

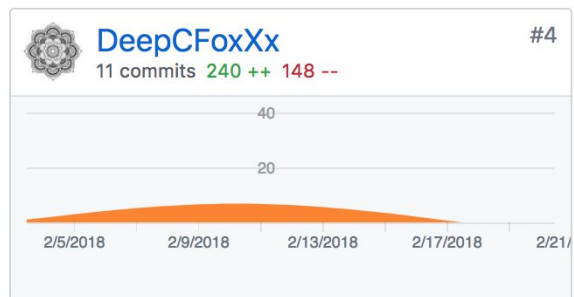
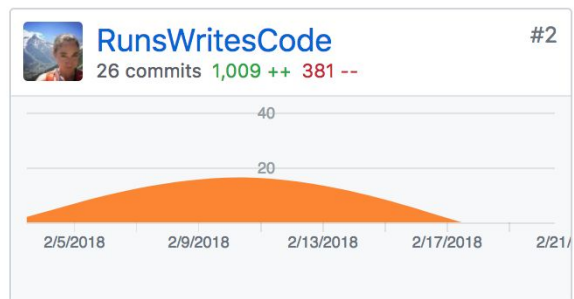
Evidence for project unit

Mark Conroy

E17

P.1 Github contributors page

Contributions to master, excluding merge commits



P.2 Project brief

Shares App

A local trader has come to you with a portfolio of shares. She wants to be able to analyse it more effectively. She has a small sample data set to give you and would like you to build a minimal viable product (MVP) that uses the data to display her portfolio in useful ways so that she can make better decisions.

MVP

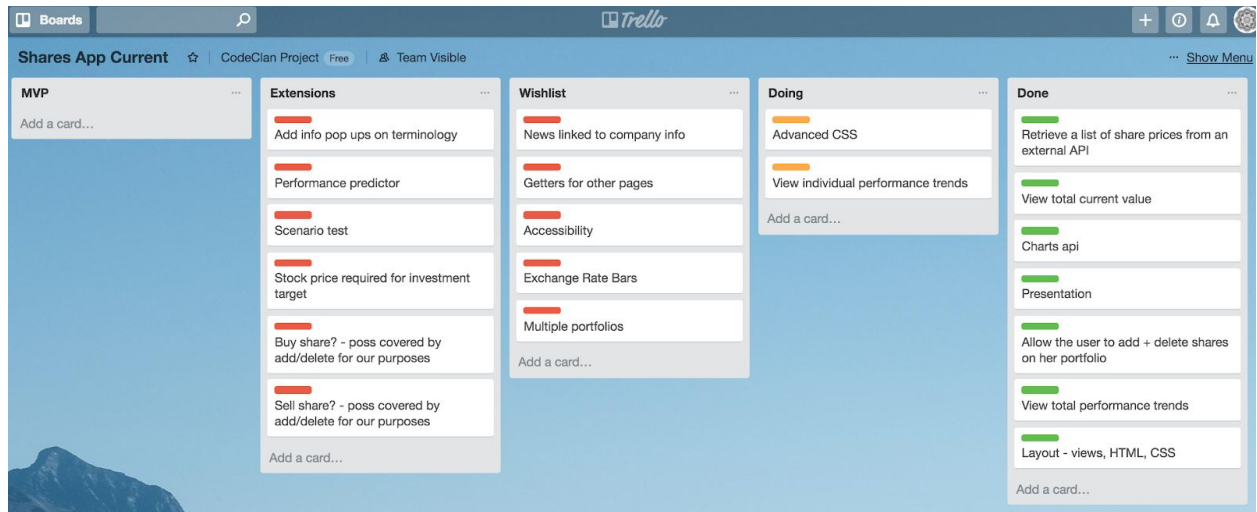
- View total current value
- View individual and total performance trends
- Retrieve a list of share prices from an external API and allow the user to add shares to her portfolio
- Provide a chart of the current values in her portfolio

Examples of further features

Speculation based on trends and further financial modelling.

- Based on the last 6 weeks of data, (e.g. it reduced by 5%) predict the performance of a stock for the next 6 weeks. Based on that what would be the sum of the total on a certain day in the future.
- She wants to know what would happen to the value of her portfolio if the price of a share changed by 10% or if she reduced her quantity by 15%.
- What price does a stock have to get to in order to meet her investment target?
- To get her £1700 worth of RBS shares what price does it have to go to?
- Buy new shares
- Sell shares

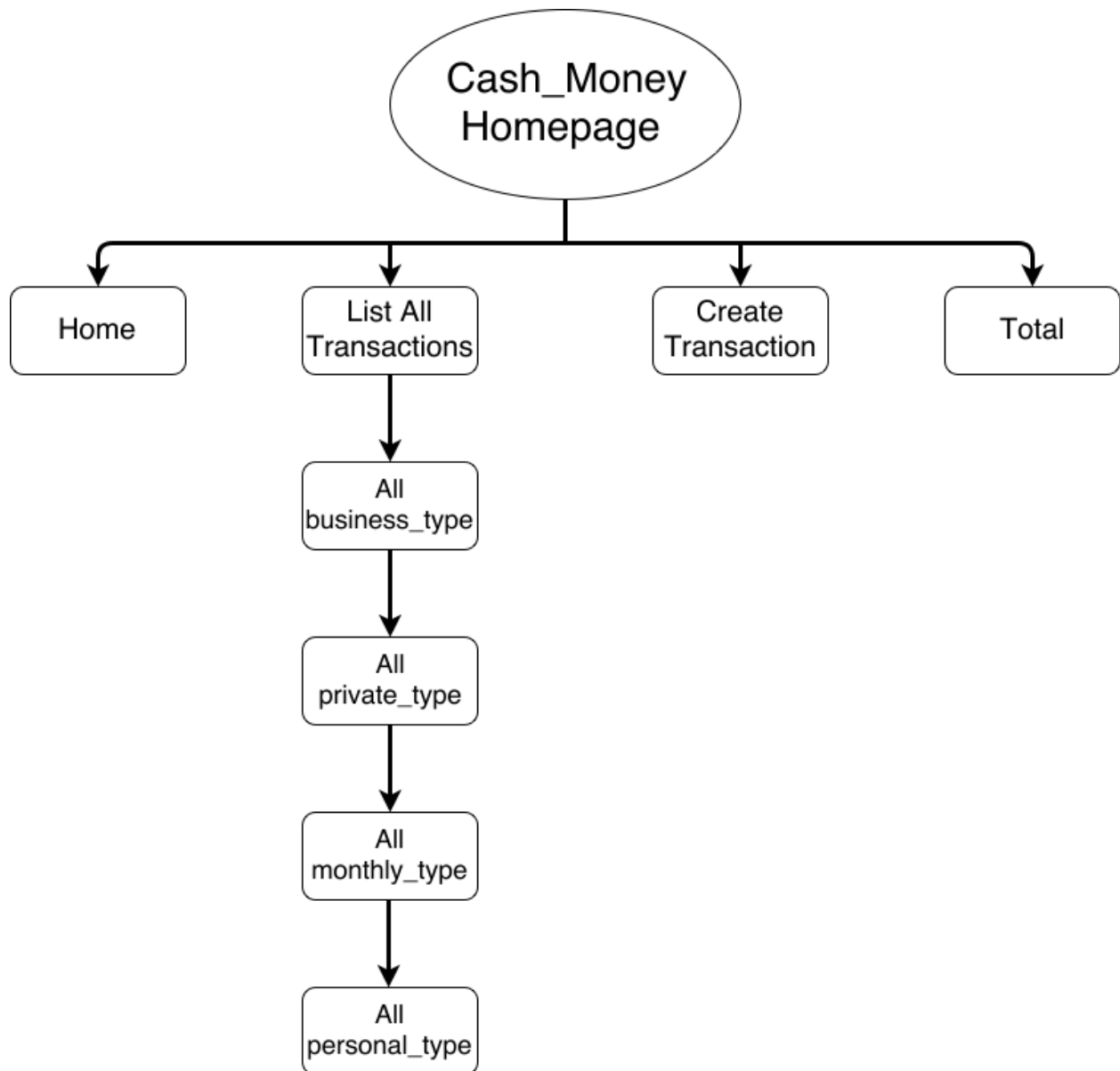
P.3 Use of Trello



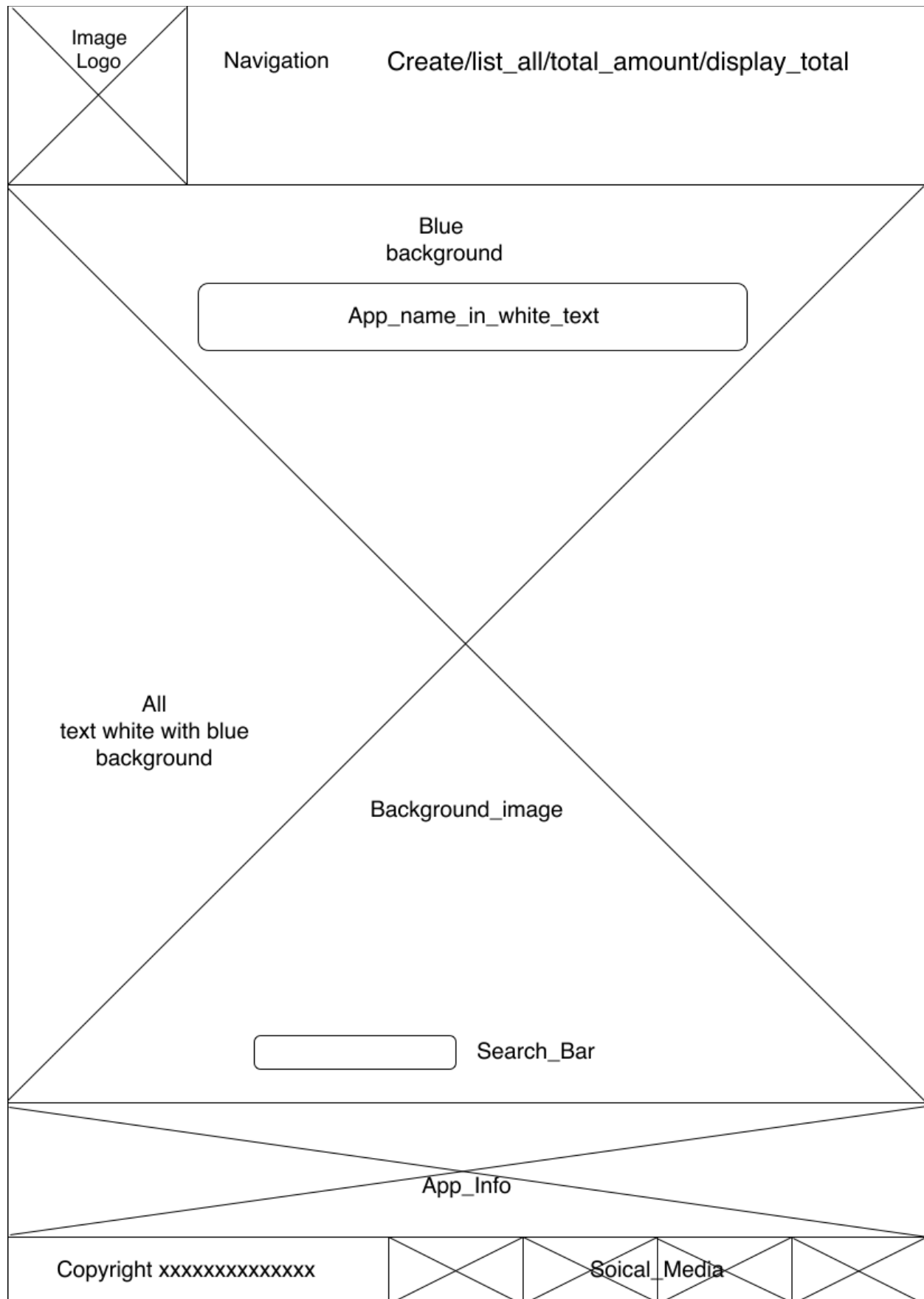
P.4 Acceptance criteria

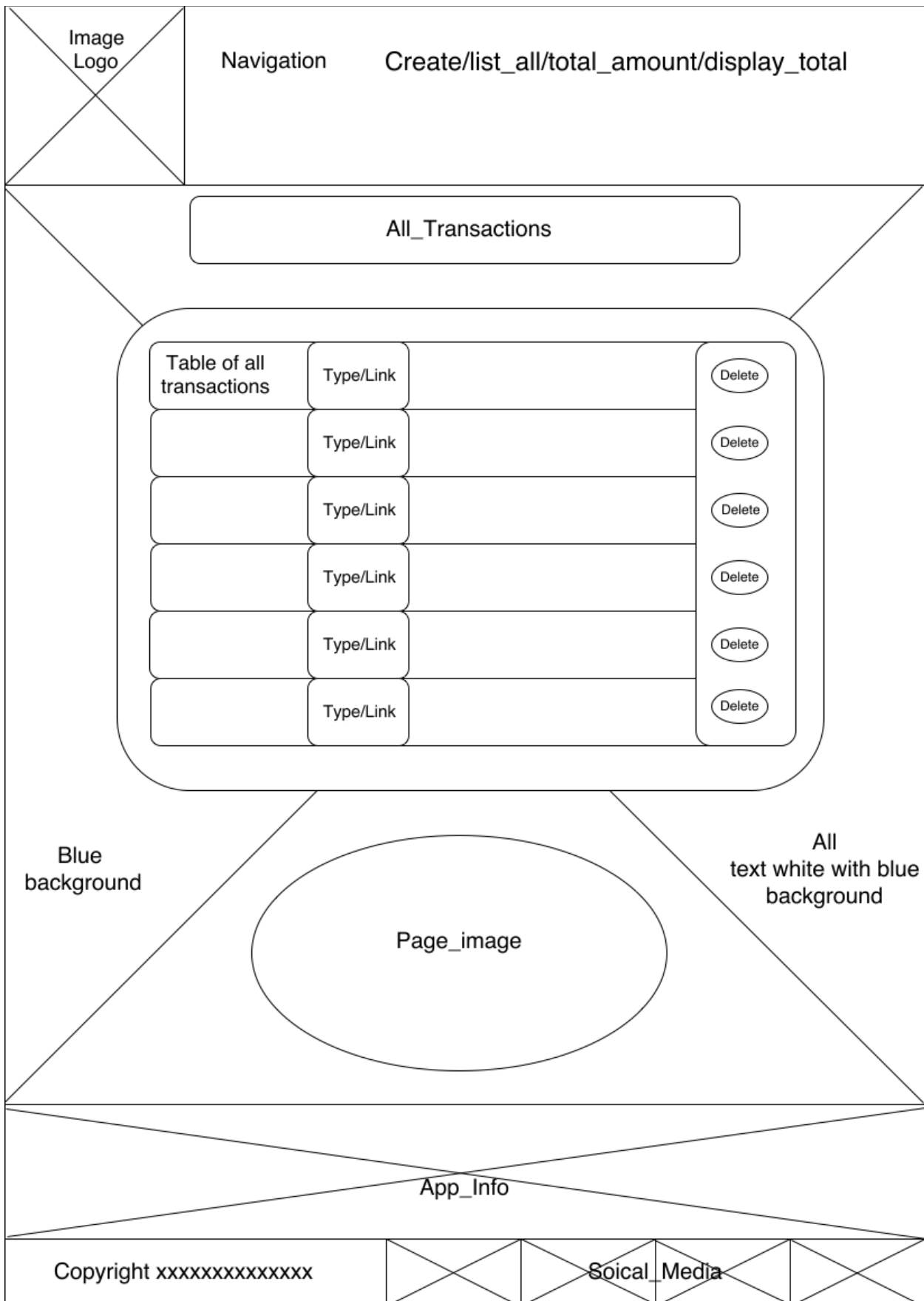
<u>Acceptance Criteria</u>	<u>Result/output</u>	<u>Pass/Fail</u>
User can select individual stock and see performance over set period of time	2nd Page shows line graph of the performance of select stock on load	Pass
When individual stock is selected news reports relevant to stock are displayed	2nd Page shows five news reports returned from news api rendered at the bottom of the page	Pass
Total portfolio value must be displayed	On load the page displays the cumulative portfolio value returned from stock api	Pass
Total loss and profit for selected stock to be displayed	2nd page on load shows total loss and profit for stock selected from users portfolio	Fail
User will be able to buy and sell stock	When stock is sold / bought current prices returned from api will be reflected in portfolio	Fail
Each individual news report will have a clickable image linked to source of story	2nd page on load shows five news reports with images returned from news api	N/A

P.5 User site map

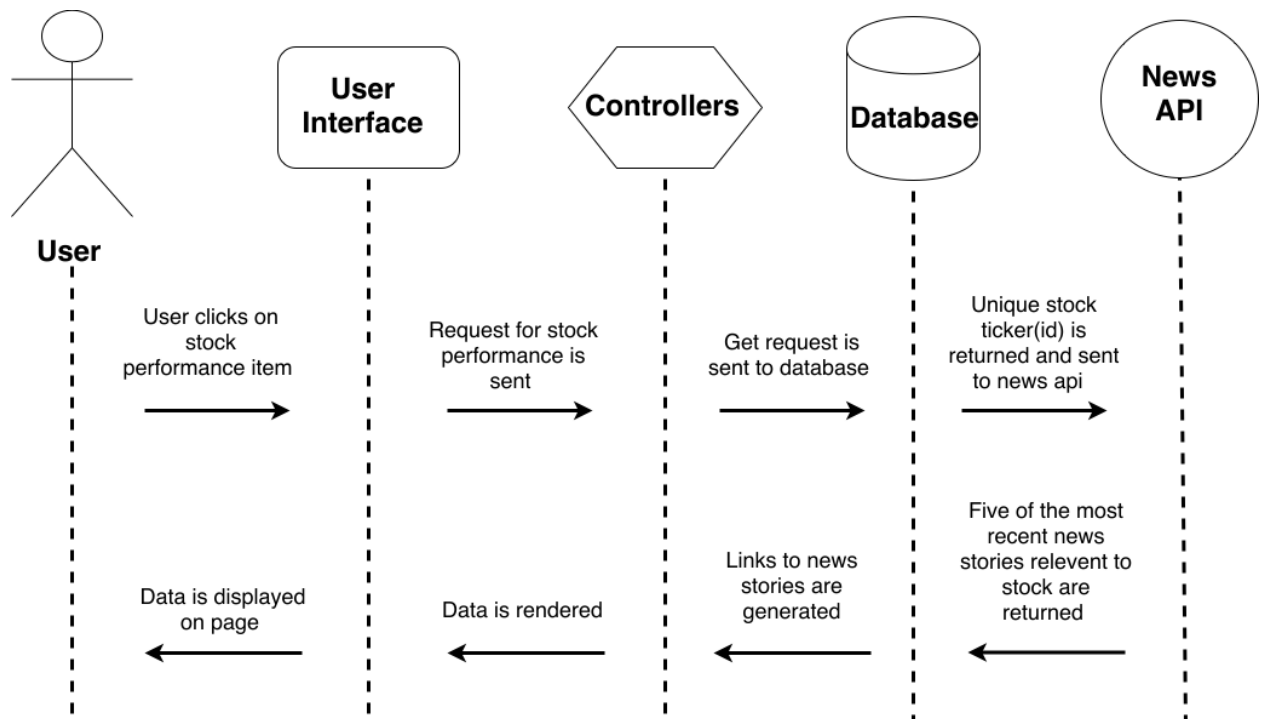
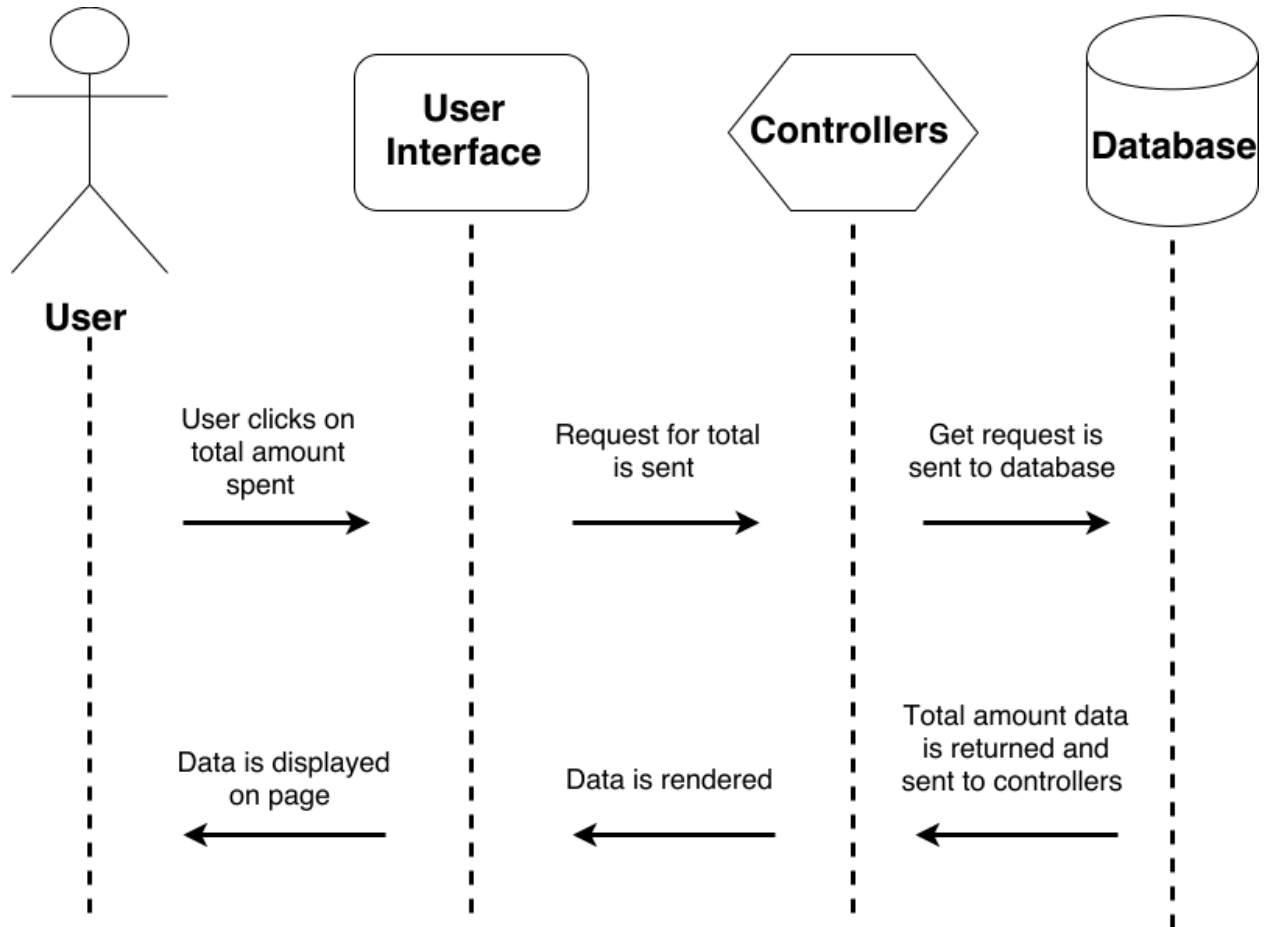


P.6 Wireframes

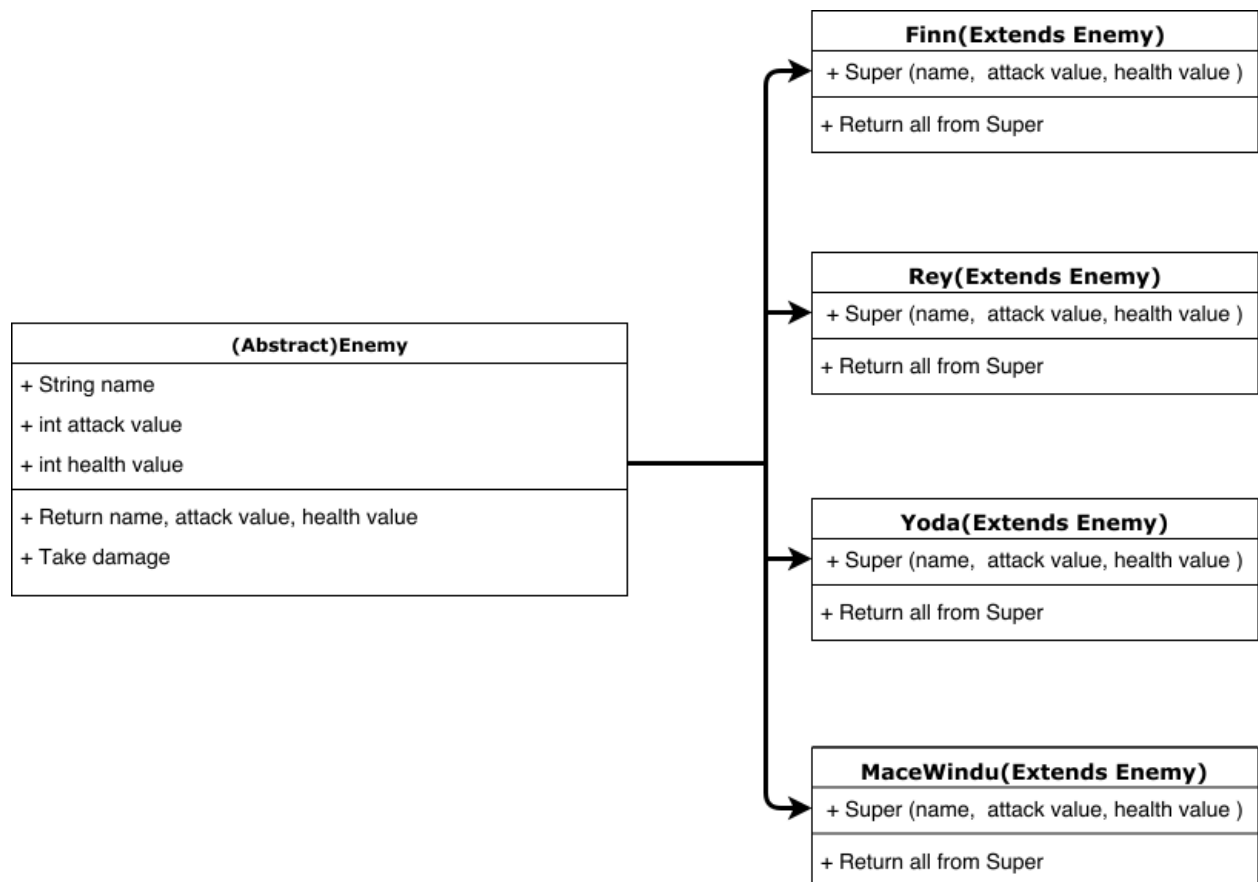
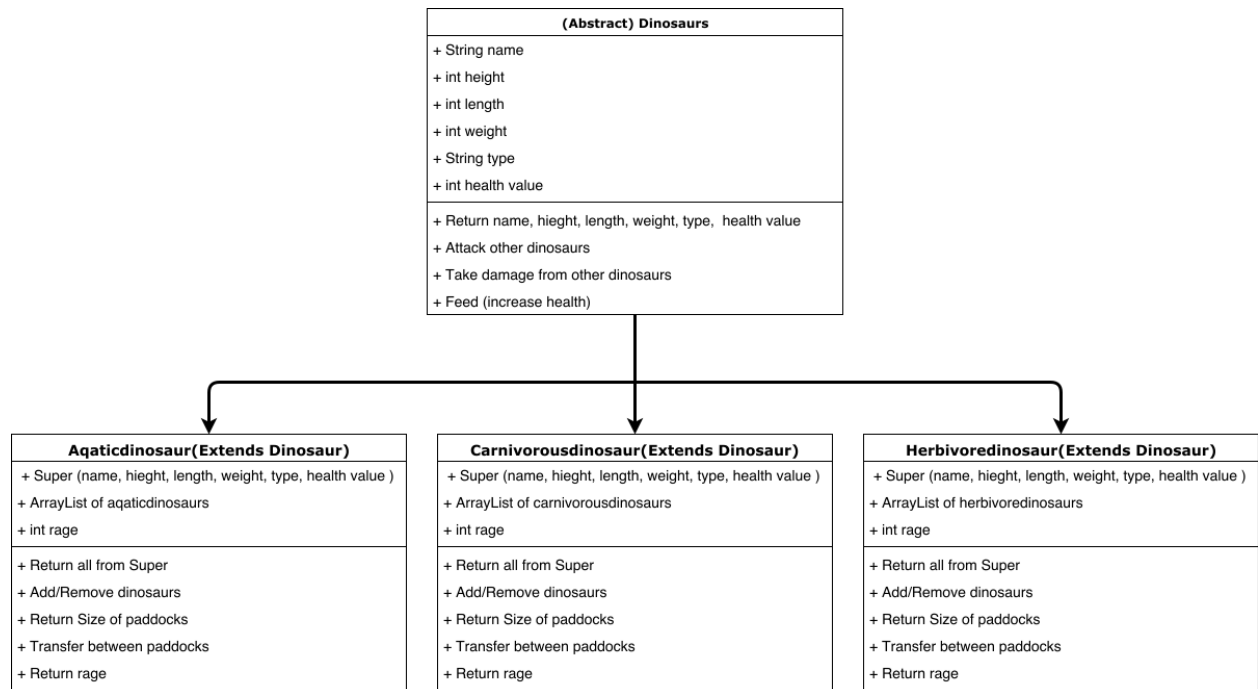




P.7 System interaction diagrams



P.8 Object diagrams



P.9 Choice of two algorithms

Algorithm_01 is sorting through an array comparing elements and switching them until they are in order of low to high.

```
def array_sort_algorithm(array)
  n = array.length

  loop do

    swapped = false

    (n-1).times do |i|
      if array[i] > array[i + 1]
        array[i], array[i + 1] = array[i + 1], array[i]
        swapped = true
      end
    end

    break if not swapped
  end

  array
end
```

The reason I chose to use this algorithm was to get a better understanding of algorithms in general and sorting methods in ruby. Having done that I feel this example has given me a better understanding of how to switch elements in an array and how ruby goes through an array comparing elements within it to output a result.

Algorithm_02 is using ruby's built in date function to find how many sundays fall on the first of the month from the the start date to the end date.

```
require 'date'

start_date = Time.local(1987, 07, 23)
end_date = Time.local(2018, 02, 25)
sunday_counter = 0

while end_date >= start_date
  if end_date.strftime("%A") == "Sunday" && end_date.strftime("%d") == "01"
    sunday_counter += 1
  end
  end_date -= 86400
end
```

The reason I chose to use this algorithm was to complete an exercise in finding the number of sundays that fall on the first of the month from your birthday (start_date) to the present day (end_date). I feel this example would have limited use outside of the exercise but feel it was worth doing to get a better understanding of built in ruby functions such as date / time I also learned how to use strftime (string to time).

P.10 Example of pseudocode

```
<div class="field-group">
  <label for="-field" class="label">Merchant:</label>
  <div class="field">
    <select name="merchant_id">
      <!--Need to access all merchant names in a drop down-->
      <% @merchants.each do |merchant| %>
        <!--Display each name individually and make them clickable-->
        <option value="<%= merchant.id %>"><%= merchant.merchant_name %></option>
        <!--Make them available and useable linked with id-->
      <% end %>
    </select>
  </div>
</div>
```

P.11 Github link to personal project

https://github.com/DeepCFoxXx/Sith_Adventure_Game

DeepCFoxXx / Sith_Adventure_Game

Watch

0

Star

0

Fork

0

<> Code

Issues 0

Pull requests 0

Projects 0

Wiki

Insights

Settings

No description, website, or topics provided.

Edit

Add topics

60 commits

1 branch

0 releases

1 contributor

Branch: master

New pull request

Create new file

Upload files

Find file

Clone or download

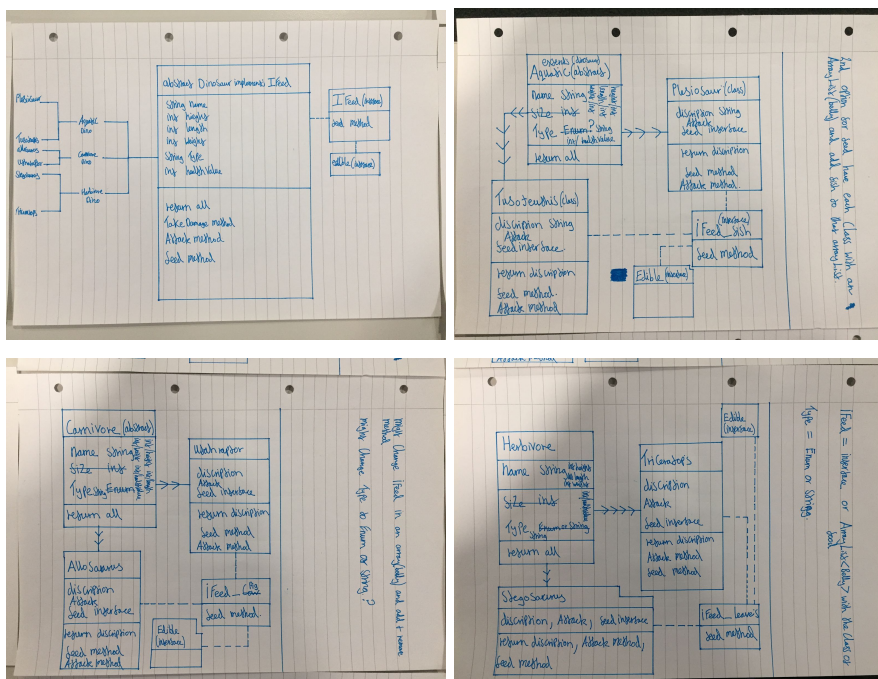
DeepCFoxXx Added Read.Me

Latest commit 9f2984c 15 days ago

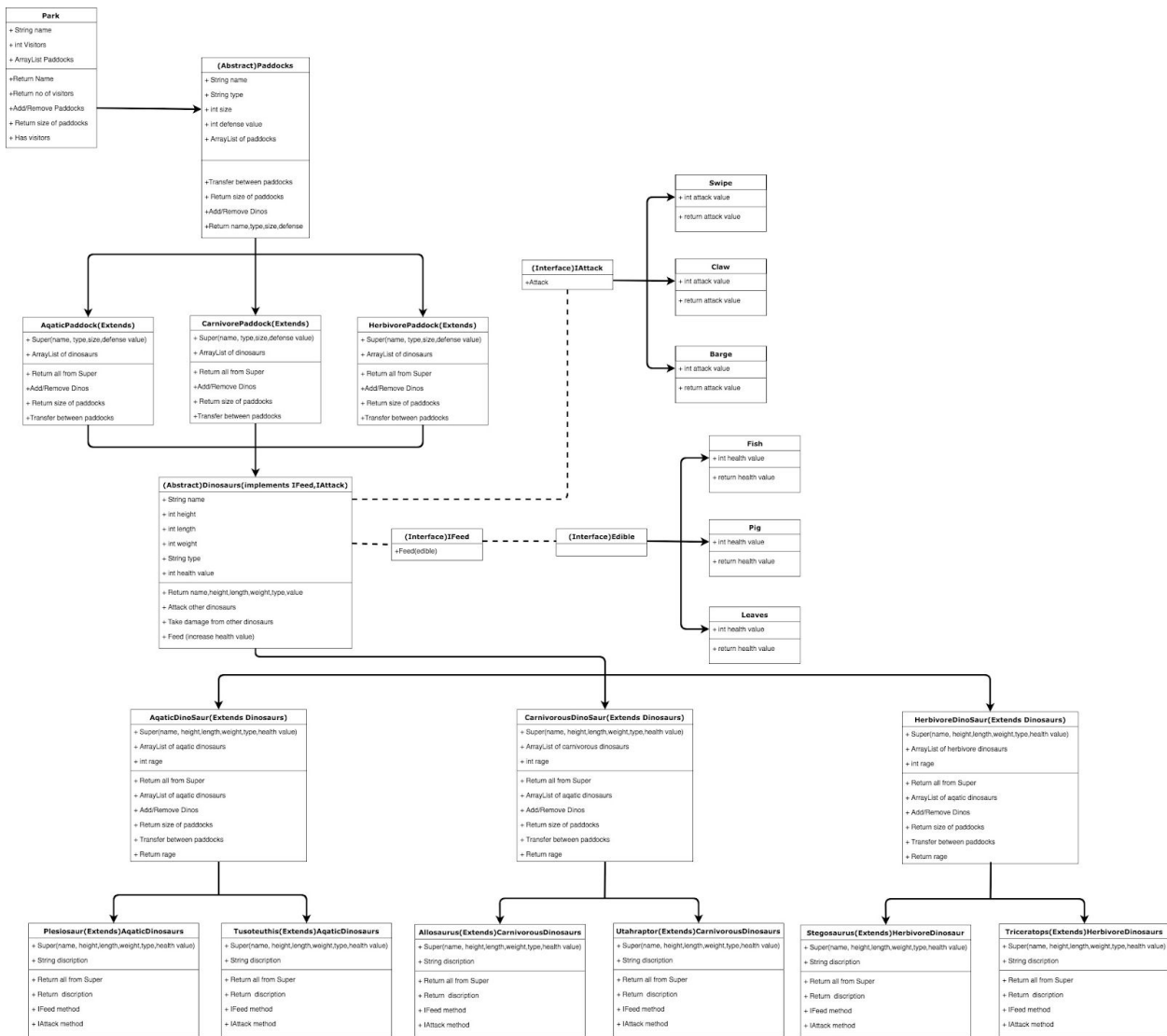
.gradle	Player class started	2 months ago
.idea	Added Read.Me	15 days ago
READ.ME	Added Read.Me	15 days ago
gradle/wrapper	Player class started	2 months ago
out	Refactoring	a month ago
src	Refactoring	a month ago
build.gradle	Player class started	2 months ago
gradlew	Player class started	2 months ago
gradlew.bat	Player class started	2 months ago
settings.gradle	Player class started	2 months ago

P.12 Planning in different stages

Before



After



P.13 User input

User inputting 2000 in to a new transaction

Create Transaction

Merchant:

Tesco

Transaction Type:

Monthly

Amount:

2000

Create



Input being added to total amount by type (Tesco/Monthly)

Total By Type

Total Amount By Transaction Type = £2030



P.14 Interaction with data persistence

Data being inputted into program

Create Transaction

Merchant:

M n S ▾

Transaction Type:

Personal ▾

Amount:

1000

Create



Conformation of data being saved

All Transactions

Value	Type	Merchant	Delete
£20	Business	ASDA	Delete
£15	Personal	Tesco	Delete
£50	Private	Morrisons	Delete
£30	Monthly	M n S	Delete
£1000	Personal	M n S	Delete



P.15 User output result

The user requesting information or an act to be performed (Delete last transaction)

All Transactions

Value	Type	Merchant	Delete
£20	Business	ASDA	Delete
£15	Personal	Tesco	Delete
£50	Private	Morrisons	Delete
£30	Monthly	M n S	Delete
£1000	Personal	M n S	Delete



The request being carried out

All Transactions

Value	Type	Merchant	Delete
£20	Business	ASDA	Delete
£15	Personal	Tesco	Delete
£50	Private	Morrisons	Delete
£30	Monthly	M n S	Delete



P.16 API being used in a program

The code that uses the api

```
var app = function() {  
  var url = 'https://restcountries.eu/rest/v2';  
  makeRequest(url, requestComplete);  
  var selection = document.querySelector('#countries-dropdown');  
  selection.addEventListener('change', displayInfo);  
  console.log(getCountryByAlphaCode("FRA"));  
};  
  
window.addEventListener('load', app);
```

The API being used by the program while running



P.17 Bug report

User needs to view total portfolio value	Fail	Used returned stock id from database to request info from API	Pass
Modal box closes from clicking on home page	Fail	Added function to listen for event	Pass
User must be able to select stock to add to portfolio	Fail	Added search box that displays 10 companys when input is given	Pass
Piechart must display percentage of portfolio values	Fail	Added more functionality to charts controller	Pass

P.18 Testing your program

Test code

```
it('does task have difficulty level', function{
  assert.strictEqual(task1.difficultyLevel, 5);
})

it('tasks starts off empty', function(){
  assert.strictEqual(hero1.tasks.length, 0)
})

it('can add to tasks', function(){
  hero1.addTasks(task1);
  assert.strictEqual(hero1.tasks.length 1)
})

it('food can heal hero', function(){
  food1 = Food('Apple', 1);
  hero1.eatFood(food1);
  assert.strictEqual(hero1.health, 29);
})
```

The test code failing

```
→ Heroes_and_Rats git:(master) ✗ npm tes
> heroes_and_rats@1.0.0 test /Users/markconroy/codeclan_work/home_work/week_11/day_05/Heroes_and_Rats
> mocha specs

/Users/markconroy/codeclan_work/home_work/week_11/day_05/Heroes_and_Rats/specs/hero_spec.js:70
  it('does task have difficulty level', function{
  ^
SyntaxError: Unexpected token {
    at new Script (vm.js:51:7)
    at createScript (vm.js:138:10)
    at Object.runInThisContext (vm.js:199:10)
    at Module._compile (module.js:624:28)
    at Object.Module._extensions..js (module.js:671:10)
    at Module.load (module.js:573:32)
    at tryModuleLoad (module.js:513:12)
    at Function.Module._load (module.js:505:3)
    at Module.require (module.js:604:17)
    at require (internal/module.js:11:18)
    at /Users/markconroy/codeclan_work/home_work/week_11/day_05/Heroes_and_Rats/node_modules/mocha/lib/mocha.js:231:27
    at Array.forEach (<anonymous>)
    at Mocha.loadFiles (/Users/markconroy/codeclan_work/home_work/week_11/day_05/Heroes_and_Rats/node_modules/mocha/lib/mocha.js:228:14)
    at Mocha.run (/Users/markconroy/codeclan_work/home_work/week_11/day_05/Heroes_and_Rats/node_modules/mocha/lib/mocha.js:536:10)
    at Object.<anonymous> (/Users/markconroy/codeclan_work/home_work/week_11/day_05/Heroes_and_Rats/node_modules/mocha/bin/_mocha:582:18)
    at Module._compile (module.js:660:30)
    at Object.Module._extensions..js (module.js:671:10)
    at Module.load (module.js:573:32)
    at tryModuleLoad (module.js:513:12)
    at Function.Module._load (module.js:505:3)
    at Function.Module.runMain (module.js:701:10)
    at startup (bootstrap_node.js:193:16)
    at bootstrap_node.js:617:3
npm ERR! Test failed.  See above for more details.
```

Test code with errors corrected

```
it('does task have difficulty level', function(){
  assert.strictEqual(task1.difficultyLevel, 5);
})

it('tasks starts off empty', function(){
  assert.strictEqual(hero1.tasks.length, 0)
})

it('can add to tasks', function(){
  hero1.addTasks(task1);
  assert.strictEqual(hero1.tasks.length, 1)
})

it('food can heal hero', function(){
  food1 = new Food('Apple', 1);
  hero1.eatFood(food1);
  assert.strictEqual(hero1.health, 29);
})
```

Test code passing

```
→ Heroes_and_Rats git:(master) ✗ npm tes  
]
> heroes_and_rats@1.0.0 test /Users/markconroy/codeclan_work/home_work/week_11/day_05/Heroes_and_Rats
> mocha specs

hero
  ✓ does task have difficulty level
  ✓ tasks starts off empty
  ✓ can add to tasks
  ✓ food can heal hero

4 passing (6ms)

npm update check failed
Try running with sudo or get access
to the local update config store via
sudo chown -R $USER:$(id -gn $USER) /Users/markconroy/.config
→ Heroes_and_Rats git:(master) ✗
```