BUILDING MOBILE APPLICATIONS

FINAL PROJECT

MOVIE EXPLORER

Faculty of Information and Management Systems

Program : Computer Science

Name: Umera Noor Mohammed Syed

Group: 108158-1

Date of Submission: 17 January 2025

Introduction:

The **Movie Explorer app** is a SwiftUI-based project designed to help users explore a curated list of popular movies. It provides an easy-to-use interface where users can browse through a list of movies, view detailed information about each movie, and manage a **"Watch Later"** list by adding or removing movies. This app is a beginner-friendly demonstration of SwiftUI concepts, including navigation, lists, state management, and UI customization.

Development Steps:

Step 1: Setting Up the Project in Xcode

1. Installed Xcode on macOS.
2. Created a new project and selected **App** as the template under **iOS**.
3. Named the project **Movie Explorer** and ensured the **SwiftUI** framework was selected.
4. Organized the project into three files:

    - Movie.swift: Contains the model for movies.
    - ContentView.swift: Defines the main page of the app.
    - MovieDetailView.swift: Displays the details of each movie.

Step 2: Adding the Main View & Navigation

1. Designed the main page using a **NavigationView** for smooth navigation between views.
2. Used a **List** to display movies, which included the title and release year.
3. Implemented a **NavigationLink** for each movie, allowing users to tap and navigate to the detail page.
4. Customized the background color and list row appearance to improve the user interface.

Step 3: Creating the Movie Model

1. Defined a **Movie** struct in **Movie.swift** to represent a movie with the following properties:

    - Title

- Genre
- Release Year
- IMDb Rating

2. Added a **sampleMovie** property to provide dummy data for testing previews.

Step 4: Designing the Detail View & Adding a Button

1. Created **MovieDetailView.swift** to display detailed information about a selected movie, such as:

    - Title
    - Genre
    - Release Year
    - IMDb Rating

2. Added a **Button** to toggle a "Watch Later" status. The button changes its label and color dynamically when clicked:

    - If the movie is added to the list, it displays "Added to Watch Later."
    - If removed, it displays "Removed from Watch Later."

3. Used @State variables to manage the button's state and success messages.

Step 5: Testing and Debugging

1. During the testing phase, the app was run in the iPhone simulator to ensure functionality and compliance with project requirements. The following issues were encountered and resolved:

    1.1. Incorrect Formatting of Release Year

    - **Issue**: The release year displayed with commas (e.g., 1,999 instead of 1999).
    - **Solution**: Converted the year to a plain string using String(movie.releaseYear) to prevent unnecessary formatting.

    1.2. Preview Not Working

    - **Issue**: The preview for MovieDetailView failed to render properly.

- **Solution**: Added a sampleMovie in the Movie struct to provide dummy data for SwiftUI previews.

1.3. Visual Customization

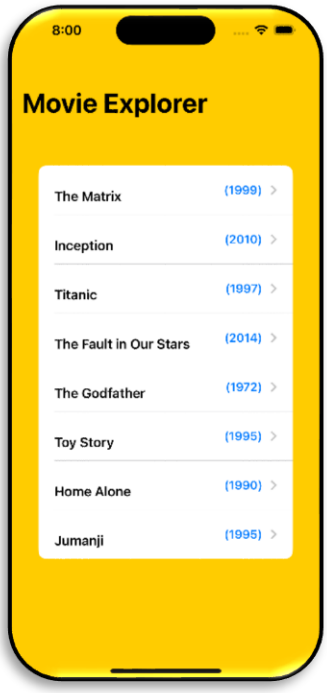
- **Issue**: The app appeared plain with default SwiftUI styling.
- **Solution**: Enhanced the UI by adding a colored background, custom fonts, and corner radius for a more polished appearance.

Step 6: Finalizing the App

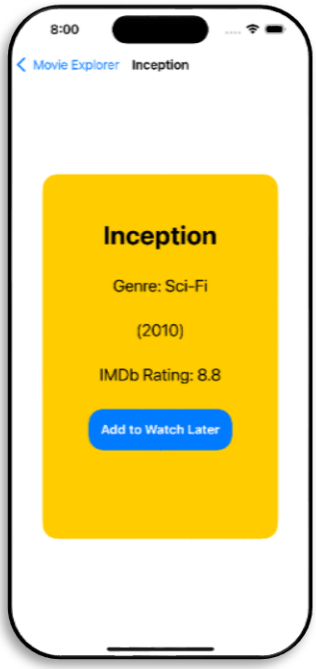
1. Ensured the app met all project requirements, including:
    - Two functional pages (main list and detail view).
    - Interactive button functionality.
    - Clean, user-friendly design.
2. Captured screenshots of the app for documentation.

Screenshots:

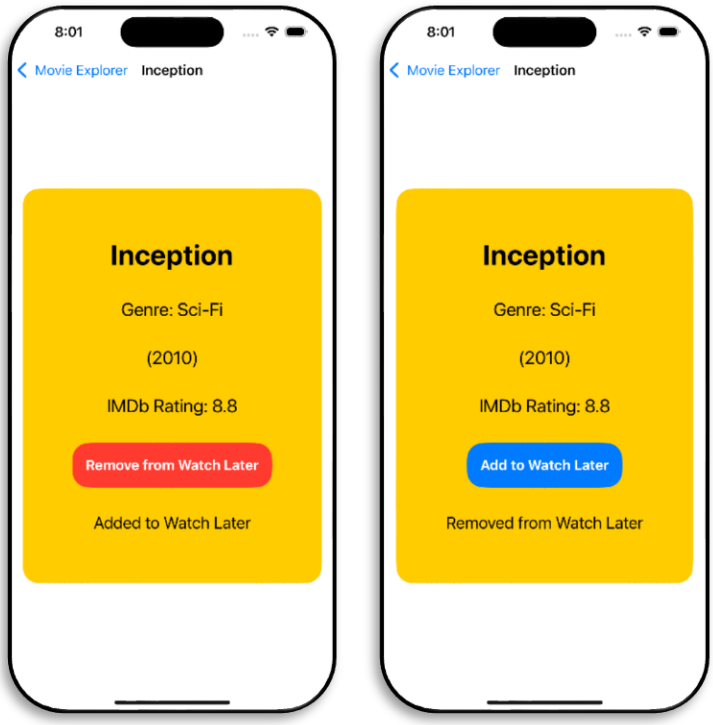
The screen showing the list of movies with their titles and release years



A detailed view of a movie with its genre, release year, IMDb rating, and the "Add to Watch Later" button.



Upon clicking the button, the following message appears on the screen along with the change in button display.



Conclusion:

The **Movie Explorer app** is a simple yet effective demonstration of SwiftUI's capabilities, designed to meet the requirements of a beginner-level project. It successfully combines essential features like navigation, lists, state management, and user interaction within a clean and user-friendly interface.

Through this project, I gained valuable experience in building iOS applications and deepened my understanding of SwiftUI concepts, such as creating views, managing states, and handling navigation. Additionally, this project gave me the opportunity to learn and work with Xcode, Apple's powerful development environment for building iOS apps. I explored its various features, including the interface builder, preview functionality, and the iPhone simulator, which made testing and debugging more efficient.

The app not only fulfills its intended purpose of exploring movies and managing a "Watch Later" list but also lays a strong foundation for future enhancements, such as adding advanced features like search functionality or persistent data storage. This project has been an insightful learning journey, showcasing how even small applications can deliver meaningful functionality with thoughtful design and efficient coding practices.

****