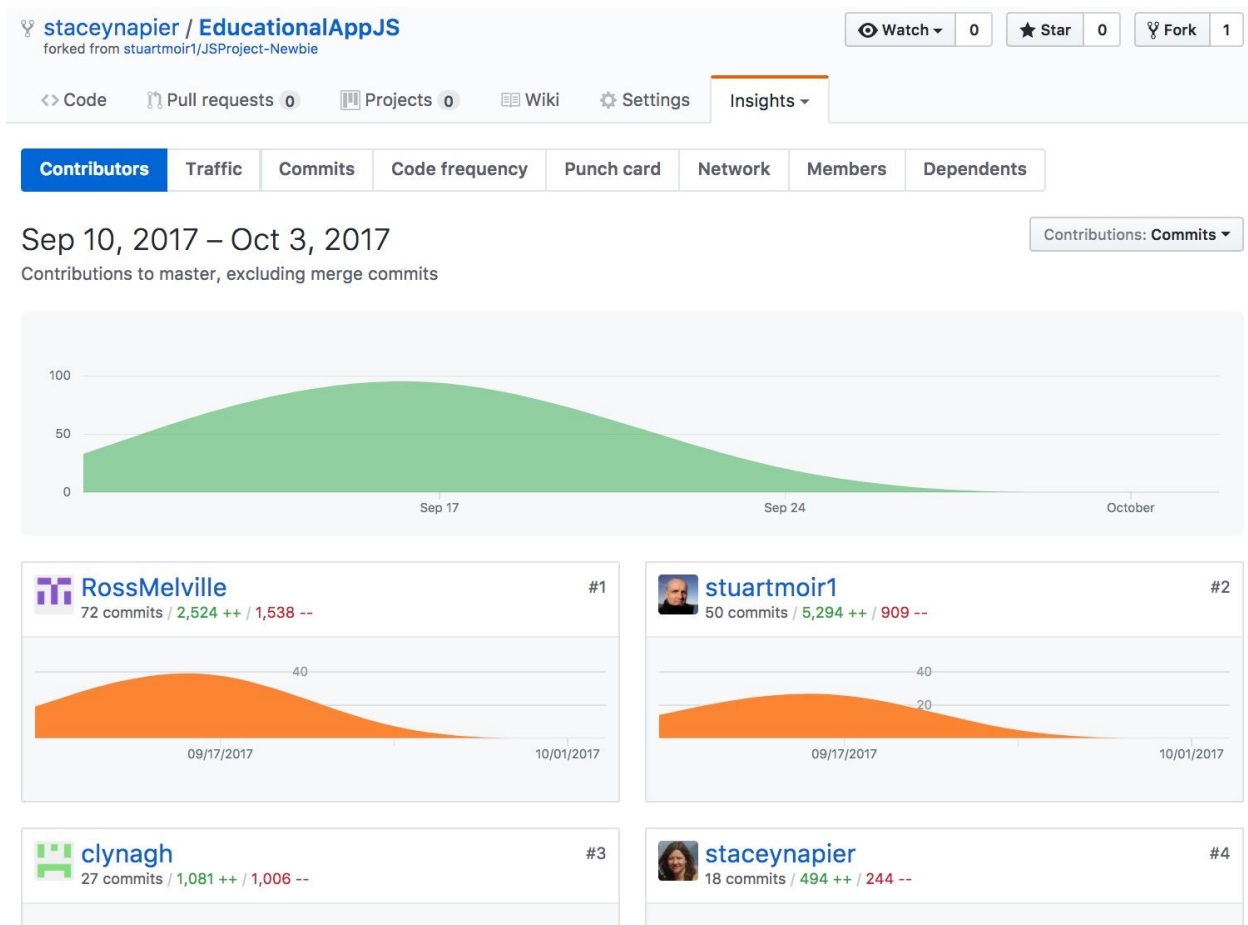


Stacey Napier Project Evidence

P1 - Group GitHub



P2 - Project Brief

Create an online educational tool which is fun and interactive which will assist the user in understanding programming principles.

The app should allow the user to search for keywords that they are looking to have a greater understanding of.

The search should provide the user with a definition of the subject and subsequent options to allow them to further their learning.

MVP

Ability to search db and get definition.

User activity by interactive options.

P3 - Planning

CodeCler E14 Javascript Project ☆ Private

Development Images

Flow

User Journeys

Completed Tasks

- Audible
- css refactor
- Description fade and return
- Random Button
- alert for wrong word entry and move input to lowercase
- Merge code (GitHub) and debug
- Definition key terms fade
- Hamburger Dropdown

Must

- User Journey
- Dropdown list for terms
- Implement design choices
- External API - Audio
- Explore the idea of a right/left arrow at the side of the tests or
- Additional information pop up
- HTML & CSS
- Stuarts search bar into answers of test
- Error messages if input is not in db

Could

- Tags
- Language/Topic index page
- Revisit the icons and links at the footer
- Make it flashier

Should

- Display recent searches
- Auto Complete Search
- External API - Audio

Would

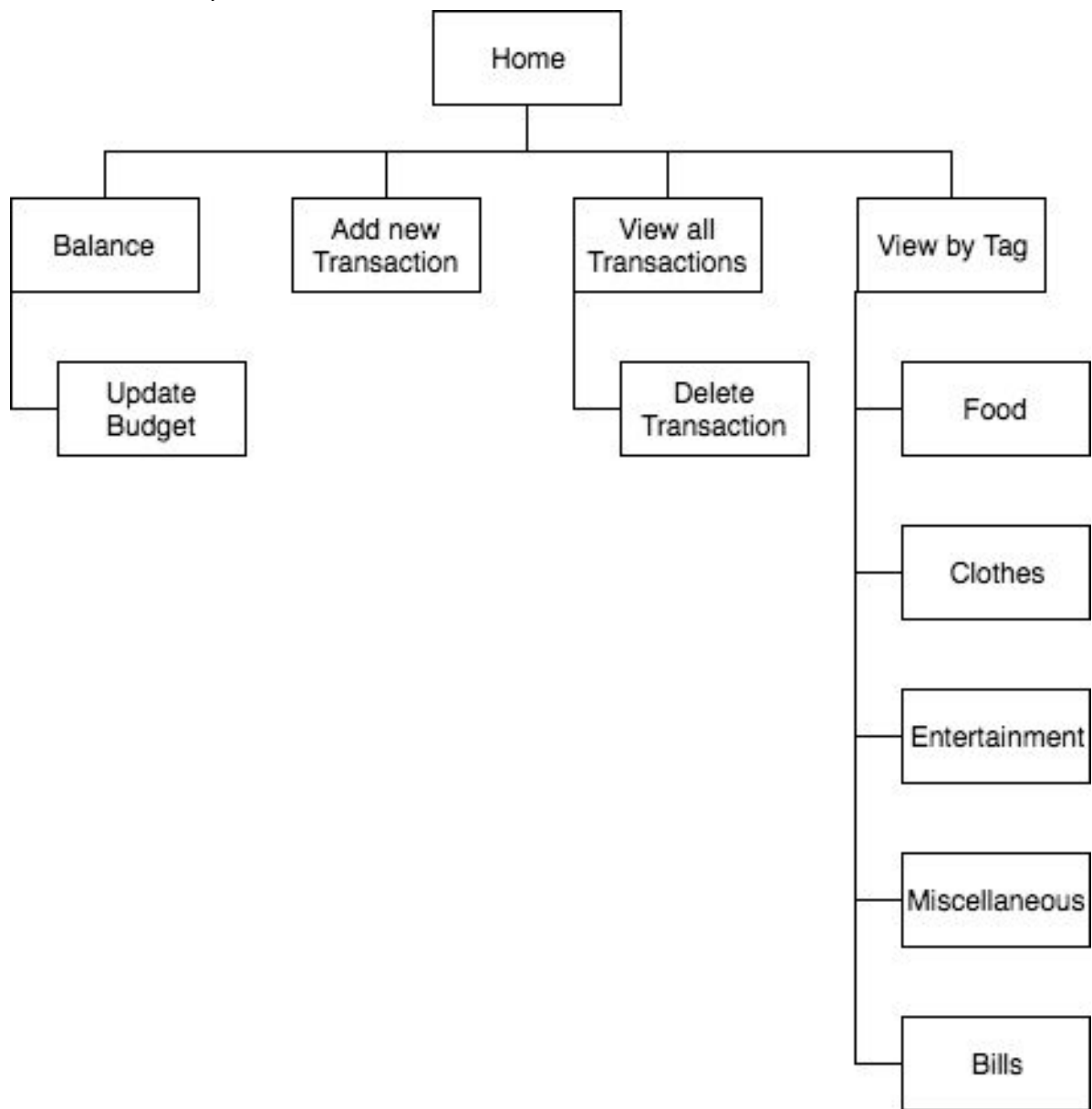
- Textua
- Access: 0/1

Add a card...

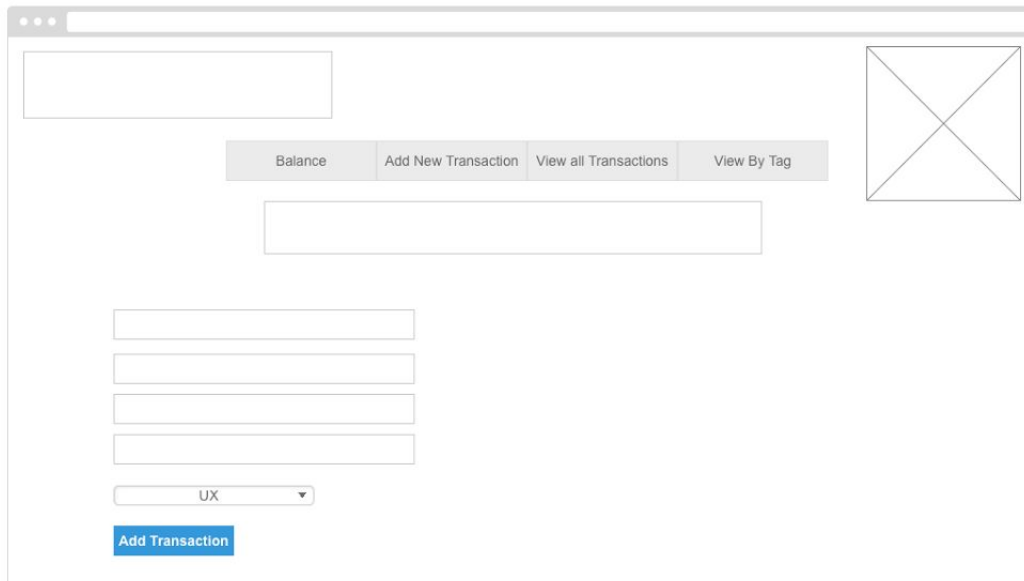
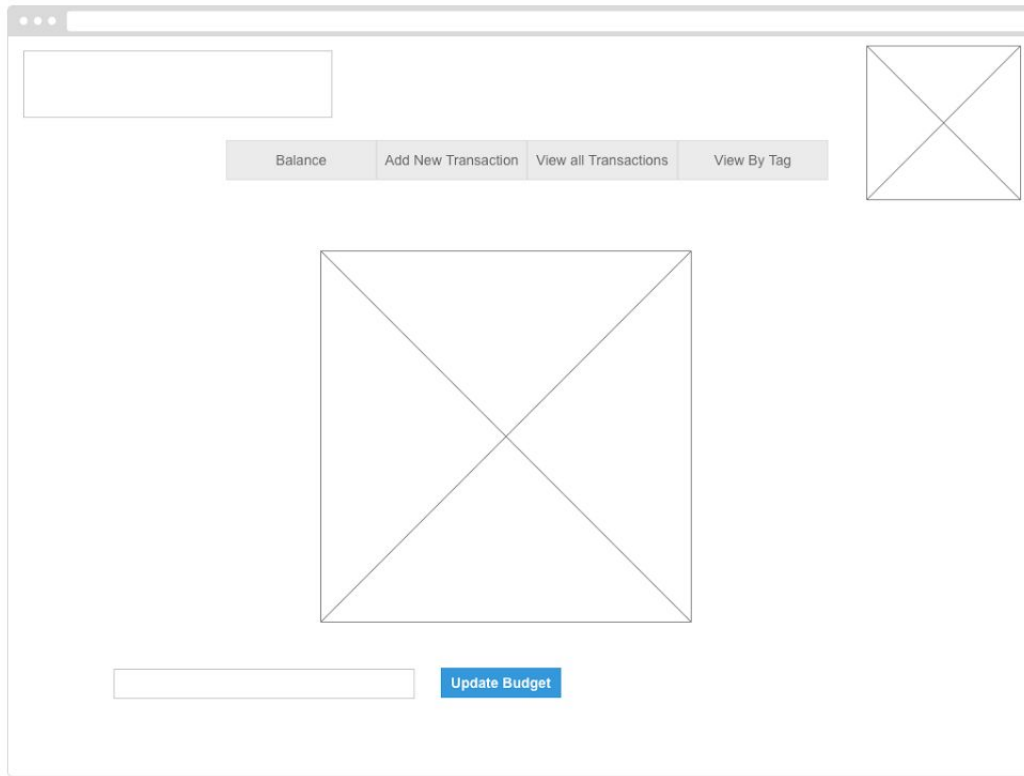
P4 - Acceptance Criteria

Acceptance Criteria	Expected Result/Output	Pass/Fail
A user is able to search for a particular keyword	Word is displayed	Pass
A user is able to use a menu to view list of keywords	Menu displays showing keywords	Pass
A user is able to check their understanding through an interactive test.	User should have to input data which is checked and confirmed if correct	Pass
A user can view more information about a word	A button press will present further information in a pop up	Pass

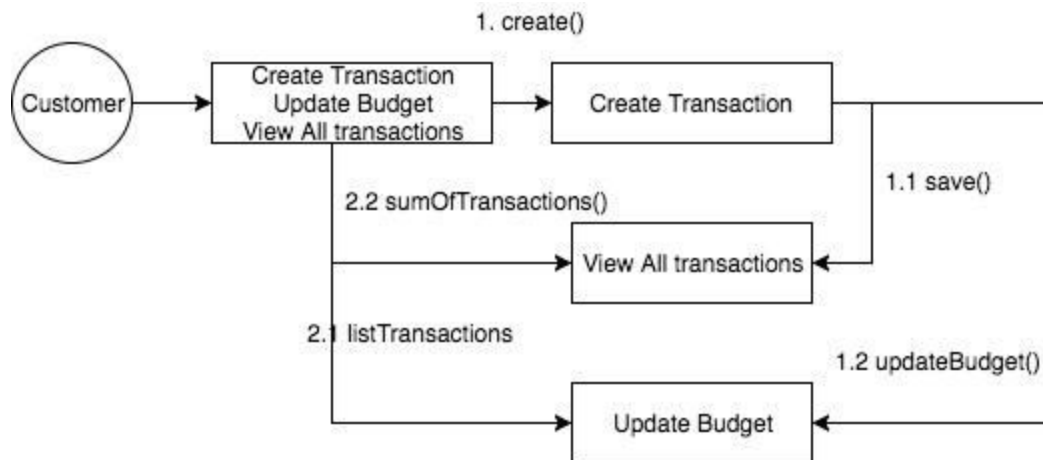
P5 - User Sitemap



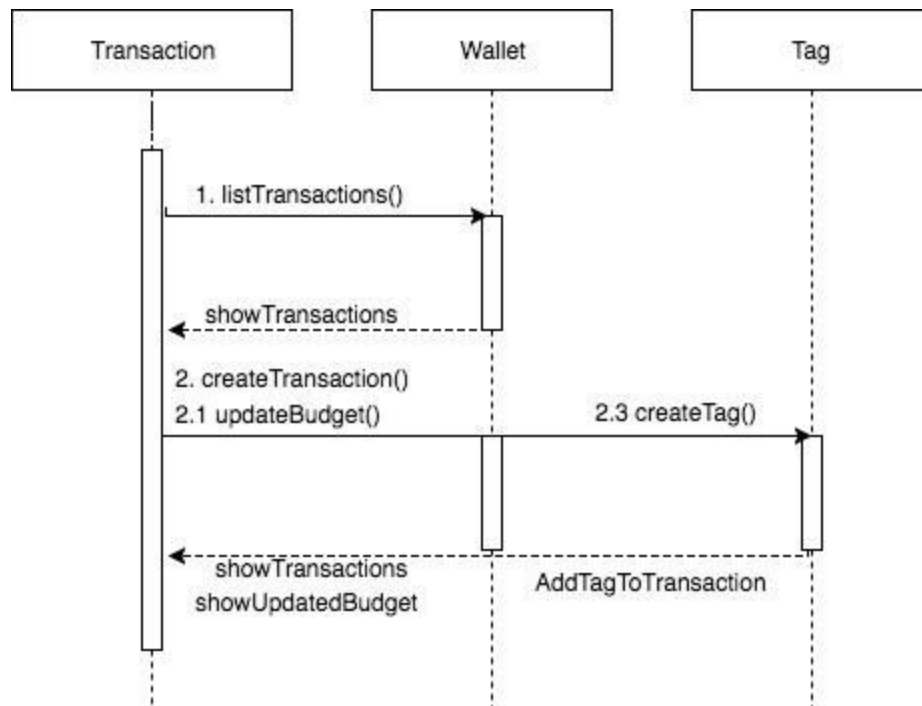
P6 - Wireframe



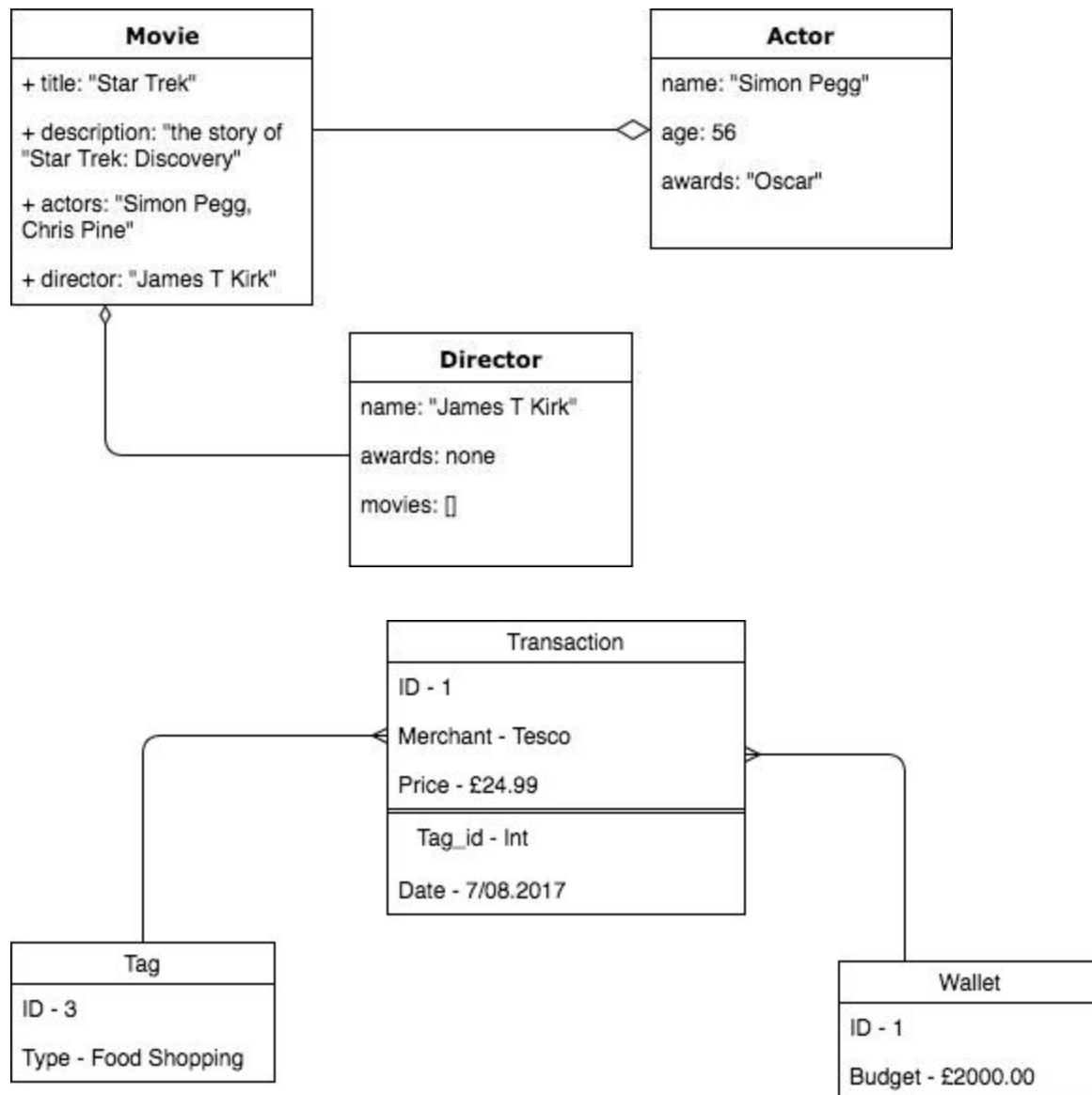
P7 Collaboration Diagram



Sequence Diagram



P8 - Object Diagram

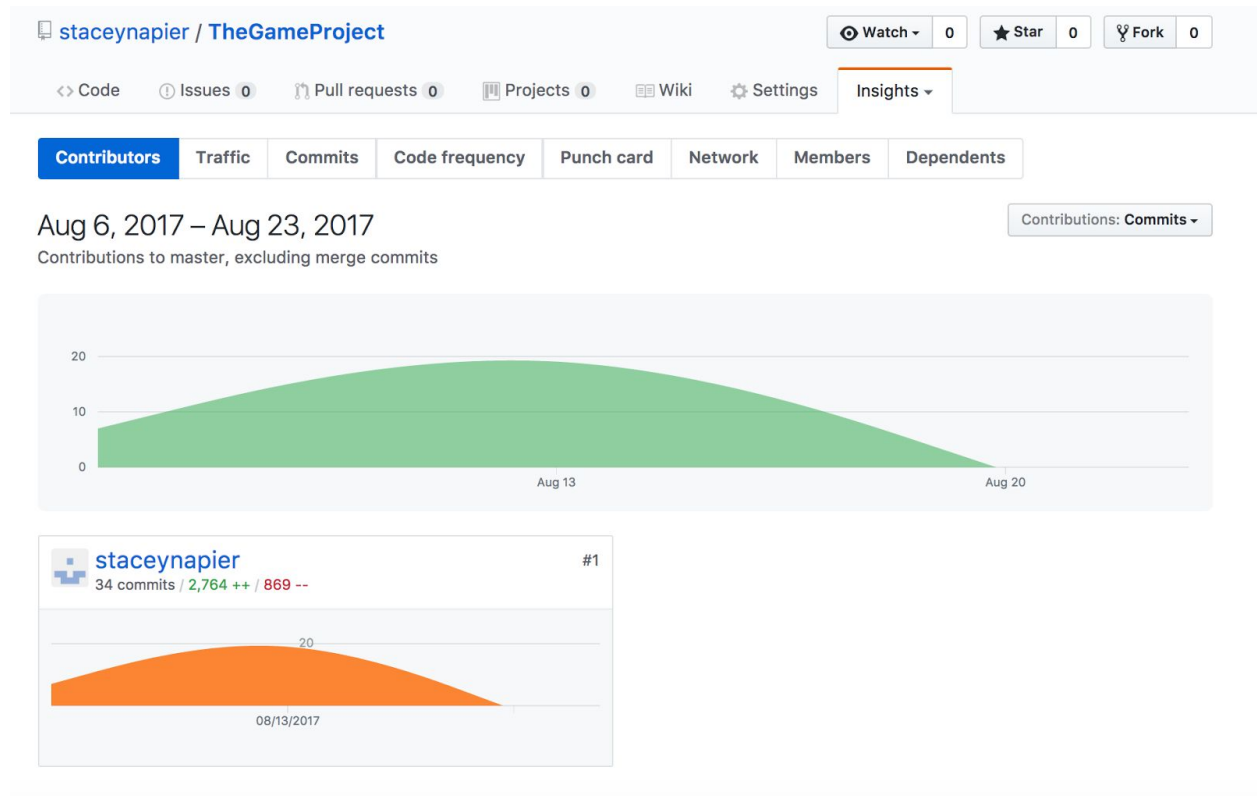


P10 - Pseudocode

```

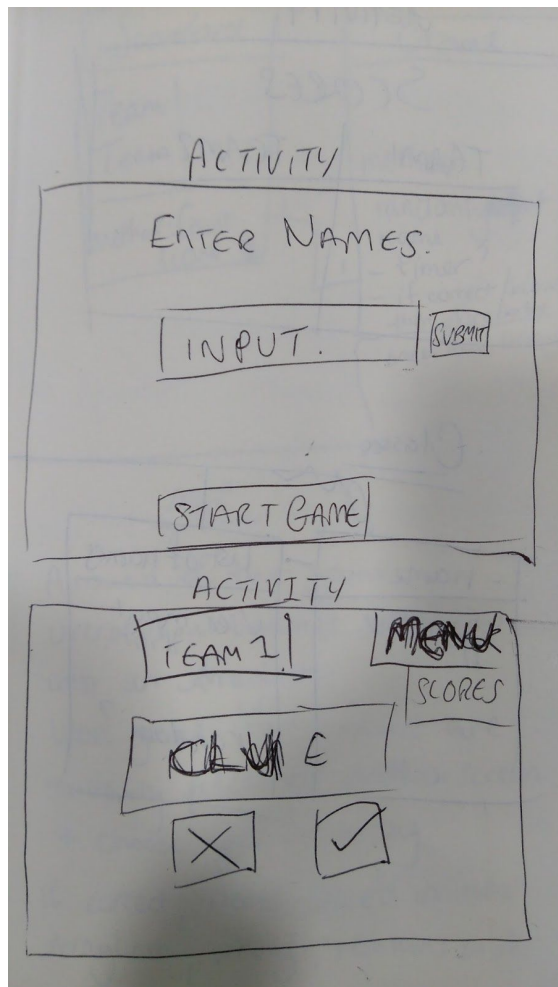
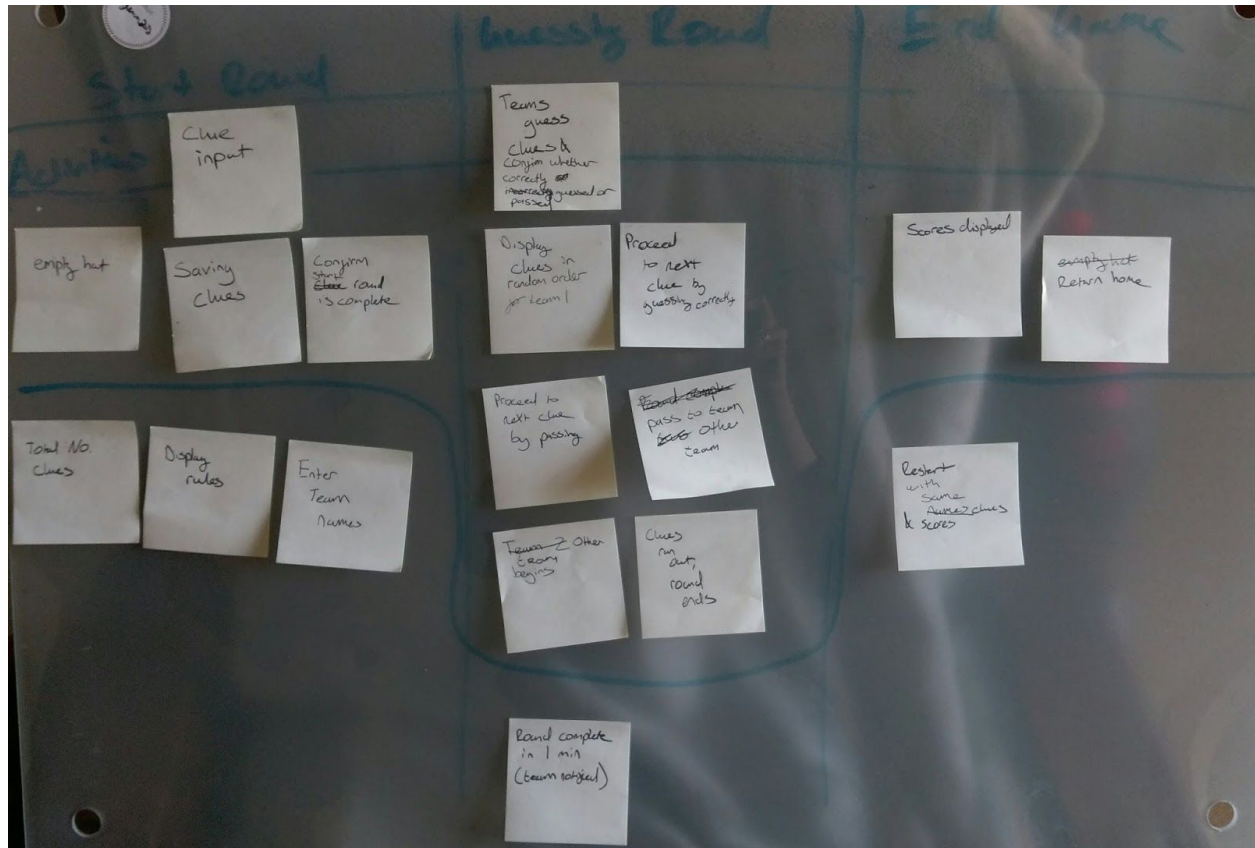
def self.find_all
  #select all from the tags table in the database
  # run the sql runner
  # return the results in ruby by mapping the array.
  sql = "SELECT * FROM tags";
  tags = SqlRunner.run(sql)
  results = tags.map { |tag| Tag.new(tag) }
  return results
end
  
```

P11 Solo Project



<https://github.com/staceynapier/TheGameProject>

P12 - Planning




The image above indicates a planning session, in which I noted down all of the steps that would be required for the game to work. From here, I was able to establish which steps were necessary for the MVP - indicated by the blue line.

P13

User inputs a new budget which alters the available balance left.

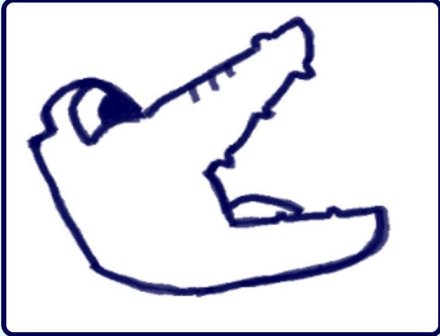
Snappier Finance



[Balance](#) | [Add New Transaction](#) | [View All Transactions](#) | [View By Tags](#)

A snappier way to budget your finances


Current Remaining Balance: £1837.71



Your initial balance was £2000.00

Update balance: £ [Submit](#)

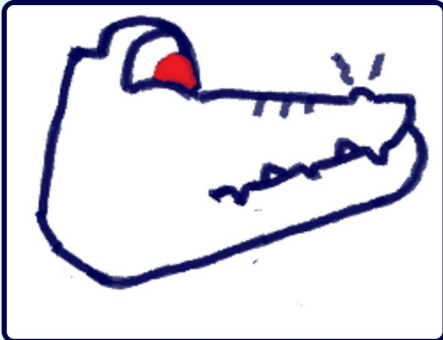
Snappier Finance



[Balance](#) | [Add New Transaction](#) | [View All Transactions](#) | [View By Tags](#)

A snappier way to budget your finances

Current Remaining Balance: £-12.29




Your initial balance was £150.00

Update balance: £ [Submit](#)

P14

User inputs new transaction details. Showing all transactions confirms transaction saved in the database.

Snappier Finance



[Balance](#) | [Add New Transaction](#) | [View All Transactions](#) | [View By Tags](#)

Enter New Transaction Details

Merchant:


Description:

Value: £

Date of Transaction:

Tag:

Snappier Finance



[Balance](#) | [Add New Transaction](#) | [View All Transactions](#) | [View By Tags](#)


Transactions

Merchant	Description	Value	Transaction Date	Type	
Amazon	camera	£45.32	2017-07-26	Entertainment	<input type="button" value="Delete"/>
Amazon	present for ma	£34.98	2017-07-25	Miscellaneous	<input type="button" value="Delete"/>
Ovo energy	gas bill	£72.0	2017-07-21	Bills	<input type="button" value="Delete"/>
Chanter	pint	£3.0	2017-07-18	Entertainment	<input type="button" value="Delete"/>
Co-op	Tuesday dinner	£6.99	2017-07-15	Food	<input type="button" value="Delete"/>

Total of all transactions = £162.29

P 15

User selects a tag and is taken to a new page with a list of all the transactions with that tag.

Snappier Finance

Balance | Add New Transaction | View All Transactions | View By Tags

Tags

Food







Rent


Entertainment

Clothes

Bills

Miscellaneous



Snappier Finance

Balance | Add New Transaction | View All Transactions | View By Tags

Transactions

Merchant	Description	Value	Transaction Date	Type	
Chanter	pint	£3.0	2017-07-18	Entertainment	Delete
Amazon	camera	£45.32	2017-07-26	Entertainment	Delete

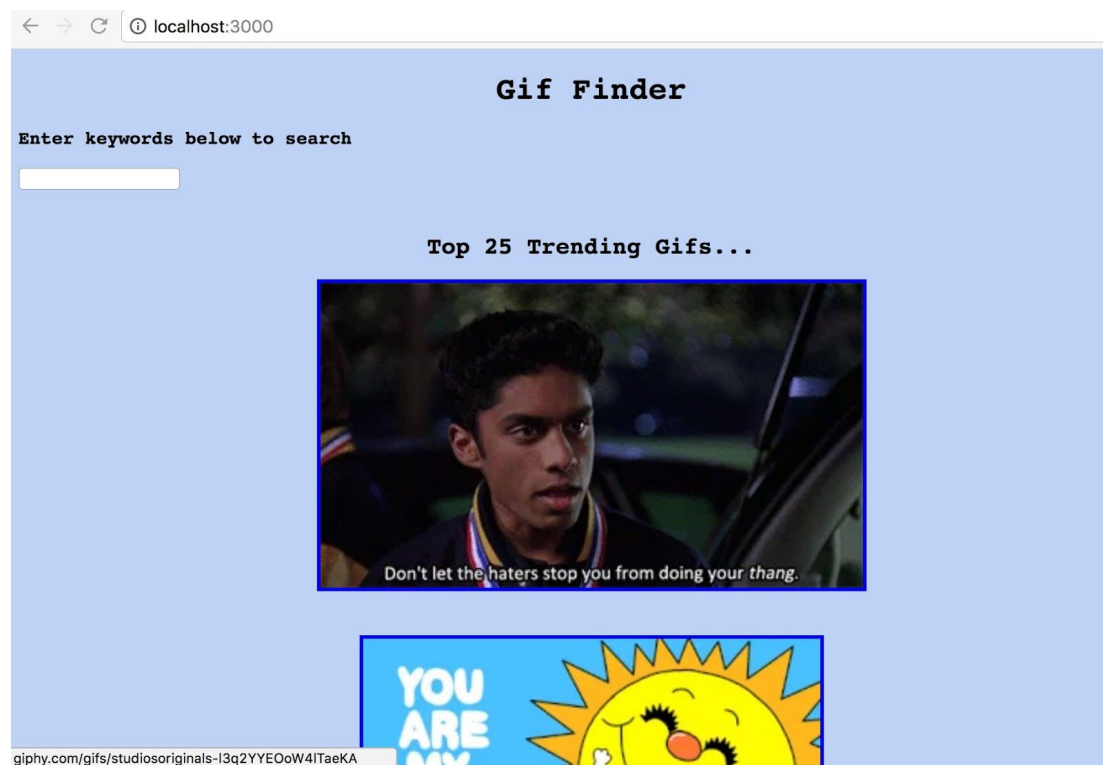
Total Entertainment transactions = £48.32

Total of all transactions = £162.29

```

1 var app = function(){
2   var url = 'https://api.giphy.com/v1/gifs/trending?api_key=77f26d5aac2243618618a35dee280226&limit=25&rating=G';
3   makeRequest(url, requestComplete);
4 }
5
6 var makeRequest = function(url, callback){
7   var request = new XMLHttpRequest();
8   request.open('GET', url);
9   request.addEventListener('load', callback);
10  request.send();
11 }
12
13 var makeSearchRequest = function(callback) {
14   var searchData = document.getElementById("input").value;
15   var apiUrl = 'https://api.giphy.com/v1/gifs/search?api_key=77f26d5aac2243618618a35dee280226&q='
16   + searchData + '&limit=5&offset=0&rating=G&lang=en';
17   var request = new XMLHttpRequest();
18   request.open('GET', apiUrl);
19   console.log(apiUrl);
20   request.addEventListener('load', callback);
21   request.send();
22 }
23
24 var requestComplete = function(){
25   console.log("Request Successfully Completed!");
26   if(this.status !== 200) return;
27   var jsonString = this.responseText;
28   var gifs = JSON.parse(jsonString);
29   console.log(gifs.data);
30   localStorage.setItem('gifs', gifs.data);
31   loopThrough(gifs.data);
32 }

```



P18 Testing

Test code

```
public class GameTest {

    Game game;

    @Before
    public void before(){
        Clue clue = new Clue("Donald Trump");
        Clue clue1 = new Clue("Theresa May");
        Clue clue2 = new Clue("Kim Jong Un");
        game = new Game();
        game.addClue(clue);
        game.addClue(clue1);
        game.addClue(clue2);
    }

    @Test
    public void hasList() { assertEquals(2, game.getList().size()); }

    @Test
    public void testLength() { assertEquals((Integer)2, game.getLength()); }

    @Test
    public void canEmptyList(){
        game.empty();
        assertEquals(1, game.getList().size());
    }
}
```

Tests failing

```
6 tests done: 5 failed - 75ms

"/Applications/Android Studio.app/Contents/jre/jdk/Contents/Home/bin/java" ...

java.lang.AssertionError:
Expected :1
Actual   :2
<Click to see difference>

<1 internal calls>
  at org.junit.Assert.failNotEquals(Assert.java:834) <2 internal calls>
  at com.example.user.thegame.GameTest.canRemoveAtIndex(GameTest.java:69) <28 internal calls>

java.lang.AssertionError:
Expected :2
Actual   :3
<Click to see difference>

<1 internal calls>
  at org.junit.Assert.failNotEquals(Assert.java:834) <2 internal calls>
  at com.example.user.thegame.GameTest.hasList(GameTest.java:34) <28 internal calls>
```

Code after changes

```

public class GameTest {

    Game game;

    @Before
    public void before(){
        Clue clue = new Clue("Donald Trump");
        Clue clue1 = new Clue("Theresa May");
        Clue clue2 = new Clue("Kim Jong Un");
        game = new Game();
        game.addClue(clue);
        game.addClue(clue1);
        game.addClue(clue2);
    }

    @Test
    public void hasList() { assertEquals(3, game.getList().size()); }

    @Test
    public void testLength() { assertEquals((Integer)3, game.getLength()); }

    @Test
    public void canEmptyList(){
        game.empty();
        assertEquals(0, game.getList().size());
    }

    @Test
    public void canGetAnswerAtIndex(){
        Clue result = game.getAnswerAtIndex(1);
        assertEquals("Theresa May", result.getName());
    }

    @Test
    public void canGetRandomClue() { assertNotNull(game.getRandomClue()); }
}

```

Tests passing


All 6 tests passed - 6ms

"/Applications/Android Studio.app/Contents/jre/jdk/Contents/Home/bin/java" ...

Process finished with exit code 0