GRIFFITH COLLEGE

Assignment Cover Sheet

Learner Name: **Joshan John**

Learner Number: 3092883

Faculty: Computing

Programme: BSCH Stage/Year: 4

Module: BSCH-MD

Study Mode: Full Time

Lecturer Name: Tracey Cassells

Assignment Title: Mobile Development – Milestone 2

Additional Relevant Information (e.g. number of pieces submitted, etc.):

_____

AI in Learner Assessment Policy Indicate here applicable categories allowed for this assignment.	Category of AI Use	Allowed or not allowed
	No AI Use	No
	AI for Planning	No
	AI for Editing	No
	AI for Support Tasks	No
	AI for Collaboration	No
	Full AI Use	**(NOT ALLOWED)**

Academic Integrity Honour Code:

I submit this work in line with the principles of Academic Integrity as appearing in the Academic Integrity policy, the assessment description(s), and the relevant AI Assessment Scale in the AI in Learner Assessment

I affirm that I have not given or received any *unauthorised* help, from a person or through unauthorised content generation on this assignment, and that this work is my own.

I understand that penalties may be imposed if this assessment is in breach of the Academic Integrity and Misconduct policy.

Signature: Joshan John Date: 29/11/2025

Please note: Learners **MUST** retain a hard/soft copy of all assignments and must have the lecturer/member of Faculty acknowledge/sign as proof of submission.

TABLE OF CONTENT

Table of Contents

LUCKY WHEEL
Mobile Development - Milestone 2

Project Details

Project Name	Lucky Wheel
Sensor's used	Accelerometer
minSDK	Android SDK 26
Firebase services	Authentication (email and password + Google Auth), Real-time Database
Data preferences	Data Store (preference)
Fonts (Google Fonts)	Inter, Knewave, Merienda, Space Grotesk
GitHub links	• https://github.com/joshanjohn/luckywheel_3092883 • Milestone 2 closed Issues (click to view)
Ai prompt for logo	https://gemini.google.com/share/5c64652d1232

Main Goal

Lucky Wheel is a mobile gaming app for a fun and competitive experience, where players collect gold by spinning a wheel using their device's sensor. Each spin lands the pointer on a random segment of the wheel, allowing players to win or lose gold based on where it stops. The app also features a real-time leaderboard, showcasing players with the highest gold counts, making the game engaging and competitive.

Milestone 1

Features and Improvements

- Email and password based authentication using firebase.
- Implement preference for auto login using the datastore preferences.
- Use an accelerometer sensor to spin the wheel and the stopping mechanism.
- Make the spinning of the wheel feel natural, so that it starts with a varying acceleration and ends with some natural, smooth friction.
- Implemented a gold scoring system and a real-time leaderboard.

- Implemented a navigable screen to switch between the gold wheel section and the custom game.
- Implement a custom wheel with varying percentages of pi size, with an option to edit wheel item properties.

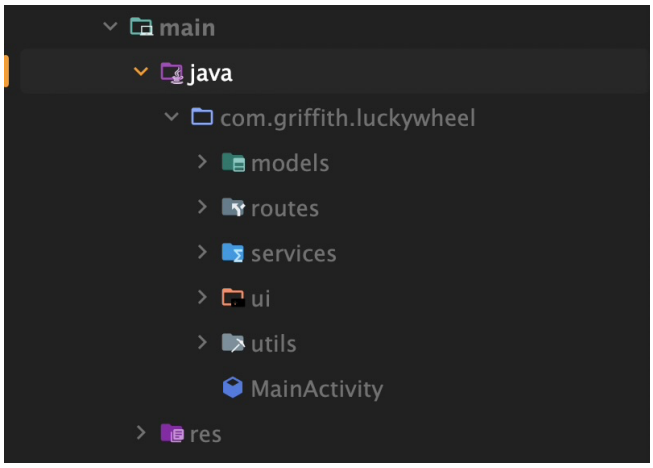
Milestone 2

Features and Improvements

- **Improved authentications**.
    - Support's Google Authentication for google account user's.
    - Support firebase password recovery, password reset link to each user.
- **Improved spinning wheel**.
    - Support dynamic label colour based on background wheel item colour.
    - Larger label names as compared to previous.
- **Improved custom wheel game**.
    - Player can save the customised game wheel to the firebase storage.
    - Player can access the saved customized game from firebase storage.
    - Player can update an saved game after editing the wheel items.
    - Limit the number of custom wheel item, and all items are visible on adding initial.
- **Improved game Setting Screen**.
    - Improve the UI design and icons for better user interaction.
    - Fully functional settings buttons.
    - Player have access to settings options, which allow player to navigate to various game sections.
- **Dynamic result popups**
    - Awesome result UI based on each spinning result.
    - In custom game, result dialogue shows matching colour based result's.
- **Profile and account management**
    - Player can edit the username or player name.
    - Player can delete the account and all the data associated to the player.
- **Improved logo and system performance**
    - New unique app logo for milestone 2.
    - Improve all the service performance by using IO threads.

Project Architecture

The project is developed in an agile SDCL method. The project uses the Model View Controller (MVC) 3-level design architecture pattern.



All the model classes, defined as data classes in Kotlin, are organized within the data package. The services package contains all service and controller-related code, including the Firebase service, authentication service, and data store service. The UI elements, such as themes, screens, components, and UI logic, are placed in the UI package, along with the themes.

App UI Theme

The reason for choosing the current app UI theme is that it brings an arcade-playing feel. Just like a traditional red and black roulette wheel. For instance, in the gold collection game uses an alternative dark and light green colours, which mirrors the classic roulette wheel colouring pattern.

Some Key design considerations include:

- **Darker background**: Helps the user focus more on the spinning wheel while maintaining an immersive feel. The buttons also carry a green accent, maintaining visual consistency.
- **Interactive button design**: The press-and-hold button is circular, making it comfortable to use in any device orientation and reducing accidental touches. Additionally, the button changes to a light green shade when pressed, providing immediate visual feedback to the user.

Colour Theme

The App uses the following colours for the app scaffold background colour, in a linear gradient fashion. For some elements, it has a golden yellow colour.

	Colour	Hex value
	darkGreenColor	0xFF0BA136
	darkerGreenColor	0xFF092609
	extraDarkerGreenColor	0x85071507
	lightGreenColor	0xFF20B652

Font Theme

The application uses the Google Fonts service for its typography. The primary fonts are **Inter (**Anon, n.d.), which is used for main UI text, body content, and labels, and **Space Grotesk** (Anon, n.d.), which is applied to secondary headings and highlighted sections.

For decorative titles and special headers, the app uses the **Knewave** (Anon, n.d.) font, while instructional text within the playground area is styled with **Merienda** (Anon, n.d.). These fonts are all sourced from Google Fonts.

App Icon

Lucky Wheel app uses a custom logo, which is unique and stylish, rather than normal android icon.

```
<application
    android:allowBackup="true"
    android:dataExtractionRules="@xml/data_extraction_rules"
    android:fullBackupContent="@xml/backup_rules"
    android:icon="@mipmap/lucky_wheel_logo"
    android:label="@string/app_name"
    android:roundIcon="@mipmap/lucky_wheel_logo_round"
```

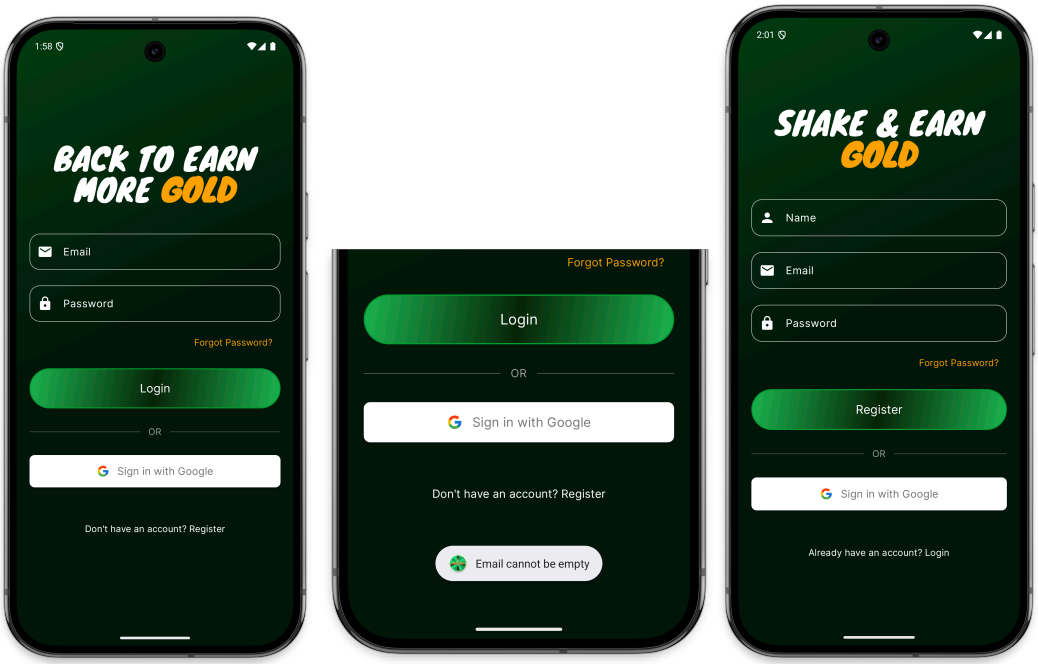
App UI Design

Authentication Screens

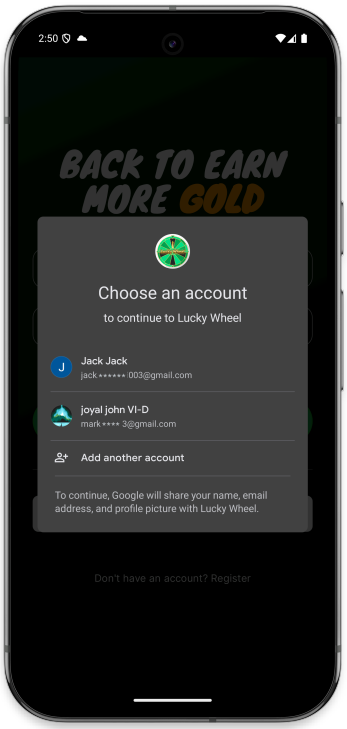
The app allows new user to register using their name, email, and a password. The existing user could log in. Once the player logs in, the login info will be stored in the data store as a preference, so next time when the user comes back after a pause or closing the app, the app will still be logged in.

Both the login and register text fields are sanitised and validated before authorization. Any validation error will be displayed as a toast at the bottom.

Player has an option to recovery the forgotten password and Google authentication option.

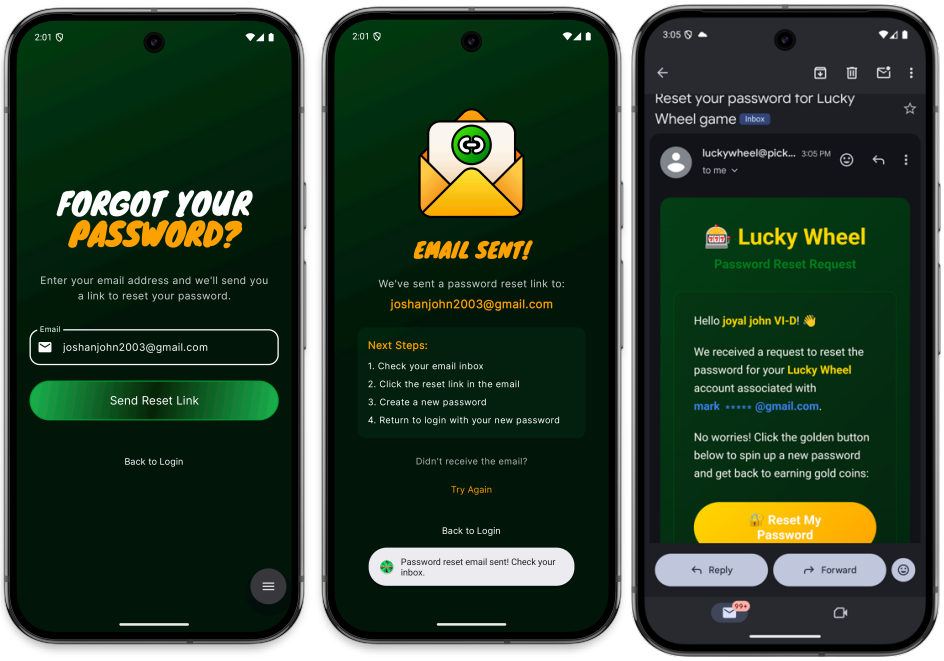


On pressing the Google signin button it show the list of available or already registred google accounts in the current device.



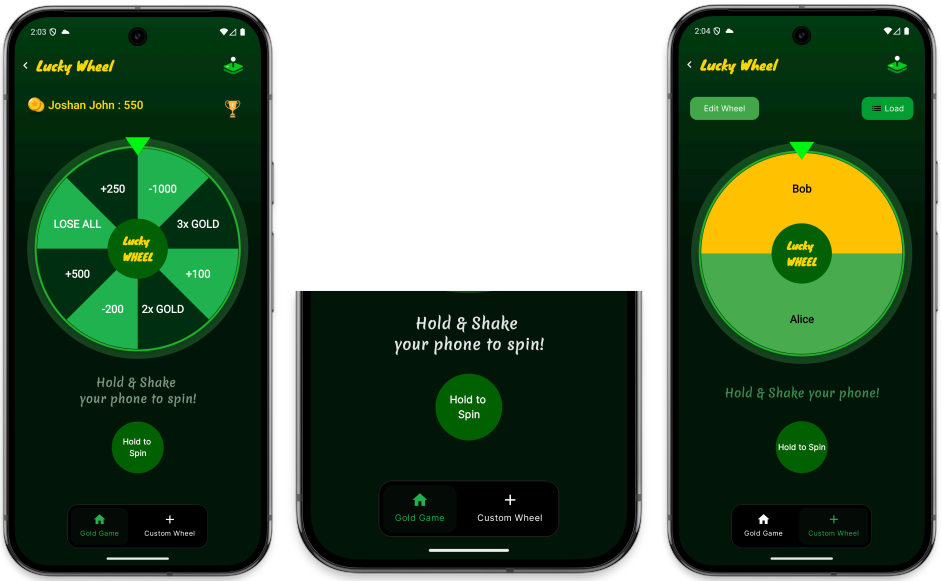
Forgot Password Screens

From authentication screens if the player choose the forgot password option it will take to forgot password screen where user provided respective registered email. On submitting the email, if the email is registered on firebase authentication, it will send a password recovery link to respective mail address. Plus on Ui it shows guide on how to reset the password.



Playground Screens

Inside the playground, the player could switch between the gold game and the custom player spinning wheel game. On top right corner it has options to game setting/control.

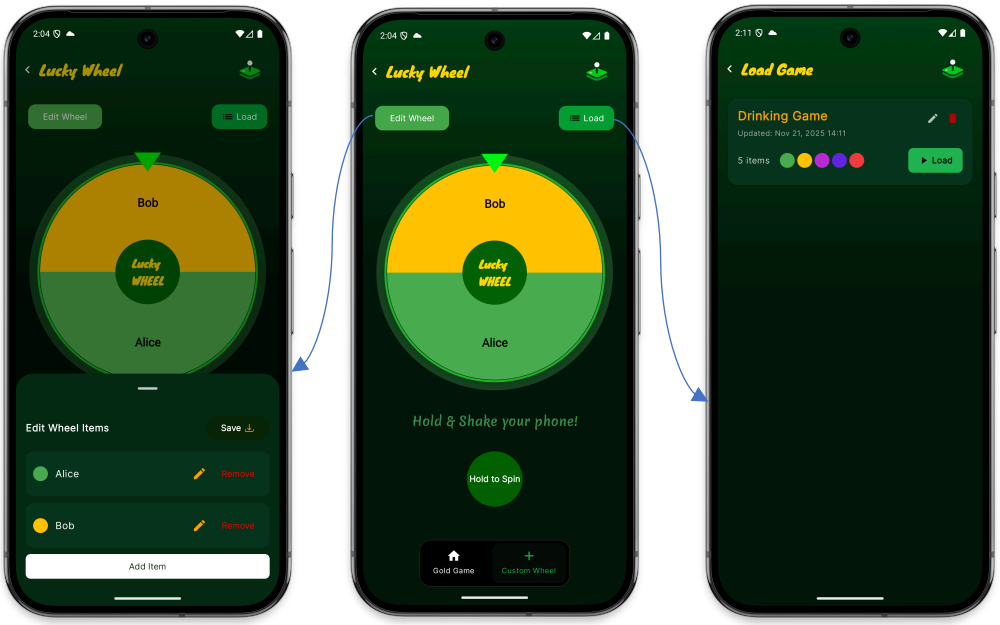


Gold Wheel Section

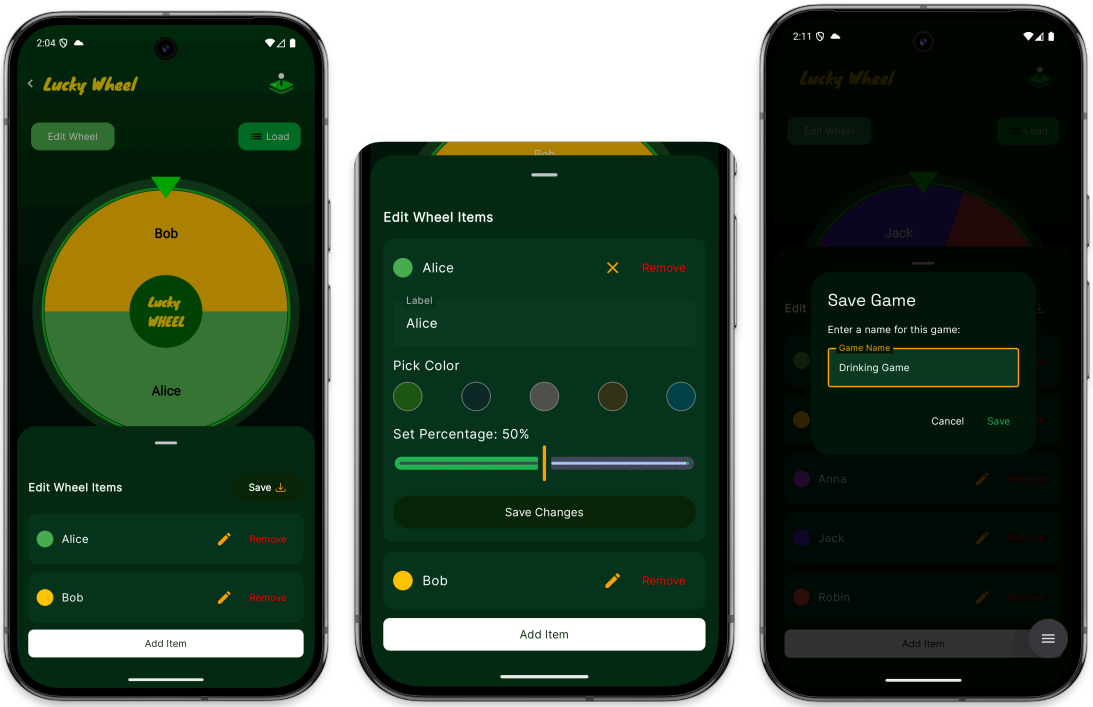
Player spins the lucky gold wheel by pressing and holding the green bottom button and shaking to spin the wheel. The spinning result shows as soon as the wheel stops as a pop-up window. New result popup dialogue looks beautiful and information centric

Custom Wheel Section

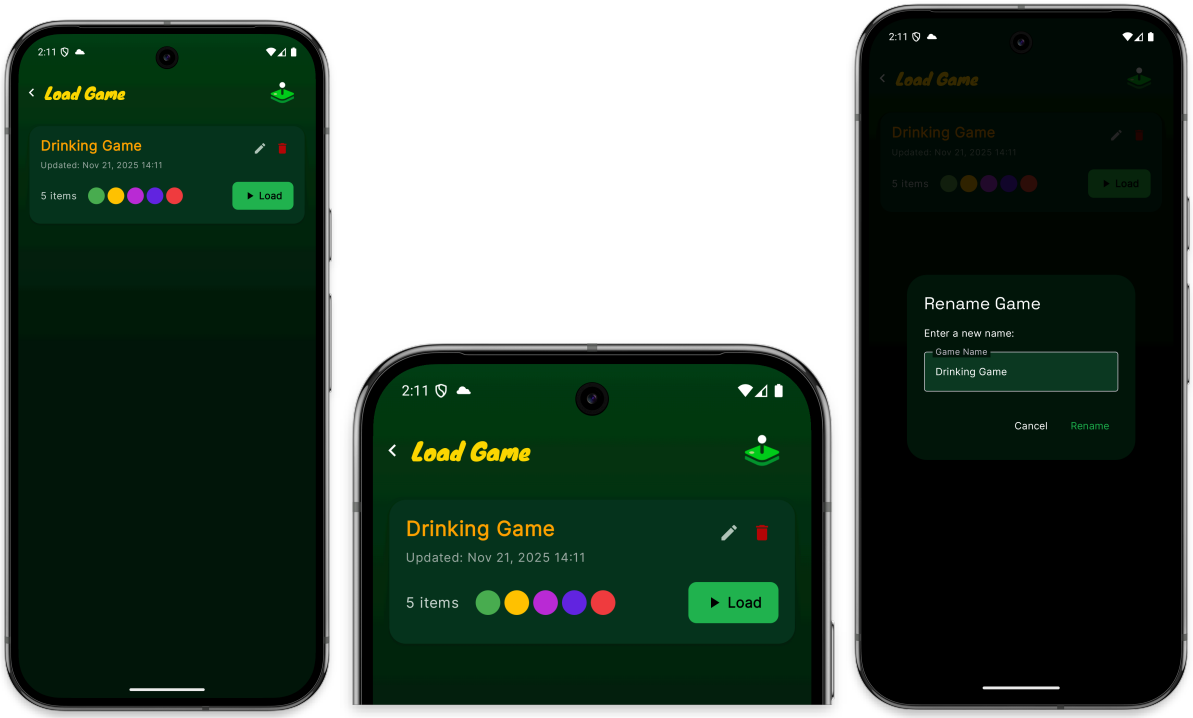
The player can use the same press and shake feature to spin the user-customizable wheel.



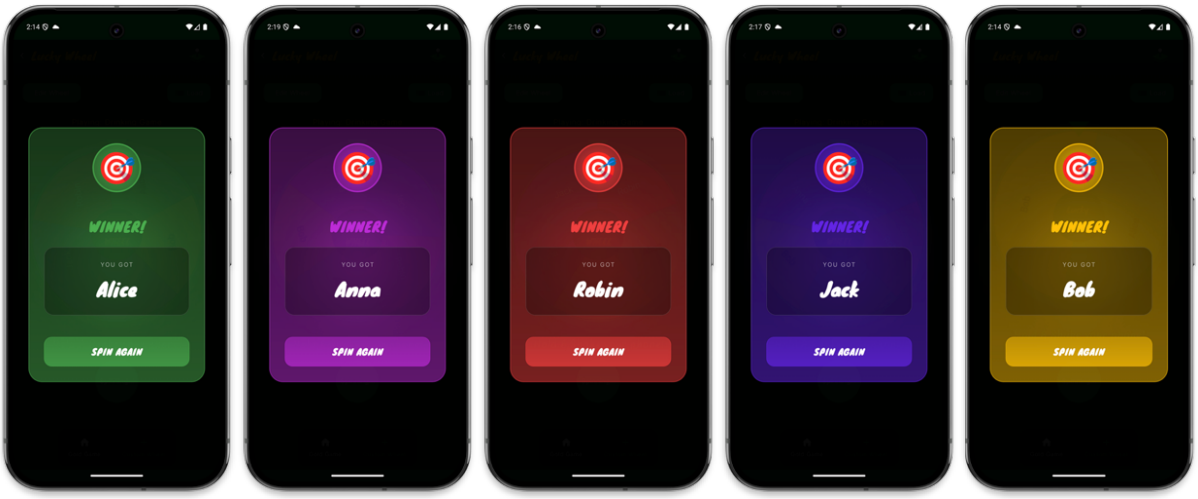
On clicking the "**Edit Wheel**" button, the user can add a new item to the wheel and adjust the percentage and colour of the pie. The user can also rename the label. If needed user can also remove the item from the pie. After editing player can save the existing game with a name.



On Clicking “**Load**” button it shows all the previously saved custom games which player can resume playing. Player can also edit the save games name and delete entirely.



After each spinning the result dialogue popup, the colour of dialogue screen depends on chosen wheel item.



Leaderboard Screen

Players can view the top-ranked players by clicking the trophy icon.

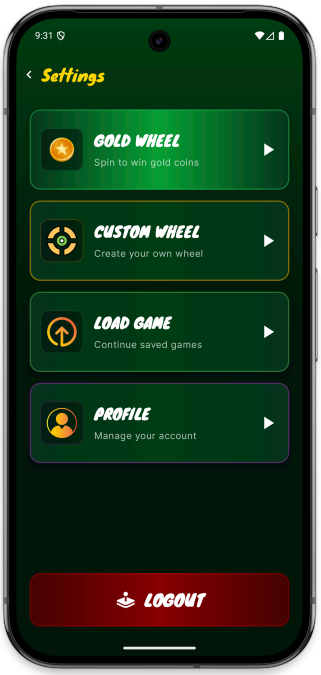
Overall, the app has a green arcade-style theming with a golden splash on some UI components.

Settings Screen

The settings screen has buttons which allows player to navigate and control the game.

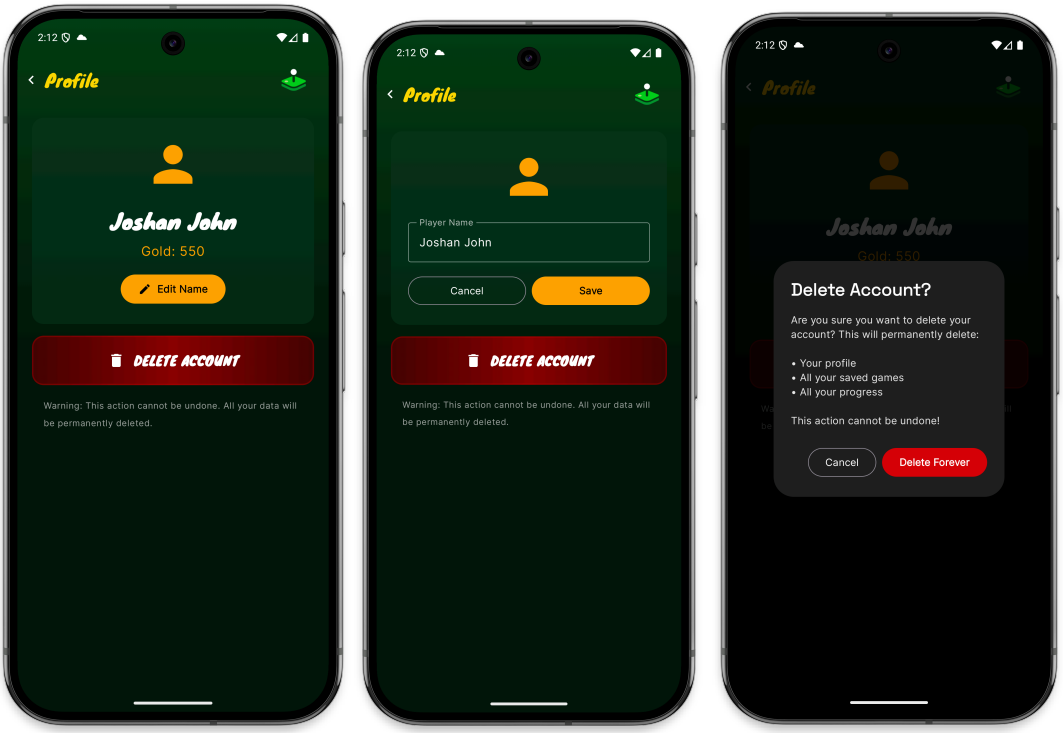
Some of the options include

- Navigation to gold wheel game.
- Navigation to custom wheel game.
- Load existing games.
- View account and profile settings.



Profile Screen

Player can view their name and gold count with option to delete the account. Player has option to edit the player or user name. If player wish to delete the account and press "**Delete Account**" button, it shows a confirmation popup, to avoid accidental account deletion and to confirm the player or user choice.

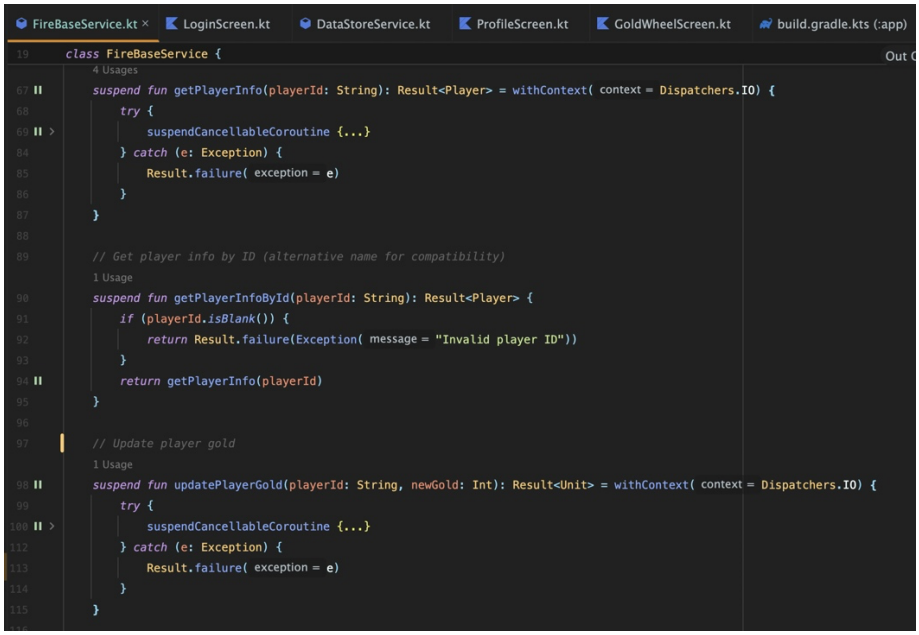


Threading

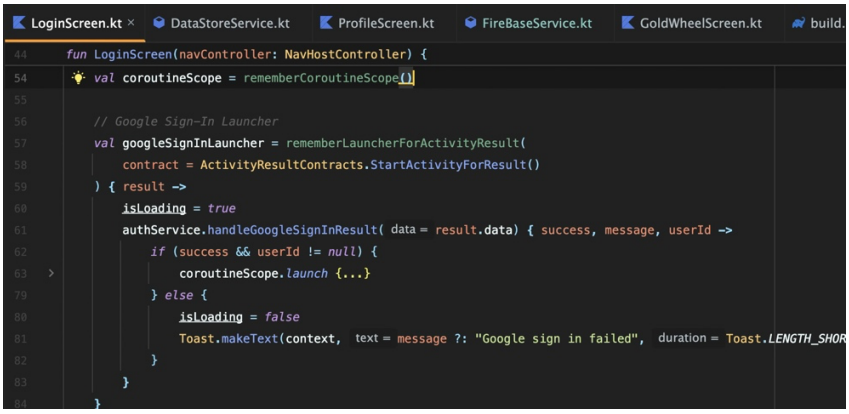
The game utilize the threading functionality using coroutine functions. Mainly used for separating the service works from the main UI thread. Mainly the firebase and datastore service uses the threading. All the firebase and datastore services methods are wrapped using “**suspend**” which automatically managed the threading and for UI uses. Additionally on UI layer uses “**coroutineScope”** on using the suspend functions**.**

Some of the thread safeties or aware used in all services class methods:

- **suspendCancellableCoroutine:** helps to properly bridges callback-based APIs to coroutines (Carrión, 2025).
- **withContext(Dispatcher.IO):** will ensure the threads are using IO thread pool and not blocking any CPU thread.



An **rememberCoroutineScope** is used to use all the suspended service functions or methods. As it bind with the composition point, and automatically kill or cancell any running coroutine threads the the composition ends or disposed (Sayed, 2025).



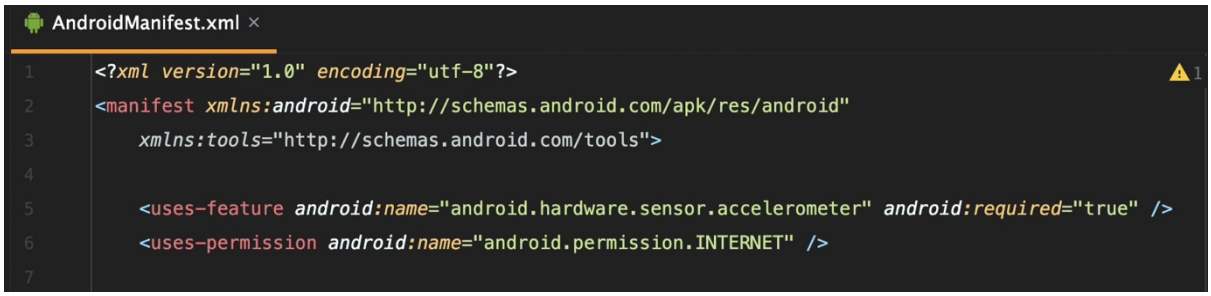
Permissions and License

All the app assets are sourced from icons8 and are free assets. No copyright issues and are licensed.

- Click here to view all icons8 asset licensing (Anon, n.d.).
- Click here to view all freepik asset licensing (Anon, n.d.).

Note: The image icons is AI generated (Google's Gemini) using nano banana model. Here is the link to chat prompt for generating the same click here to view.

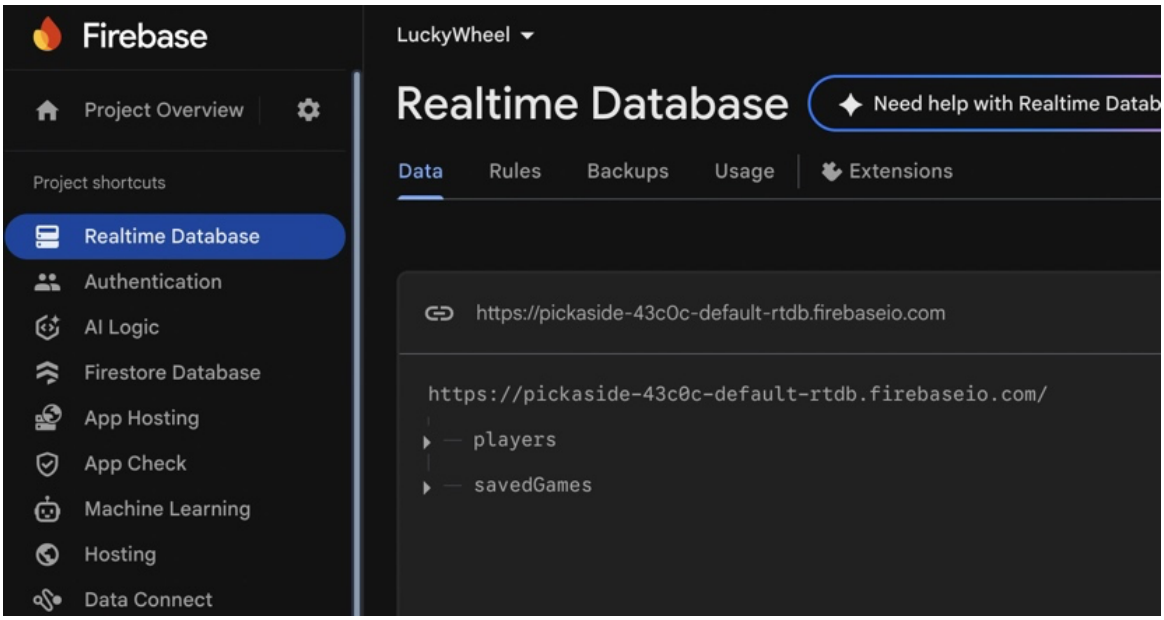
Declared the app uses the accelerometer mobile hardware sensor and permission for using the network in the ***AndroidManifest.xml***, and it is required to run the application.



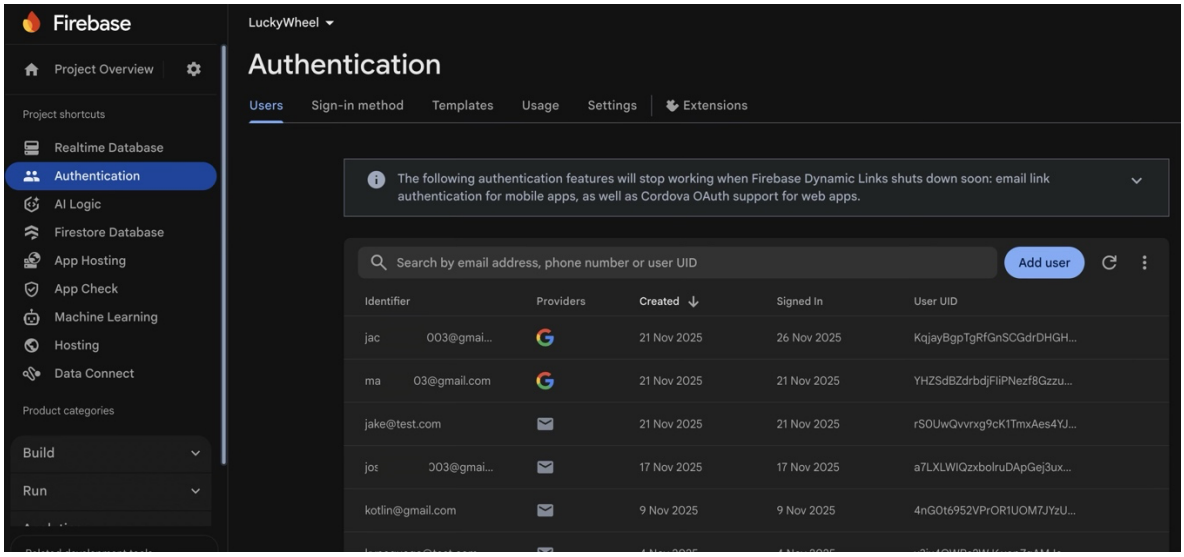
Backend Management

Currently, Firebase is primarily used as a backend service. This includes Firebase authentication for secure use, registration, and logins. Also, the application uses Firebase real-time database to fetch and store player info and gold count in real time with no delay.

For scalability the it stores player data on **players** collection and the each player's associated saved custom game on **savedGames** collection. Where both collection has associated player id. So that if needed scalability and has to migration from a firebase database to other database like PostgreSQL, make super easier with existing schema.



Also uses the data store preferences (Anon, n.d.) for storing the basic player data if they logged in. It is used for the auto login feature, so that the user doesn't need to log in again after closing or pausing the application. It checks if there is any preference stored and redirects to the playground screen. If the player logs out, then all the datastore preferences are removed.



References

App Architecture: Data Layer - DataStore. *Android Developers*. Available at: https://developer.android.com/topic/libraries/architecture/datastore (Accessed: 8 November 2025a).

Carrión, I. (2025) *Converting Callbacks to Coroutines and Flows in Kotlin*. *Carrion.dev*. Available at: https://carrion.dev/en/posts/callback-to-flow-conversion/ (Accessed: 26 November 2025).

Inter. *Google Fonts*. Available at: https://fonts.google.com/specimen/Inter (Accessed: 5 November 2025b).

Knewave. *Google Fonts*. Available at: https://fonts.google.com/specimen/Knewave (Accessed: 5 November 2025c).

Merienda. *Google Fonts*. Available at: https://fonts.google.com/specimen/Merienda (Accessed: 5 November 2025d).

Sayed, R. (2025) *Understanding rememberCoroutineScope in Jetpack Compose: Principles and Best Practices*. *Medium*. Available at: https://medium.com/@ramadan123sayed/understanding-remembercoroutinescope-in-jetpack-compose-principles-and-best-practices-a2d9033eb17a (Accessed: 26 November 2025).

Space Grotesk. *Google Fonts*. Available at: https://fonts.google.com/specimen/Space+Grotesk (Accessed: 5 November 2025e).

Terms of Use. *Freepik*. Available at: https://www.freepik.com/legal/terms-of-use (Accessed: 27 November 2025f).

Universal Multimedia License Agreement for Icons8 | Icons8 Help Center. Available at: https://intercom.help/icons8-7fb7577e8170/en/articles/5534926-universal-multimedia-license-agreement-for-icons8 (Accessed: 27 November 2025g).