Aplikacja do przeglądania filmów i seriali połączona z API i bazą danych

**Mateusz Ambroży 264240**

1. **Cel projektu**

Aplikacja mobilna stworzona w ramach projektu na przedmiot "Aplikacje mobilne". Głównym celem aplikacji jest umożliwienie użytkownikom przeglądania i zarządzania listą obejrzanych filmów oraz seriali pozyskanych z API, a także ocenianie i komentowanie oglądanych produkcji i przeglądanie opinii innych użytkowników o obejrzanych filmach i serialach.

1. **Technologie użyte w projekcie:**

* Retrofit: Biblioteka do komunikacji z API.
* Picasso: Biblioteka do ładowania obrazów.
* Cloudinary: Zewnętrzny serwis do przechowywania i zarządzania obrazami.
* Django: Framework użyty do stworzenia backendu aplikacji.
* REST API: Interfejs używany do komunikacji między aplikacją mobilną a serwerem.
* Java: język programowania w którym napisano aplikację.

1. **Struktura aplikacji:**

Aplikacja składa się z kilku głównych komponentów, które realizują poszczególne funkcje.

* *MainActivity:*

Ekran główny aplikacji, na którym użytkownik może zobaczyć opcje takie jak logowanie, rejestracja oraz przejście do listy obejrzanych filmów i seriali.

Obraz zawierający tekst, zrzut ekranu, Tablet, Telefon komórkowy

Opis wygenerowany automatycznie

* *AccountActivity:*

Aktywność, która umożliwia użytkownikowi przejście do kolejnych akcji w jego koncie

Obraz zawierający tekst, zrzut ekranu, Czcionka, design

Opis wygenerowany automatycznie

* *LoginActivity:*

Aktywność odpowiedzialna za logowanie użytkownika. Użytkownik wprowadza swoje dane, które są wysyłane do serwera w celu autoryzacji.

Obraz zawierający tekst, zrzut ekranu, Czcionka, design

Opis wygenerowany automatycznie

* *RegisterActivity:*

Aktywność odpowiedzialna za rejestrację nowych użytkowników. Dane użytkownika są przesyłane do serwera, gdzie następuje ich zapisanie w bazie danych.

Obraz zawierający tekst, zrzut ekranu, Czcionka

Opis wygenerowany automatycznie

* *MovieDetailActivity:*

Aktywność, która wyświetla szczegółowe informacje o wybranym filmie. Użytkownik może zobaczyć tytuł, rok produkcji, fabułę, plakat, a także oceny i opinie innych użytkowników.

Obraz zawierający tekst, zrzut ekranu

Opis wygenerowany automatycznie

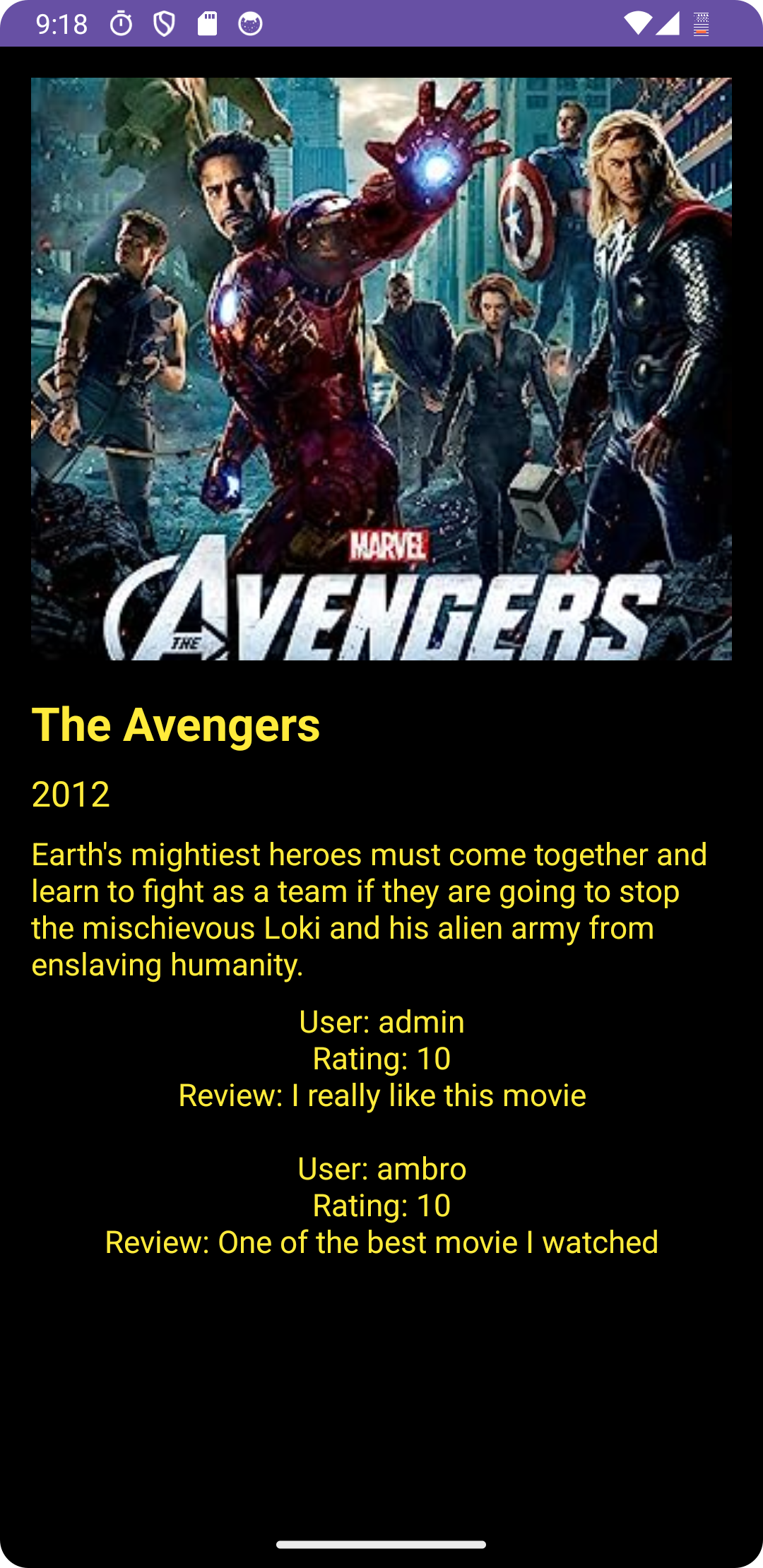
* *WatchedActivity:*

Aktywność odpowiedzialna za wyświetlanie listy obejrzanych filmów i seriali przez użytkownika.

Obraz zawierający tekst, zrzut ekranu, projekt graficzny, rafa

Opis wygenerowany automatycznie

* *MoviesWatchedDetailsActivity*  
  Aktywność w której można przeglądać szczegóły filmów i seriali (w tym opinie i oceny innych użytkowników), które zostały już obejrzane.



* *Backend (Django):*

Backend aplikacji został stworzony przy użyciu frameworka Django. Główne elementy backendu to:

**Modele:**

*User*: Model użytkownika aplikacji.

*Profile*: Profil użytkownika, zawierający dodatkowe informacje takie jak lista obejrzanych filmów i seriali.

*Movie*: Model filmu zawierający informacje o tytule, roku produkcji, fabule, plakacie itd.

*Review*: Model recenzji filmu, zawierający opinię i ocenę użytkownika.

1. **Problemy i ich rozwiązania:**

Podczas tworzenia aplikacji natrafiono na kilka problemów, m.in.:

* **Błąd z Cloudinary**: Początkowo aplikacja zgłaszała błąd związany z konfiguracją Cloudinary. Problem został rozwiązany przez dodanie odpowiednich zmiennych środowiskowych w pliku .env i prawidłową ich konfigurację w pliku settings.py w Django.
* **Komunikacja z API**: Podczas przesyłania danych między aplikacją a serwerem napotkano problemy związane z autoryzacją. Zostały one rozwiązane przez prawidłową konfigurację JWT w Django oraz poprawne implementowanie obsługi tokenów w aplikacji mobilnej.
* **Wyświetlanie danych w MovieDetailActivity**: Aby poprawnie wyświetlać dane filmu oraz recenzje użytkowników, do aktywności dodano odpowiednie pola oraz logikę pobierania danych z API.
* **Resetowanie się tokenów JWT:** access token oraz refresh token przy zamykaniu widoku MainActivity. Problem polegał na automatycznym wylogowywaniu się z aplikacji przy jej zamknięciu powodował usuwanie tokenów przy przechodzeniu między innymi aktywnościami.

1. **Wnioski**  
   Projekt aplikacji mobilnej pozwolił na praktyczne wykorzystanie umiejętności zdobytych podczas kursu. Implementacja obejmowała zarówno frontend (aplikacja mobilna) jak i backend (serwer Django), co umożliwiło pełne zrozumienie procesu tworzenia aplikacji od podstaw do jej wdrożenia. Dzięki temu projektowi możliwe było nabycie cennych doświadczeń w zakresie integracji różnych technologii oraz rozwiązywania realnych problemów programistycznych