Pub Golf App

The app creates a route of 9 different locations in Portsmouth and a set of 9 challenges is created, these are stored in the app. The app currently stores 15 different locations and 14 different challenges. The user adds and removes players from the scorecard and adds / edits individual player scores, these are all stored by the app. Lastly, the user is able to create a new game which will clear the scorecard, route and challenges. The app will then create a new route and set of challenges for the user.

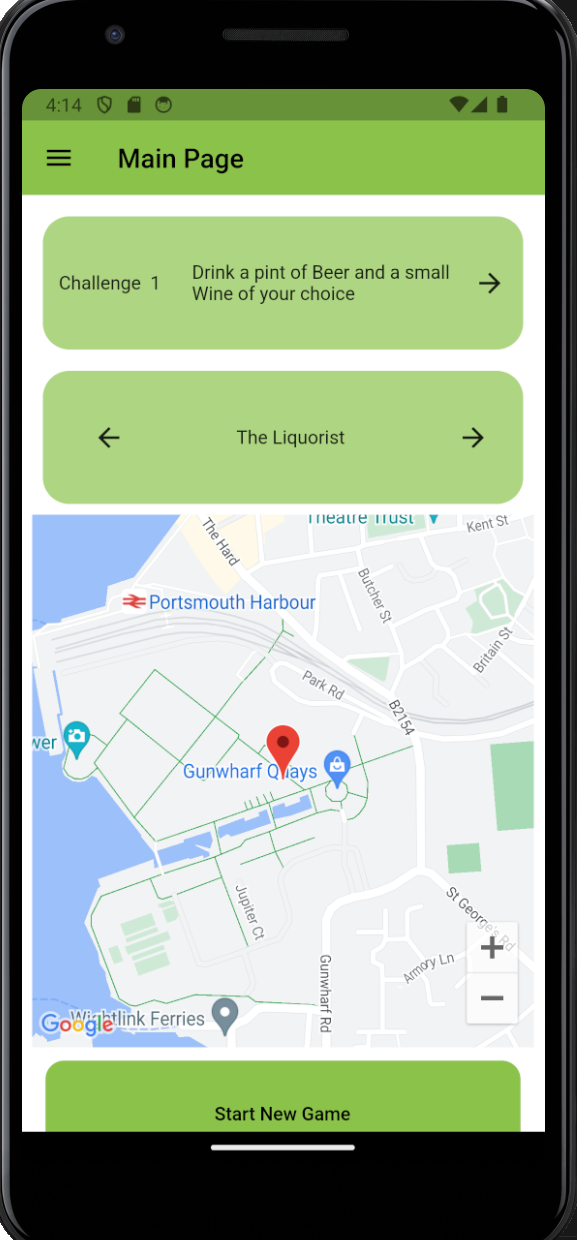
The App uses the following technologies:

Language: Dart

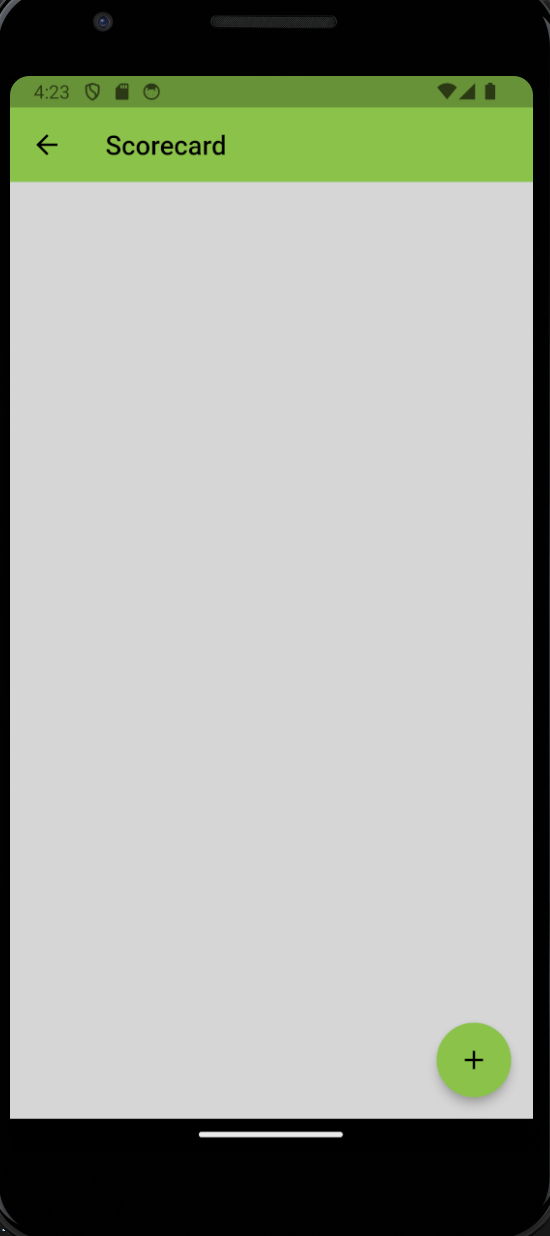
Libraries / Packages:

1. Flutter
2. Hive
3. Google Maps Flutter
4. Flutter Slidable

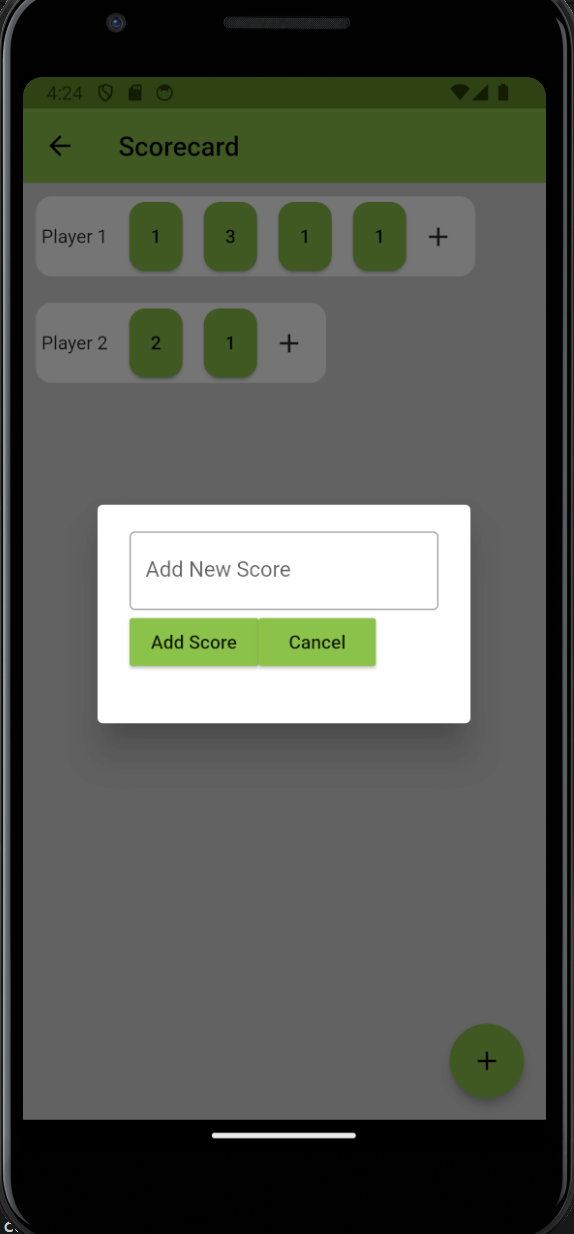
Screenshots:



* App Bar: Displays the Page name and Navigation Drawer
* Challenge Container: Displays the challenge number, the current challenge and an icon button to go to the next challenge
* Location Container: Displays the current location and two icon buttons to navigate to the next location or previous location
* Google maps container shows the current location
* New Game button: the user presses the button to create a new game.

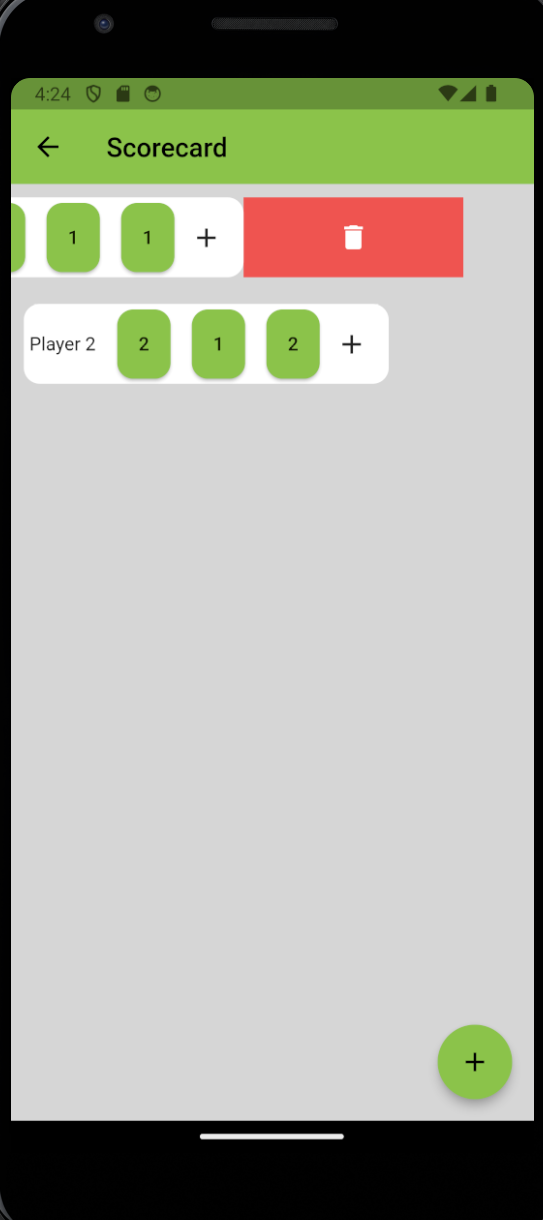


* Scorecard Page
* This is the initial display when the user first loads up the app
* Arrow in the App Bar used to navigate to the Main page
* Floating action button used to add new players

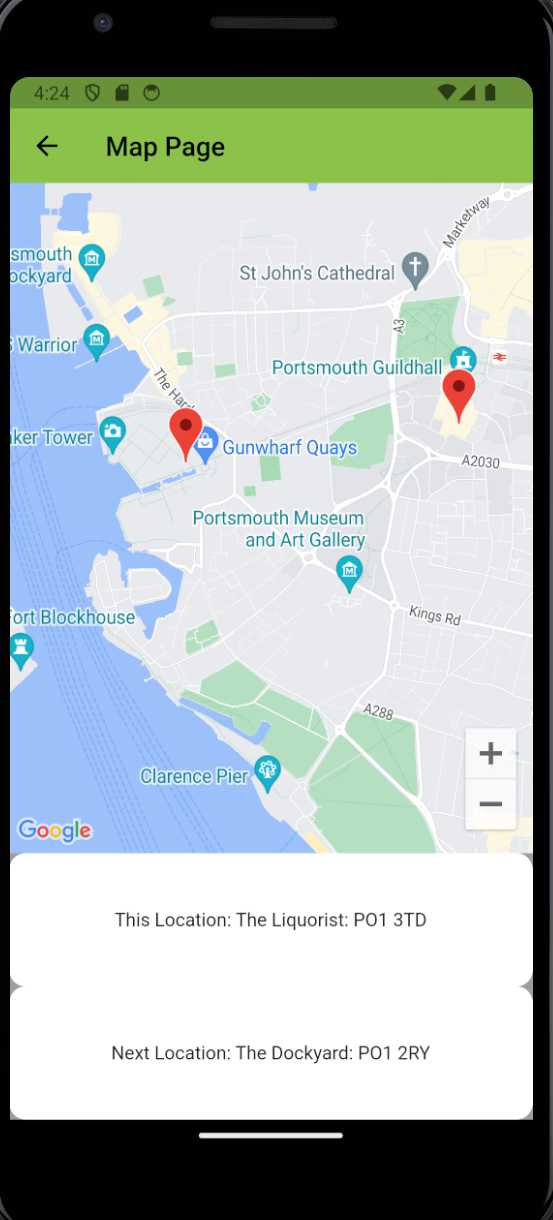


* Container shows each players name and scores for each location
* Add Button icon is used to add new scores
* Each score can be pressed, this allows the user to edit each individual score.

* The user can add / edit the score using the dialog box and save it, this will add / update the score to the correct player. This is the same dialog box used when adding new players.



* Each Container that stores the players name and score has a horizontal scroll, this means the user can look at all the scores stored
* Each container is slidable, this allows the user to delete a player from the scorecard.



* App bar displays the page name and navigation arrow to the main page

* Google maps inside a container allows the users to see a bigger map than on the main page.
* The map uses markers to show the current location and the next location
* Container shows the current location and its corresponding postcode
* Container shows the next location and its corresponding postcode, unless the route is finished. The container displays the message “End of Route”

Functions and Explanations

Main Page:

InitState() - initialises the app when it is first loaded up

nextChallenge() - increments the challengeIndex and updates the state

setLocations() - sets the current location and next location

getNextLocation() - increments the location index, calls the set location function and updates the state

getLastLocation() - decrements the location index, calls the set location function and updates the state

createNewGame() - creates a new game, resets the index pointers and calls the set locations function

Scorecard Page:

addNewPlayer() - adds a player via a dialog box where the user inputs a name

savePlayer() - saves the name from the users input and updates the state of the app

removePlayer() - removes the selected player from the scorecard and updates the state

addNewScore() - adds a new score to the player from a user input via a dialog box

editScore() - allows the user to edit a score via an input on a dialog box

saveEditedScore() - saves the edited score from the users input and updates the state

saveScore() - saves the score from the users input and updates the state

Database:

createNewGame() - creates a new game by clearing the scorecard data and calling the createNewRoute and createNewChallengeSet methods

getData() - gets all the data stored in the database

setData() - saves all the data stored in the database

createNewRoute() - creates a new route by removing the old route and storing the new route in the database

createNewChallengeSet() - creates a new set of challenges by removing the old challenges and storing the new challenges in the database.