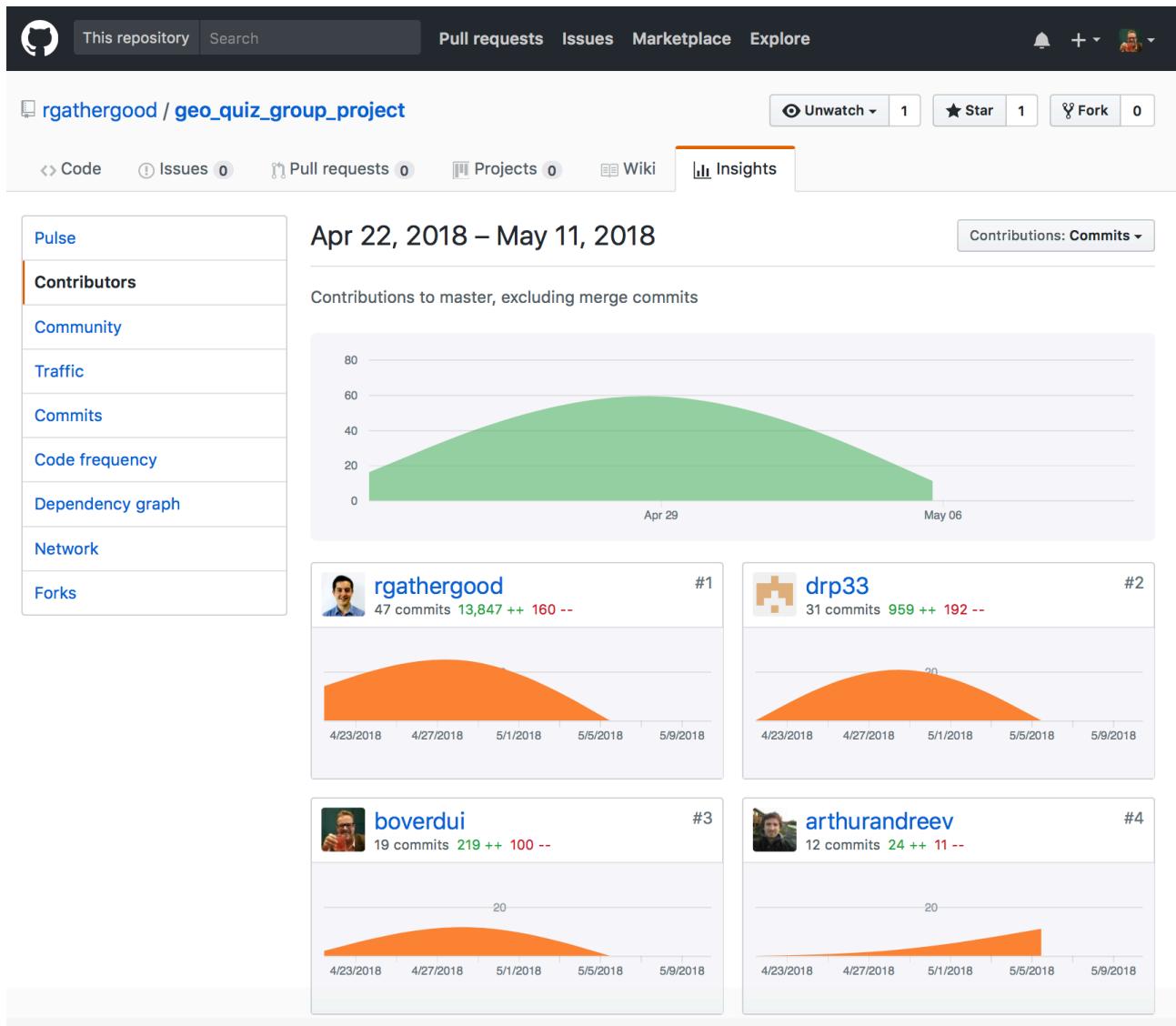


Evidence for Project Unit.

Bert Overduin
Cohort E19

P.1 GitHub contributors page



P.2 Project brief

README.md

Vex Yourself - Flag Quiz - Group Project

Week 14 Group Project

Educational App

The BBC are looking to improve their online offering of educational content by developing some interactive apps that display information in a fun and interesting way.

Your task is to make an MVP to put forward to them - this may only be for a small set of information, and may only showcase some of the features to be included in the final app. You might use an API to bring in content or a database to store facts. The topic of the app is your choice, but here are some suggestions you could look into:

- Interactive timeline, e.g. of the history of computer programming
- Interactive map of a historical event - e.g. World War 1, the travels of Christopher Columbus

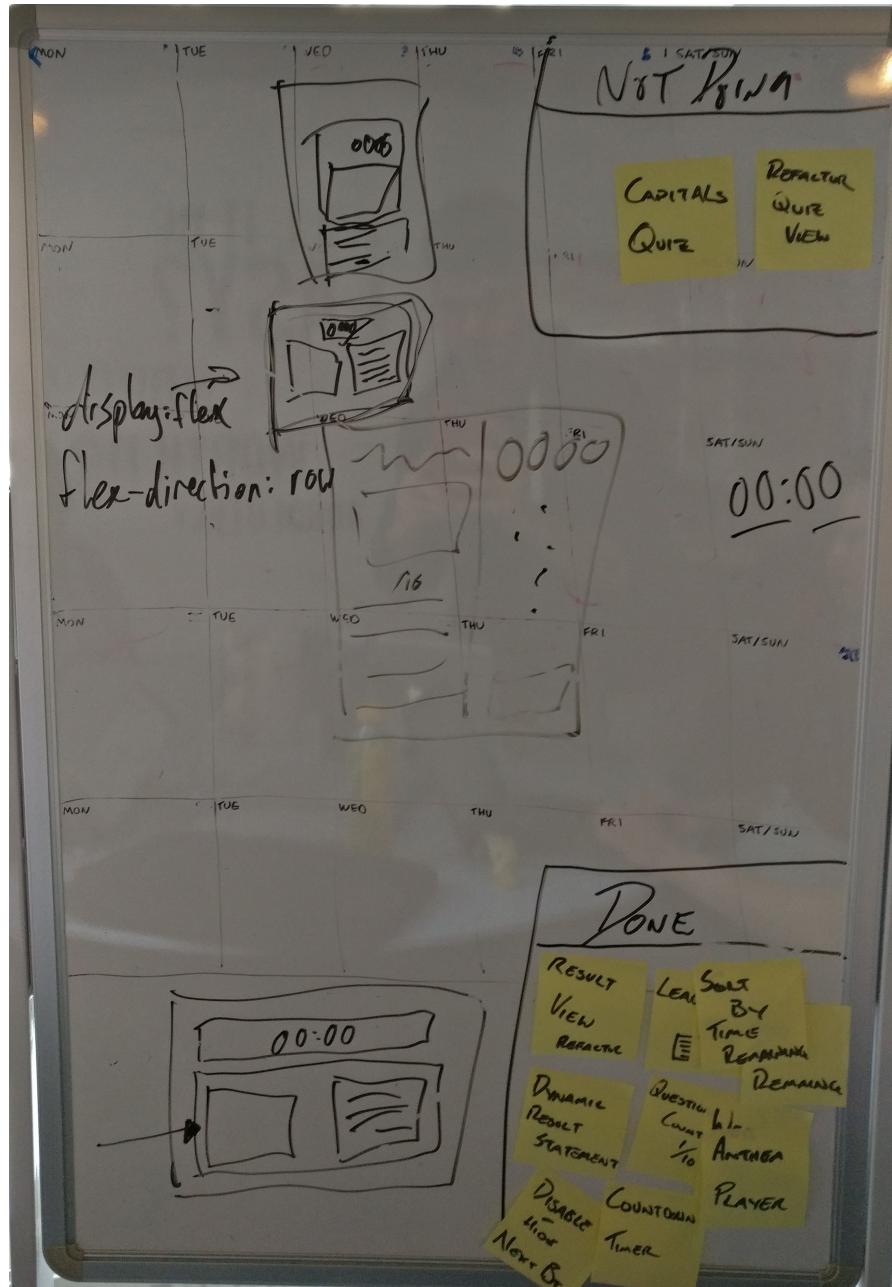
MVP

- Display some information about a particular topic in an interesting way
- Have some user interactivity using event listeners, e.g to move through different sections of content

P.3 Use of Trello

Note:

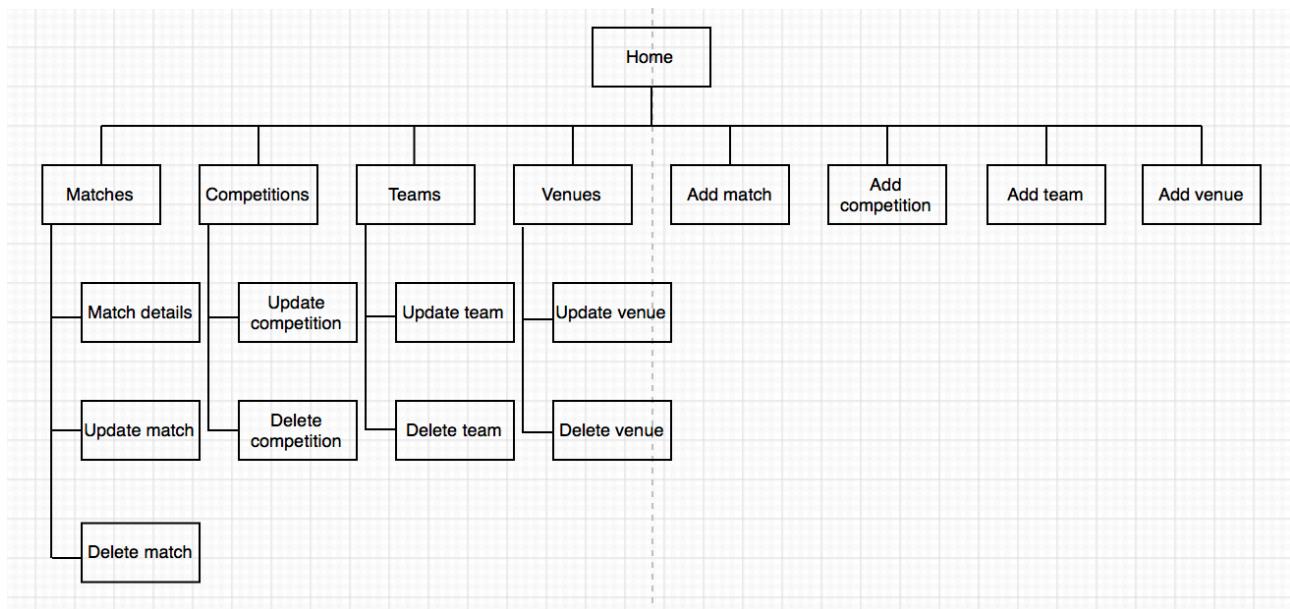
During the group project we were working all the time physically in the same room (at the same table). Therefore we decided, after a while, to move from Trello to white board and Post-it Notes. We all agreed that the physical act of moving a Post-it Note when a task has been completed is much more satisfying than moving or marking a box on Trello



P.4 Acceptance criteria

Acceptance Criteria	Expected Result / Output	Pass / Fail
A user is able to access a list of tasks, sorted by priority.	A list of tasks, sorted by priority, is displayed when the app is opened.	Pass
A user is able to add a new task.	When user clicks the ADD A TASK button, a page is displayed where task details (name, description, priority) can be entered. When user clicks the ADD button, the task details are saved.	Pass
A user is able to access the details (name, description, priority) of a task.	When user clicks on a task, a page with the task details is displayed	Pass
A user is able to update the details of a task.	When user changes any task details on the task details page and clicks the UPDATE button, the task details are updated.	Pass
A user is able to mark a task as completed.	When user ticks the Completed tick box on the task details page and clicks the UPDATE button, the task is marked as completed.	Pass
A user is able to delete a task.	When user clicks on the DELETE button on the task details pages and confirms the resulting pop up message, the task is deleted.	Pass

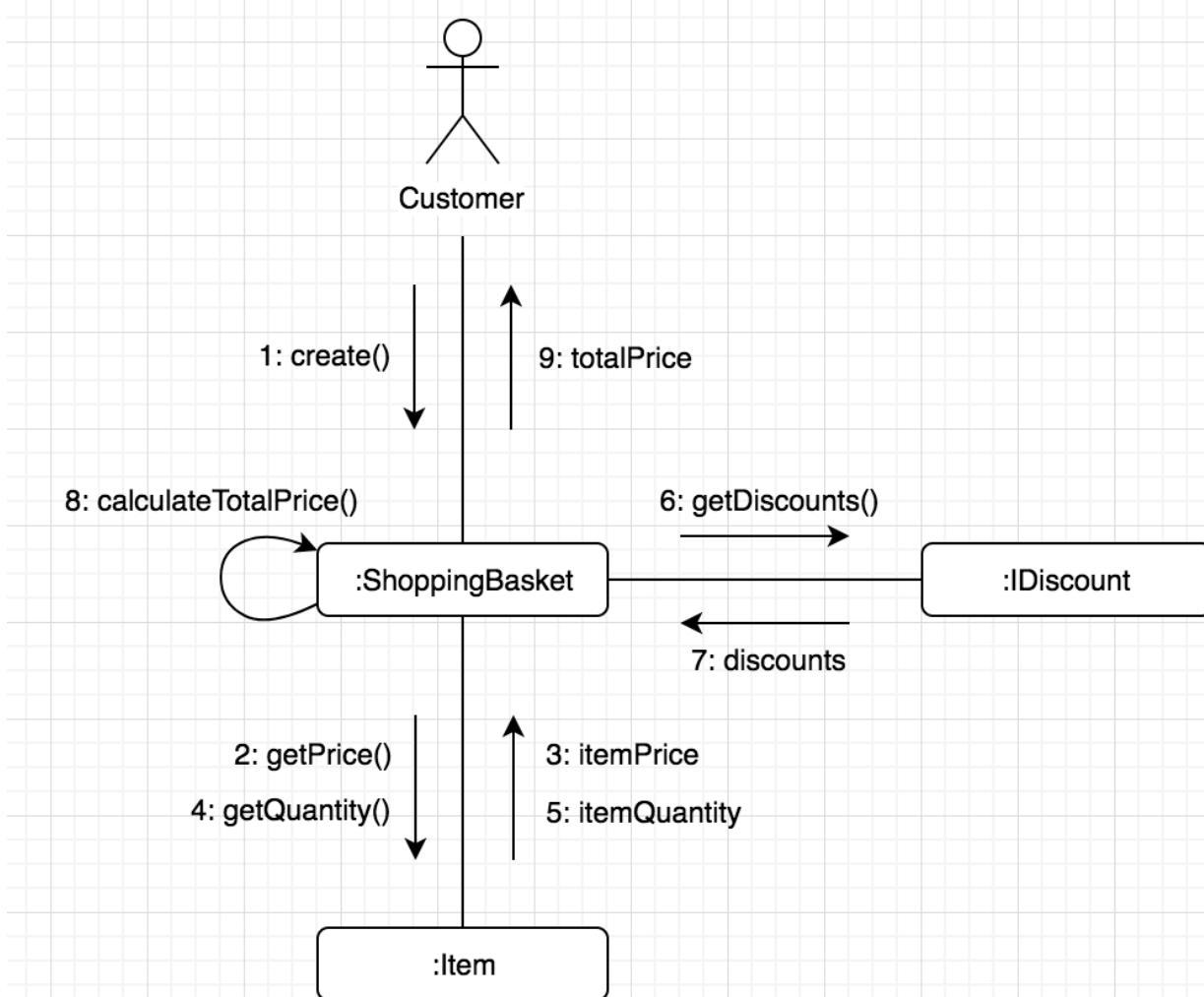
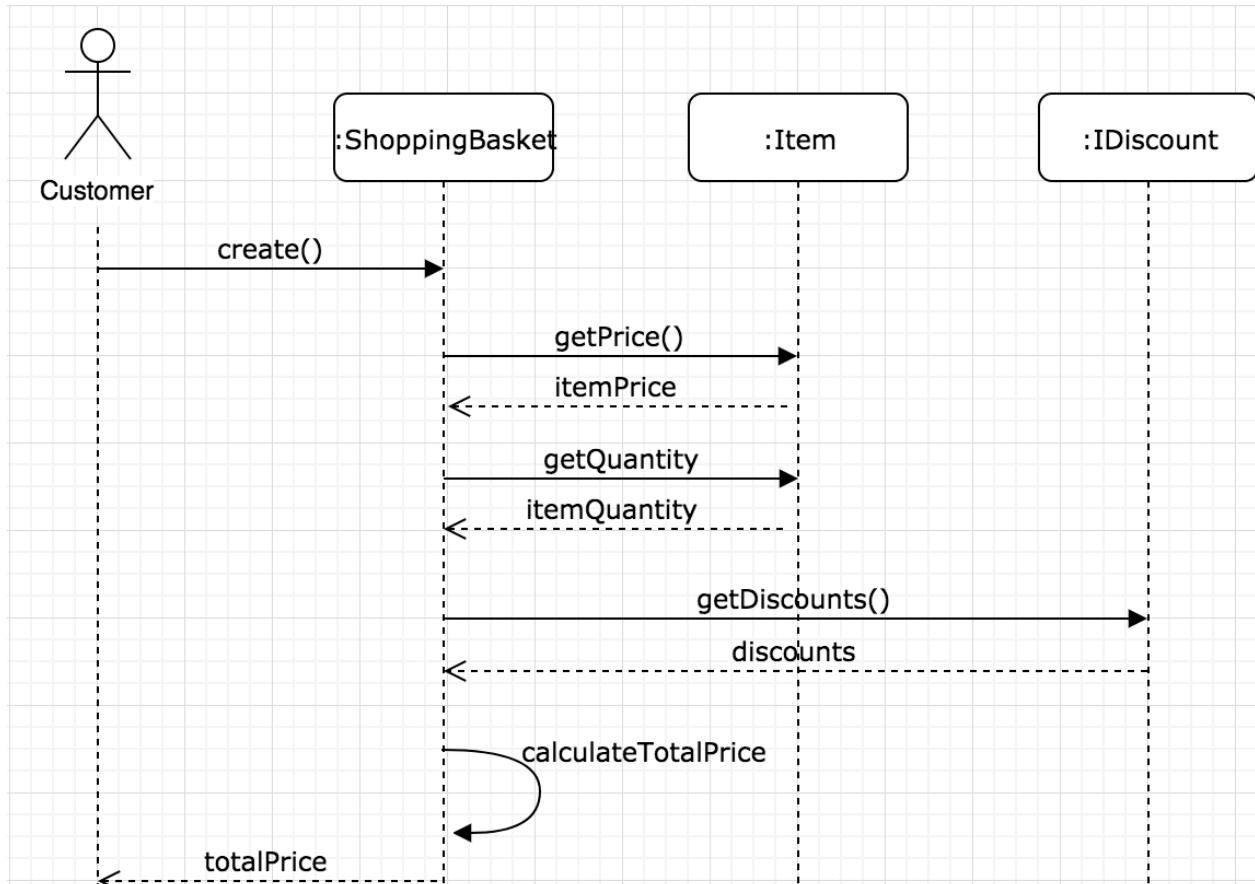
P.5 User sitemap



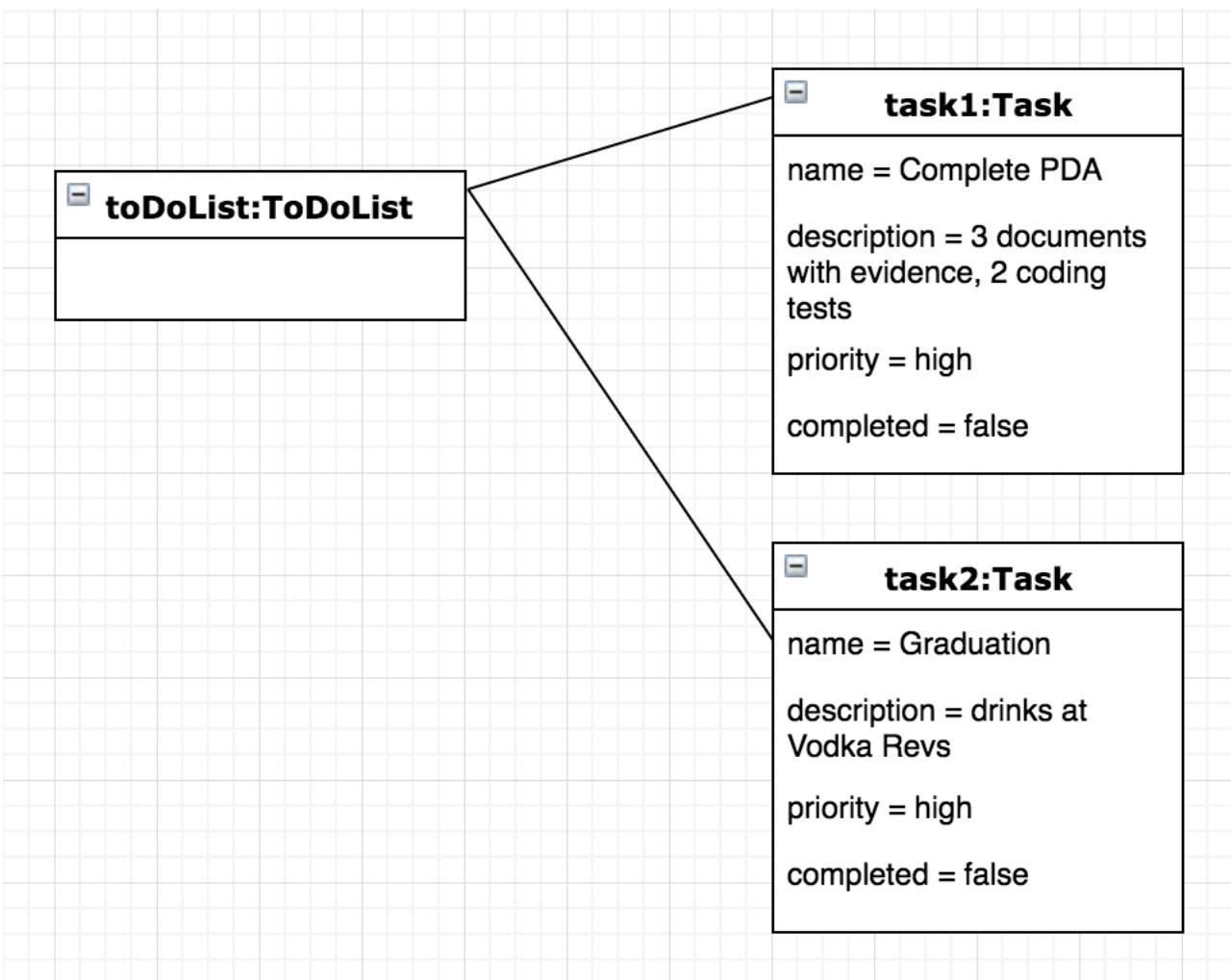
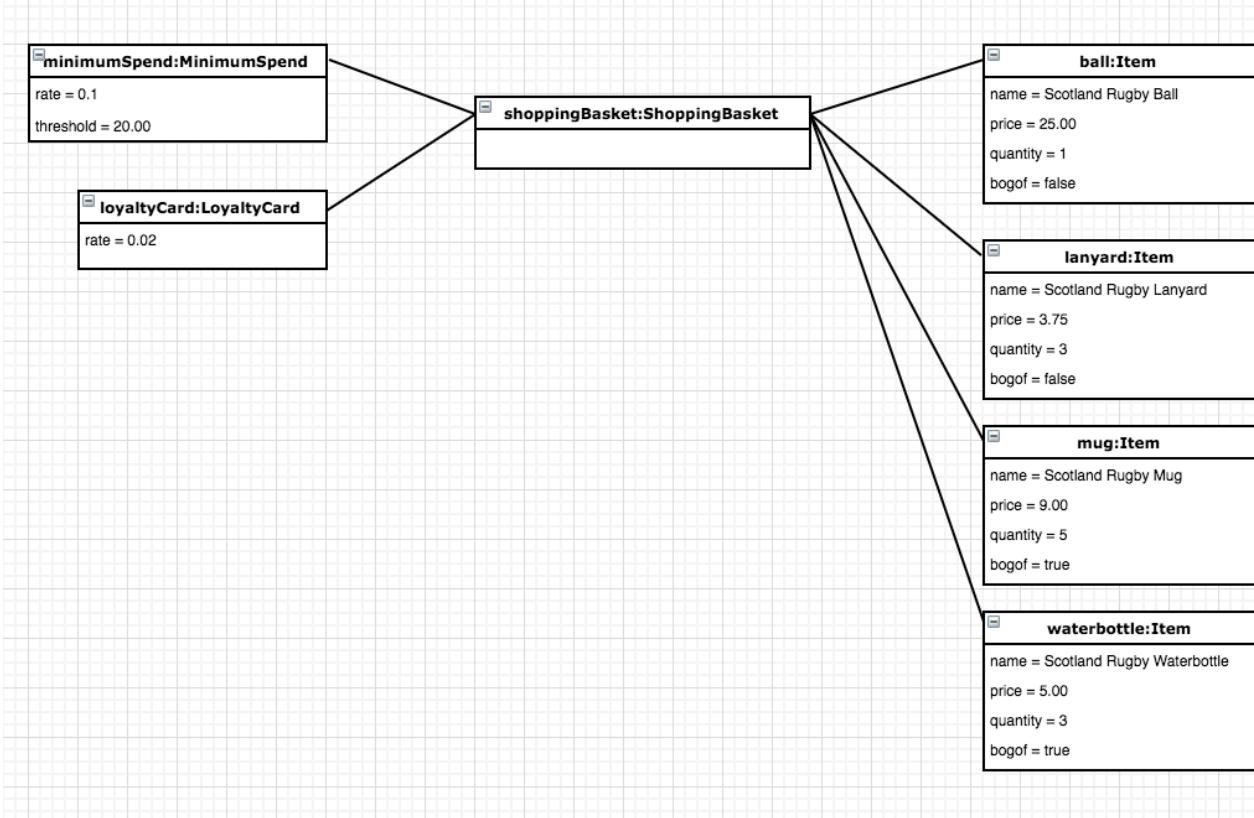
P.6 Two wireframe designs



P.7 Two system interactions diagrams



P.8 Two object diagrams



P.9 Choice of two algorithms

```
❖ string.rb
1 class String
2
3   def sanitize()
4     return self.gsub(/[A-Za-z]+/, &:capitalize).gsub(/\sOf\s/, &:downcase)
5   end
6
7 end
```

The **sanitize** algorithm turns any given string into Title Case. If the word “of” occurs in the string it is converted to lowercase.

This algorithm was used in the “Score Board” project to sanitize rugby team, competition and venue names. Downcasing the word “Of” was added as it occurs regularly in rugby team names (e.g. University of Edinburgh”). Note: to allow for names that not conform to the Title Case pattern (e.g. “PRO14” or “USA”), this algorithm is only used when names are initially entered, not when they are updated, so the user can override the Title Case pattern if desired.

The way it works:

1. A global substitution of any combination of upper-and lowercase letters by its capitalized equivalent.
2. A global substitution of any occurrences of the word “Of” by its downcased equivalent.

```
6 BingeWatch.prototype.calculateDays = function () {
7   const duration_minutes = this.tvShow.calculateTotalDuration();
8   const duration_hours = duration_minutes / 60;
9   const duration_days = Math.ceil(duration_hours / this.hours);
10  return duration_days;
11 }
```

The **calculateDays** algorithm calculates the number of days it will take to bingewatch a tv show, given how much time one wants to spend watching per day (this.hours).

The way it works:

1. The duration of the series in minutes (duration_minutes) is calculated using calculateTotalDuration.
2. The duration in minutes (duration_minutes) is divided by 60 to give the duration in hours (duration_hours).
3. The duration in hours (duration_hours) is divided by the number of hours one wants to spend watching per day (this.hours) and rounded up to the nearest integer (using Math.ceil) to give the duration in days (duration_days).

P.10 Example of pseudocode

```
1 # play_game function
2
3 # input: player hand and computer hand
4
5 # if the player hand beats the computer hand:
6 #     print message that the player wins
7 #     increase player score with 1 point
8 # else if the computer hand beats the player hand:
9 #     print message that the computer wins
10 #    increase computer score with 1 point
11 # else:
12 #     print message that it is a draw
13 #     keep scores as they are
14 # end
```

P.11 Github link to one of your projects

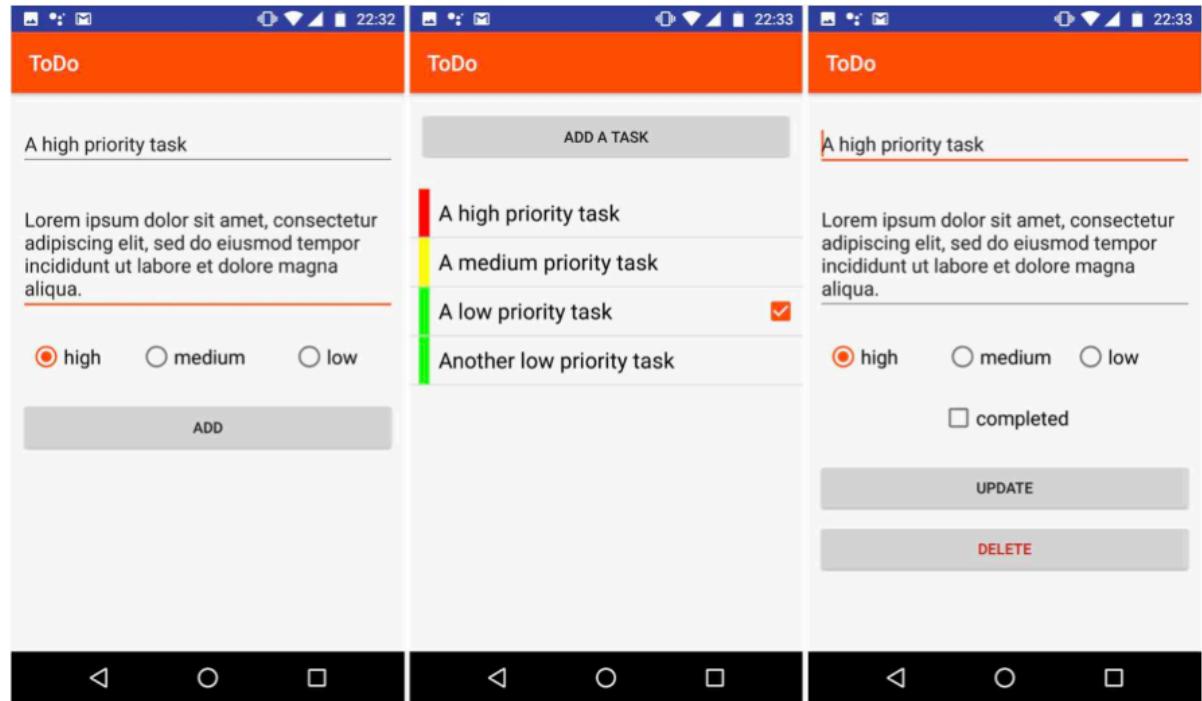
https://github.com/boverdui/to_do_list_project

The screenshot shows the GitHub repository page for 'boverdui/to_do_list_project'. The top navigation bar includes links for 'This repository', 'Search', 'Pull requests', 'Issues', 'Marketplace', and 'Explore'. On the right, there are icons for notifications, a plus sign, and a user profile. Below the header, the repository name 'boverdui / to_do_list_project' is displayed, along with buttons for 'Unwatch' (with a count of 1), 'Star' (with a count of 0), and 'Fork' (with a count of 0). A navigation bar below the repository name includes 'Code' (selected), 'Issues 0', 'Pull requests 0', 'Projects 1', 'Wiki', 'Insights', and 'Settings'. A note below the navigation bar states 'No description, website, or topics provided.' with an 'Edit' button. Below this, a summary bar shows '20 commits', '1 branch', '0 releases', and '1 contributor'. A 'Branch: master' dropdown is followed by buttons for 'New pull request', 'Create new file', 'Upload files', 'Find file', and a prominent green 'Clone or download' button. The main content area displays a list of recent commits:

Author	Commit Message	Date
boverdui	Update README.md	Latest commit 5b71ce3 28 days ago
ToDoList	Added screenshots for readme	28 days ago
README.md	Update README.md	28 days ago

Below the commits, a section titled 'Java / Android' is shown. It includes a 'Goal:' section with the text 'Practice what we learned in the Java / Android weeks and research how to persist data.', an 'MVP' section with the text 'You are required to write an Android app which will allow the user to maintain a task list.', and a 'Project Extensions' section with two bullet points: 'Allow the user to prioritise and/or categorise tasks.' and 'Use the calendar to set due dates on tasks.'

Screenshots of the Finished Product:



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

This repository Search Pull requests Issues Marketplace Explore

boverdui / to_do_list_project

Code Issues 0 Pull requests 0 Projects 1 Wiki Insights Settings

android_to_do_list_project Updated just now

Column	Tasks
Must haves (MVP)	<ul style="list-style-type: none"> Set up database. Add functionality to add a task. Create view displaying a list of tasks showing only the bare details of each task. Add functionality to select a task from the list. Create view displaying the full details of a selected task. Add functionality to mark a task as completed. Add functionality to update a task. Add functionality to delete a task.
Should haves (Extensions)	<ul style="list-style-type: none"> Add functionality to prioritise tasks. Add functionality to categorise tasks. Add functionality to set a due date for a task.
Could haves	<ul style="list-style-type: none"> Add functionality to send a notification when the due date for a task has been reached.
In progress	<ul style="list-style-type: none"> Add functionality to send a notification when the due date for a task has been reached.
Done	

This repository Search Pull requests Issues Marketplace Explore

boverdui / to_do_list_project

Code Issues 0 Pull requests 0 Projects 1 Wiki Insights Settings

android_to_do_list_project Updated just now

Column	Tasks
Must haves (MVP)	<ul style="list-style-type: none"> Add functionality to update a task. Add functionality to delete a task.
Should haves (Extensions)	<ul style="list-style-type: none"> Add functionality to prioritise tasks. Add functionality to categorise tasks. Add functionality to set a due date for a task.
Could haves	<ul style="list-style-type: none"> Add functionality to send a notification when the due date for a task has been reached.
In progress	<ul style="list-style-type: none"> Add functionality to mark a task as completed.
Done	<ul style="list-style-type: none"> Set up database. Create view displaying a list of tasks showing only the bare details of each task. Add functionality to add a task. Add functionality to select a task from the list. Create view displaying the full details of a selected task.

This repository Search Pull requests Issues Marketplace Explore

boverdui / to_do_list_project

Code Issues 0 Pull requests 0 Projects 1 Wiki Insights Settings

android_to_do_list_project Updated just now

0 Must haves (MVP) + ...

2 Should haves (Extensions) + ...

1 Could haves + ...

0 In progress + ...

9 Done + ...

Add functionality to categorise tasks.
Added by boverdui

Add functionality to set a due date for a task.
Added by boverdui

Add functionality to send a notification when the due date for a task has been reached.
Added by boverdui

Set up database.
Added by boverdui

Create view displaying a list of tasks showing only the bare details of each task.
Added by boverdui

Add functionality to add a task.
Added by boverdui

Add functionality to select a task from the list.
Added by boverdui

Create view displaying the full details of a selected task.
Added by boverdui

Add functionality to mark a task as completed.
Added by boverdui

Add functionality to update a task.
Added by boverdui

Add functionality to delete a task.
Added by boverdui

Add functionality to prioritise tasks.
Added by boverdui

0 Must haves (MVP) + ...

2 Should haves (Extensions) + ...

1 Could haves + ...

0 In progress + ...

9 Done + ...

Add functionality to categorise tasks.
Added by boverdui

Add functionality to set a due date for a task.
Added by boverdui

Add functionality to send a notification when the due date for a task has been reached.
Added by boverdui

Set up database.
Added by boverdui

Create view displaying a list of tasks showing only the bare details of each task.
Added by boverdui

Add functionality to add a task.
Added by boverdui

Add functionality to select a task from the list.
Added by boverdui

Create view displaying the full details of a selected task.
Added by boverdui

Add functionality to mark a task as completed.
Added by boverdui

Add functionality to update a task.
Added by boverdui

Add functionality to delete a task.
Added by boverdui

Add functionality to prioritise tasks.
Added by boverdui

P.13 User input / P.14 Interaction with data persistence

Score Board

Matches Competitions Teams Venues Add a match Add a competition Add a team Add a venue

104 matches Filter... win loss draw

Date	Venue	Competition	Home team	Away team	Actions
2018-03-24	St James' Park	Premiership Rugby	Newcastle Falcons	25 22	Northampton Saints
2018-03-17	Stadio Olimpico	Six Nations	Italy	27 29	Scotland
2018-03-03	NRCA Stadium	Rugby Europe Trophy	Netherlands	71 30	Poland
2018-02-24	Murrayfield Stadium	Six Nations	Scotland	25 13	England
2018-02-23	Myreside Stadium	U20 Six Nations	Scotland U20	24 17	England U20
2018-02-11	Murrayfield Stadium	Six Nations	Scotland	32 26	France
2018-02-09	Myreside Stadium	PRO14	Edinburgh Rugby	29 24	Leinster Rugby
2018-01-12	Murrayfield Stadium	European Rugby Challenge Cup	Edinburgh Rugby	34 33	Stade Francais
2017-12-23	Murrayfield Stadium	PRO14	Edinburgh Rugby	18 17	Glasgow Warriors
2017-12-15	Murrayfield Stadium	European Rugby Challenge Cup	Edinburgh Rugby	78 0	Krasny Yar
2017-11-25	Murrayfield Stadium	Autumn Internationals	Scotland	53 24	Australia
2017-11-18	Murrayfield Stadium	Autumn Internationals	Scotland	17 22	New Zealand
2017-11-11	Murrayfield Stadium	Autumn Internationals	Scotland	44 38	Samoa
2017-09-23	Murrayfield Stadium	Other	The University of Edinburgh	31 7	University of St Andrews
2017-09-15	Myreside Stadium	PRO14	Edinburgh Rugby	17 20	Benetton Rugby
2017-09-10	Bridgnorth Park	Other	Glasgow Warriors	14 19	Northampton Saints

Score Board

Matches Competitions Teams Venues Add a match Add a competition Add a team Add a venue

Add a new match:

Date: 31/03/2018

Venue: Murrayfield Stadium

Home team: Edinburgh Rugby

Home team score: 6

Away team score: 20

Away team: Cardiff Blues

Competition: European Rugby Challenge Cup

Comments:

Score Board

Matches Competitions Teams Venues Add a match Add a competition Add a team Add a venue

105 matches Filter... win loss draw

Date	Venue	Competition	Home team	Away team	Actions
2018-03-31	Murrayfield Stadium	European Rugby Challenge Cup	Edinburgh Rugby	6 20	Cardiff Blues
2018-03-24	St James' Park	Premiership Rugby	Newcastle Falcons	25 22	Northampton Saints
2018-03-17	Stadio Olimpico	Six Nations	Italy	27 29	Scotland
2018-03-03	NRCA Stadium	Rugby Europe Trophy	Netherlands	71 30	Poland
2018-02-24	Murrayfield Stadium	Six Nations	Scotland	25 13	England
2018-02-23	Myreside Stadium	U20 Six Nations	Scotland U20	24 17	England U20
2018-02-11	Murrayfield Stadium	Six Nations	Scotland	32 26	France
2018-02-09	Myreside Stadium	PRO14	Edinburgh Rugby	29 24	Leinster Rugby
2018-01-12	Murrayfield Stadium	European Rugby Challenge Cup	Edinburgh Rugby	34 33	Stade Francais
2017-12-23	Murrayfield Stadium	PRO14	Edinburgh Rugby	18 17	Glasgow Warriors
2017-12-15	Murrayfield Stadium	European Rugby Challenge Cup	Edinburgh Rugby	78 0	Krasny Yar
2017-11-25	Murrayfield Stadium	Autumn Internationals	Scotland	53 24	Australia
2017-11-18	Murrayfield Stadium	Autumn Internationals	Scotland	17 22	New Zealand
2017-11-11	Murrayfield Stadium	Autumn Internationals	Scotland	44 38	Samoa
2017-09-23	Murrayfield Stadium	Other	The University of Edinburgh	31 7	University of St Andrews
2017-09-15	Murrayfield Stadium	PRO14	Edinburgh Rugby	47 33	Bath Rugby

P.15 User output result

Score Board

localhost:4567/matches

Matches Competitions Teams Venues Add a match Add a competition Add a team Add a venue

105 matches Filter... win loss draw

Date ▾▼	Venue ▾▼	Competition ▾▼	Home team ▾▼	Away team ▾▼	Actions
2018-03-31	Murrayfield Stadium	European Rugby Challenge Cup	Edinburgh Rugby	6 20 Cardiff Blues	
2018-03-24	St James' Park	Premiership Rugby	Newcastle Falcons	25 22 Northampton Saints	
2018-03-17	Stadio Olimpico	Six Nations	Italy	27 29 Scotland	
2018-03-03	NRCA Stadium	Rugby Europe Trophy	Netherlands	71 30 Poland	
2018-02-24	Murrayfield Stadium	Six Nations	Scotland	25 13 England	
2018-02-23	Myreside Stadium	U20 Six Nations	Scotland U20	24 17 England U20	
2018-02-11	Murrayfield Stadium	Six Nations	Scotland	32 26 France	
2018-02-09	Myreside Stadium	PRO14	Edinburgh Rugby	29 24 Leinster Rugby	
2018-01-12	Murrayfield Stadium	European Rugby Challenge Cup	Edinburgh Rugby	34 33 Stade Francais	
2017-12-23	Murrayfield Stadium	PRO14	Edinburgh Rugby	18 17 Glasgow Warriors	
2017-12-15	Murrayfield Stadium	European Rugby Challenge Cup	Edinburgh Rugby	78 0 Krasny Yar	
2017-11-25	Murrayfield Stadium	Autumn Internationals	Scotland	53 24 Australia	
2017-11-18	Murrayfield Stadium	Autumn Internationals	Scotland	17 22 New Zealand	
2017-11-11	Murrayfield Stadium	Autumn Internationals	Scotland	44 38 Samoa	
2017-09-23	Murrayfield Stadium	Other	The University of Edinburgh	31 7 University of St Andrews	
2017-09-16	Murrayfield Stadium	PRO14	Edinburgh Rugby	17 20 Benetton Rugby	

Score Board

localhost:4567/matches/67?

Matches Competitions Teams Venues Add a match Add a competition Add a team Add a venue

Match:

Date:
2018-02-24

Venue:
Murrayfield Stadium, Edinburgh, Scotland

Competition:
Six Nations

Home team:
Scotland

Home team score:
25

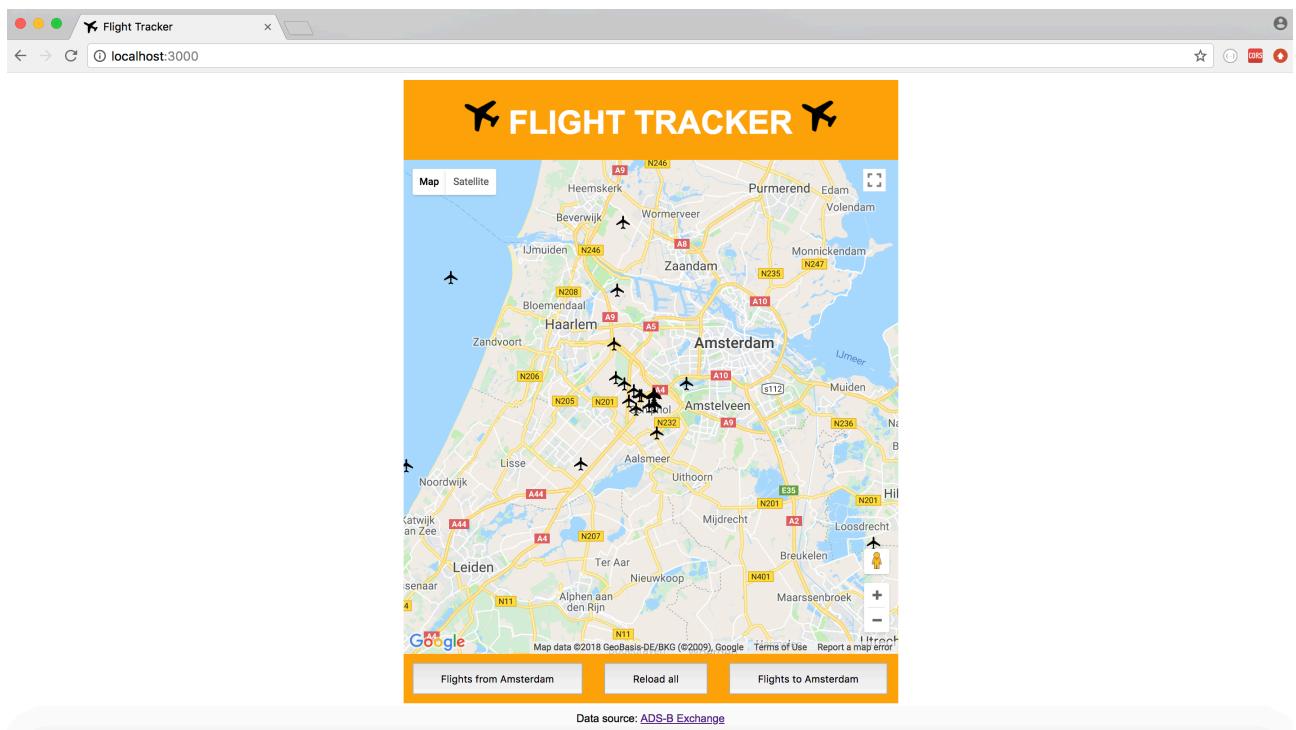
Away team score:
13

Away team:
England

Comments:
Referee: Nigel Owens. Scotland: tries by Huw Jones (2x) and Sean Maitland; conversions by Greig Laidlaw (2x); penalties by Greig Laidlaw and Finn Russell. England: try by Owen Farrell; conversion by Owen Farrell; penalties by Owen Farrell (2x).

P.16 Show an API being used within your program

```
JS flight_data.js
1 const Request = require('../helpers/request.js');
2
3 const FlightData = function() {
4     this.url = 'http://public-api.adsbexchange.com/VirtualRadar/AircraftList.json';
5     this.data = null;
6 }
7
8 FlightData.prototype.getData = function(onComplete) {
9     const request = new Request(this.url);
10    request.get((data) => {
11        this.data = data;
12        onComplete(data);
13    });
14 }
15
16 module.exports = FlightData;
```



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

Most recent match should be shown at the top of the list.	Failed	Order matches by date descending.	Passed
Names of competitions, teams, and venues should always be in Title Case, no matter how the user entered them.	Failed	Added name sanitize method.	Passed
User should be able to override the Title Case requirement for competition, team, and venue names if necessary (e.g. for competition "PRO14" or team "USA").	Failed	Don't invoke name sanitize method when updating competition, team, and venue names.	Passed
It should not be possible to add an already existing competition, team, or venue.	Failed	Added a check whether the competition, team, or venue already exists.	Passed
It shouldn't be possible to delete a competition, team, or venue when it's used in any of the existing matches.	Failed	Disable delete button for competitions, teams, and venues that are used in any of the existing matches	Passed

P.18 Testing your program

```
1  require("minitest/autorun")
2  require_relative("../card")
3  require_relative("../cardgame")
4
5  class TestCardGame < MiniTest::Test
6
7    def setup()
8      @cardgame = CardGame.new()
9      @card_1 = Card.new("Spades", 1)
10     @card_7 = Card.new("Spades", 7)
11     @card_10 = Card.new("Spades", 10)
12     @cards = [@card_1, @card_7, @card_10]
13   end
14
15  def test_check_for_ace_ace()
16    assert_equal(true, @cardgame.check_for_ace(@card_1))
17  end
18
19  def test_check_for_ace_not_ace()
20    assert_equal(false, @cardgame.check_for_ace(@card_7))
21  end
22
23  def test_highest_card_card1_higher()
24    assert_equal(@card_7.value, @cardgame.highest_card(@card_7, @card_1))
25  end
26
27  def test_highest_card_card2_higher()
28    assert_equal(@card_10.value, @cardgame.highest_card(@card_7, @card_10))
29  end
30
31  def test_check_cards_total()
32    assert_equal("You have a total of 18", @cardgame.cards_total(@cards))
33  end
34
35 end
```

```
[→ code git:(master) ✘ ruby specs/cardgame_spec.rb
Run options: --seed 31130

# Running:

EEEEEE

Finished in 0.001337s, 3739.7160 runs/s, 0.0000 assertions/s.

  1) Error:
TestCardGame#test_highest_card__card2_higher:
NoMethodError: undefined method `highest_card' for #<CardGame:0x007f88ec8d8638>
  specs/cardgame_spec.rb:28:in `test_highest_card__card2_higher'

  2) Error:
TestCardGame#test_check_for_ace__not_ace:
NoMethodError: undefined method `check_for_ace' for #<CardGame:0x007f88ed0e7b58>
  specs/cardgame_spec.rb:20:in `test_check_for_ace__not_ace'

  3) Error:
TestCardGame#test_check_cards_total:
NoMethodError: undefined method `cards_total' for #<CardGame:0x007f88ed0e5b50>
  specs/cardgame_spec.rb:32:in `test_check_cards_total'

  4) Error:
TestCardGame#test_check_for_ace__ace:
NoMethodError: undefined method `check_for_ace' for #<CardGame:0x007f88ed0e45e8>
  specs/cardgame_spec.rb:16:in `test_check_for_ace__ace'

  5) Error:
TestCardGame#test_highest_card__card1_higher:
NoMethodError: undefined method `highest_card' for #<CardGame:0x007f88ed22fc90>
  specs/cardgame_spec.rb:24:in `test_highest_card__card1_higher'

5 runs, 0 assertions, 0 failures, 5 errors, 0 skips
```

```
1 class CardGame
2
3     def check_for_ace(card)
4         if card.value == 1
5             return true
6         else
7             return false
8         end
9     end
10
11    def highest_card(card1, card2)
12        if card1.value > card2.value
13            return card1.value
14        else
15            return card2.value
16        end
17    end
18
19    def cards_total(cards)
20        total = 0
21        for card in cards
22            total += card.value
23        end
24        return "You have a total of " + total.to_s
25    end
26
27 end
```

```
[→ code git:(master) ✘ ruby specs/cardgame_spec.rb
Run options: --seed 53263
```

```
# Running:
```

```
.....
```

```
Finished in 0.001363s, 3668.3788 runs/s, 3668.3788 assertions/s.
```

```
5 runs, 5 assertions, 0 failures, 0 errors, 0 skips
```

