Final Project Proposal

Team 17: Charlie Say, Toby Parrish, Ramish Mohammad, Mason Mann

Requirements

- A list of all of the members of your team. You should aim to have 3 or 4 people on your team.
- A high-level description of what your app will do.
- A link to the specific 3rd-party API your app will use, an explanation of the API methods your app will call, and a description of how the data from those methods will be used in your app or what data will be sent from your app to the API. The final project description contains a list of APIs that could work well for this project.
- A description of how the UI for your app will be organized, including a list of your app's activities, a description of how the user will navigate between those activities, a description of any major non-navigation interactions available to the user, and a description of any notifications the app will display.
- A description of the additional feature not covered in class that your app will implement.
- Mocks (i.e. visual prototypes) for each of your app's major activities. These can be as simple as paper-and-pencil sketches that you scan and include into the report, or you can use a prototyping tool like Figma (and if you do, remember that you can apply for a free Figma education plan since you're a student).
- A description of the division of labor for the project between the members of your team (see below for more info about this).

Description of App

The application will require authentication of the user associated with a STEAM account in order to use the application. Once they are confirmed with a legitimate account, they will be redirected to the "Profile" screen where this activity will contain a "Wishlist", "Save Later", and "Compare" list that will be scrollable. Pressing the icons/buttons on the lists will bring up the "Game Details" screen where there will be more information about the selected game as well as have the option to add to additional lists. Adding or removing a game from a list will cause a notification to appear on the screen. The "Game Details" will also allow the user to activate an implicit intent to watch the game trailer on YouTube.

The main functionality of the application is to be able to compare games beside each other in order to make a decision on whether or not to add to the "Wishlist". Important features in order to accomplish this is to have a "Search" functionality along with a "Results" page. The "Search" activity will also have suggestions and autocomplete features. The "Results" screen will have

the results of the search query with possibly a recent game section where previous searches were made.

Some other activities and features include a navigation bar at the top and bottom of each activity. The top navigation will include a "back" and "logout" button". While the bottom navigation will have a "home", "explore", "search", and "settings" button. The "home button will take the user back to the "profile" screen, "explore" will take the user to the "Explore" screen where the user can discover new games based on recommendations and most popular, "search" will allow the user to look for games to add to their wishlist, save later, or compare list, and "settings" will have preferences for the user to filter games.

Link to Specific 3rd-Party API

https://steamcommunity.com/dev/registerkey
https://developer.valvesoftware.com/wiki/Steam_Web_API

Description of UI

Organization - Navigation bar at the bottom (Home, Explore, Search, Settings), all activities will be initiated with buttons and icons on each page, there will also be a "logout" and back button on several pages for ease of navigation.
Activities - Login activity page, profile activity page, search page to find games and compare them, exploring game activity page and a settings page
Navigation - Will have designated buttons and maybe also swipe features
Major non-navigation interactions - Adding/removing games from the compare/saved games list, searching for games, sign-in/registration activity
Notifications - When a game is added to your wishlist

Description of Additional Features (Not all features will be implemented)

Jetpack Composable - compose.animations, compose.material/3, compose.ui
Material Design Components - Modular and customizable Material Design UI components for Android.
Credentials Library - For registration and app sign-in (might not need because of Steam API authentication)
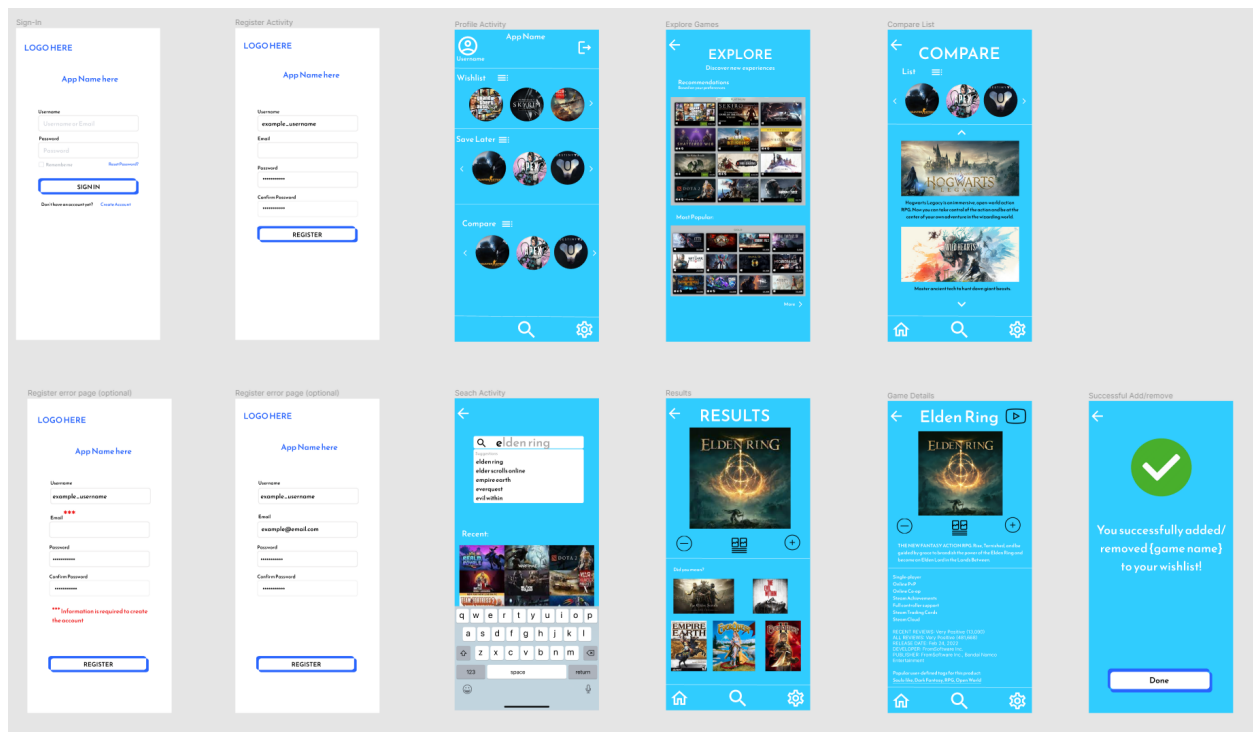Drawerlayout - Implement a Material Design widget
Gridlayout - Implement a grid layout
Navigation - Build and structure your in-app UI, handle deep links, and navigate between screens.
Room - Create, store, and manage persistent data backed by a SQLite database.

Mocks (i.e. Visual Prototypes)

https://www.figma.com/file/cyY7V2Y52sA2gviJA5tAnt/Mobile-App-Prototype?node-id=0%3A1&t=VX5yO7CVzXHgJJ3t-1

Dividing Project Labor

- Charlie Say - UI components for all activities, implement lifecycle ViewModel architecture
- Ramish Mohammad - Compare selected games and also saved games / wishlist
- Mason Mann - Steam WebAPI integration, Steam Sign-In, Login/Register Logic
- Toby Parish - UI components for all activities, implement lifecycle ViewModel architecture