# SDET13 Codebar UI Testing

**Team: Nicolas Rothera, Katie Frost, Daniel Raymond**

# Table of Contents

# 1.0 Introduction

This test plan includes the objectives, scope, schedule, risks and approach to testing which will be taken. This test plan has been created to communicate these factors to all the team members and will clearly identify test deliverables.

The brief was to create an automation suite for the codebar website, to analyze the current test coverage over the site and decide where best to continue automation testing.  Doing exploratory testing on the website and analyzing the current tests in the code, it was decided that the website could benefit from UI tests as there are none of these currently and this is where the most value could be added.

## 1.1 Objectives

- Create user interface tests for the codebar website.
- Full automate tests
- Maximise test coverage
- Make tests readable and reusable
- Log bugs and write steps to recreate them
- Multi browser testing

# 2.0 Scope

Sprint 1

- Two days to write out user stories in the given/when/then syntax.
- Prioritise the use cases in terms of importance, 13 being most important and 1 being least.
- 3 days to write tests
- Have all 5+ priority rated use cases tested by end of sprint 1.

Sprint 2

- Enable tests to work in multiple browsers
- Write tests for the rest of the use cases with lower priority

- Make sure tests all work collectively and independently

# 3.0 Assumptions/Risks

## 3.1 Assumptions

- We can use already written methods to generate random data for our cucumber tests.
- We can make assumptions about how the website should be used and how users will interact with the website.

## 3.2. Risks

This section covers any risks that have been identified prior to the project and the appropriate action to reduce their impact on the project.

| # | Risk | Impact | Trigger | Mitigation Plan |
|---|---|---|---|---|
| 1 | Getting carried away with number of tests written and wanting to increase number of tests beyond the scope, with risk of missing deadline or tests not working. | High | Tests within the scope being written. | Priorities will be set before beginning the project. Once tests are written focus on integrating them with the rest of the tests and only when they all pass considering writing tests not initially set up. |
| 2 | Interacting with the source code | High | Changing or adding code to files already written | Write code in files independent from source files |

# 4.0 Test Approach

This project is using an agile approach with weekly iterations. Tasks will be equally split and designated to each team member to complete as part of the first sprint which will finish at the end of the week. At the end of each week the requirements identified will be delivered and the team will hold a retrospective to discuss what they can improve on/keep doing for the next sprint.

# 5.0 Test Environment

The tests are set up on Mac OS and can be run on the chrome and firefox browsers.

# 6.0 Bugs and Defects

## Sign up Github Bug

- Create Github account
- Sign up to codebar
- Sign out of codebar
- Delete Github account
- Re-create Github account
- Sign up to codebar, error occurs

## Sign up to course as a coach

- Sign in as coach on codebar
- Go to events and click on a course
- Click RSVP
- Click get your ticket, no functionality

## Inconsistent student-coach order on subscriptions page

- Go to subscriptions page

- Order of Students subscribe and Coach Subscribe is inconsistent

Cancelling an Event alert

- Events page
- Attend an event as a coach or student
- Click cancel my spot
- Alert will pop up

# 7.0 Tools

- MacBook Air with Google Chrome and Firefox
- Google docs for test plan, user stories
- Trello for sprint organisation
- Ruby for test writing
- Cucumber for test framework

# 8.0 Entry and Exit Criteria

**Entry**

- User stories written and prioritised
- Create use case diagrams
- Sprint planning

**Exit**

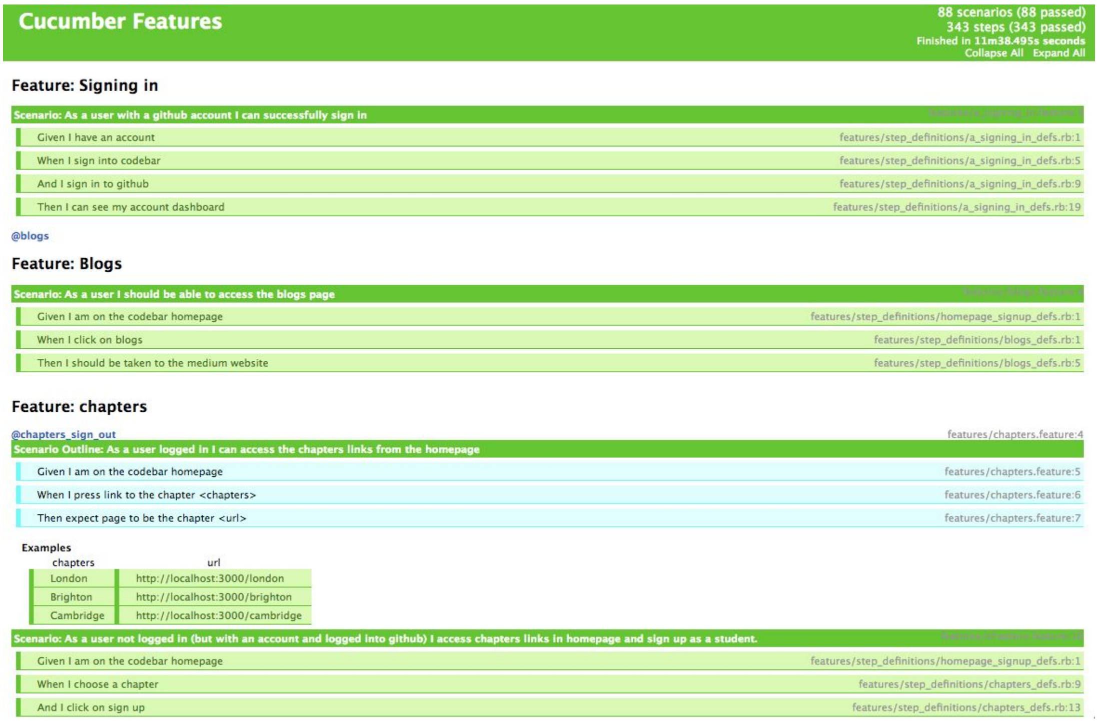
- All user stories completed
- 100% test pass rate
- Multi browser testing

# 9.0 Deliverables

## 6.1. Test Schedule

| **Task Name** | **Start** | **Finish** | **Effort/Amount of time to spend** |
|---|---|---|---|
| Writing use cases | 15/01/2018 | 16/01/18 | 2 days |
| Prioritise use cases | 16/01/18 | 16/01/18 | 1 hour |
| Splitting up tasks | 16/01/18 | 16/01/18 | 30 minutes |
| Writing Tests for sprint 1 | 17/01/18 | 19/01/18 | 3 days |
| Enabling sprint 1 tests to work together | 19/01/18 | 19/01/18 | 1 day |
| Writing tests for sprint 2 | 22/01/18 | 23/01/18 | 2 days |
| Getting tests to work in multiple browsers | 19/01/18 | 23/01/18 | 4 days |
| Making sure sprint 1 and sprint 2 tests all run together and pass | 23/01/18 | 24/01/18 | 1 day |
| Full documentation | 15/01/18 | 25/01/18 | On going throughout project. |

# 10.0 Test Report

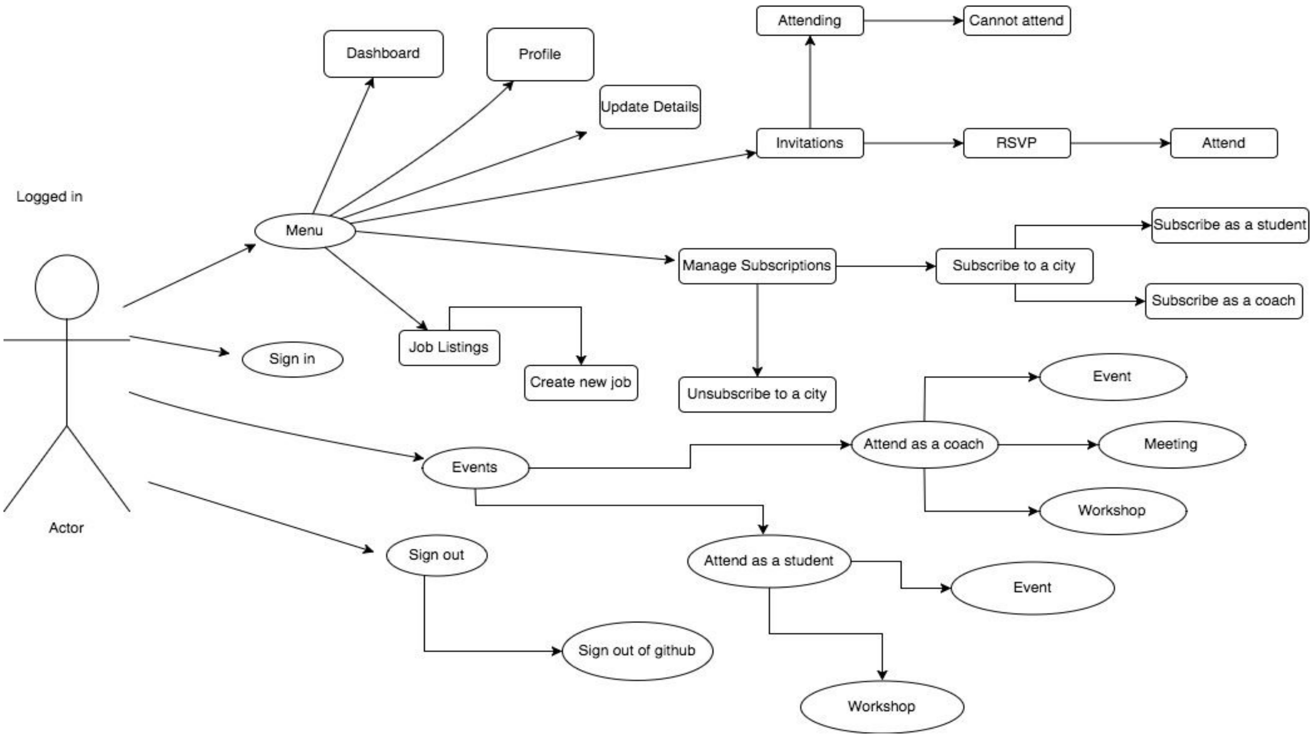


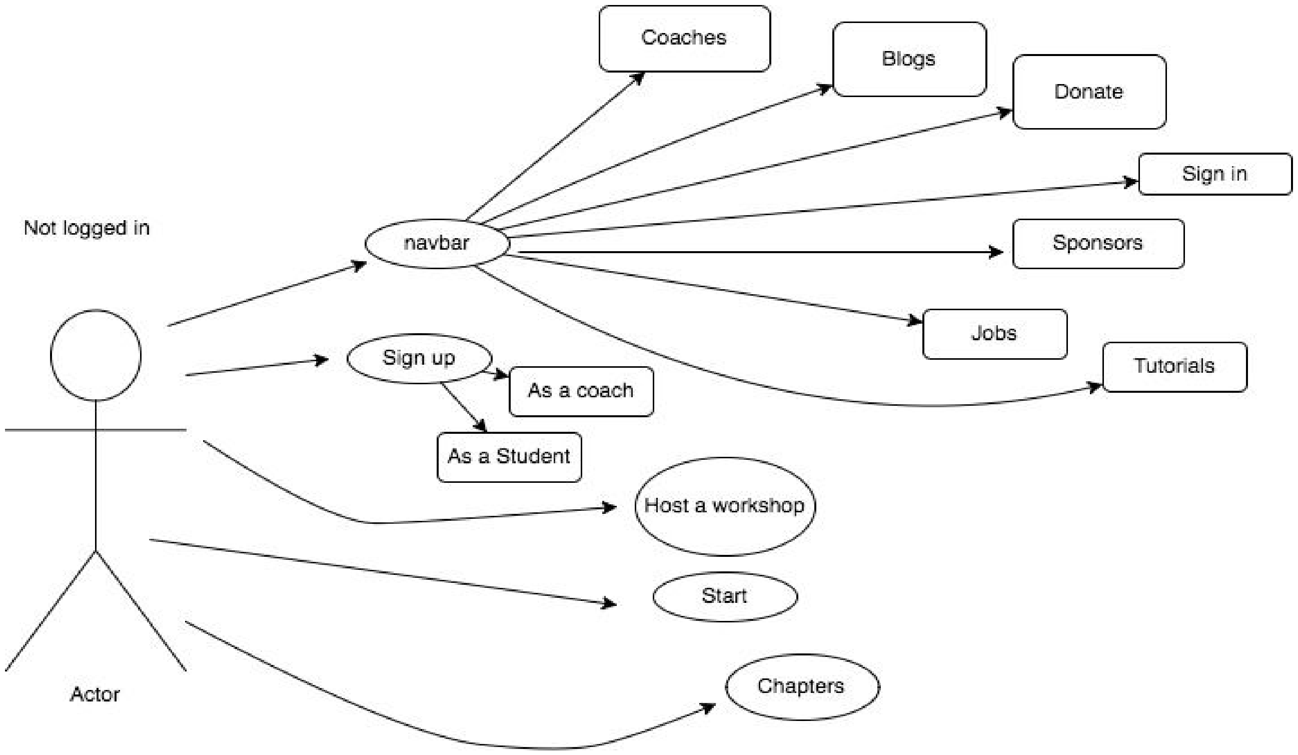
```
Cucumber Features                                                        88 scenarios (88 passed)
                                                                         343 steps (343 passed)
                                                                Finished in 11m38.495s seconds
                                                                        Collapse All  Expand All

Feature: Signing in

Scenario: As a user with a github account I can successfully sign in        features/a_signing_in.feature
  Given I have an account                         features/step_definitions/a_signing_in_defs.rb:1
  When I sign into codebar                        features/step_definitions/a_signing_in_defs.rb:5
  And I sign in to github                         features/step_definitions/a_signing_in_defs.rb:9
  Then I can see my account dashboard             features/step_definitions/a_signing_in_defs.rb:19

@blogs
Feature: Blogs

Scenario: As a user I should be able to access the blogs page                features/blogs.feature
  Given I am on the codebar homepage              features/step_definitions/homepage_signup_defs.rb:1
  When I click on blogs                           features/step_definitions/blogs_defs.rb:1
  Then I should be taken to the medium website    features/step_definitions/blogs_defs.rb:5

Feature: chapters

@chapters_sign_out                                                           features/chapters.feature:4
Scenario Outline: As a user logged in I can access the chapters links from the homepage
  Given I am on the codebar homepage              features/chapters.feature:5
  When I press link to the chapter <chapters>     features/chapters.feature:6
  Then expect page to be the chapter <url>        features/chapters.feature:7

  Examples
```

| chapters | url |
|---|---|
| London | http://localhost:3000/london |
| Brighton | http://localhost:3000/brighton |
| Cambridge | http://localhost:3000/cambridge |

```
Scenario: As a user not logged in (but with an account and logged into github) I access chapters links in homepage and sign up as a student.    features/chapters.feature
  Given I am on the codebar homepage              features/step_definitions/homepage_signup_defs.rb:1
  When I choose a chapter                         features/step_definitions/chapters_defs.rb:9
  And I click on sign up                          features/step_definitions/chapters_defs.rb:13
```

# 11.0 Use Case Diagrams




Not logged in
Actor
navbar
Coaches
Blogs
Donate
Sign in
Sponsors
Jobs
Tutorials
Sign up
As a coach
As a Student
Host a workshop
Start
Chapters