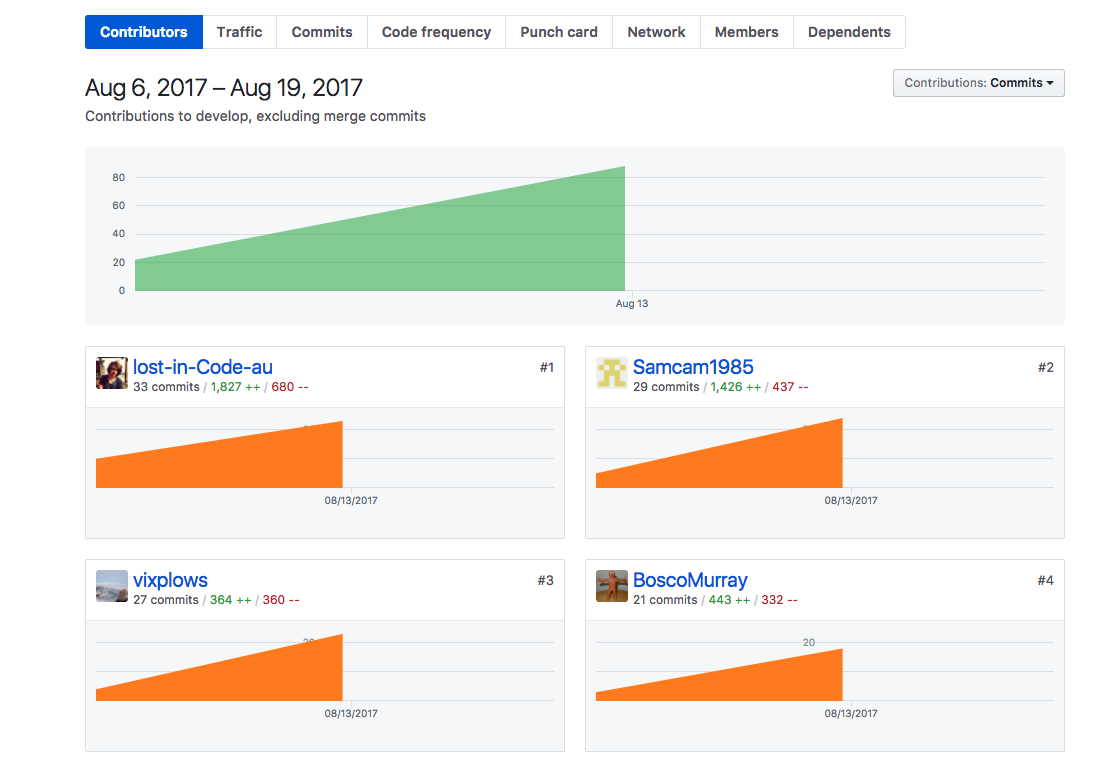
**Evidence for Project Unit**

Victoria Plows

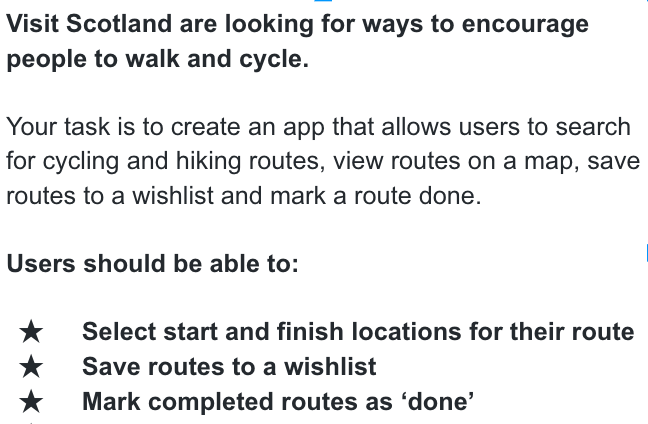
E13

23/06/2017

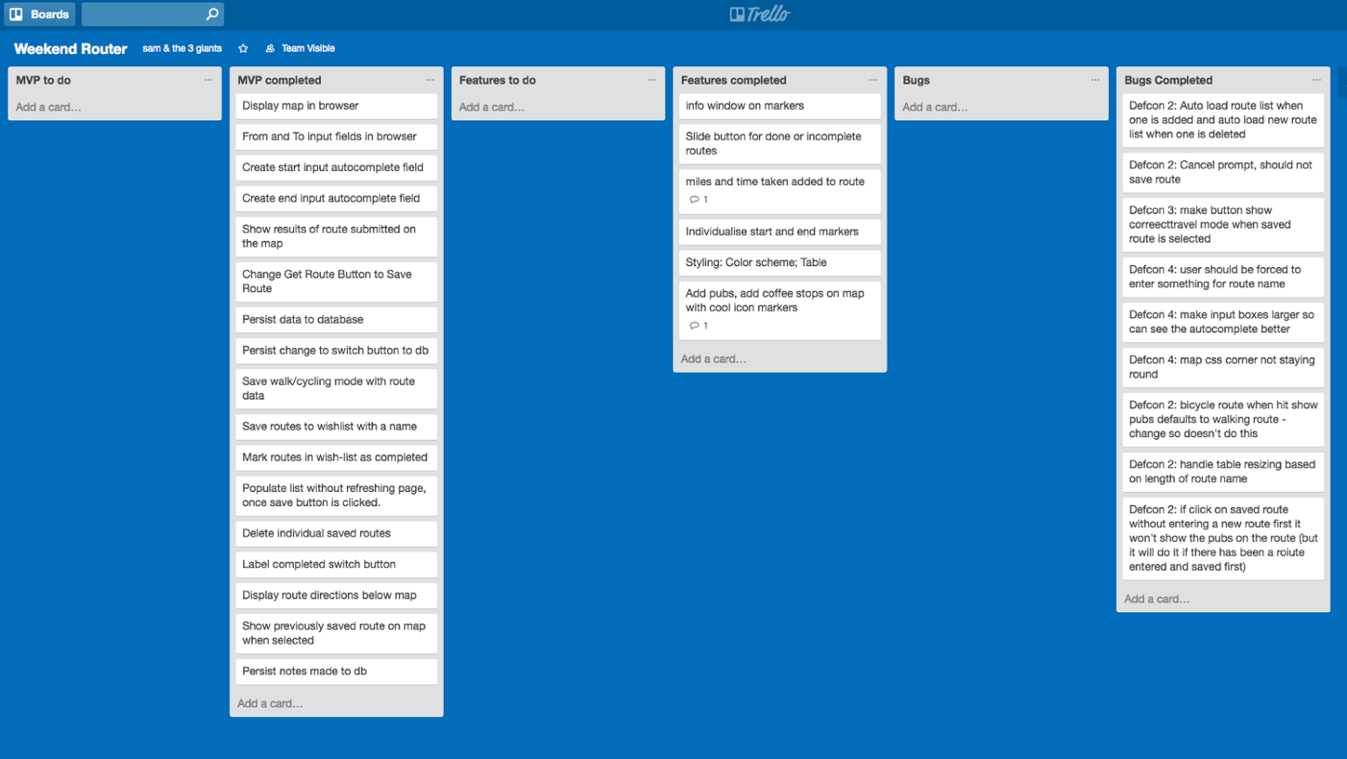
**P- 1 Github Contributors page**



**P- 2 Project Brief**



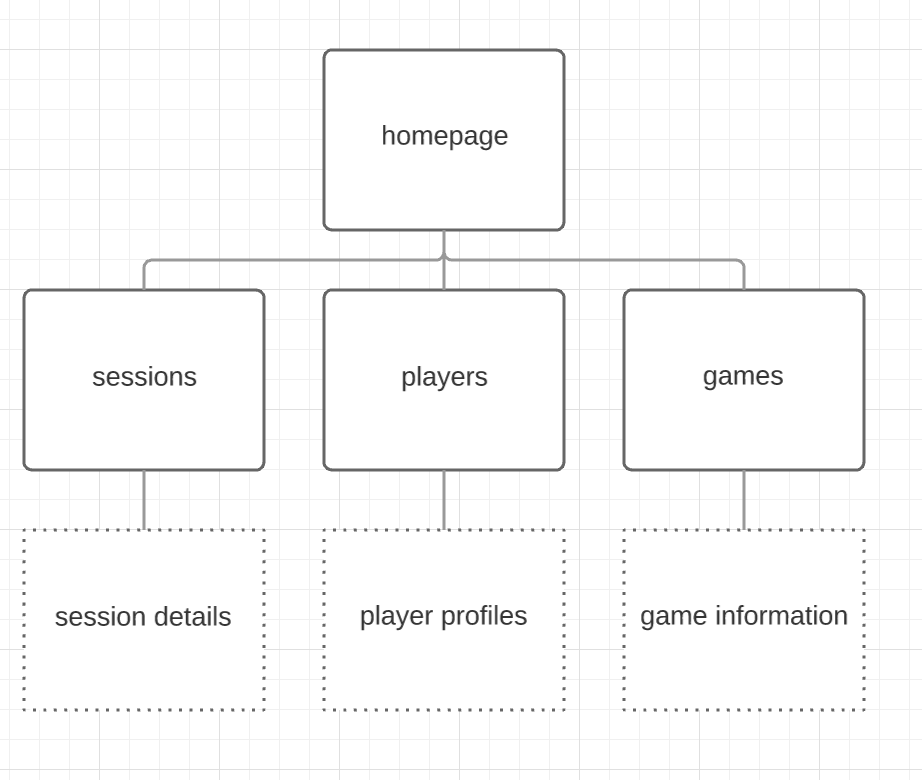
**P-3 Use of Trello**



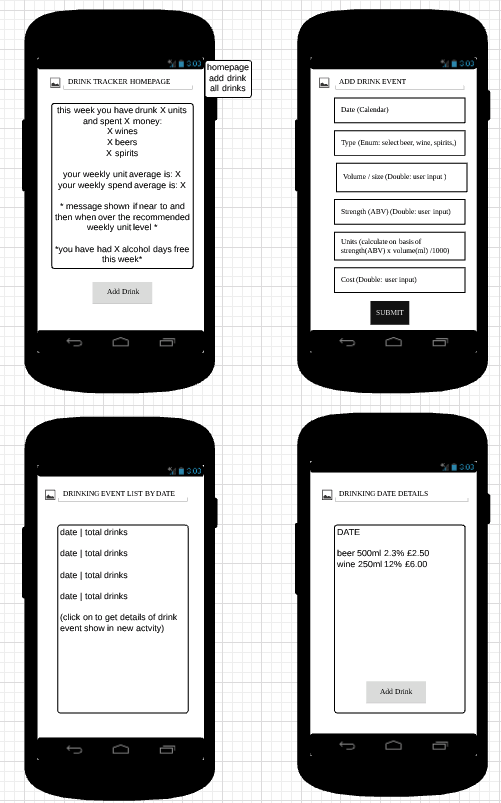
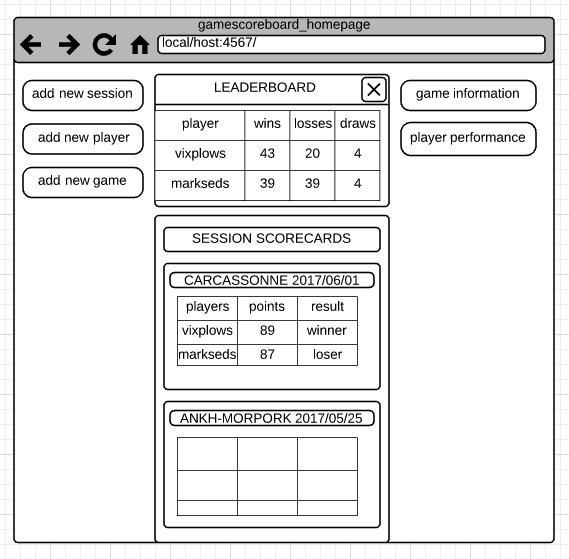
**P-4 Acceptance Criteria**

|  |  |  |
| --- | --- | --- |
| **Acceptance Criteria** | **Expected Result** | **Pass/Fail** |
| **User can sign up to register for application** | When a user enters an email address and password to sign up for an account they will see a pop-up confirming they have registered. | Pass |
| **User can sign in and out of application** | When a user signs in using their email address and password that will be able to view their account. When a user signs out clicking the sign out button they will no longer see their account until they sign in again. | Pass |
| **User can navigate from homepage to view own player profile** | When a user clicks on the show player profile button they will be shown their player profile page, | Pass |
| **User can navigate from homepage or own profile page to view player profiles of other users** | When a user clicks on the view players button they will see the list of other players. | Pass |
| **User can select other users to follow** | When a user selects another player to follow by ticking the box next to the players name that player will be saved and shown in their players following list. | Pass |
| **User can see list of users following** | When a user clicks on the players I follow button a list of the players they are following will be displayed. | Pass |
| **User can see list of users that are following them** | When a user clicks on show players that are following me button a list of player that are following them will be displayed. | Pass |
| **User can edit and save profile information** | When a user clicks on the edit profile button they will be able to change their information and when they hit save this will be saved to the database and their revised profile displayed to them. | Pass |

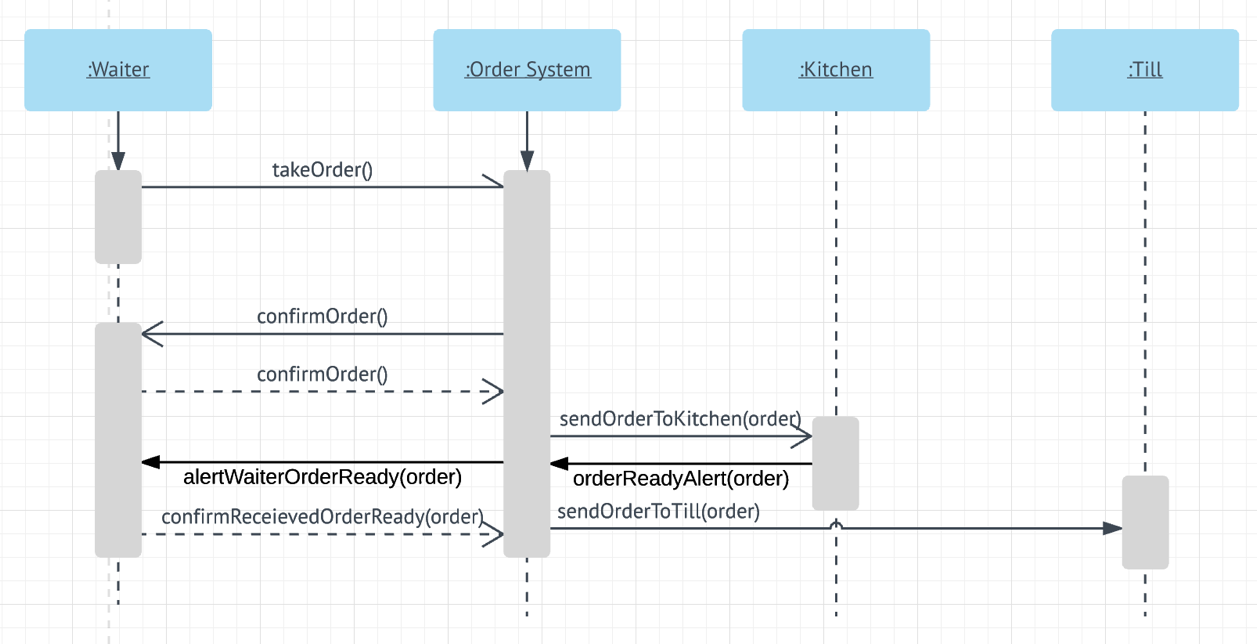
**P-5 User sitemap**

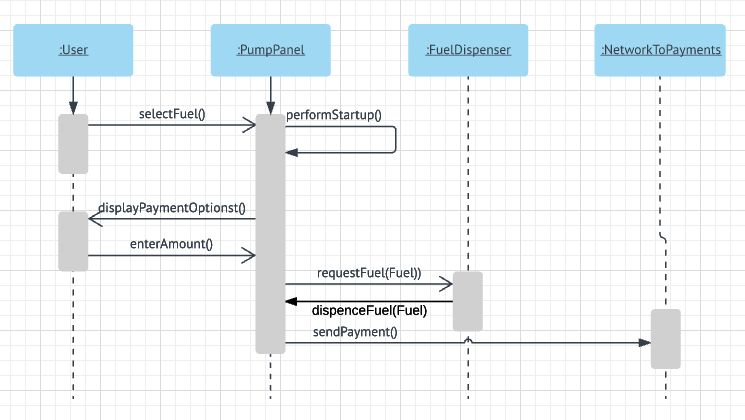
****

**P-6 Wireframes designs**

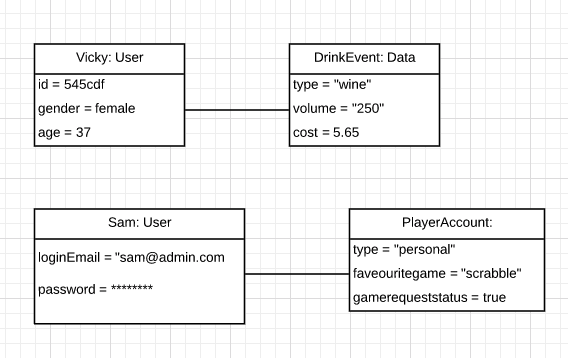
****

**P-7 System interactions diagrams**

****

****

**P-8 Two Object Diagrams**

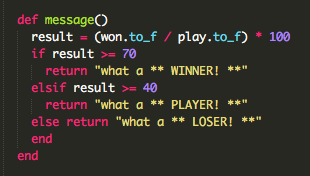
****

**P- 9 D.T.- a Choice of two algorithms (find the algorithms on a program you might have written, show the code you have used. )**

A - Algorithm- In my DrinkTracker Project I needed to be able to add a drink event to a user’s profile by date creating a new date if no drink events existed already or adding to existing drink events on a date if some already existed. To do this, I used an algorithm to search through the existing keys (dates) in the drinkEventHashMap and if the key (date) did not already exist a key, of that date, was added and the drink event added to an arraylist associated with that key (date). If the key (date) already existed the drink event was added to the existing arraylist for that key (date).

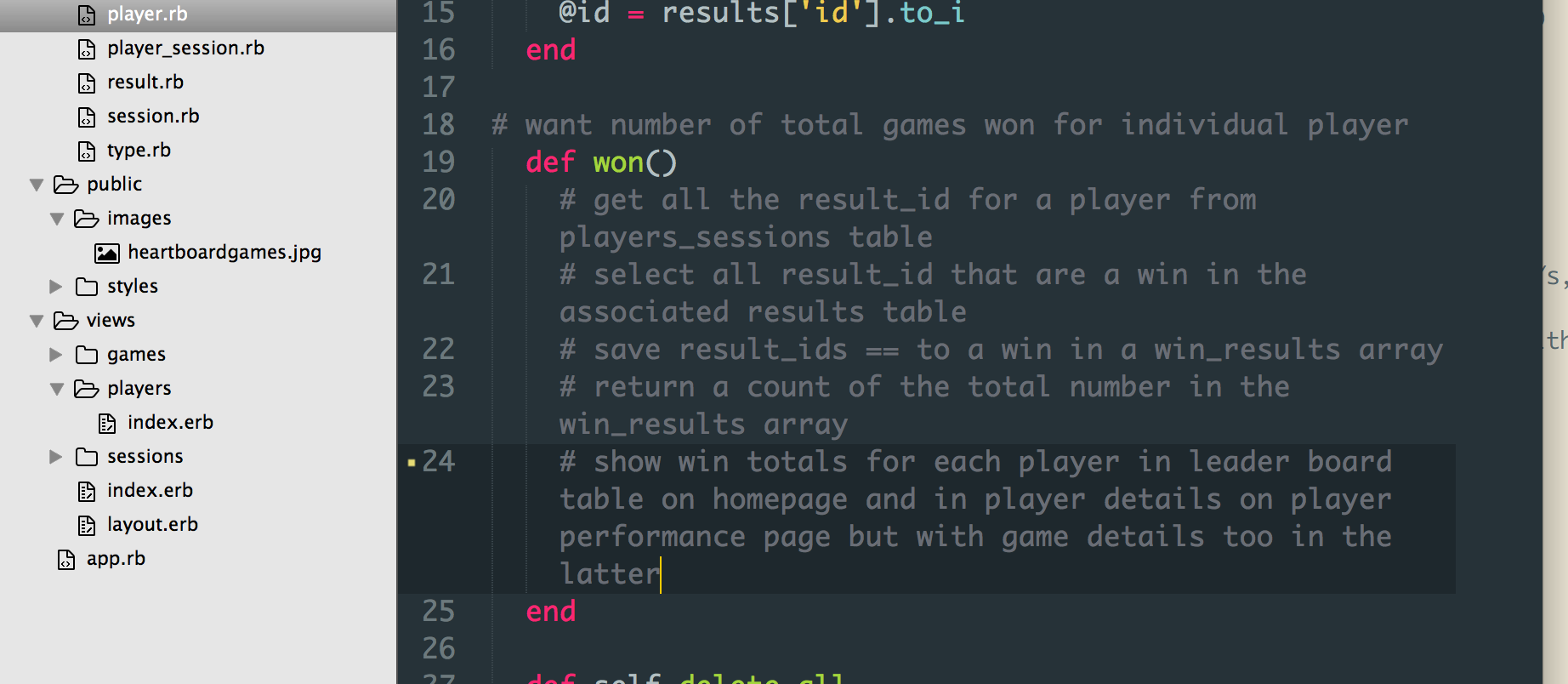
****

B - Algorithm - In my GameScoreBoard project, I needed to be able to display a message about the players dependent on the percentage of games they had won that updated automatically. The following algorithm enabled me to display a message on the player’s page that they were a “winner” if they had won 70% or more of the games they had played, if it was 40% or less that they were a “loser” and anything in between that they were a “player”:

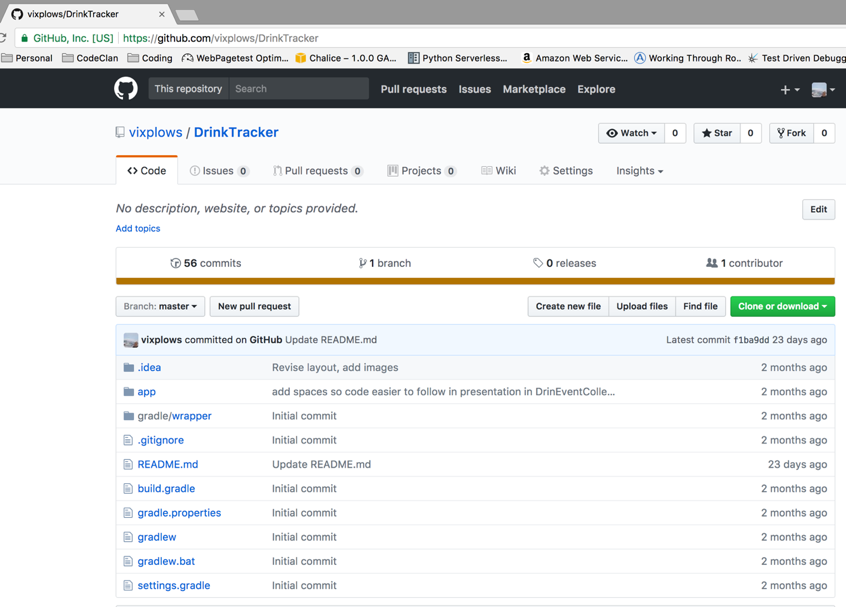
****

**../../Desktop/Screen%20Shot%202017-09-20%20at%2020.39.50.pn**

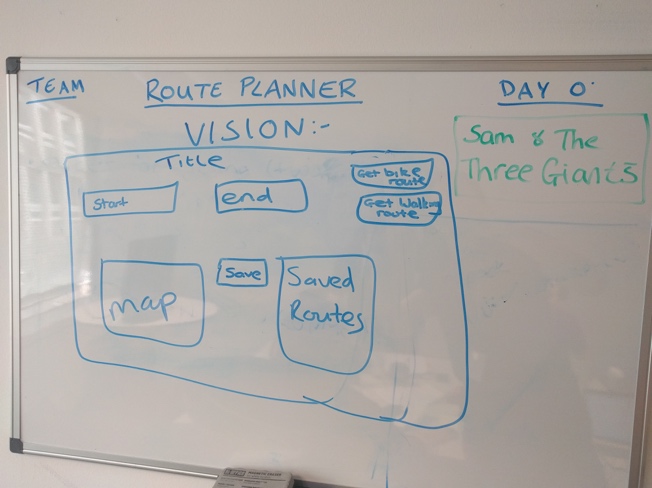
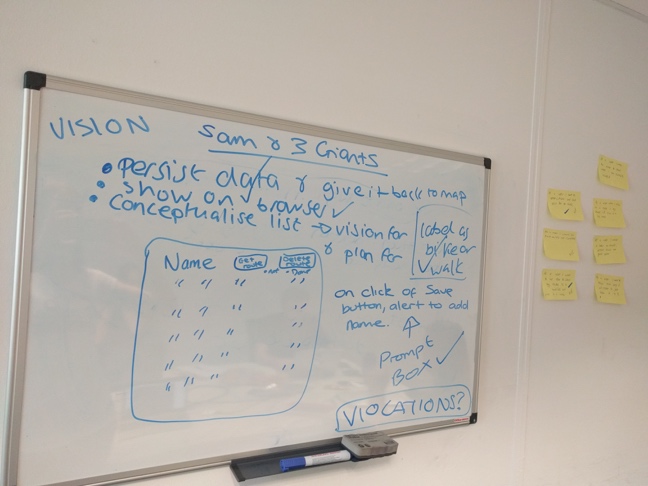
**P - 10 Example of Pseudocode**

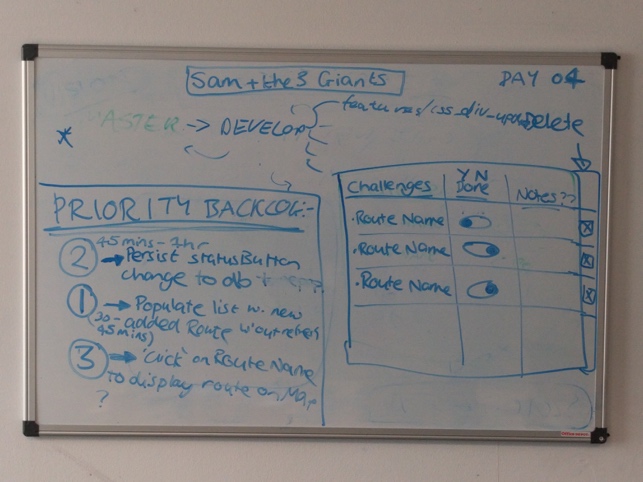
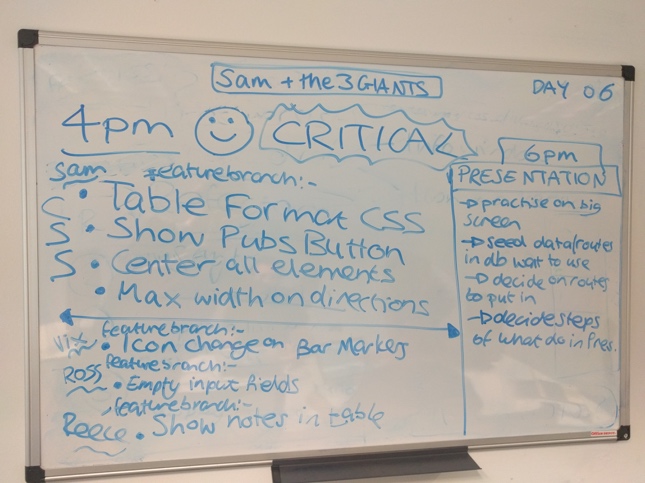
****

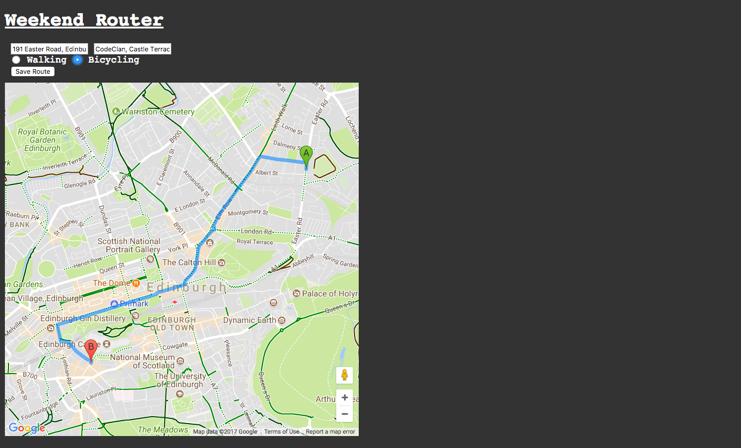
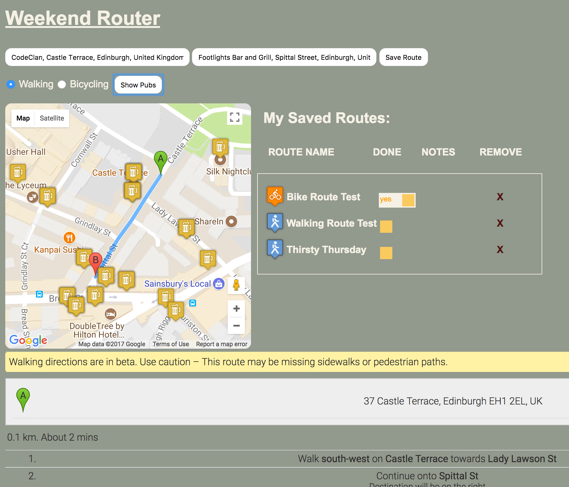
**P - 11 Github link to one of your projects**

****

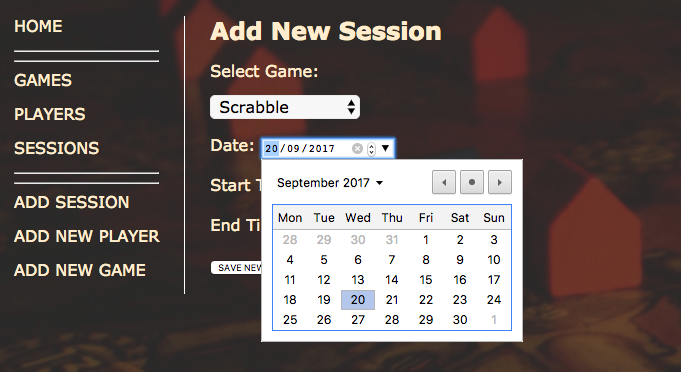
**P - 12 Screenshot of your planning and the different stages of development to show changes.**

** **

** **

** **

**P - 13 User input**

****

**P - 14 Interaction with data persistence**

****

**P - 15 User output result**

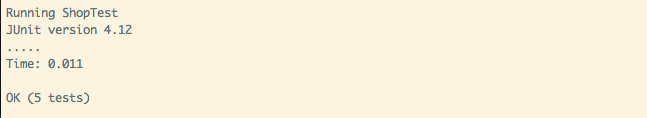
****

**P - 16 Bug tracking report showing the errors diagnosed and corrected.**

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
| **Shop has to be able to add stock** | **Failed** | **Create add stock method, enabling stock to be added** | **Passed** |
| **Shop has to be able to remove stock** | **Failed** | **Create remove stock method, enabling stock to be removed** | **Passed** |
| **Shop has name** |  |  | **Passed** |
| **Shop stock starts empty** |  |  | **Passed** |
| **Shop can get total potential profit calculated as pounds and pence** | **Failed** | **Changed integers to doubles for stock items buy and sell prices** | **Passed** |

**P -17 Testing your program**

****

****