

Android TV News App

Features

- Jetpack Compose
- Material3 Components
- Retrofit
- MVVM Architecture
- Shimmer Effects for Loading
- Immersive List for better User Experience
- Category Change Using Row Tabs
- Refresh on Long Press D-pad down
- Custom Icon and Splash Screen

Steps Taken

1. Setup Empty Compose Activity for TV Screens (1080p).
2. Added Gradle Dependencies for Android TV Development.
3. Created Folder Structure Following MVVM Architecture.
4. Initialized Empty Git Repository, connected to Remote Repo to begin to track changes.
5. Created API key from [NEWS API](#).
6. Created API_KEY variable in [local.properties](#) and imported in gradle file using :

```
android{
```

```

val localProperties = Properties()
val localPropertiesFile = File( rootDir , "local.properties")
if( localPropertiesFile.exists() && localPropertiesFile. isFile ){
    localPropertiesFile. inputStream (). use {
        localProperties.load( it )
    }
}

buildTypes {
    release {
        buildConfigField(
            "String",
            "NEWS_API_KEY",
            localProperties.getProperty("NEWS_API_KEY")
        )
    }
    debug {
        buildConfigField(
            "String",
            "NEWS_API_KEY",
            localProperties.getProperty("NEWS_API_KEY")
        )
    }
}
}

```

7. Used JSON from News API example to build Network Model using Extension "Kotlin data Class From JSON".
8. Created Constants Class and initialized BASE_URL, SEARCH_DELAY, QUERY_PAGE_SIZE and API_KEY.
9. Added new Dependencies : Retrofit, okhttp3, okthhp logging interceptor, gson.
7. Created NewsAPI interface using retrofit2.
7. Built the Singleton Retrofit Client.

7. Started to Work on HomeScreen Composable.
8. Created a first version Working UI to display the News, UI does not look professional as of now.
9. Loading Screen utilized Circular Indicator, looked buggy, so created custom Shimmer Effect Extension function for Modifier Class.
10. Created Custom Card Component Using Box and Composables.
11. The First Version utilized lazy Column and with a overall vertical Layout.
12. Switched to Lazy Row with Custom Immersive List Components.
13. Changed Shimmer Effect to match Horizontal Layout.
14. Created Tab Row for Category Change.
15. Used Annotated String to Customise the First Letter of the Heading.
16. Added Custom Typography and Theming.
17. Changed Overall Theme to Purple and Black.
18. Changed font sizes according to Usual Android TV Standards.
19. Added Refresh on Long D-Pad Down Key

Problems:

1. Did not make the ImageURL Nullable, caused few crashes, but fixed it.
2. Called .length on Null value causing the app to crash.

3. The UI Code became too messy to handle in one file.
4. Circular Loading Seemed too old school and boring.
5. Vertical Layout did not seem intuitive for a TV Screen.
6. Adding Tab Rows Disrupted the Overall Layout.
7. The Selected Tab does not trigger Api call
8. Removed News Items Showed up and disrupted to overall feel.
9. **NEWS API not fetching NEWS for countries other than "us".**

Fixes:

1. Learnt more about Defensive Programming, increased the usage of Exception handling, Elvis Operator, Safe dot, and the null assertion.
2. Created a Components package inside the UI package to separate reusable components following the DRY principle, contributing to a cleaner and maintainable codebase.
3. Created Custom Shimmer Effect Extension function for Modifier Class.
4. Changed Layout to be Horizontal.
5. Adjusted the Whole layout to look Modern and Slick.
6. Used mutable state to store Selected Category which causes the updated API call and trigger Recomposition.
7. Added Check if the Title == "[Removed]" it won't be displayed in the lazy Row.