DPS926 Assignment 2 Report – Movie Space App

Harsh K - 127318194

1. What is the final app idea and functionality?

The idea of this application is to help users **search, discover, and keep track of all the latest movies**. It aims to help users stay in the know through discovery and recommendations. Users can find what's playing in cinemas as well as what's trending and upcoming. Movie Space provides all the information users would need for a movie and it presents that information in a seamless and stylized user interface. Users can **search for a movie** or **explore various genres** to find their next movie. Users can **add movies to their library, set watching statuses, and leave a score** to keep track of movies on their list as they go. The user can set statuses between "Watching", "Completed", and "Plan to Watch", and also give the movie a score from 1-10 stars. Movie Space uses a **recommendation algorithm** to suggest movies based on movies the user has completed. The algorithm analyzes at most 4 movies (if the user has more, it analyzes 4 at random each time) and provides around 10-20 recommendations for each movie. The user can then access these recommendations using the option in the explore tab and can add movies they like into their library. Throughout the application there are also various quality of life features such as, when searching for movies or viewing lists of movies, the user can easily **sort their search results** using various sort parameters; or being able to **play a movie's trailer** or **sharing a movie** they found with their friends; or **visiting the webpage of the movie** so that they can know more about the movie or book tickets, etc. I still have more plans for the application and have published it to the play store, where it is currently under review. Once it's approved, I will add a link to the application on google play in my GitHub Repo and YouTube Demo description.

2. What is the used web services?

The web service / API I have used for this application is The Movie Database (TMDB). This API offers all the information required for this application. It has enough options to the point where it has inspired me to consistently increase the scope of the assignment after seeing how simple and easy it was to use. It offers collections of movies that are “Now Playing”, “Trending”, “Popular”, “Upcoming”, etc. We can also search for a movie and get all its details, or search movies by genre, or find recommendations for a single movie. This method of finding recommendations is also what was utilized for creating the recommendation algorithm. This API was utilized using a NetworkingManager class and used to retrieve Movie collections, genres, movie details, trailers, etc.

3. What is the data that is stored locally in SQLite DB?

At first, only the basic “Movie” object and user watch status and ratings were stored locally, as that was simpler to implement. However, to reduce API calls I decided to store the complete “MovieDetails”, also containing the user watch status and ratings. The application works in such a way that on the main pages, a simpler “Movie” model is utilized to display information on the screen. When requesting for collections of movies from TMDB, it does not return all the information of a movie (missing some information like genre names, runtime, budget, revenue, etc.). To counter this I implemented a MovieDetails class which inherits from Movie and whose model is created when a Movie is clicked on. Thus when the user saves a movie to their library, this MovieDetails model is what is saved. This allows the user to access the movies in their library and obtain all the necessary information of the movie without the need of additional API calls. The CRUD of MovieDetails information is handled using an SQLite DB which is stored locally on the user’s device with no information collected or shared with a third party.