



Sparta Global - Codebar Website UI Test Plan

BY NATHAN ROBERTS, STEVEN OLUFEMI & JONATHAN DONALDSON

Description/Intro

The reason we chose these things was because if Codebar was to grow, they had to be able to inform people of what they offer and how they could make use of their services. These things were integral to the success of Codebar. The tests conducted focussed on the user interface and recording the journeys of a sample user. The journeys and the procedure of tests and functions used were also created before any tests were executed.

UI tests were created as this was identified as something that codebar currently lacked.

Analysis of what was already there

- UI tests were already present, however there was not much detail in terms of user journeys
- The descriptions of tests were not easily readable to non-technical people
- UI tests only checked that links went to the right page
- No Page Object Model was present that enabled people to reuse functions

Objectives

- Performing UI testing on the codebar site
- Creating tests based on common user behaviour and valuable features of the site
- This included things such as being able to know about upcoming workshops and being able to register an RSVP or being able to create a workshop as a coach

Scope

- We plan to test the website by conducting automated user interface tests.(UI)
- We divided the work load in to sprints of specific work. Taking into account priority, difficulty and cost/value to the website.
- The sprints divided into user stories which were recorded in the log file. A short description or comment of any work that needed adding or recording in case of affecting future tests.
- Work was divided equally among the team by difficulty, priority and workload. This resulted in efficient test writing and Page-Object-Model creation.
- To review progress we conducted team meetings with and without the project lead. This allowed any issues to be quickly raised and resolved.

Methods

- Firstly make a list of all the pages that the website has.
- Next create use cases
- Then create a page object model based on the use cases
- Finally create Test automation methods for each page in a way that they could easily be reused and were name them clearly to ensure duplicate methods would not be created in future

Planning

Use of shared google docs for planning and logging

TASK	PERSON	DATE STARTED	DATE COMPLETED	COMMENT				
Write Use cases for signing in and signing up as a student & coach, and the side bar features	Steven	15/01/2018	16/01/2018	All possible routes being covered				
Write Use cases for posting a job as a coach	Jon	15/01/18	15/01/18	All possible routes being covered				
Write Use cases for using the footer links	Nathan	16/01/18	16/01/18	All possible routes being covered				
Write Use cases for accessing tutorials	Nathan	16/01/18	16/01/18	All possible routes being covered				
Set up initial cucumber framework (creating empty page object model)	Group	16/01/18	16/01/18	cucumber frameworkfully setup. Pages folder contains nessarcay info only				
Creating first cucumber test to ensure it works	Nathan	16/01/18	16/01/18	First tests created and destroyed and used for making sure POM functions work.				

Planning poker for workload allocation

Google docs for POM planner

Google docs for writing test cases

Project Risks

Risk	Description	Prevented by
Coordinating on git	Working in a group may cause merge conflicts	Keeping code organised and having good planning to

		minimise overlapping work
Interfering with existing code	New code may cause existing code to malfunction	Creating a UI test folder to separate all of our code
Creating unnecessary tests	Wasting time and resources on tests that add no value or are repeated	Planning and examining existing tests

Tools

Instructions for setup taken from readme

The tools that we need to perform these test shall included Cucumber, Selenium, Capybara.

Bugs Found (Frequency:F Impact:I out of 10)

Blog link in header takes you to a different url compared to the footer. F:5 I:1

Access the codebar homepage. Click on the blog link in the nav bar

(<https://medium.com/@codebar>) and the blog link in the footer

(<https://medium.com/the-codelog>). The pages are slightly different.

Signing out doesn't allow others to sign in on same computer. F: 3, I: 7

Sign in as one user, sign out then sign in again to reproduce