

## **Evidence for Project Unit**

Raul Ruiz

Cohort 15

17/09/17

## P- 1 Github Contributors page

View History Bookmarks People Window Help

Contributors to rauletxailarena x Startpoint

https://github.com/rauletxailarena/group-project/graphs/contributors

Codeclan Jobs to apply Temporada 5 200 universities just..

Pulse

Contributors

Community

Traffic

Commits

Code frequency

Dependency graph

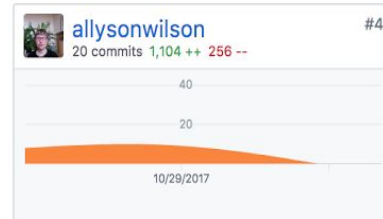
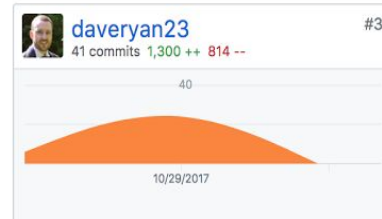
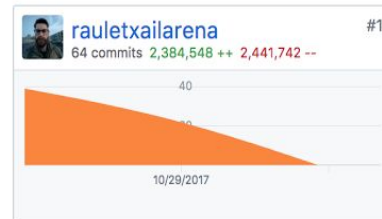
Network

Forks

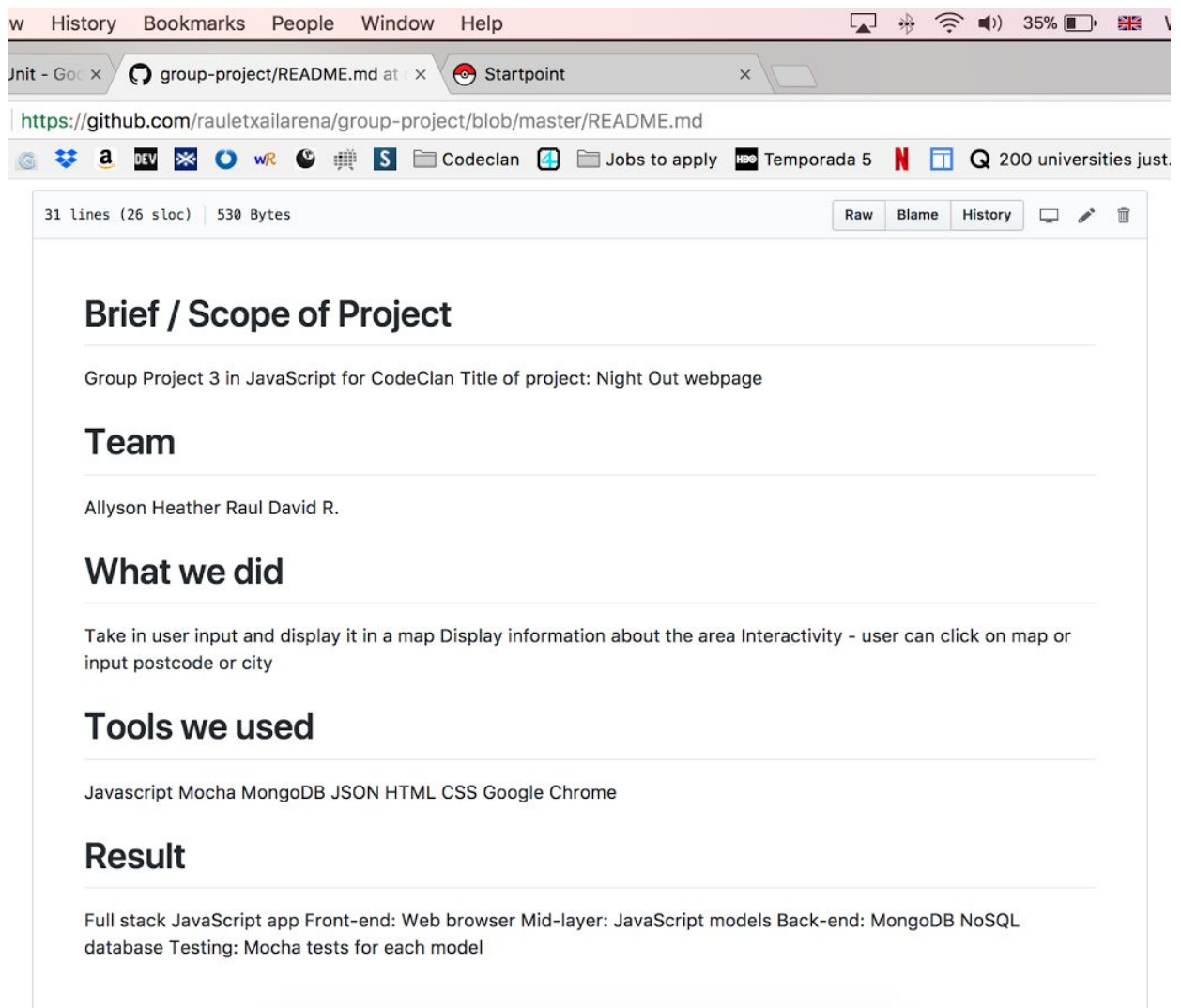
Oct 22, 2017 – Nov 8, 2017

Contributions: Commits

Contributions to master, excluding merge commits



## P- 2 Project Brief



The screenshot shows a web browser window with the address bar displaying the URL <https://github.com/rauletxailarena/group-project/blob/master/README.md>. The browser's address bar also shows the text "group-project/README.md at Startpoint". The browser's tabs include "Jnit - Go", "group-project/README.md at Startpoint", and "Startpoint". The browser's address bar also shows the text "200 universities just".

The README file content is displayed in a light gray box with a white background. It has a header "31 lines (26 sloc) | 530 Bytes" and a "Raw" button. The content is structured as follows:

### Brief / Scope of Project

Group Project 3 in JavaScript for CodeClan Title of project: Night Out webpage

### Team

Allyson Heather Raul David R.

### What we did

Take in user input and display it in a map Display information about the area Interactivity - user can click on map or input postcode or city

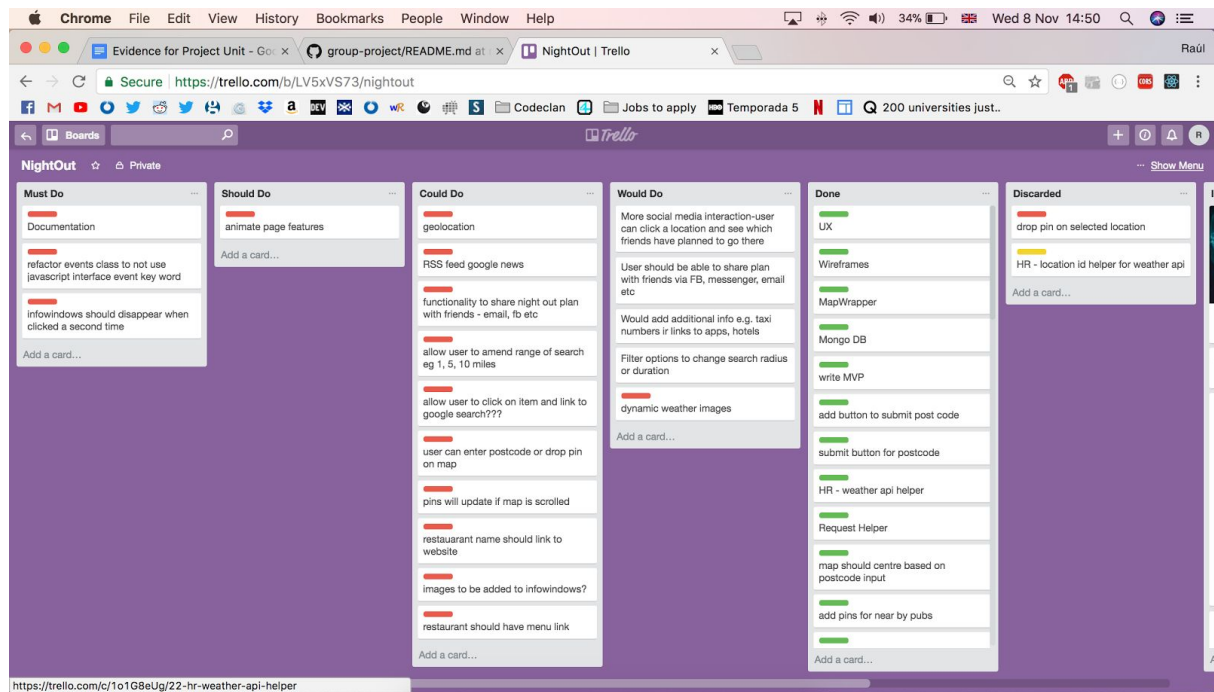
### Tools we used

Javascript Mocha MongoDB JSON HTML CSS Google Chrome

### Result

Full stack JavaScript app Front-end: Web browser Mid-layer: JavaScript models Back-end: MongoDB NoSQL database Testing: Mocha tests for each model

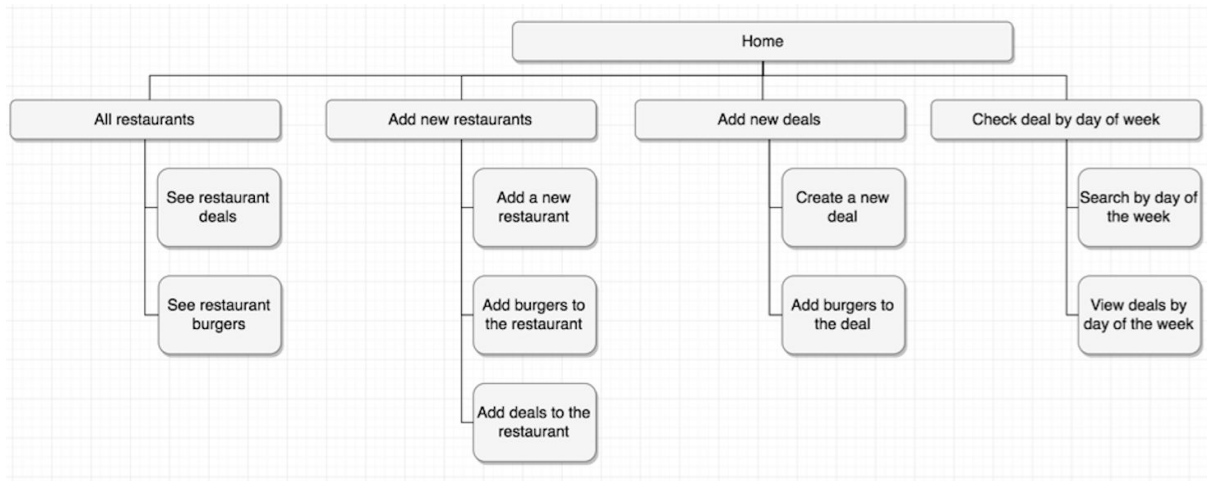
## P-3 Planning during the group project



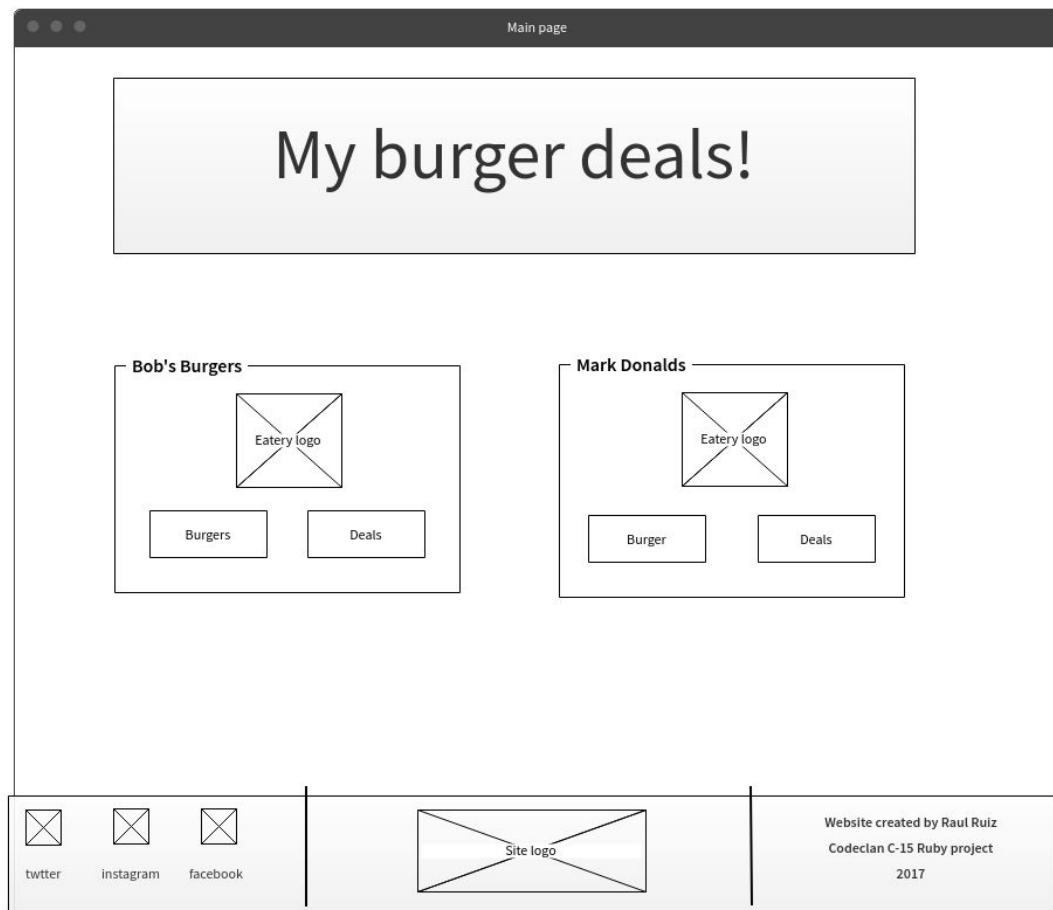
## P-4 Acceptance Criteria

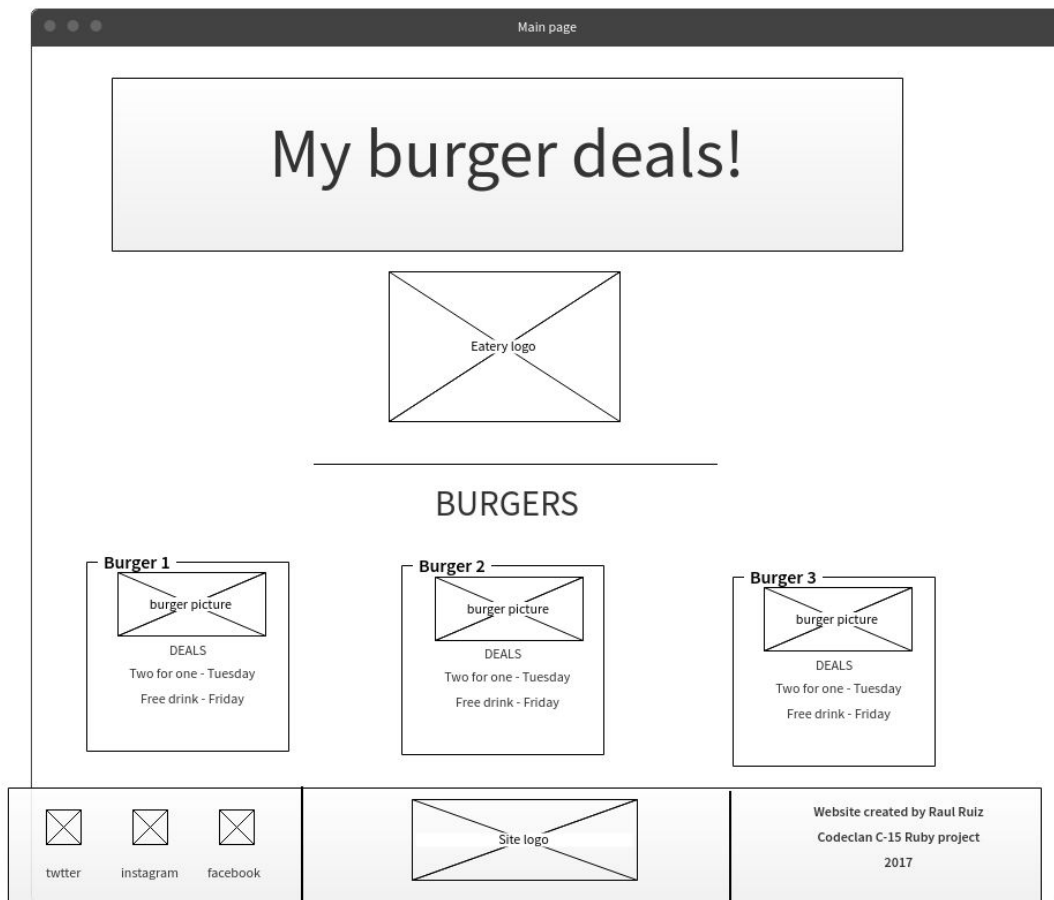
| Acceptance Criteria                                    | Expected Result  | Pass/Fail |
|--|--|-----------|
| User can search by postcode                            | User will obtain data based on a postcode                                      | Pass      |
| Data filtered by postcode                              | The information displayed must be bound to the postcode entered by the user    | Pass      |
| Users can manage their preferences                     | Users can add and remove places to their <i>favourites</i> list                | Pass      |
| User can display their favourite places                | Users can see their favourite places in a map                                  | Pass      |
| Users can obtain additional information about the area | Users receive weather conditions from the specific place and the specific date | Pass      |

## P-5 User sitemap



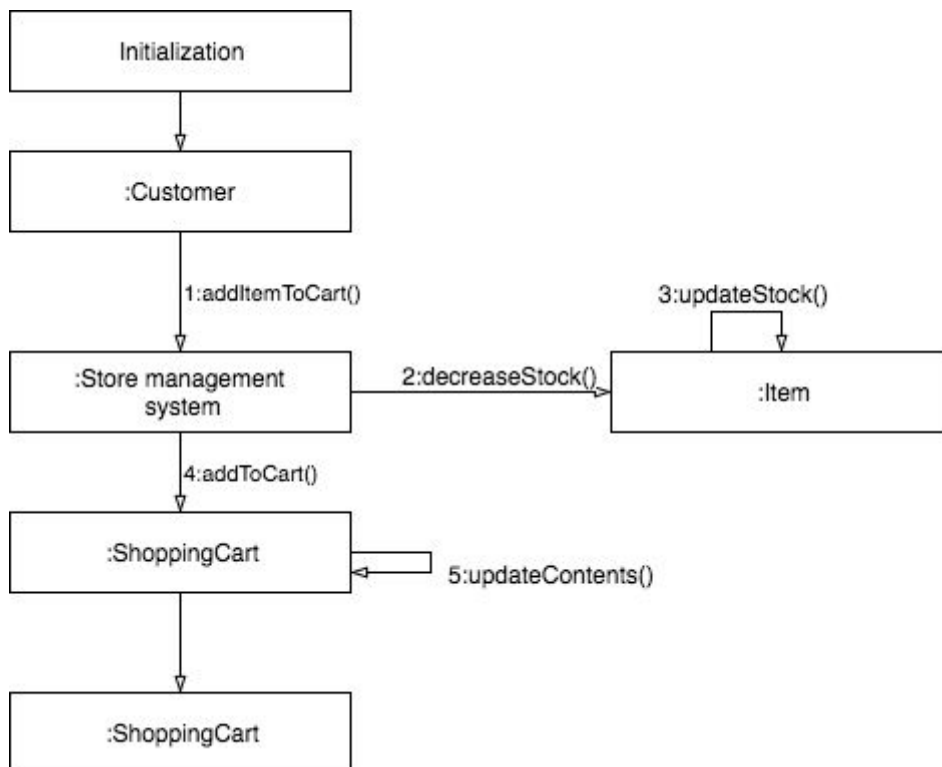
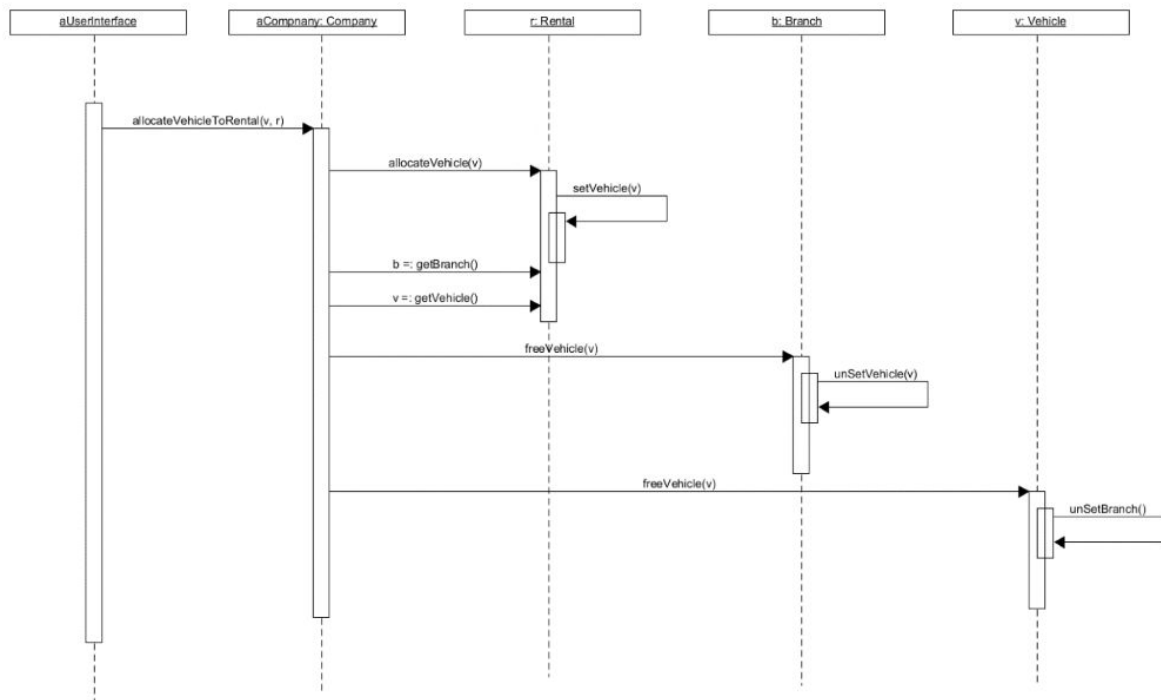
## P-6 Wireframes designs



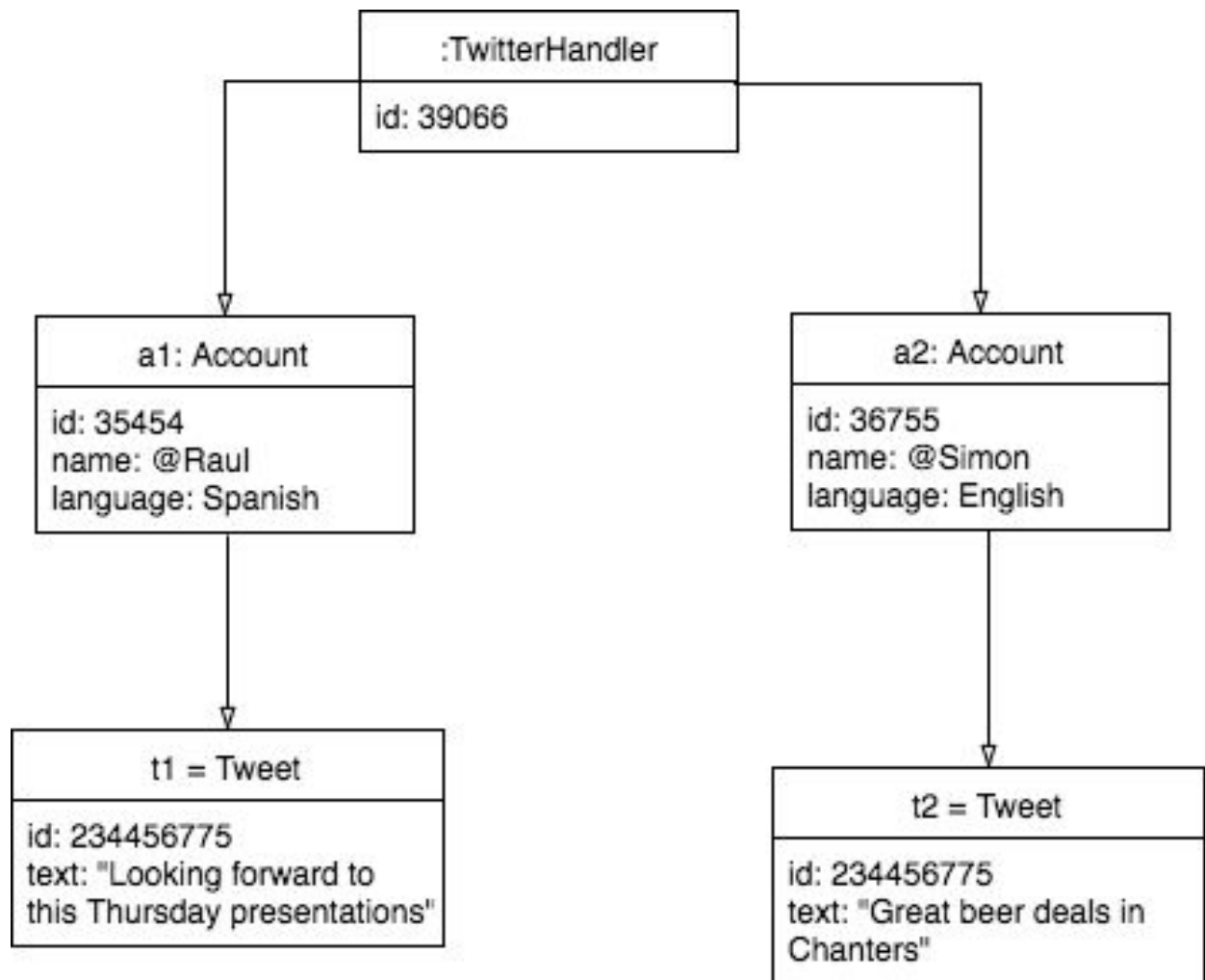


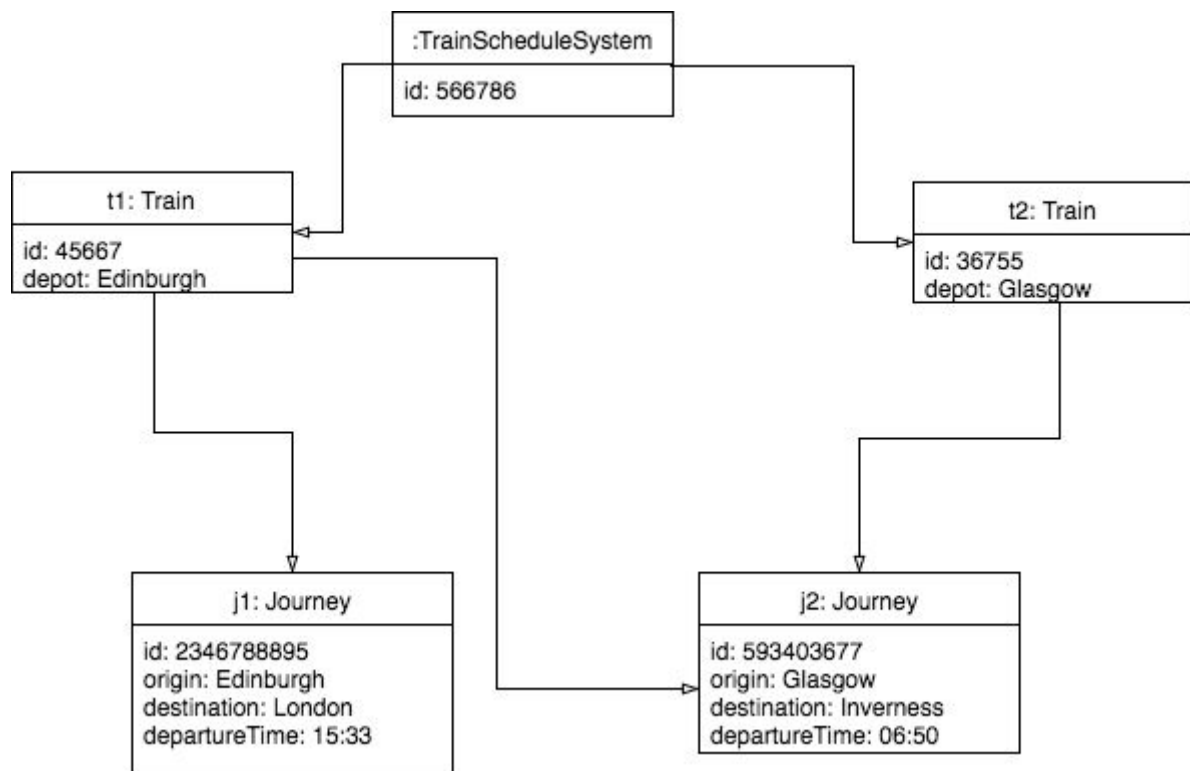


## P-7 System interactions diagrams



## P-8 Two Object Diagrams





## P- 9 Choice of two algorithms

### Algorithm 1: Pokemon move selector

```
untitled  app.js
72
73 var getPokemonAttacksArray = function(pokemonObject, pokemon){
74   var attacksArray = [];
75   for (var i = 0; i < 4; i++){
76     var move = pokemonObject.moves[Math.floor(Math.random() * pokemonObject.moves.length)].move;
77     attacksArray.push(move);
78   }
79   pokemon.moves = attacksArray
80   console.log("Pokemon attacks:", pokemon.moves);
81 }
```

I used this algorithm to loop through a Pokemon object (obtained from an API request), select four moves randomly, and assign those four attacks to my model of that pokemon object.

### Algorithm 2: Prime number function

```
13 var isPrime = function(number){
14   if (number === 1) return false;
15   for (var i = (number - 1); i > 1; i--){
16     if (number % i === 0) {
17       return false;
18     }
19   }
20   return true;
21 }
```

This algorithm was used to determine if a specific number was prime or not. The way it works is as follows: If the number is 1, because it is an exception to the prime rule, it returns false. Else, the function will loop through each number from 1 to  $n - 1$  and assign its value to  $i$ , divide the number by the value of  $i$  and return false only if the modulo value of the division is 0. Else, it will return true, which means that the number is prime.

## **P - 10 Example of Pseudocode**

```
def find_deals_by_day_of_the_week(day_of_the_week)  
# each time this function is called there should be a series of eateries, a series of  
burgers and a series of deals  
# the function should be able to loop through the eateries  
# within the first loop it should also loop through all the deals  
# when looping through the deals it should check for the day of the week that the deal  
applies to  
# if the deal applies to the target day of the week, the function should add that deal to  
an array  
# after both loops finish, the function should return an empty array if there are no  
deals for the target day of the week of populated with deals if there are  
end
```

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

The image displays two screenshots. The top screenshot shows a web application titled "Burger deals tracker" running on a local host. It features four restaurant cards: Bob's Burgers, McDonald's, Krusty Burger, and The Krusty Krab. Each card lists burgers and deals, with buttons to edit or remove the restaurant. The bottom screenshot shows the GitHub repository for the project, "rauletxailarena/burger\_tracker\_project". It includes statistics like 28 commits, 1 branch, and 0 releases, along with a list of recent commits and a button to add a README.

**Burger deals tracker**

See all the restaurants   Add a new restaurant   Add a new deal   Check today's deals!

**Bob's Burgers**

**Burgers**

SYMPATHY FOR THE DEVILED EGG BURGER

SWEET HOME AVOCADO BURGER

AS GOUDA AS IT GETS

**Deals**

2x1 Tuesday

Bring your cat - get 1/3 off your burger

**McDonald's**

**Burgers**

McCHICKEN

SZECHUAN BURGER

McCHICKEN DeLuxe

**Deals**

2x1 Tuesday. Now on Friday!

Clean your own tray - get one burger for free

**Krusty Burger**

**Burgers**

DEEP FRIED KRUSTY BURGER

SLIDESHOW BOB FOOT LONG

HAGGIS BURGER

**Deals**

All you can eat!

Free cursed frogurt

**The Krusty Krab**

**Burgers**

KRABBY PATTY BURGER

CHICKEN SEA BURGER

HEARTY SPONGE

**Deals**

Krabbing for a burger deal

Bikini bottom's 2x1 day

**rauletxailarena/burger\_tracker\_project**

28 commits   1 branch   0 releases   1 contributor

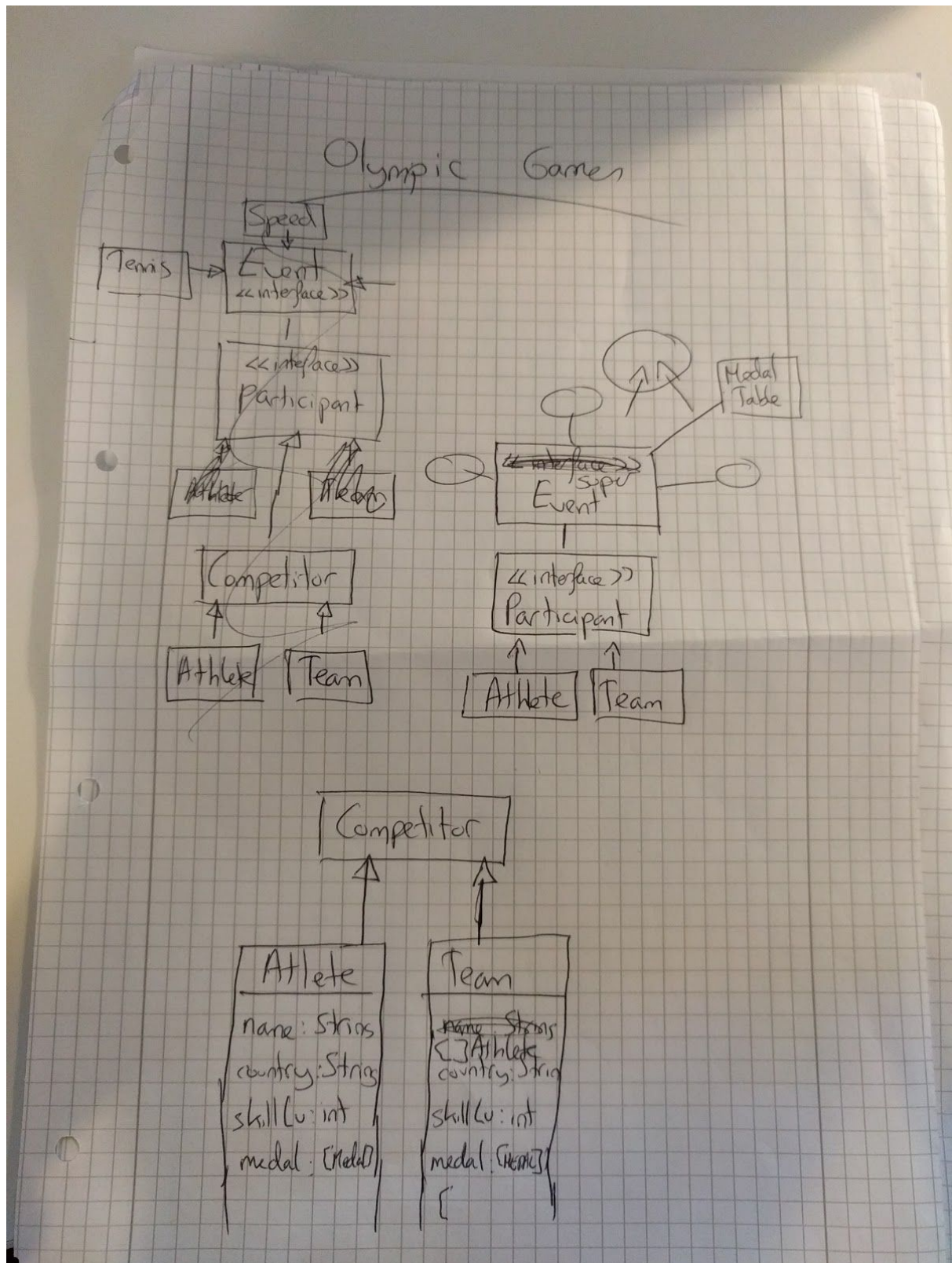
Branch: master   New pull request   Create new file   Upload files   Find file   Clone or download

**rauletxailarena** Fix some typos   Latest commit 88c940b 12 days ago

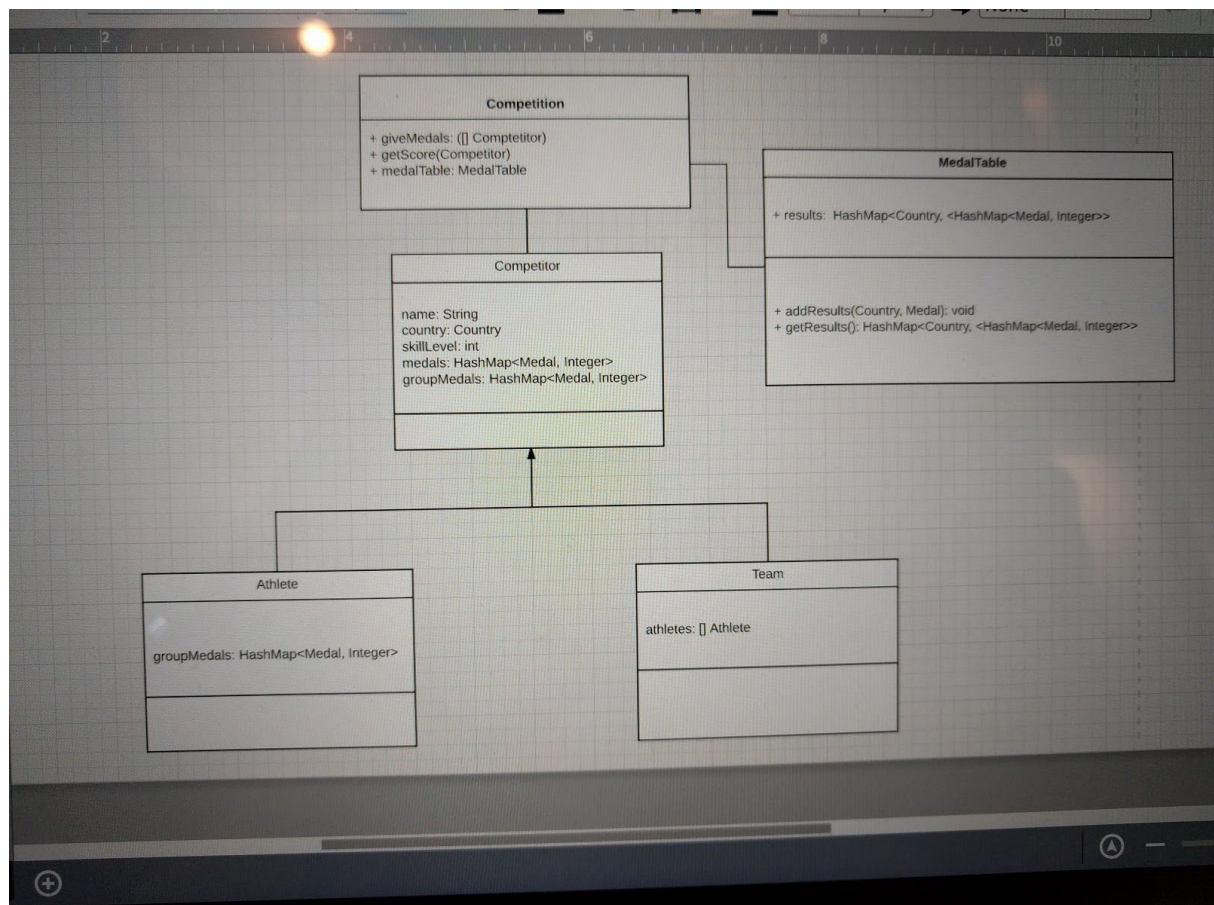
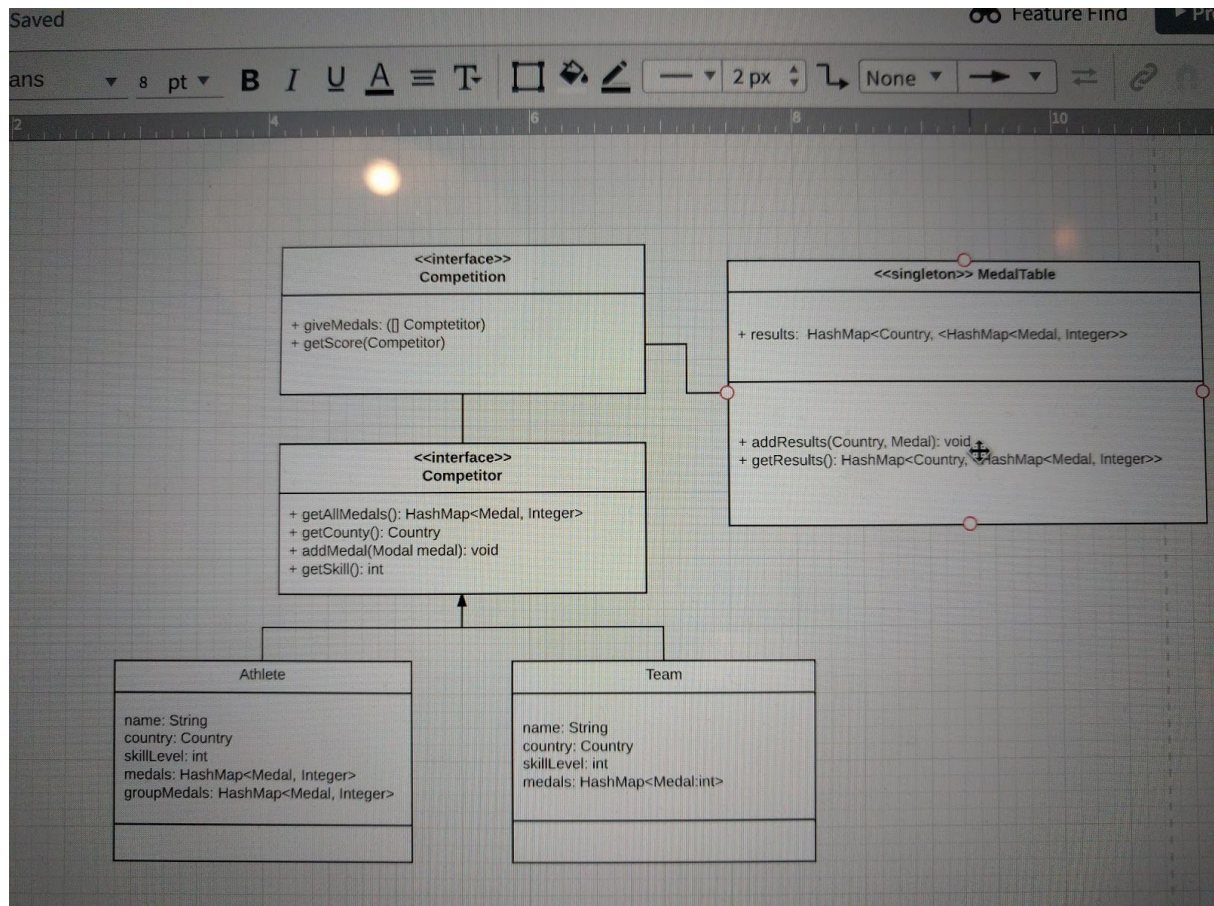
| File        | Commit Message   | Time Ago    |
|-------------|--|-------------|
| controllers | Add a default picture for new eateries                                   | 13 days ago |
| db          | Fix the position of a button   | 13 days ago |
| models      | Add delete burger from deal  | 13 days ago |
| public      | Fix some typos   | 12 days ago |
| views       | Fix some typos   | 12 days ago |
| wireframes  | Add wireframes, two controllers and All() method in Eatery.rb            | 18 days ago |
| TODO.md     | Add a few small details  | 13 days ago |
| app.rb      | Add a find by day of the week method and controller                      | 16 days ago |
| projects.md | Add method burgers() to Deal, method burgers() to eatery and deals() ... | 19 days ago |

Help people interested in this repository understand your project by adding a README.   Add a README

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










## P - 13 User input

# Burger deals tracker

Add a new restaurant Add a new deal

Restaurant:

Logo URL (optional):



McCHICKEN

SZECHUAN BURGER

McCHICKEN DeLuxe

### Deals

*2x1 Tuesday. Now on Friday!*

*Clean your own tray - get one burger  
for free*

EDIT RESTAURANT

REMOVE RESTAURANT

DEEP FRIED KRUSTY BURGER

THE MOTHER NATURE BURGER

SLIDESHOW BOB FOOT LONG

### Deals

*All you can eat!*

*Free cursed frogurt*

EDIT RESTAURANT

REMOVE RESTAURANT



## CODECLAN BURGERS


### Burgers

### Deals

EDIT RESTAURANT

REMOVE RESTAURANT

## P - 14 Interaction with data persistence



ADD MORE BURGERS

GO BACK

**DEEP FRIED KRUSTY BURGER**  
**£6.95**

---

*All you can eat!*  
33% off  
**£4.66**

*Free cursed frogurt*  
40% off  
**£4.17**

EDIT BURGER REMOVE BURGER

**THE MOTHER NATURE BURGER**  
**£6.5**

---

*All you can eat!*  
33% off  
**£4.36**

EDIT BURGER REMOVE BURGER

**SLIDESHOW BOB FOOT LONG**  
**£9.95**

---

*Free cursed frogurt*  
40% off  
**£5.97**

EDIT BURGER REMOVE BURGER

restaurants

Add a new restaurant

Add a new deal

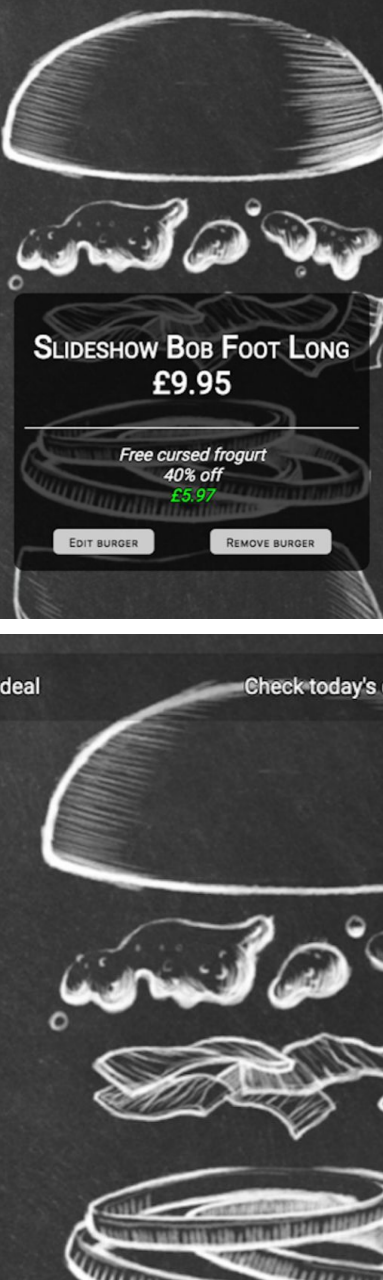
Check today's deals

Restaurant:

Burger name:

Burger price:

EDIT BURGER





ADD MORE BURGERS

GO BACK

## DEEP FRIED KRUSTY BURGER £6.95

All you can eat!  
33% off  
£4.66

Free cursed frogurt  
40% off  
£4.17

EDIT BURGER

REMOVE BURGER

## SLIDESHOW BOB FOOT LONG £9.95

Free cursed frogurt  
40% off  
£5.97

EDIT BURGER

REMOVE BURGER

## HAGGIS BURGER £3.5

All you can eat!  
33% off  
£2.34

EDIT BURGER

REMOVE BURGER



**P - 15 User output result**



The image shows a dark gray rectangular area with a subtle texture. In the center, the text "Select a day of the week:" is displayed in a white, sans-serif font. To the right of this text is a white dropdown menu with a blue border, containing the word "Thursday" and a small upward and downward arrow icon. Further to the right is a light gray button with rounded corners and the text "FIND DEALS!" in a bold, black, sans-serif font.



Today's deals:

**BOB'S BURGERS**

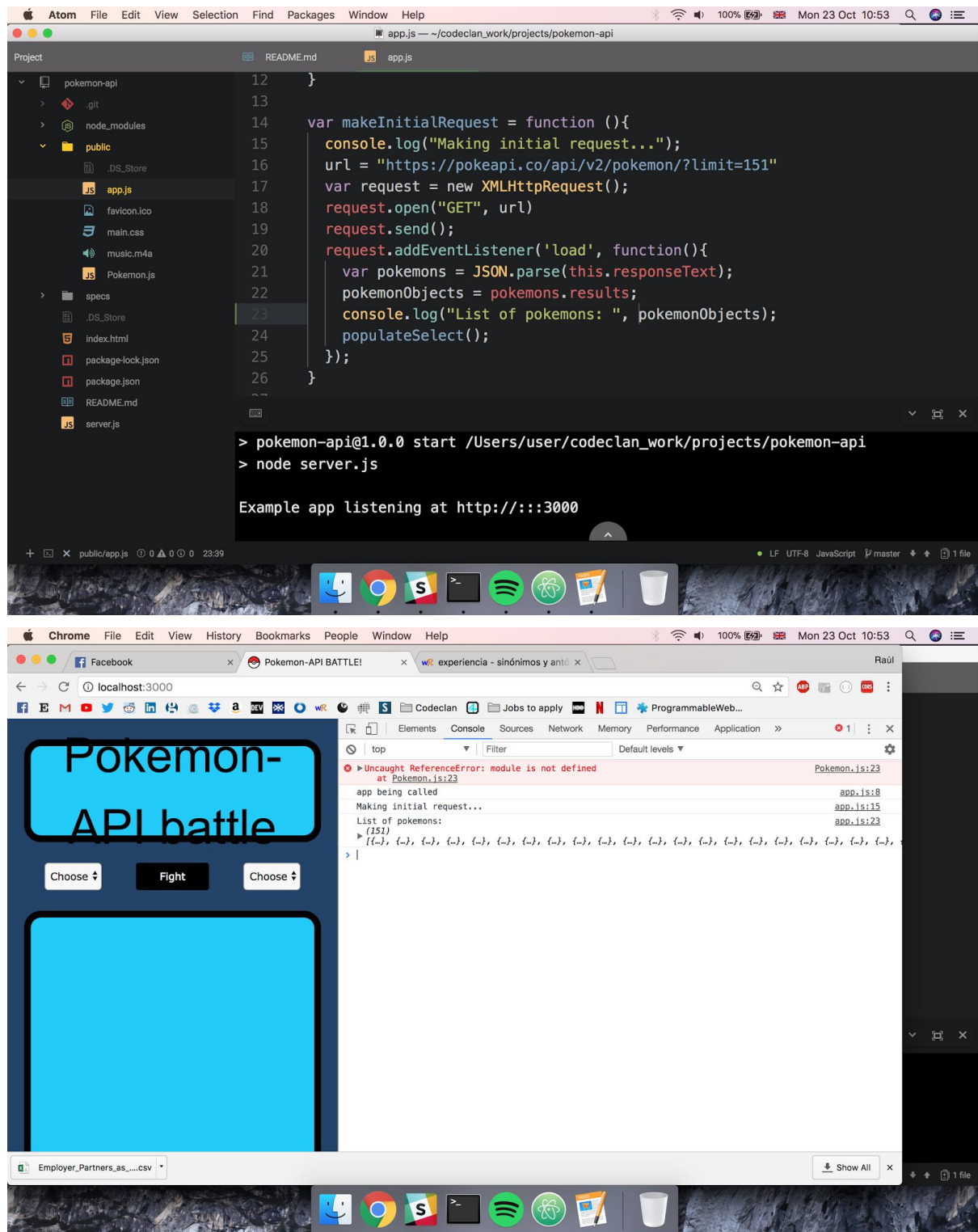
Bring your cat - get 1/3 off your burger



Krabbing for a burger deal

Deals available on Thursday

**P - 16 Show an API being used within your program.**



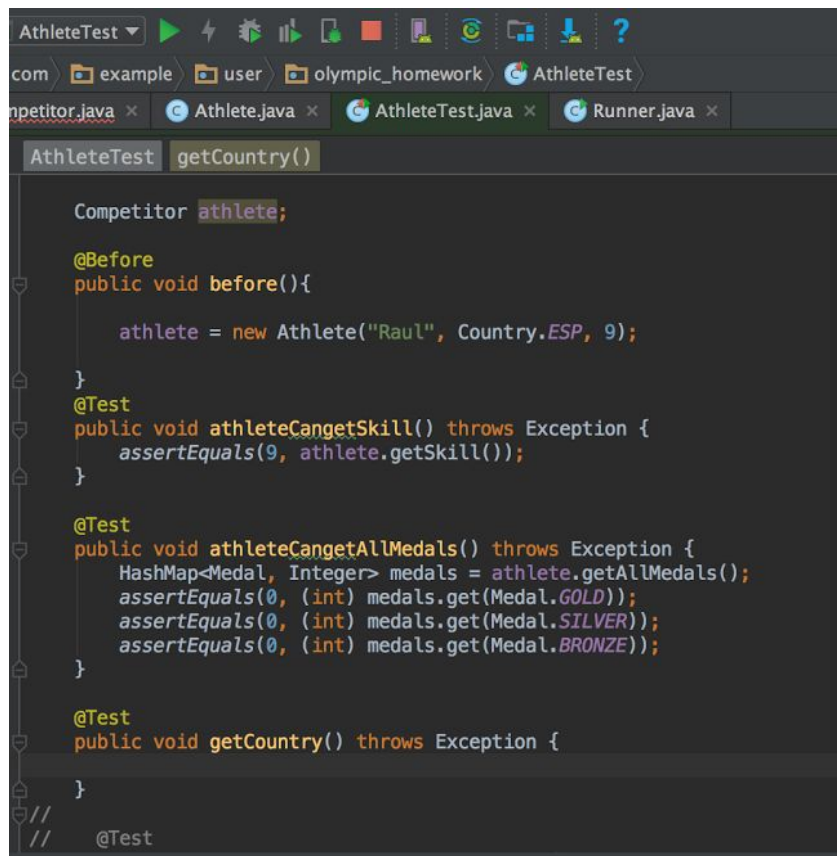
**P -17 Produce a bug tracking report**

|   |        |   |        |
|---|--------|---|--------|
| User must be able to look up by postcode                              |        |   | Passed |
| Users must be able to add favourites to their account                 | Failed | Implement database entry for each user                          | Passed |
| Wrong postcodes should not be processed                               | Failed | Add verification system to ensure correction                    | Passed |
| Marker should disappear when user removes destination from favourites | Failed | Update code to synchronize map markers with user database entry | Passed |
| User should be able to see deleted destinations                       | Failed | Added a list to keep track of deleted destinations              | Passed |



## P - 18 Testing in a program.

First test:



```
AthleteTest
com> example> user> olympic_homework> AthleteTest
competitor.java x Athlete.java x AthleteTest.java x Runner.java x
AthleteTest getCountry()

Competitor athlete;

@Before
public void before(){
    athlete = new Athlete("Raul", Country.ESP, 9);
}

@Test
public void athleteCanGetSkill() throws Exception {
    assertEquals(9, athlete.getSkill());
}

@Test
public void athleteCanGetAllMedals() throws Exception {
    HashMap<Medal, Integer> medals = athlete.getAllMedals();
    assertEquals(0, (int) medals.get(Medal.GOLD));
    assertEquals(0, (int) medals.get(Medal.SILVER));
    assertEquals(0, (int) medals.get(Medal.BRONZE));
}

@Test
public void getCountry() throws Exception {
}

//
// @Test
```

```
AthleteTest.getCountry
Java com.example.user.olympic_homework AthleteTest
Competitor.java Athlete.java AthleteTest.java Runner.java
AthleteTest
17 athlete = new Athlete("Raul", Country.ESP, 9);
18
19 }
20 @Test
21 public void athleteCanGetSkill() throws Exception {
22     assertEquals(9, athlete.getSkill());
23 }
24
25 @Test
26 public void athleteCanGetAllMedals() throws Exception {
27     HashMap<Medal, Integer> medals = athlete.getAllMedals();
28     assertEquals(0, (int) medals.get(Medal.GOLD));
29     assertEquals(0, (int) medals.get(Medal.SILVER));
30     assertEquals(0, (int) medals.get(Medal.BRONZE));
31 }
32
33 @Test
34 public void getCountry() throws Exception {
35     assertEquals(Country.ESP, athlete.getCountry());
36 }
37 //
38 // @Test
39 // public void addMedal() throws Exception {
40 //
41 // }
42
43 }

1 test failed - 22ms
java.lang.AssertionError:
Expected :ESP
Actual   :GBR
<Click to see difference>
<1 internal calls>
at org.junit.Assert.failNotEquals(Assert.java:834) <2 internal calls>
at com.example.user.olympic_homework.AthleteTest.getCountry(AthleteTest.java:35) <28 internal calls>

Process finished with exit code 255
d Monitor 4: Run TODO 45:1 LF UTF-8
```

```
AthleteTest.getCountry
com > example > user > olympic_homework > Competitor
Competitor.java x Athlete.java x AthleteTest.java x Runner.java x

Competitor    getCountry()
    }

    public HashMap <Medal, Integer> getAllMedals() {
        return this.medals;
    }

    Country getCountry(){
        return this.country;
    }

    private void setUpMedalCounter() {
        this.medals = new HashMap<>();
        this.medals.put(Medal.GOLD, 0);
        this.medals.put(Medal.SILVER, 0);
        this.medals.put(Medal.BRONZE, 0);
    }

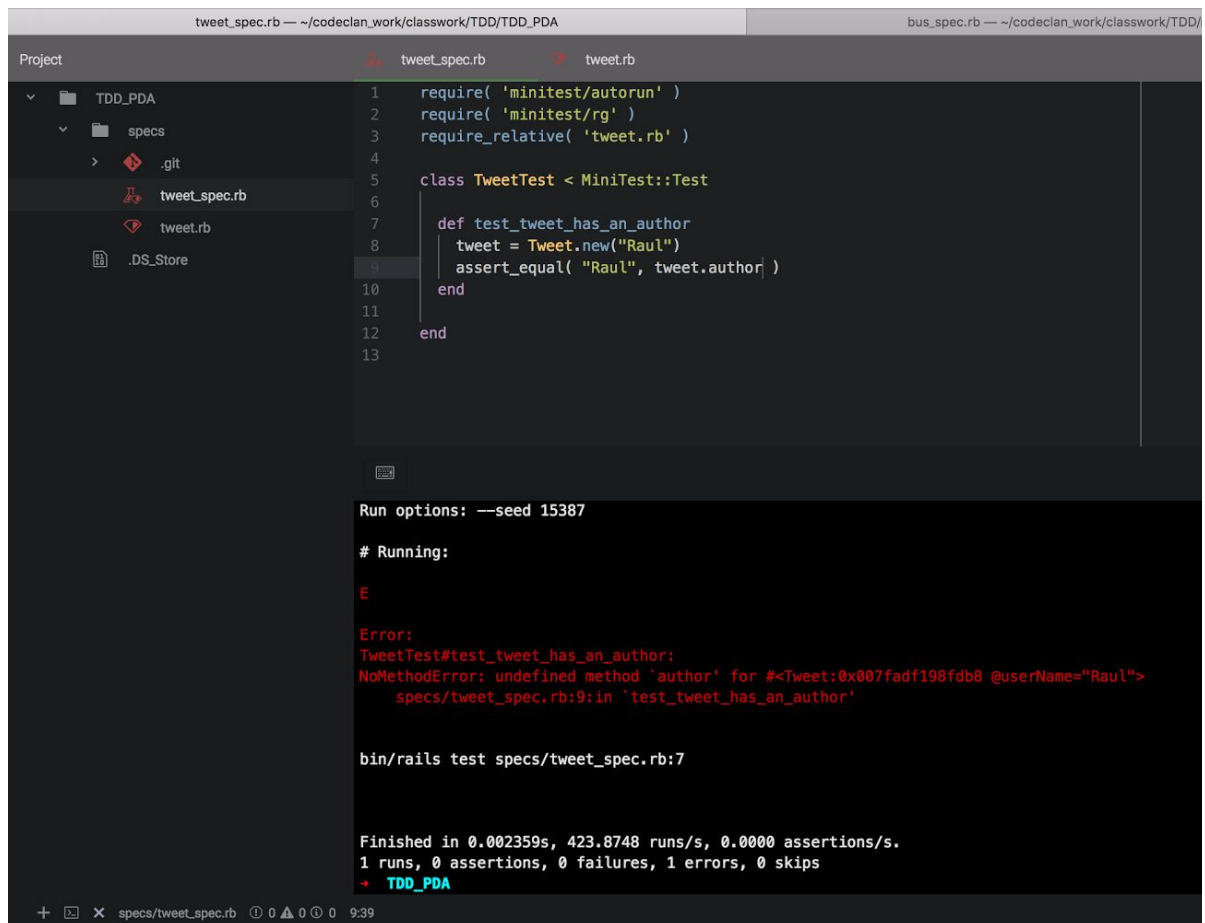
    void addMedal(Medal medal){
        int currentMedalCount = this.medals.get(medal);
        this.medals.put(medal, currentMedalCount + 1);
    }

    abstract int getSkill();
}
```

```
Olympic_homework > app > src > test > java > com > example > user > olympic_homework > AthleteTest
Android
Project
  app
    manifests
    java
      com.example.user.olympic_homework
        Athlete
        Competitor
        Country
        Medal
        Runner
  captures
  res
  Gradle Scripts
Run
  AthleteTest.getCountry
  1 test passed - 1ms
  "/Applications/Android Studio.app/Contents/jre/jdk/Contents/Home/bin/java" ...
  Process finished with exit code 0

AthleteTest
17 athlete = new Athlete("Raul", Country.ESP, 9);
18 }
19
20 @Test
21 public void athleteCanGetSkill() throws Exception {
22     assertEquals(9, athlete.getSkill());
23 }
24
25 @Test
26 public void athleteCanGetAllMedals() throws Exception {
27     HashMap<Medal, Integer> medals = athlete.getAllMedals();
28     assertEquals(0, (int) medals.get(Medal.GOLD));
29     assertEquals(0, (int) medals.get(Medal.SILVER));
30     assertEquals(0, (int) medals.get(Medal.BRONZE));
31 }
32
33 @Test
34 public void getCountry() throws Exception {
35     assertEquals(Country.ESP, athlete.getCountry());
36 }
37
38 //
39 // @Test
40 // public void addMedal() throws Exception {
41 // }
42
43 }
```

## Second test:



The screenshot shows a code editor with a project named 'TDD\_PDA'. The file explorer on the left shows the project structure: 'TDD\_PDA' (folder), 'specs' (folder), '.git' (folder), 'tweet\_spec.rb' (file), 'tweet.rb' (file), and '.DS\_Store' (file). The main editor window displays the content of 'tweet\_spec.rb'.

```
1 require( 'minitest/autorun' )
2 require( 'minitest/rg' )
3 require_relative( 'tweet.rb' )
4
5 class TweetTest < MiniTest::Test
6
7   def test_tweet_has_an_author
8     tweet = Tweet.new("Raul")
9     assert_equal( "Raul", tweet.author )
10  end
11
12 end
13
```

Below the code editor, the output of running the test is displayed:

```
Run options: --seed 15387

# Running:

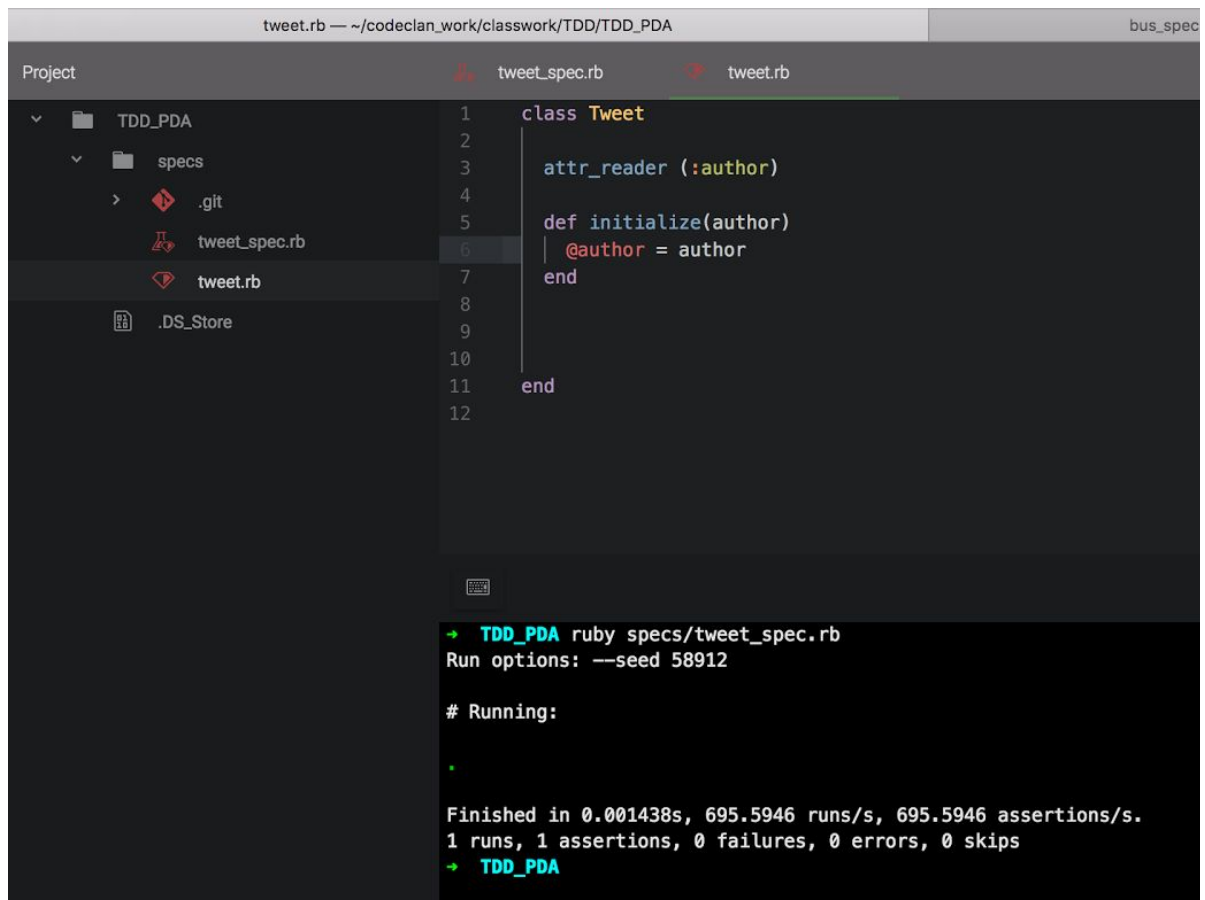
E

Error:
TweetTest#test_tweet_has_an_author:
NoMethodError: undefined method 'author' for #<Tweet:0x007fadf198fdb8 @userName="Raul">
    specs/tweet_spec.rb:9:in `test_tweet_has_an_author'

bin/rails test specs/tweet_spec.rb:7

Finished in 0.002359s, 423.8748 runs/s, 0.0000 assertions/s.
1 runs, 0 assertions, 0 failures, 1 errors, 0 skips
→ TDD_PDA
```

The status bar at the bottom shows the file path 'specs/tweet\_spec.rb', a search icon, a zoom icon, and the time '9:39'.



### Third test:

```
tweet_spec.rb — ~/codeclan_work/classwork/TDD/TDD_PDA
Project — ~/codeclan_work/c

Project
  TDD_PDA
    specs
      .git
      tweet_spec.rb
      tweet.rb
      .DS_Store

tweet_spec.rb
1 require 'minitest/autorun'
2
3 require_relative( 'tweet.rb' )
4
5 class TweetTest < MiniTest::Test
6
7   def test_tweet_has_an_author
8     tweet = Tweet.new("Raul", "testText")
9     assert_equal( "Raul", tweet.author )
10  end
11
12   def test_tweet_text_maximum_length_140_chars
13     tweet_text = "This is a tweet test to check if the function to add text"
14     tweet = Tweet.new("Raul", tweet_text)
15     assert_equal( 140, tweet.text.length)
16  end
17
18 end
19
20
Run options: --seed 30455
# Running:
.F
Failure:
TweetTest#test_tweet_text_maximum_length_140_chars [specs/tweet_spec.rb:15]:
Expected: 140
Actual: 178
bin/rails test specs/tweet_spec.rb:12
Finished in 0.001348s, 1483.2020 runs/s, 1483.2020 assertions/s.
2 runs, 2 assertions, 1 failures, 0 errors, 0 skips
→ TDD_PDA
```

tweet.rb — ~/codeclan\_work/classwork/TDD/TDD\_PDAProject

Project

TDD\_PDA

specs

.git

tweet\_spec.rb

tweet.rb

.DS\_Store

tweet\_spec.rb

tweet.rb

1class Tweet

2

3attr\_reader(:author, :text)

4

5def initialize(author, text)

6 @author = author

7 @text = setText(text)

8end

9

10def setText(text)

11 if (text.length <= 140)

12 @text = text

13 else

14 @text = text[0..139]

15 end

16end

17

→ TDD\_PDA ruby specs/tweet\_spec.rb

Run options: --seed 63376

# Running:

..

Finished in 0.001236s, 1618.2709 runs/s, 1618.2709 assertions/s.

2 runs, 2 assertions, 0 failures, 0 errors, 0 skips

→ TDD\_PDA