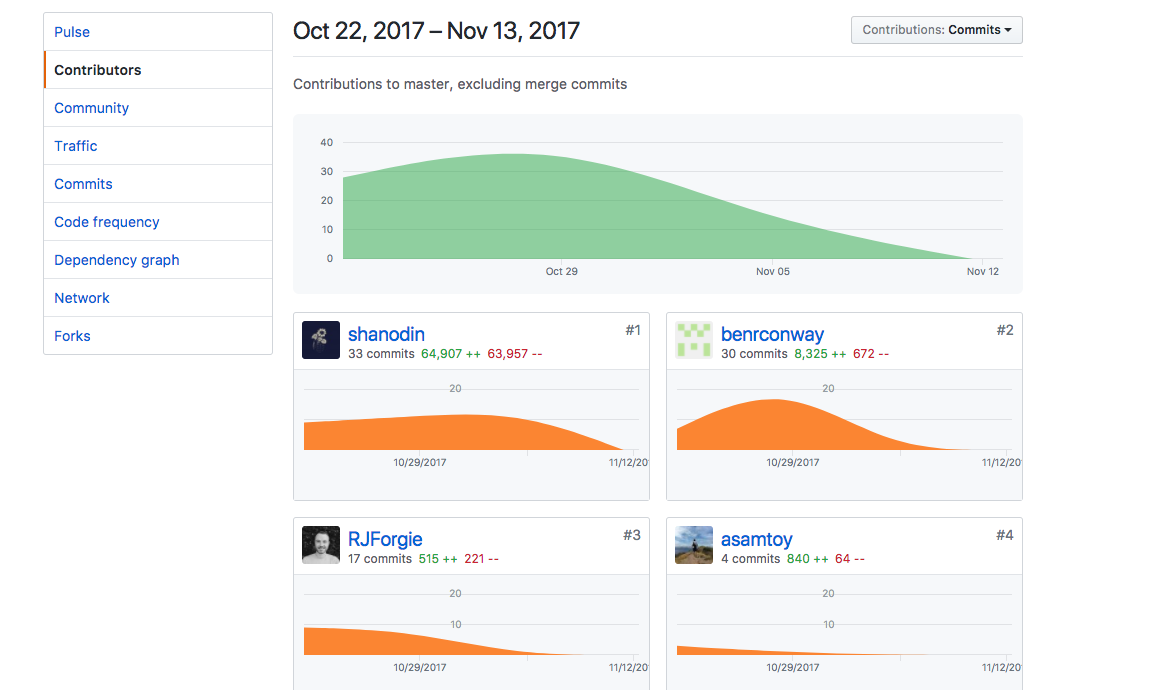
**Evidence for Project Unit**

Ryan Forgie

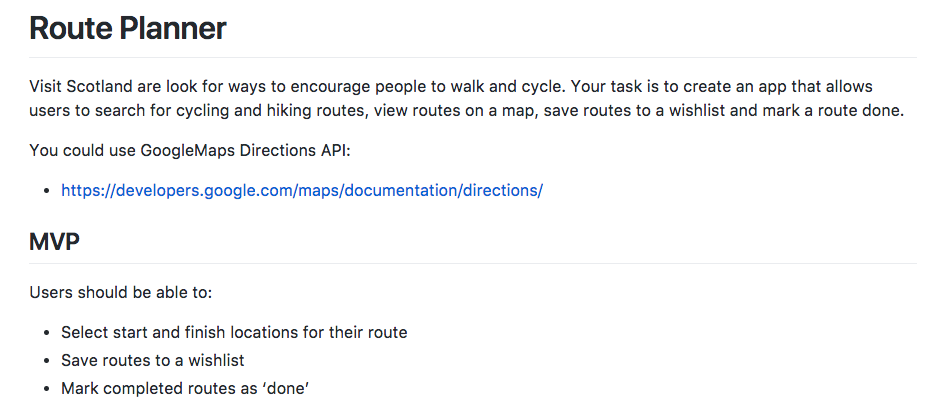
Cohort 15

Date here

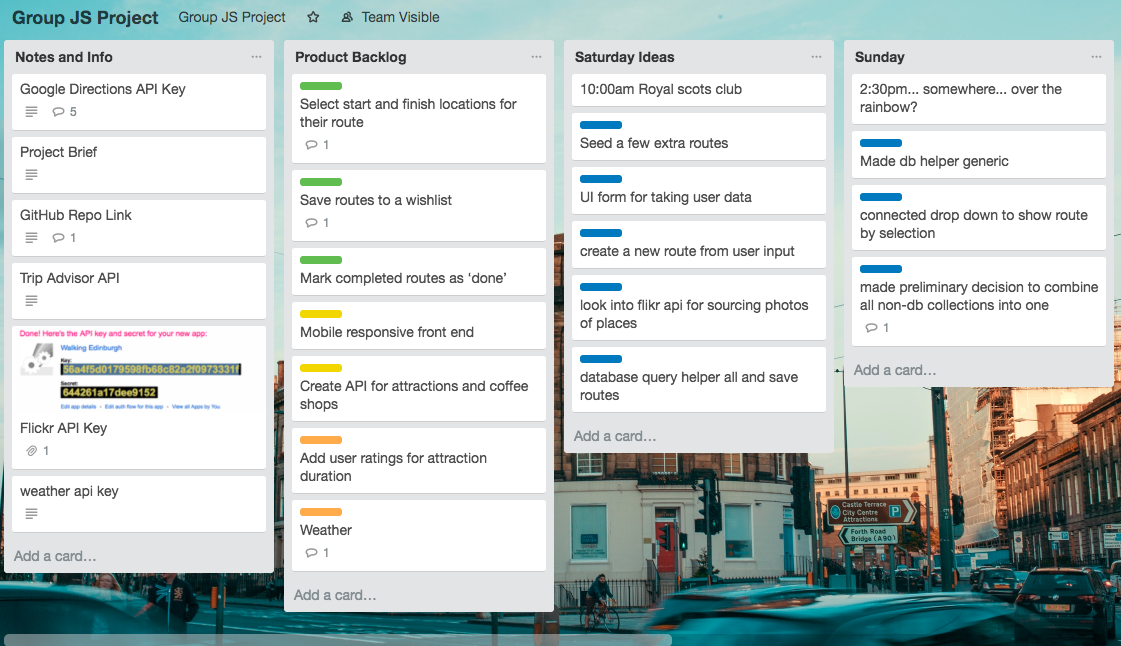
**P- 1 Github Contributors page**



**P- 2 Project Brief**



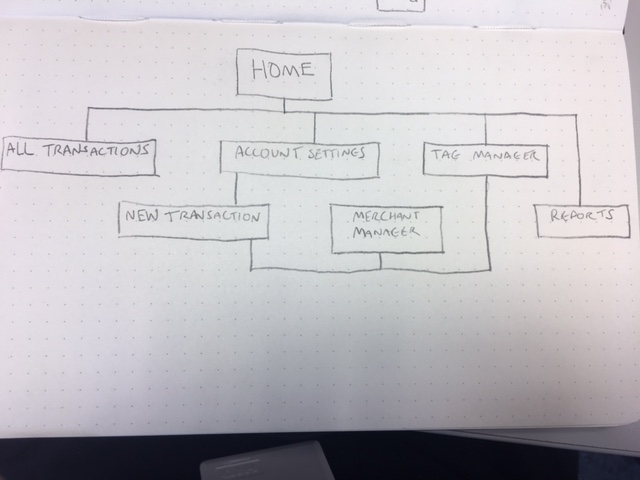
**P-3 Use of Trello**



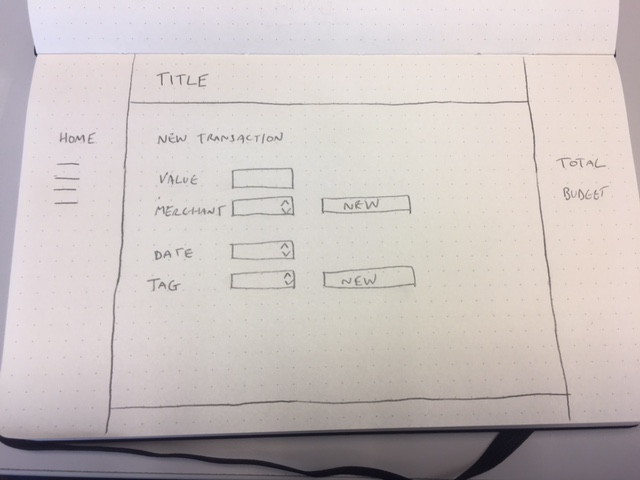
**P-4 Acceptance Criteria**

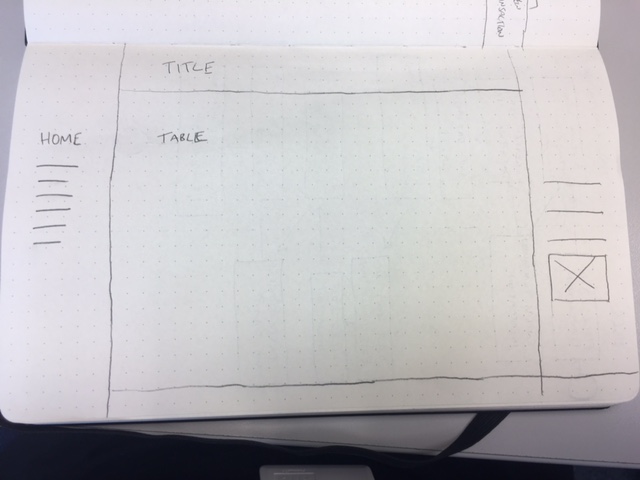
|  |  |  |
| --- | --- | --- |
| **Acceptance Criteria** | **Expected Result** | **Pass/Fail** |
| **User can enter a start and finish location for their route** | **When user enters route start and finish into autocomplete test boxes, these are assigned to new route** | **Pass** |
| **User can save routes for later viewing** | **User can create a custom route then click a save button and the route is added to the routes dropdown** | **Pass** |
| **User can click a marker and see more information on that attraction** | **User clicks marker and the name of that attraction is displayed on top of map** | **Pass** |
| **User can see today’s local weather** | **Today’s weather in Edinburgh is displayed to user in text and icon form** | **Pass** |
| **User can view photos of each attraction** | **Marker info windows will display a photo of the attraction pulled from Flickr when user clicks map marker** | **Pass** |

**P-5 User sitemap**

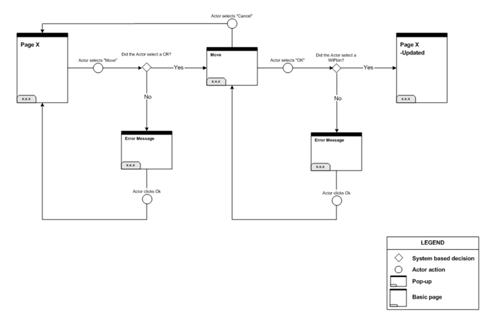
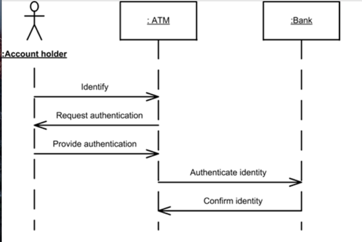
****

**P-6 Wireframes designs**

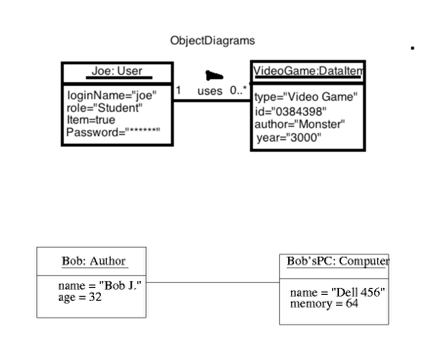
****

****

**P-7 System interactions diagrams**

****

**P-8 Two Object Diagrams**

****

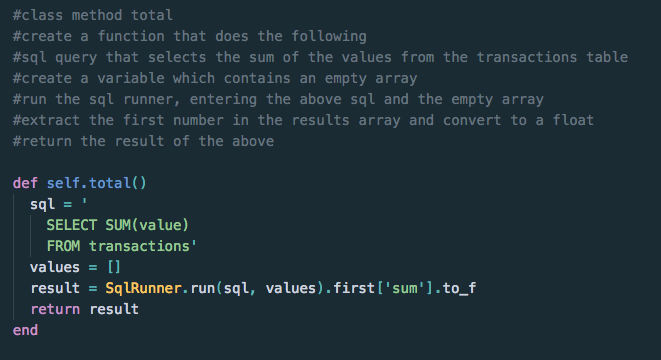
**P- 9 D.T.- a Choice of two algorithms (find the algorithms on a program you might have written, show the code you have used. )**

**On this example please take a screenshot and write what it is doing and why u decided to use it.**

**A - Search Algorithm- For one of the projects I carried out I had to find items in a warehouse, by bays and rows. The best way to do this was to use a search algorithm, where the items had an ID. I had passed the ID into the function and iterated through the items checking the ID I was looking for.**

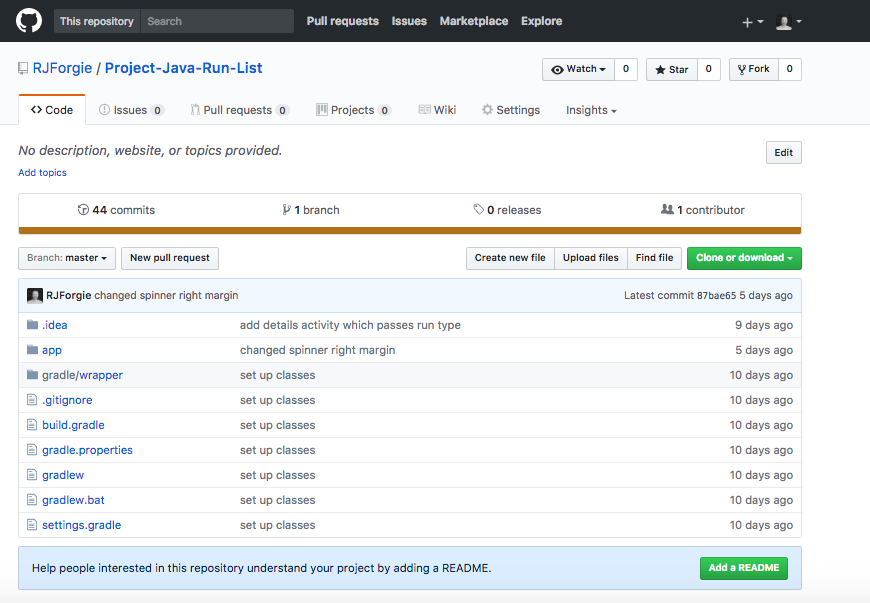
**B- Delete Algorithm - In the same project I had to delete items from the warehouse. The delete algorithm allowed me to go and find the item by ID and delete it from the array of items, in each bay.**

**P - 10 Example of Pseudocode**

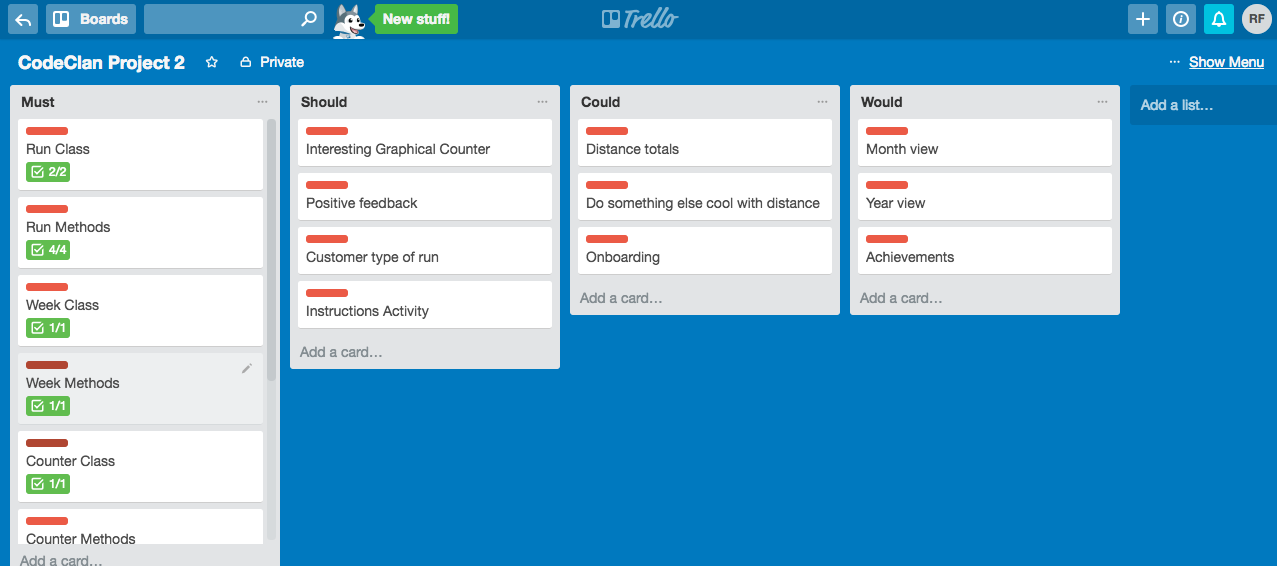
****

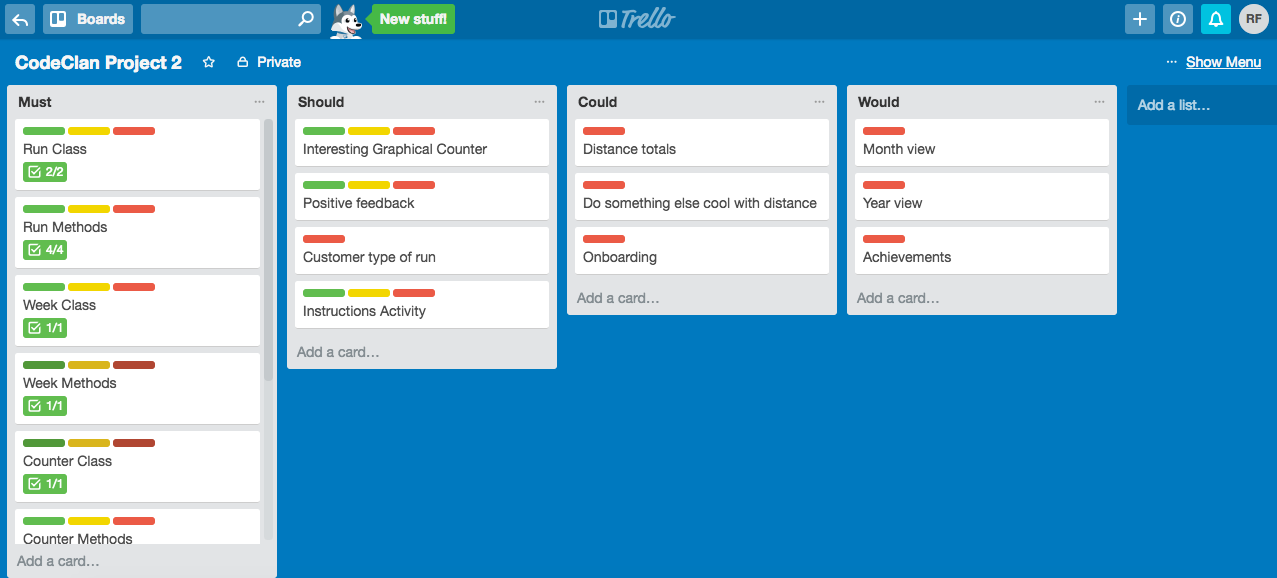
**P - 11 Github link to one of your projects**

[**https://github.com/RJForgie/Project-Java-Run-List**](https://github.com/RJForgie/Project-Java-Run-List)

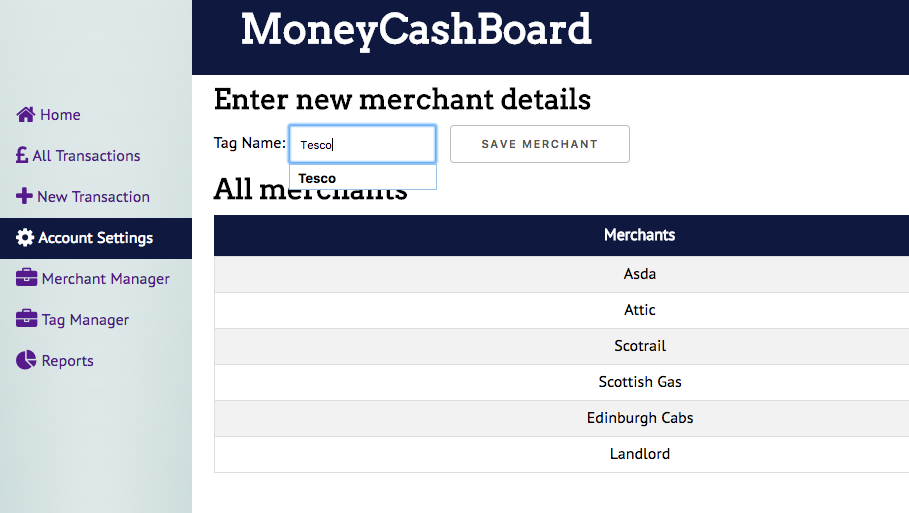
****

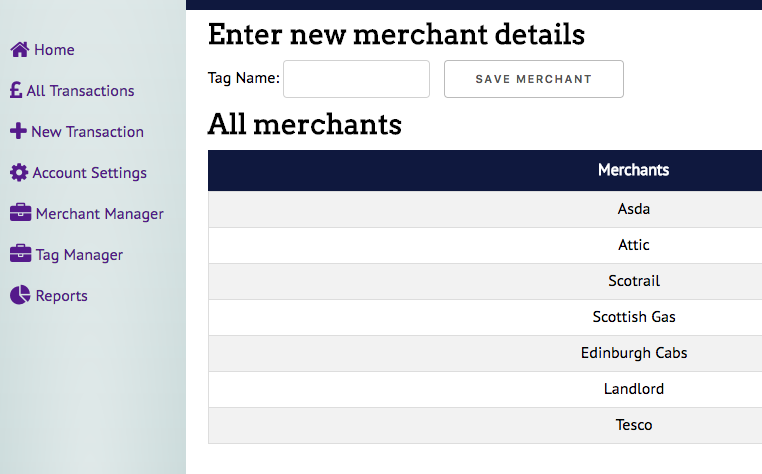
**P - 12 Screenshot of your planning and the different stages of development to show changes.**

****

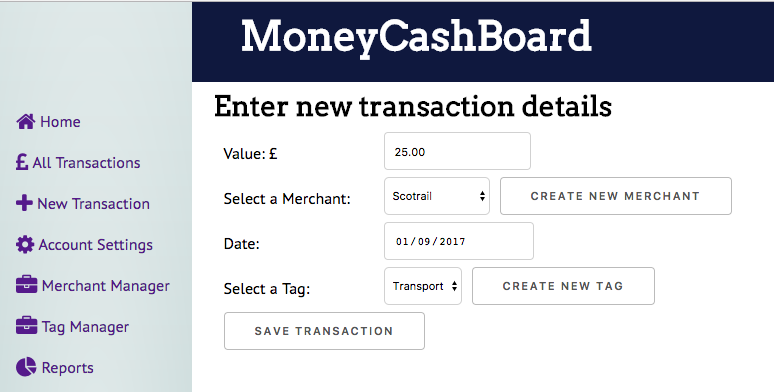
****

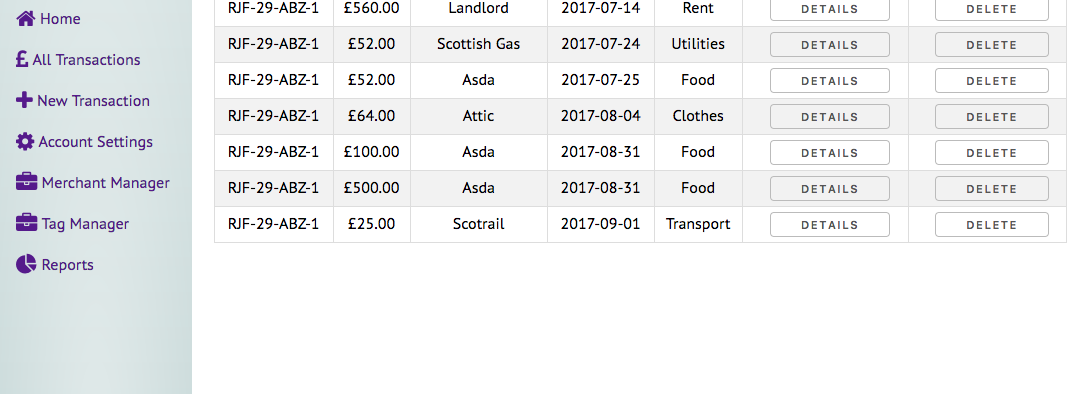
**P - 13 User input**

****

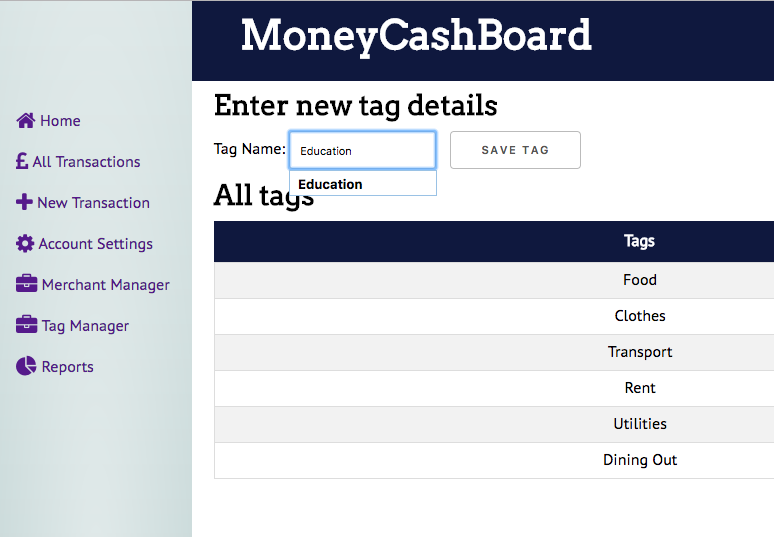
****

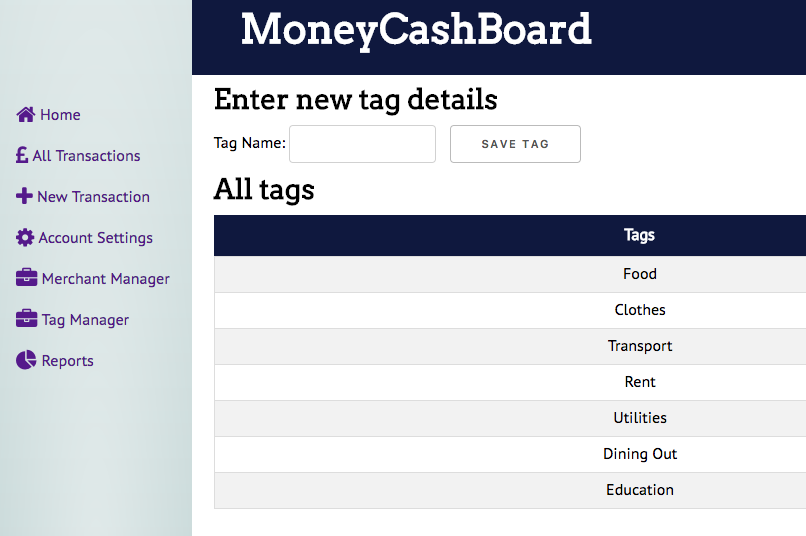
**P - 14 Interaction with data persistence**

****

****

**P - 15 User output result**

****

****

**P - 16 Use of API**

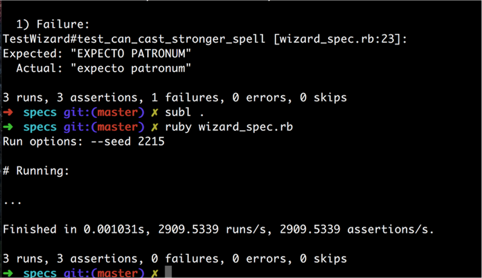
****

****

**P - 17 Bug tracking report showing the errors diagnosed and corrected.**

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
| **User must be able to add a trip** | **Failed** | **Saving a user, using the ID to assign a trip** | **Passed** |
| **Trip has a starting and end date** |  |  | **Passed** |
| **Trip date cannot be made for dates passed** | **Failed** | **Added validations to stop creation of trips with past dates** | **Passed** |
| **Trip can only have a number of available spaces** | **Failed** | **Set a number of spaces available per trip.** | **Passed** |

**P -18 Testing your program**



**Show the test not passing…..and then the test fixed.**