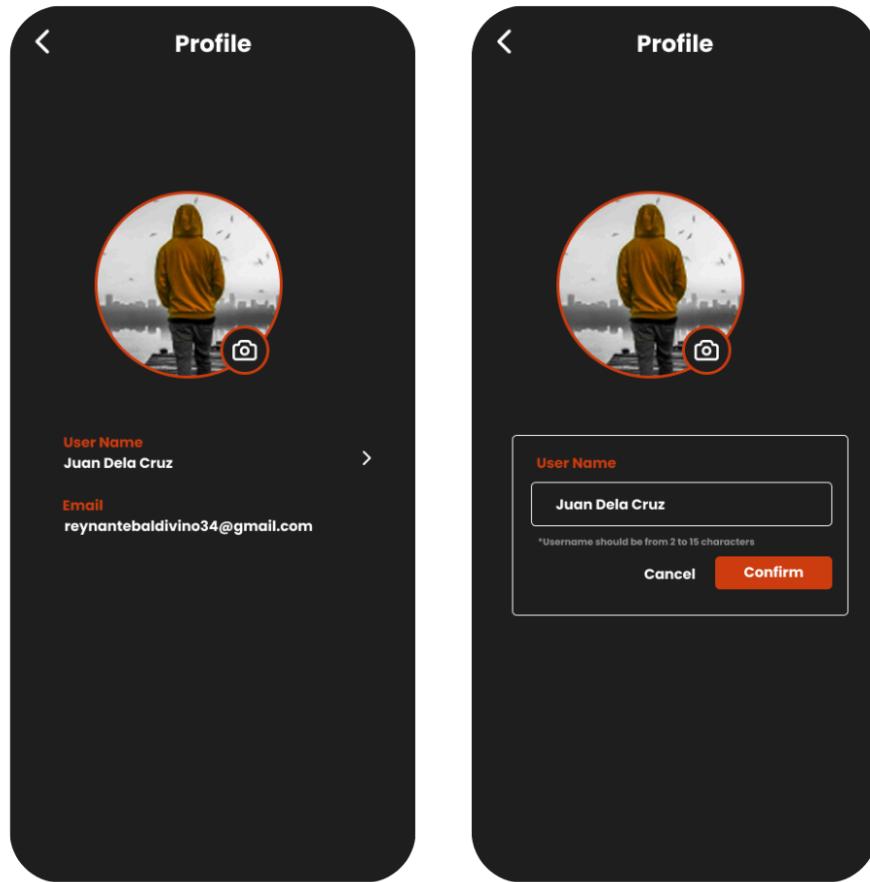


Figure 31: Profile of CineMatch

The CineMatch app's Profile management screens include both viewing and editing modes, designed with a clean, dark theme and user-friendly layout. The Profile View screen displays a circular profile photo (featuring a person in an orange hoodie), a camera icon for photo updates, and user details such as name ("username") and email. An arrow icon indicates edit capability. In the Profile Edit screen, users can update their username with guidance that it should be 3 to 15 characters long, and choose between "Cancel" or the orange "Confirm" button to save changes. Both screens maintain a minimalist, professional design with clear navigation and essential profile functions.

5.2 Navigation & Flow

5.2.1 User Flow Diagrams

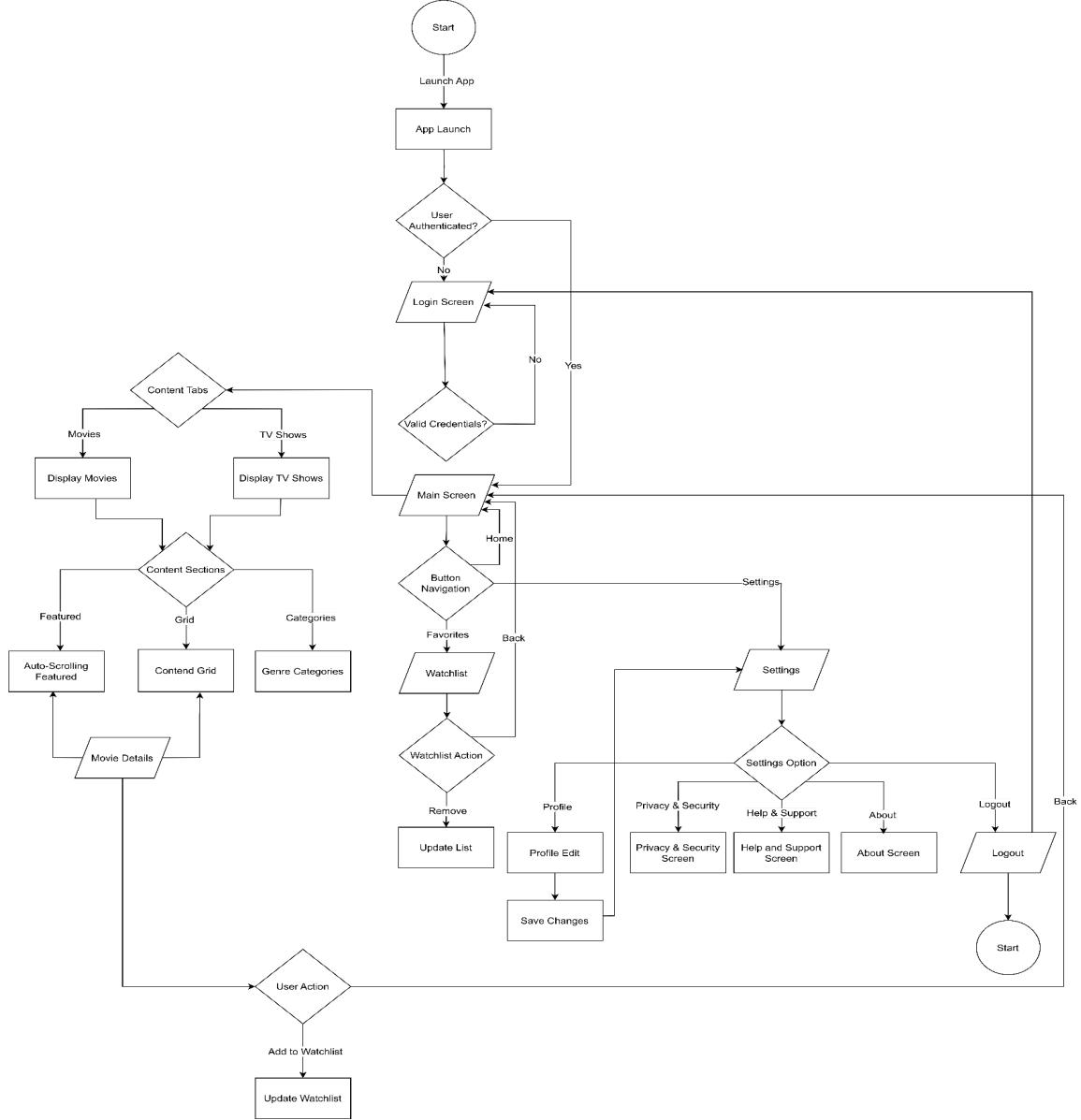


Figure 32: Database Schema of CineMatch

CineMatch offers a streamlined and intuitive user experience for discovering and managing entertainment content. Upon launching the app, users are welcomed with a clean interface that splits into two main sections: Movies and TV Shows. Each section includes Featured content, genre-based Categories, a visual Grid layout, and a Favorites list. Users

can easily navigate using consistent Back buttons and quickly access their Profile and Settings. The app supports personalized watchlists, browsing history, and smart recommendations, with separate tracking for movies and shows.

Bottom navigation provides tools for content management, user support, and privacy controls, including watchlist cleanup, secure login, and data preferences. Additional features include smart search, social sharing, offline access, and cross-device syncing. With its responsive design and focus on usability, CineMatch ensures a smooth and enjoyable entertainment experience while prioritizing user privacy and content accessibility.

5.2.2 Accessibility Guidelines

Core Accessibility Features

Table 18. Core Accessibility Features for CineMatch

Requirement	Implementation in TaskFlow	Standard
Screen Reader Support	<ul style="list-style-type: none"> All UI elements labeled for TalkBack Custom announcements for movie updates (e.g., "Added to favorites") 	WCAG 4.1.2, Android A11Y
Color Contrast	<ul style="list-style-type: none"> Minimum 4.5:1 text-to-background ratio Movie ratings use both stars and numbers Dark theme optimized for readability 	WCAG 1.4.3
Text Scaling	<ul style="list-style-type: none"> Supports 200% system font scaling Flexible layout for movie cards and descriptions 	WCAG 1.4.4

Touch Targets	<ul style="list-style-type: none"> Movie cards and buttons \geq 48x48dp 8dp spacing between interactive elements Clear spacing in navigation bar 	Material Design A11Y
---------------	--	----------------------

Navigation & Interaction

Table 19. Navigation & Interaction for CineMatch

Feature	Implementation	Tool
Keyboard Navigation	<ul style="list-style-type: none"> D-pad/Tab navigation for movie browsing Logical focus order in movie details 	Android Accessibility Scanner
Gesture Alternatives	<ul style="list-style-type: none"> All actions available via menu buttons Alternative navigation options for favorites 	TalkBack
Haptic Feedback	<ul style="list-style-type: none"> Confirmation vibrations for bookmarking Feedback for rating submission 	Android Vibration API

5.1.2.3 Platform-Specific Compliance

Table 20. Platform-Specific Compliance for C

Android Guideline	TaskFlow Implementation
Material Design Accessibility	<ul style="list-style-type: none"> Material Components with built-in accessibility High contrast theme options Consistent navigation patterns
Android Accessibility Checklist	<ul style="list-style-type: none"> Tested with TalkBack and Switch Access Content descriptions for movie posters Clear error states and feedback

VI. Deployment & Maintenance

6.1 Deployment Plan

6.1.1 Release Process

Table 21 . Release Process for CineMatch

Step	Action	Tools/Standard
Pre- release	Final code optimization and feature testing (login, search, bookmarking) Prepare documentation for setup and demonstration	Android Studio, Firebase Emulator Suite
Web Deployment	Build production-ready using github to host online.	Firebase hosting and github
Android Release	Build and sign APK for direct installation Include instructions for manual installation	Android Studio (v1.0), APK Signature Tool
Testing Phases	Developer testing across devices and browsers Peer testing for feedback and bug reports Stress testing for load performance	Multiple Android Devices, Web Browsers, Firebase Backend
Security and Access	Secure API keys with environment variables Provide demo login for easy access	Firebase Authentication
Final Submission	Deliver source code, test cases, database schema, and	Github and PDF Documentation

demo video

6.1.2 Rollout Strategy

Table 22. Rollout Strategy for CineMatch

Phase	Target Audience	Purpose	Duration
Initial Verification	Internal development team	Validate core functionality and crash reports	3 days
Limited User Evaluation	25% of potential users (classmates)	Collect usability feedback and measure performance	1 week
Web Deployment	All users (progressive rollout)	Gradually release web version and monitor via analytics	48 hours
Mobile Deployment	Selected user groups (batch release)	Distribute APK with setup instructions and observe stability	Phased rollout

VII. Appendices

7.1 Glossary

Table 23. Glossary for CineMatch

Term	Definition
API (Application Programming Interface)	A set of protocols and tools that allow different software systems to communicate with each other. CineMatch uses APIs to fetch movie data from TMDb and integrate YouTube for trailers.
Authentication	The process of verifying the identity of a user. Typically implemented through login credentials.
Android	A mobile operating system developed by Google, used as the primary platform for CineMatch.
Backend	The server-side part of the app that handles data processing, storage, and business logic (e.g., Firebase for CineMatch).
Bookmarking	A feature that lets users save movies to a personal list for later access.
CSRF (Cross-Site Request Forgery)	A security attack where unauthorized commands are sent from a trusted user. CineMatch implements protections against this.
Database	An organized collection of structured information or data, typically stored electronically. CineMatch uses a database to store user data and movie preferences.
Deployment	The act of making an application available for use. Can be done on web servers or app stores.
Firebase	A Google-backed platform providing backend services like authentication, databases, and analytics, used in CineMatch for user data management.
Frontend	The client-side part of the app that users interact with (e.g., UI screens and navigation).
MVVM	A design pattern separating UI logic from business

(Model-View-View Model)	logic, used in CineMatch's architecture.
TMDb (The Movie Database)	A third-party movie database API providing CineMatch with movie details, ratings, and metadata.
UI/UX (User Interface/User Experience)	Refers to the design and usability of the app's screens and interactions.
YouTube Integration	A feature allowing trailers to be played within the app using YouTube's embedded player.

7.2 References & Resources

Related Documentation

Table 24: Related Documentation for CineMatch

Title	Link	Relevance to TaskFlow
TMDB API Documentation	https://developers.themoviedb.org/3	Configuring database access controls
Android Room Documentation	https://developer.android.com/training/data-storage/room	Local database implementation
Firebase Authentication	https://firebase.google.com/docs/auth	User authentication and security
User	https://developer.android.com/guide/architecture/mvvm-pattern	MVVM architecture

authentication and security	droid.com/jetpack	implementation
Material Design Components	https://material.io/develop/android	UI component guidelines and implementation

External Links

Table 25. External Links for CineMatch

Resource	Link	Purpose
Material Design Guidelines	https://material.io/design	UI/UX standards for Android app
TMDB Guidelines	https://www.themoviedb.org/documentation/api/terms-of-use	API usage and compliance
Firebase Documentation	https://firebase.google.com/docs	User data management and backend
Android Developer Guides	https://developer.android.com/guide	Android development best practices
YouTube Player API	https://developers.google.com/youtube/android/player	Movie trailer integration
Google Play Console	https://play.google.com/console	App deployment and distribution
RxJava Documentation	https://github.com/ReactiveX/RxJava/wiki	Reactive programming implementation
Glide Documentation	https://bumptech.github.io/glide/	Image loading and caching