🎬 Movie Discovery App

Version: 1.0.0 **Status:** Development

📖 Overview

Movie Discovery App is a modern Android application built with **Jetpack Compose** that allows users to browse popular movies, search for specific titles, and view detailed information about them.

The app leverages the **TMDB (The Movie Database) API** to fetch real-time data and employs a clean **MVVM (Model-View-ViewModel)** architecture to ensure scalability and maintainability.

✨ Key Features

- 🏠 **Home Feed:** Browse a curated list of "Popular Movies" fetched from the API.
- 🔍 **Search:** Real-time search functionality to find specific movies.
- 📄 **Movie Details:** View in-depth information about selected movies.
- 🌓 **Dark/Light Mode:** Built-in theme switcher to toggle between dark and light themes.
- 📱 **Modern UI:** Fully built using Kotlin and Jetpack Compose with Material Design 3.

🛠️ Tech Stack

Language & Core:

- Kotlin (100%)
- Coroutines & Flow (Concurrency)

UI & Presentation:

- Jetpack Compose (Declarative UI)
- Material3 (Design System)
- Coil / Glide (Image Loading)

Architecture & State:

- **MVVM** (Model-View-ViewModel) Pattern
- ViewModel
- Navigation Compose (Single Activity Architecture)

Networking:

- Retrofit (REST Client)
- OkHttp (HTTP Client & Interceptor)
- Gson (JSON Parsing)
- Kotlinx Serialization

🏗️ Architecture

The project follows a **Feature-Based** directory structure, keeping related logic (Network, Repo, UI, ViewModel) grouped by feature.

Directory Structure

```
com.mustafa.myapplication
  ├── Details/              # Movie Details Feature
  │    ├── DetailsNetwork/   # Remote Data Source
  │    ├── DetailsRepo/      # Repository Implementation
  │    ├── DetailsUI/        # Composable Screens
  │    └── DetailsViewModel/ # State Management
  ├── Home/                 # Home/Popular Movies Feature
  │    ├── HomeNetwork/
  │    ├── HomeUi/
  │    ├── HomeViewModel/
  │    └── Repo/
  ├── Search/               # Search Feature
  │    ├── SearchNetwork/
  │    ├── SearchRepo/
  │    ├── SearchUI/
  │    └── SearchViewModel/
  ├── model/                # Data Classes (DTOs)
  ├── network/              # ApiClient & Retrofit Setup
  ├── ui.theme/             # Theme & Color Definitions
  ├── MainActivity.kt       # Entry Point & Navigation Host
  └── Routes.kt             # Navigation Routes
```

Data Flow

1. **UI (Compose)** observes state from the **ViewModel**.
2. **ViewModel** requests data from the **Repository**.
3. **Repository** fetches data from the **RemoteDataSource** (Retrofit).
4. **Retrofit** calls the **TMDB API**.
5. Data flows back up to the UI via **Kotlin Flows/State**.

🚀 Getting Started

Prerequisites

- Android Studio Koala or newer.
- JDK 11 or higher.
- A **TMDB API Key**. You can get one here.

🔑 API Key Configuration

This project keeps the API key secure in `local.properties` so it is not committed to version control.

1. Open the file `local.properties` in the root of your project.

Add your API key as follows:
sdk.dir=C\:\\Users\\YourName\\AppData\\Local\\Android\\Sdk
TMDB_API_KEY=your_actual_api_key_here

2.
3. Sync Gradle. The app will access this key via `BuildConfig.TMDB_API_KEY`.

Installation

Clone the repository
git clone
https://github.com/your-username/movie-discovery-app.git

1.
2. **Open in Android Studio**
    - File -> Open -> Select the project folder.
3. **Build and Run**
    - Connect an Android device or start an Emulator.
    - Click the **Run** button (green arrow).

📡 API Reference

This app uses the The Movie Database (TMDB) API.

Feature	Endpoint Used (Example)	Description

Popular	`GET /movie/popular`	Fetches a list of current popular movies.
Search	`GET /search/movie`	Searches for movies by query string.
Details	`GET /movie/{movie_id}`	Gets specific details for a single movie.

🤝 Contributing

1. Fork the project.
2. Create your feature branch (`git checkout -b feature/NewFeature`).
3. Commit your changes (`git commit -m 'Add NewFeature'`).
4. Push to the branch (`git push origin feature/NewFeature`).
5. Open a Pull Request.