

# **Progetto AnimeSoul**

Programmazione Dispositivi Mobili  
Appello - 16/09/2024

## **Gruppo Kong**

GABRIELE STILLO - mat. 899544

ALESSIO ARMIJO RAMIREZ MICHELE - mat. 899685

JEDURSAN SIVAKUMAR - mat. 899604

## INTRODUZIONE e FUNZIONALITA'

AnimeSoul è un'applicazione che permette agli utenti di poter scoprire, cercare e salvare dei nuovi anime oppure i propri preferiti.

Alla prima apertura dell'app saranno visualizzate varie opzioni di login(tramite Email e Password o tramite account Google) e la possibilità di registrarsi.

Una volta eseguito l'accesso, l'utente dovrà scegliere in che modo preferisce visualizzare i titoli degli anime (lingua di Default o Giapponese).

Dopodiché l'utente verrà reindirizzato alla schermata Home.

Nella sezione **Home**, l'utente, navigando nella pagina ha la possibilità di visionare una vasta selezione di anime suddivisi per generi, scegliere se aggiungerli ai **preferiti**, cliccando sul cuoricino oppure visualizzarne le informazioni **dettagliate**, cliccando su di essi.

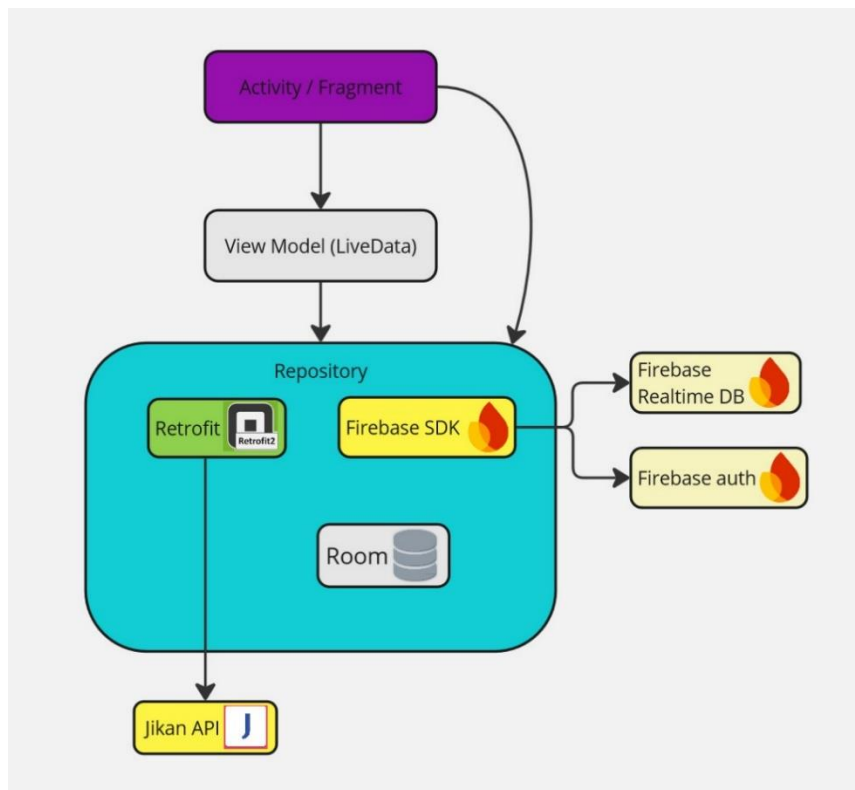
Per ogni anime è visualizzabile l'immagine/il trailer, il titolo, informazioni riguardanti l'anno di uscita, una breve trama, gli studi che lo hanno prodotto, i produttori stessi e i generi.

Per navigare nell'applicazione usiamo il Navigation Drawer situato in alto a sinistra della schermata, con il quale possibile accedere alle varie voci del menu:

- **HOME**
- **FAVORITE**
- **SEARCH**
- **TOP ANIME**
- **SETTINGS**

Che più avanti vedremo nel dettaglio.

## ARCHITETTURA DELL' APPLICAZIONE



L'applicazione segue principalmente l'architettura **MVVM (Model-View-ViewModel)**:

### Classi adapter

Le seguenti classi adapter si occupano di visualizzare gli item nelle recycler view.

- AnimeAdapter
- AnimeDetailAdapter
- GenreAnimeAdapter

GenreAnimeAdapter visualizza anche l'item\_loading che contiene una progress bar.

### Database

Queste classi si occupano di gestire la persistenza dei dati locali.

- AnimeDao: è un interfaccia che permette di interagire e modificare dati del database locale
- AnimeDatabase: inizializza il Database locale

## Classi model

Le seguenti classi model si occupano del mantenimento e della gestione dei vari dati.

- **Anime:** contiene i dati principali relativi agli anime presi da Jikan API.
- **Genres:** utile alla classe Anime per la gestione dei dati
- **GenreAnimeResponse:** contiene i dati ricevuti come risposta alla chiamata agli endpoint dei vari generi dalla Jikan API.
- **AnimeResponse:** contiene i dati ricevuti come risposta alla chiamata agli endpoint topAnime dalla Jikan API.
- **Images:** utile alla classe Anime per la gestione dei dati
- **Jpg:** utile alla classe Anime per la gestione dei dati
- **Producers:** utile alla classe Anime per la gestione dei dati
- **Trailer:** utile alla classe Anime per la gestione dei dati
- **Aired:** utile alla classe Anime per la gestione dei dati

## Repositories

Repositories usate per ottenere i vari dati dal TMDb API.

- **AnimeRepository:** ottiene gli anime per i vari fragment (getTopAnime, getAnimes, getAllFavoritesAnimes).

## Services

- **JikanApi:** interfaccia per effettuare chiamate API al servizio Jikan, che fornisce dati sugli anime come dettagli, generi, e liste degli anime popolari o consigliati.
- **RetrofitInstance:** classe che gestisce l'istanza di Retrofit, utilizzata per creare e configurare le chiamate HTTP all'API Jikan, impostando la base URL e gestendo le connessioni di rete.

## Utils

- **Constants:** contiene le costanti utilizzate in più classi.
- **Converters:** classe realizzata per la deserializzazione degli oggetti con Gson per le varie informazioni di un anime fornite dalla Jikan API.

## View model

View models che gestiscono i vari dati relativi agli Anime

- **FavoriteViewModel:** gestisce il caricamento e la gestione degli anime preferiti dell'utente da Firebase..
- **HomeViewModel:** coordina il caricamento degli anime per diverse categorie/ generi nella schermata Home.
- **LogoutViewModel:** gestisce le preferenze dell'utente, come la selezione della lingua e l'operazione di logout.
- **RecommendedViewModel:** recupera e mostra gli anime consigliati o correlati al genere dell'anime attualmente visualizzato.
- **SearchViewModel:** effettua ricerche di anime in base alle query inserite dall'utente e gestisce i risultati di ricerca.
- **TopAnimeViewModel:** si occupa del caricamento e della gestione della lista degli anime più popolari o meglio valutati.

## UI (suddivisa tra activity e fragment)

- **AnimeDetailActivity:** Mostra i dettagli completi di un singolo anime, inclusi trailer, sinossi e altri dettagli come i produttori e i generi, con una tab di consigli.
- **FavoritesFragment:** Gestisce e visualizza la lista degli anime aggiunti ai preferiti dall'utente.
- **HomeFragment:** Funge da schermata principale, mostrando varie sezioni di anime per genere o categorie, con una navigazione semplice.
- **LanguageSelectionActivity:** Permette all'utente di selezionare la lingua preferita per visualizzare i titoli degli anime (es. giapponese o lingua predefinita).
- **LoginActivity:** Gestisce l'autenticazione dell'utente, permettendo il login tramite email/password o Google, e offre il recupero della password.
- **LogoutFragment:** Fornisce opzioni per il logout e la modifica delle preferenze, come la lingua per i titoli degli anime.
- **MainActivity:** Activity principale che contiene il menu di navigazione e funge da contenitore per i vari fragment dell'app, come la home e i preferiti.
- **RegisterActivity:** Permette all'utente di creare un nuovo account tramite email e password.

- **SearchFragment:** Gestisce la funzionalità di ricerca degli anime, mostrando risultati pertinenti in base ai criteri di ricerca dell'utente.
- **TermsConditionsActivity:** Mostra i termini e le condizioni dell'applicazione all'interno di un'interfaccia testuale.
- **TopAnimeFragment:** Visualizza una lista degli anime più popolari o meglio votati, ordinati per categoria.

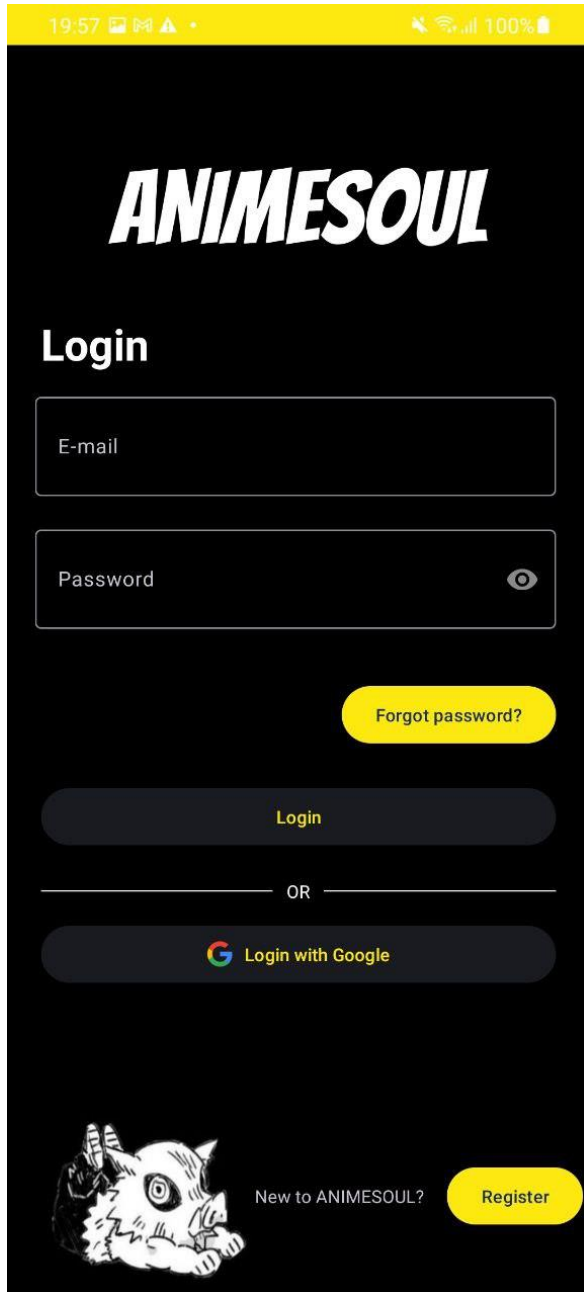
### Altro

Sono stati utilizzati:

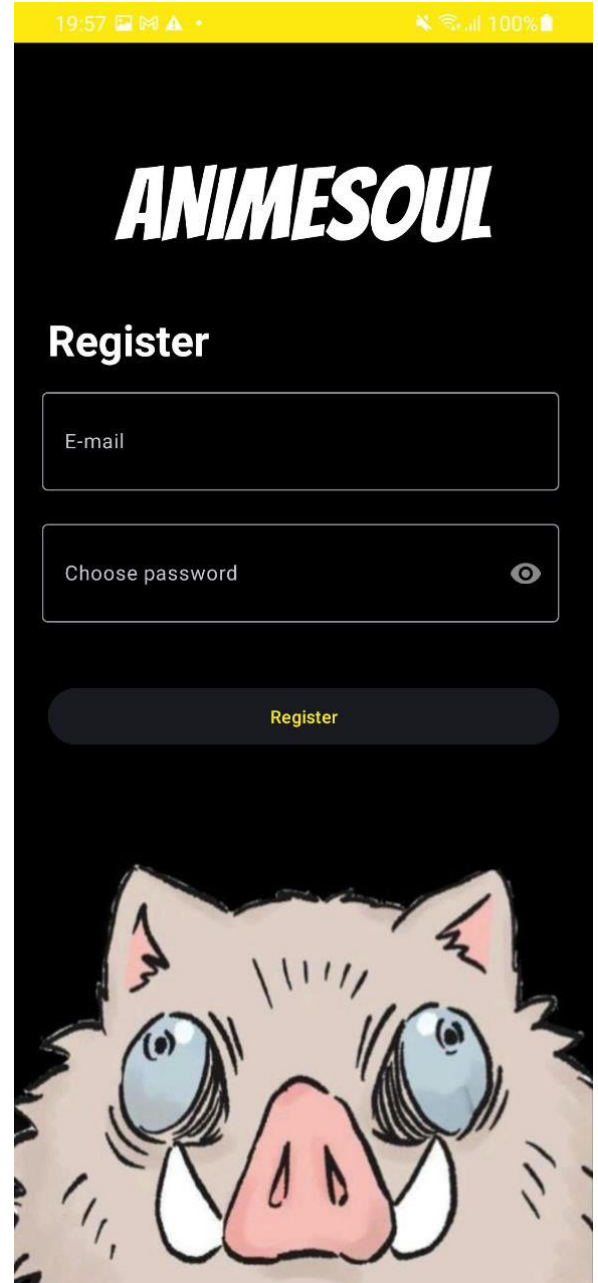
- View Binding
- Material Component 3
- Navigation Component 3
- Firebase Auth
- Realtime Firebase
- Google Services
- Retrofit
- Converter Gson
- Glide

## SCHERMATE DELL'APPLICAZIONE

### LoginActivity & RegisterActivity

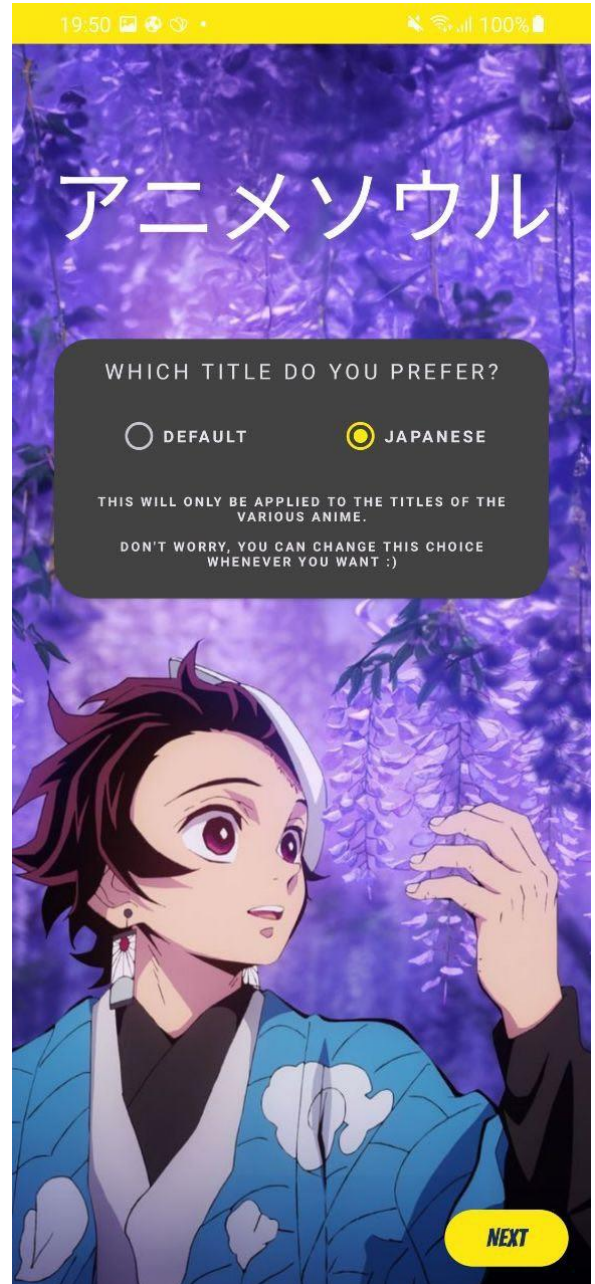
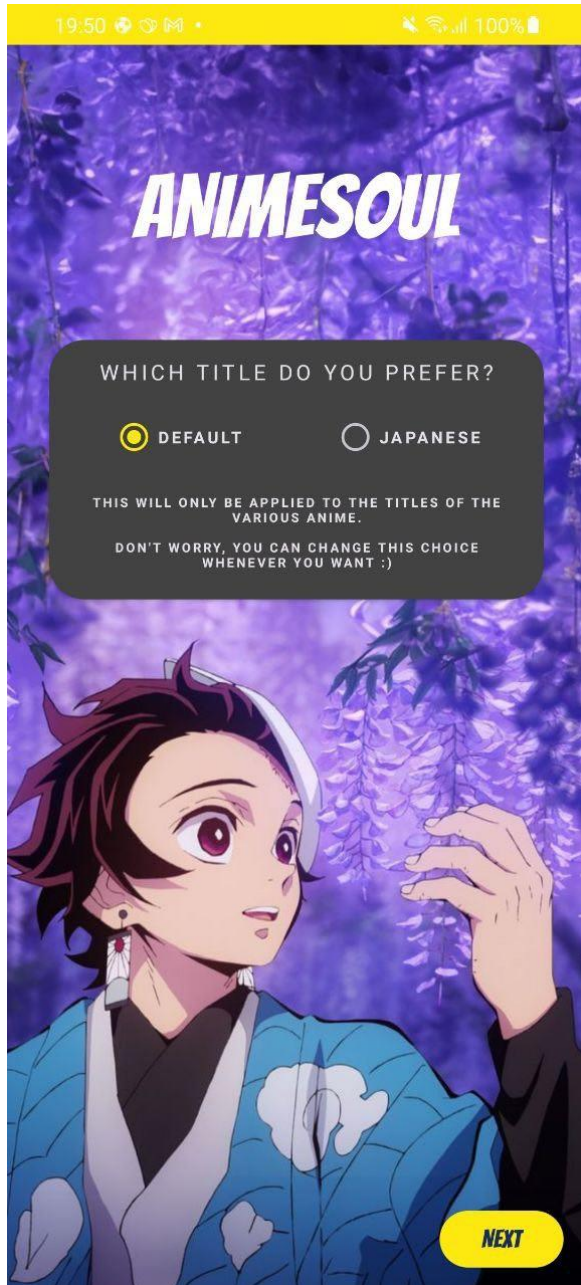


The screenshot shows the login screen of the ANIMESOUL application. At the top, the status bar displays the time 19:57, signal strength, and 100% battery. The app's logo, "ANIMESOUL", is prominently displayed in a stylized white font. Below the logo, the word "Login" is centered. There are two input fields: "E-mail" and "Password", both with white outlines and a white eye icon for password visibility. A yellow button labeled "Forgot password?" is positioned below the password field. A dark grey button labeled "Login" is centered below the input fields. Below the "Login" button, the word "OR" is centered, followed by a dark grey button labeled "Login with Google" with the Google logo. At the bottom left, there is a small cartoon illustration of a character's head. To its right, the text "New to ANIMESOUL?" is displayed, followed by a yellow button labeled "Register".



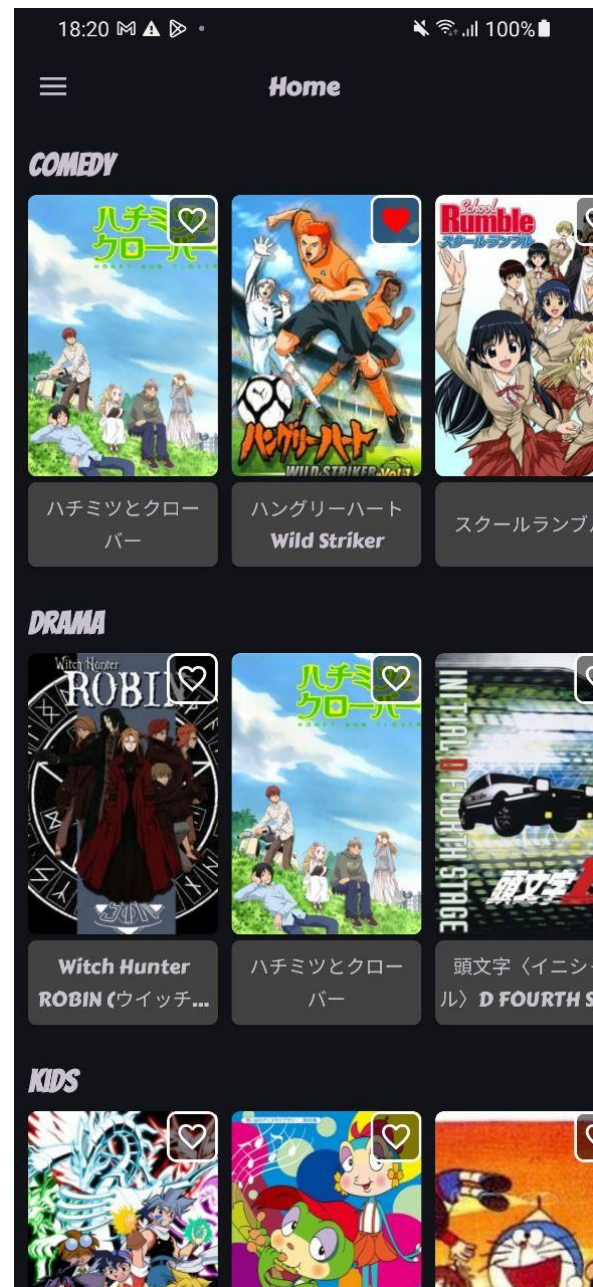
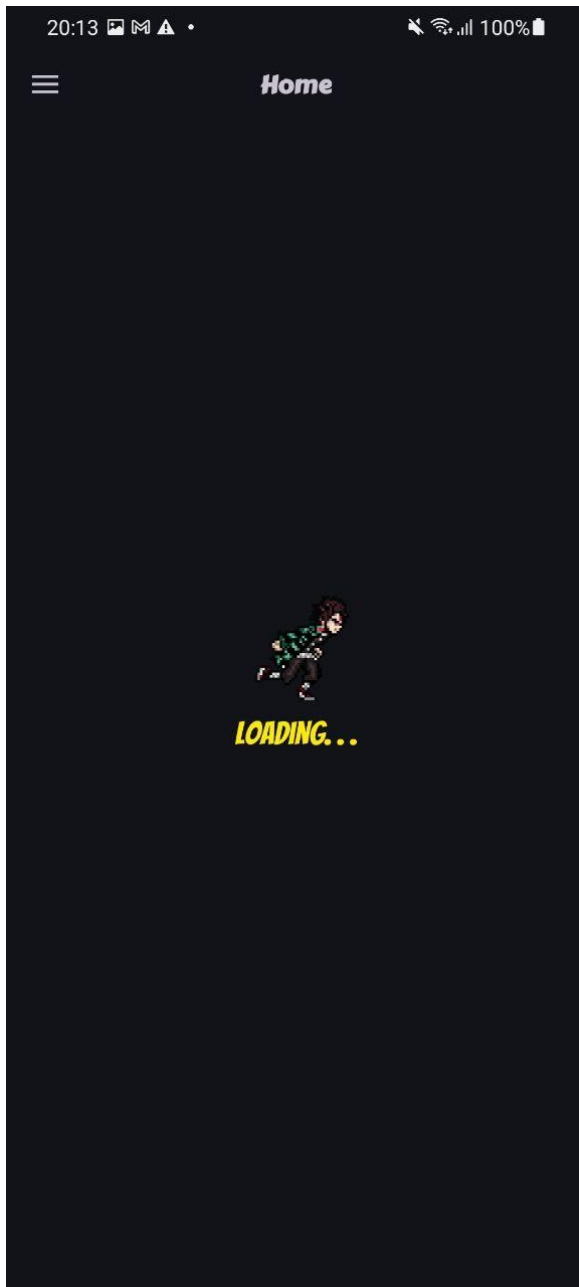
The screenshot shows the register screen of the ANIMESOUL application. At the top, the status bar displays the time 19:57, signal strength, and 100% battery. The app's logo, "ANIMESOUL", is prominently displayed in a stylized white font. Below the logo, the word "Register" is centered. There are two input fields: "E-mail" and "Choose password", both with white outlines and a white eye icon for password visibility. A dark grey button labeled "Register" is centered below the input fields. At the bottom, there is a large, detailed cartoon illustration of a character's face with large blue eyes and a wide, toothy grin.

## LanguageSelectionActivity

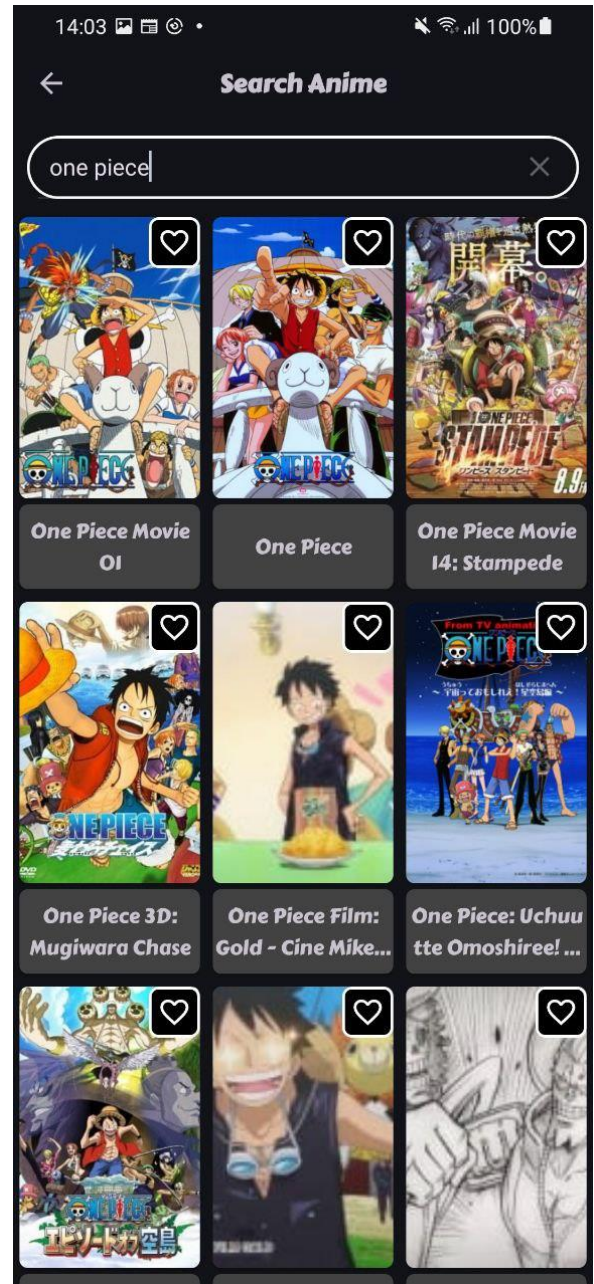
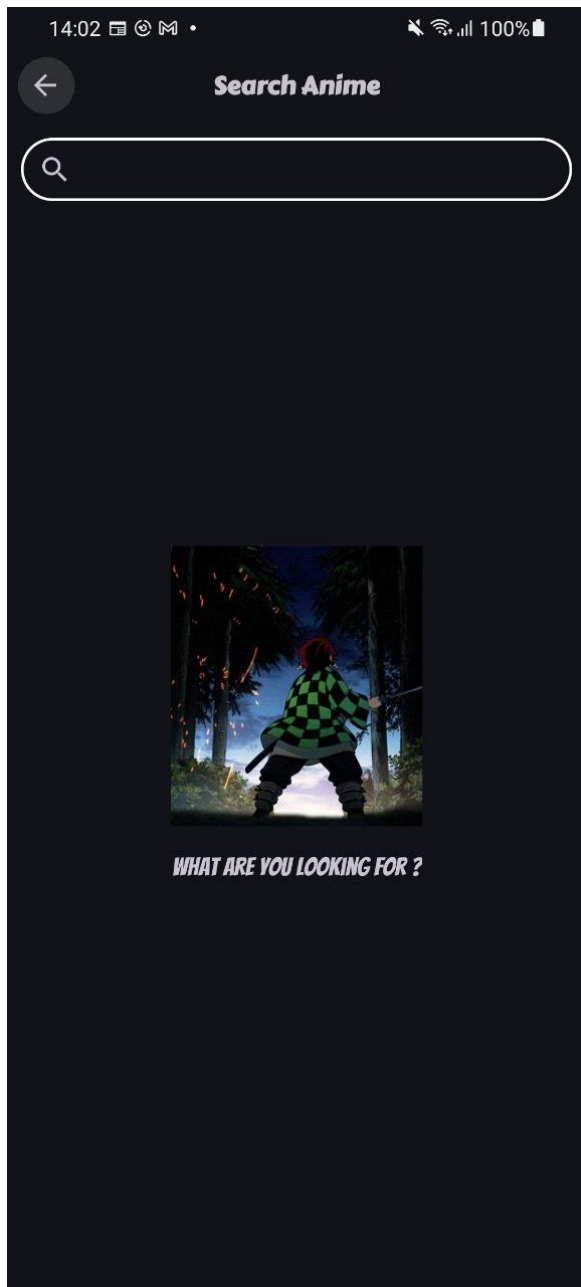




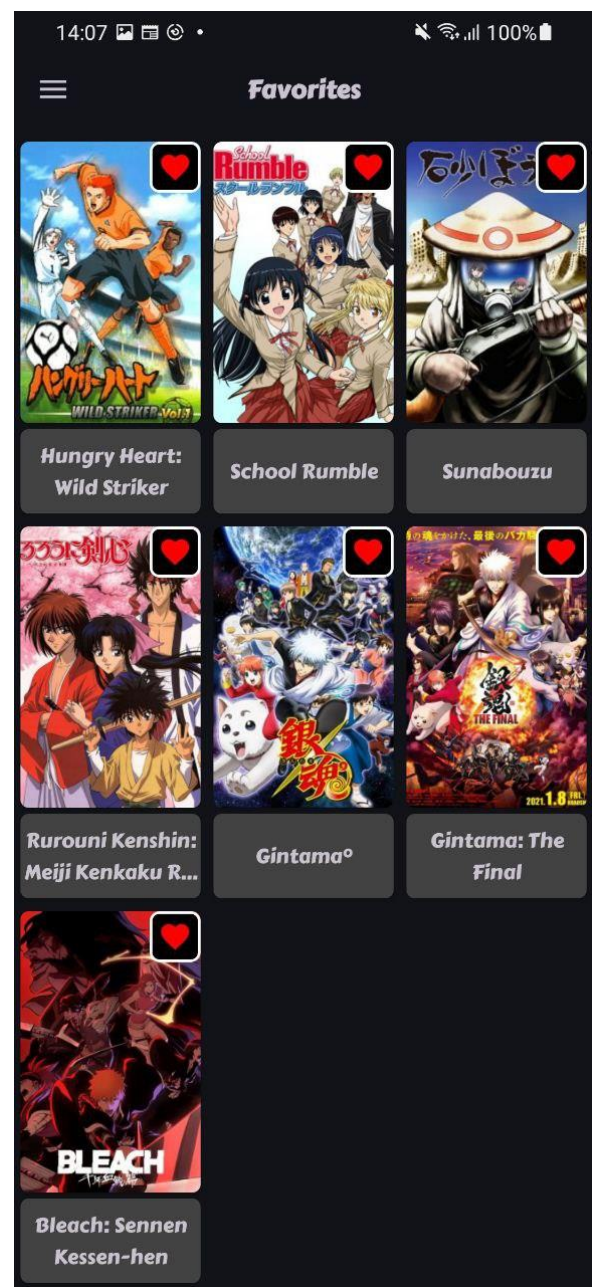
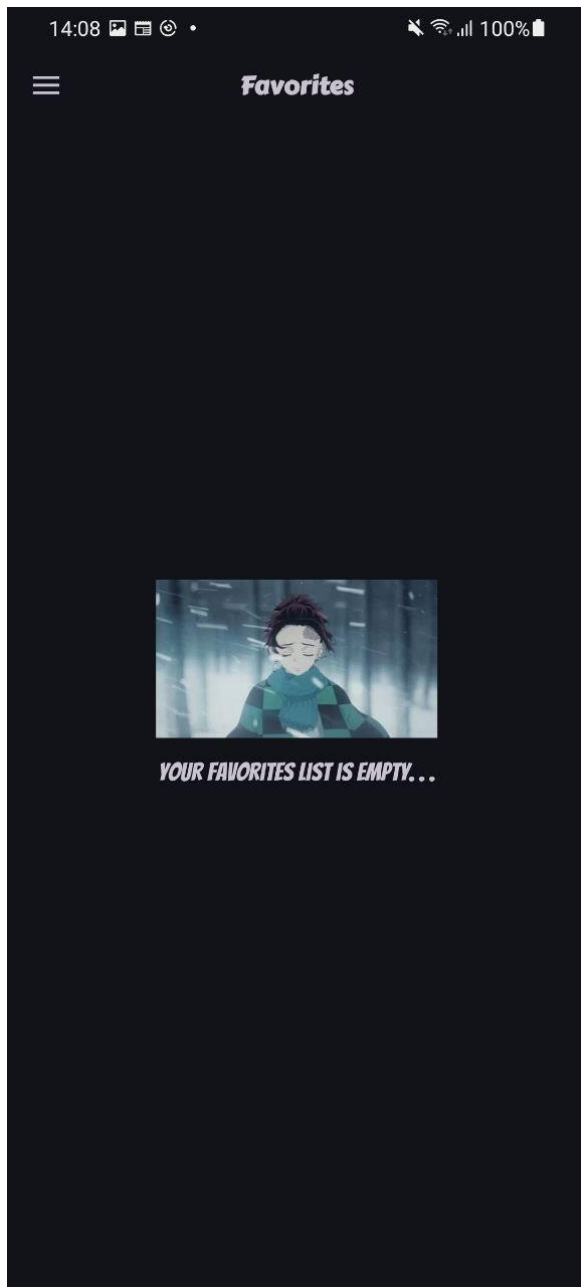
## HomeFragment



## SearchFragment

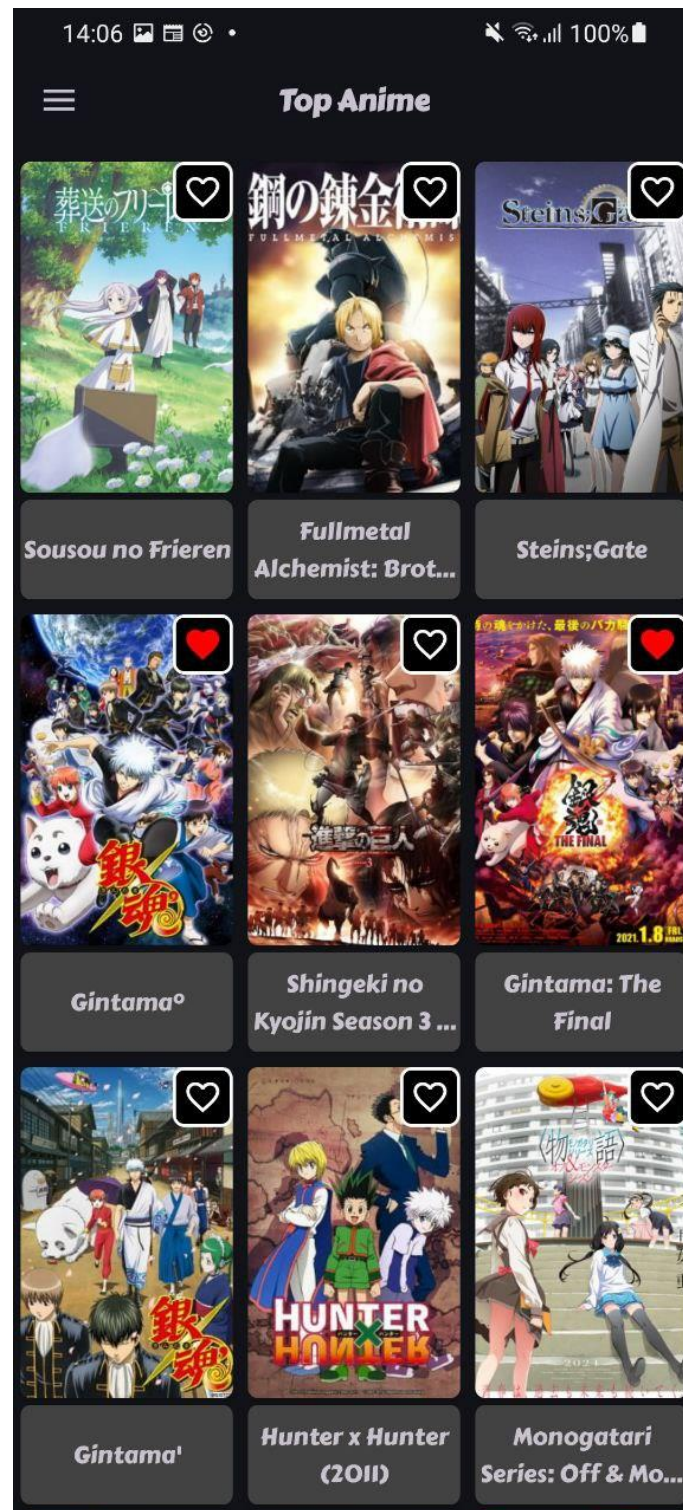


## FavoriteFragment

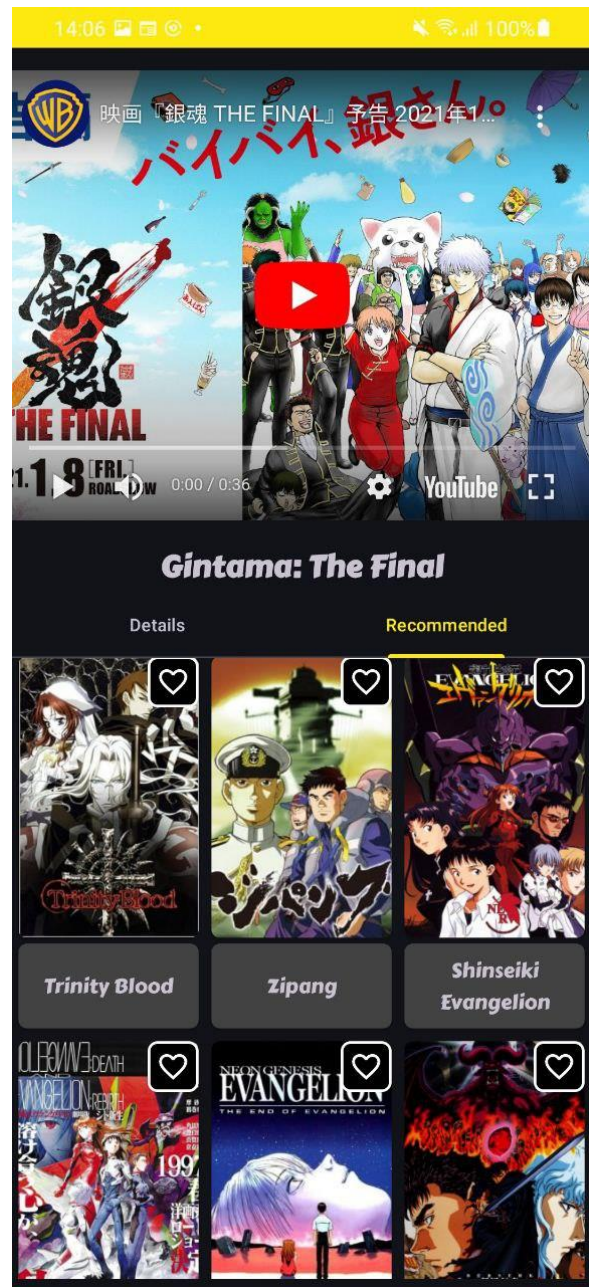
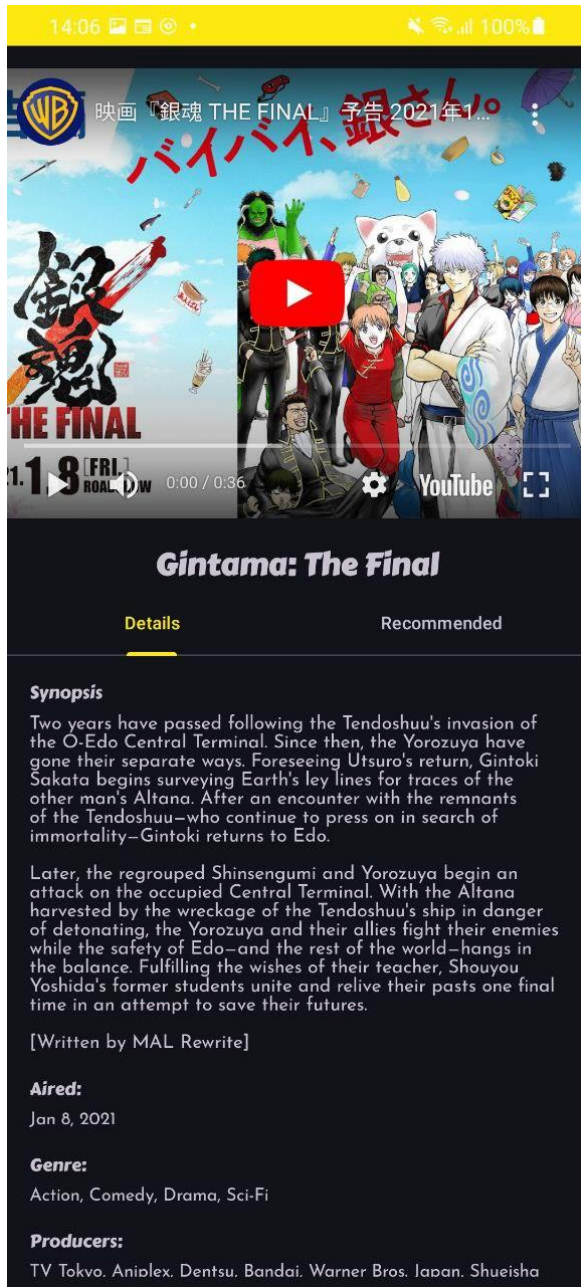




## TopAnimeFragment



## AnimeDetailActivity



## LogoutFragment (settings)

