

TABLE OF CONTENT

Table of Contents

<i>Project Details</i>	2
<i>Main Goal</i>	2
<i>Objective: Milestone 1</i>	2
<i>Project Architecture</i>	3
<i>App UI Theme</i>	3
Colour Theme	4
Font Theme.....	4
<i>App UI Design</i>	5
Authentication Screens	5
Playground Screens.....	6
Gold Wheel Section	6
Custom Wheel Section	7
Leaderboard Screen	8
Settings Screen (In progress)	8
<i>Permissions and License</i>	9
<i>Backend Services</i>	9
<i>References</i>	10

LUCKY WHEEL

Mobile Development – Milestone 1

Project Details

Project Name	Lucky Wheel
Sensor's Used	Accelerometer
Firebase Services	Authentication (email and password), Real-time Database
Data Preferences	Data Store (preference)
Fonts (Google Fonts)	Inter, Knewave, Merienda, Space Grotesk

Main Goal

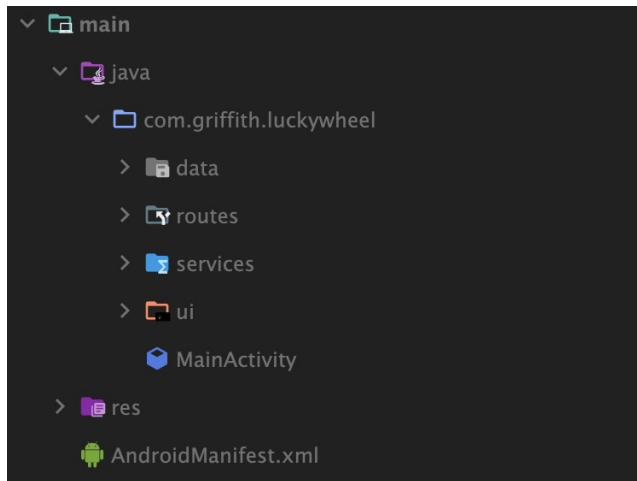
Lucky Wheel is a gaming app. The main goal of the app is to have a competitive gameplay for collecting the gold by spinning a wheel using a mobile sensor. The spinning wheel will land the pointer on a random choice (arch) on the spinning wheel. The player can earn or lose gold by spinning the lucky wheel. The app shows Real-time data ranking for players with high gold count.

Objective: Milestone 1

- Build a user authentication and registration system 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.

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.

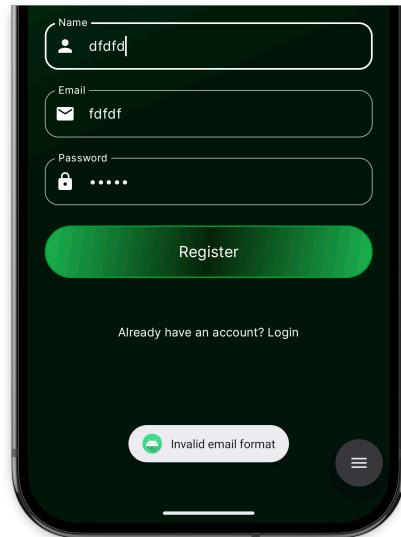
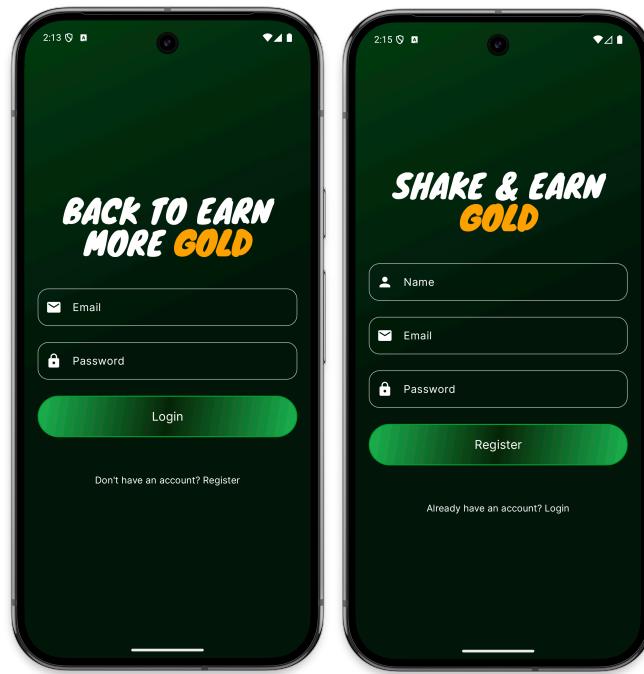
For decorative titles and special headers, the app uses the **Knewave** (Anon, n.d.) font, while instructional text within the playground area is styled with **Merienda** (Anon, n.d.). These fonts are all sourced from Google Fonts.

App 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.



Playground Screens

Inside the playground, the player could switch between the gold game and the custom player spinning wheel game.



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.



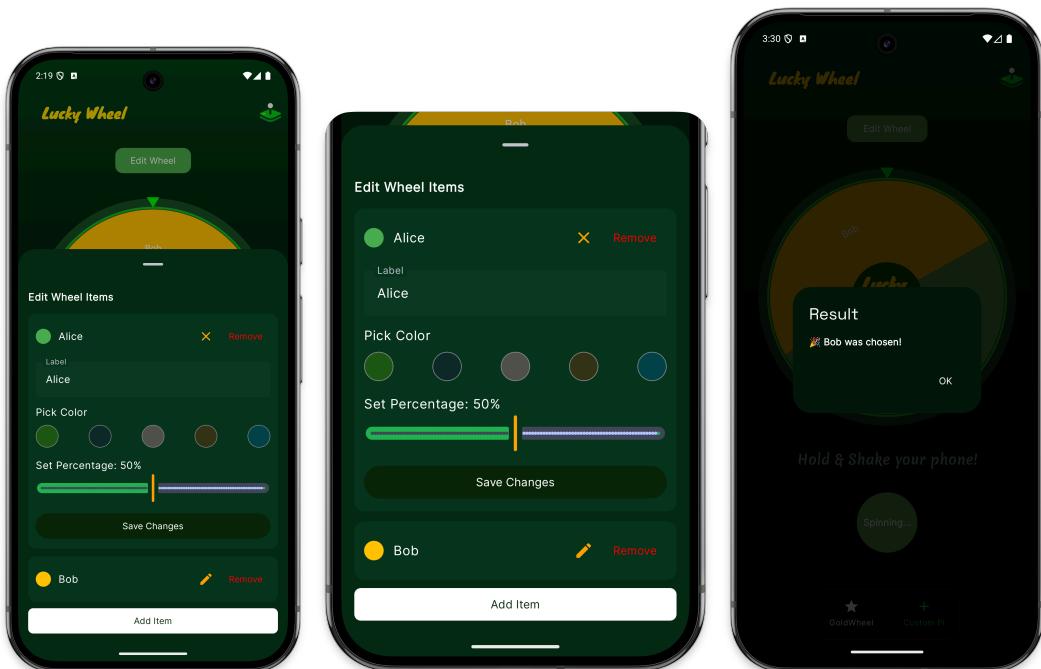
Joshan John
3092883

Custom Wheel Section

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



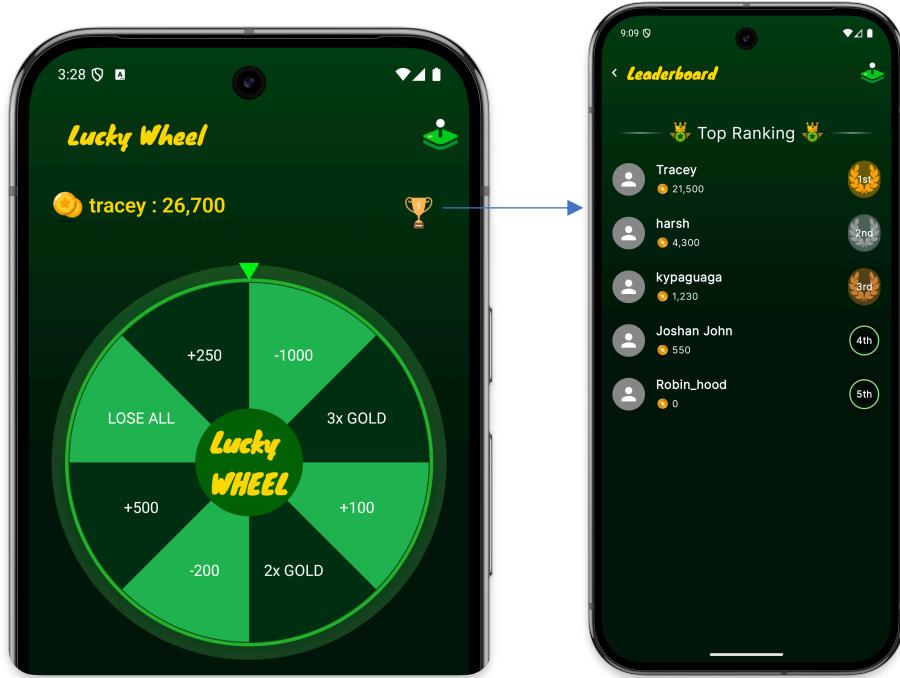
On clicking the “**Edit Wheel**” button, the user can add a new item to the pie and adjust the percentage of the pie. The user can also rename the label, edit the color. If needed user can also remove the item from the pie.



Joshan John
3092883

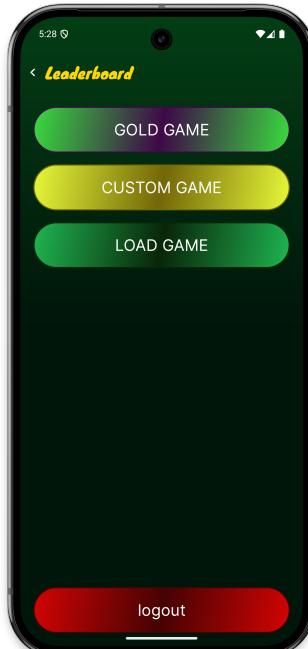
Leaderboard Screen

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 (In progress)



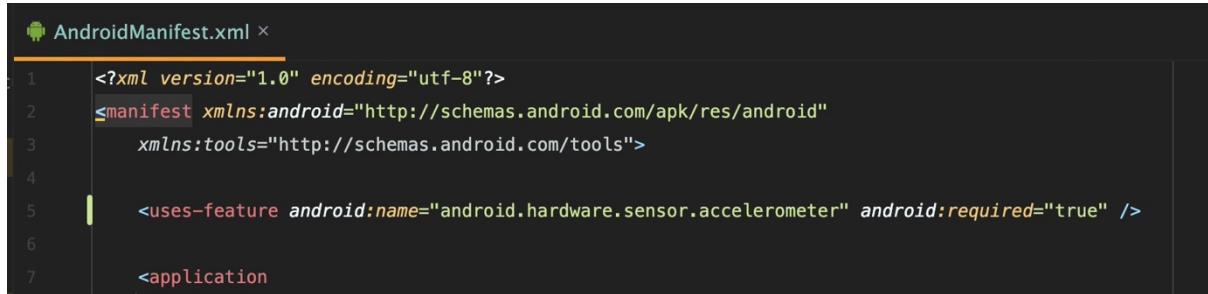
Joshan John
3092883

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](#)

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



```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools">

    <uses-feature android:name="android.hardware.sensor.accelerometer" android:required="true" />

    <application>
```

Backend Services

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.

The screenshot shows the Firebase Authentication console for the 'LuckyWheel' project. The left sidebar includes 'Project Overview', 'Realtime Database', 'Authentication' (which is selected), 'AI Logic', and 'Firestore Database'. The main area is titled 'Authentication' with tabs for 'Users', 'Sign-in method', 'Templates', 'Usage', 'Settings', and 'Extensions'. A message at the top states: 'The following authentication features will stop working when Firebase Dynamic Links shuts down soon: email link authentication for mobile apps, as well as Cordova OAuth support for web apps.' Below this is a search bar and a table of users:

Identifier	Providers	Created	Signed In	User UID
kypaguaga@test.com	✉️	4 Nov 2025	4 Nov 2025	x3ix40WBs2WJ6xepZgAMJeK...
robin@test.com	✉️	4 Nov 2025	4 Nov 2025	rvnHDrcC14PAVrOKPxUS93ZM...
test@test.com	✉️	4 Nov 2025	6 Nov 2025	m7ivzKlnsnPHcFyc4JBHBVwd...
joshan@test.com	✉️	4 Nov 2025	4 Nov 2025	vu1dkqEVD60I0bZPjVyYncTj9...
harshpatil02072002@gmail.com	✉️	4 Nov 2025	4 Nov 2025	InL6eONciIZqFXXsPXJxUmaV...

Buttons for 'Add user' and 'Rows per page' (50) are visible.

The screenshot shows the Firebase Realtime Database console for the 'LuckyWheel' project. The left sidebar includes 'Project Overview', 'Realtime Database' (selected), 'Authentication', 'AI Logic', and 'Firestore Database'. The main area is titled 'Realtime Database' with tabs for 'Data', 'Rules', 'Backups', 'Usage', and 'Extensions'. A message at the top says: 'Protect your Realtime Database resources from abuse, such as billing fraud or phish'. Below this is a preview of the database structure:

```

https://pickaside-43c0c-default-rtbd.firebaseio.com/
  players
    lnL6eONciIZqFXXsPXJxUmaVCeP2
    m7ivzKlnsnPHcFyc4JBHBVwdF8C2
      gold: 64600
      playerId: "m7ivzKlnsnPHcFyc4JBHBVwdF8C2"
      playerName: "Tracey"
    rvnHDrcC14PAVrOKPxUS93ZMEJv1
    vu1dkqEVD60I0bZPjVyYncTj9Uy2
    x3ix40WBs2WJ6xepZgAMJeKI6u72
  
```

References

Inter. Google Fonts. Available at: <https://fonts.google.com/specimen/Inter> (Accessed: 5 November 2025a).

Knewave. Google Fonts. Available at: <https://fonts.google.com/specimen/Knewave> (Accessed: 5 November 2025b).

Joshan John
3092883

Merienda. Google Fonts. Available at: <https://fonts.google.com/specimen/Merienda> (Accessed: 5 November 2025c).

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