**Microtask #2: Writing node.js tests**

**1. Steps in writing Node.js tests for Newcomer Homepage**

**1.1. Studying required skills**

As I was new to Selenium as well as Node.js (JavaScript), I had to start from scratch. It was quite difficult to have a clear direction of what to learn and how to learn at the very beginning. Based on skill description on project, at first, I decided to get familiar with JavaScript, Node.JS and then spend few days for learning Selenium WebDriver. After finishing learning, I reviewed Node.js test code of MediaWiki and realized it was not used Selenium WebDriver. Spending some time studying and searching the Internet, I found out that WebDriverIO was used instead of Selenium WebDriver. So I spent more time for learning WebDriverIO. With basic knowledge gained, I studied Node.js test code of MediaWiki to understand it and get familiar with the coding style.

**1.2. Study Newcomer Homepage functionality**

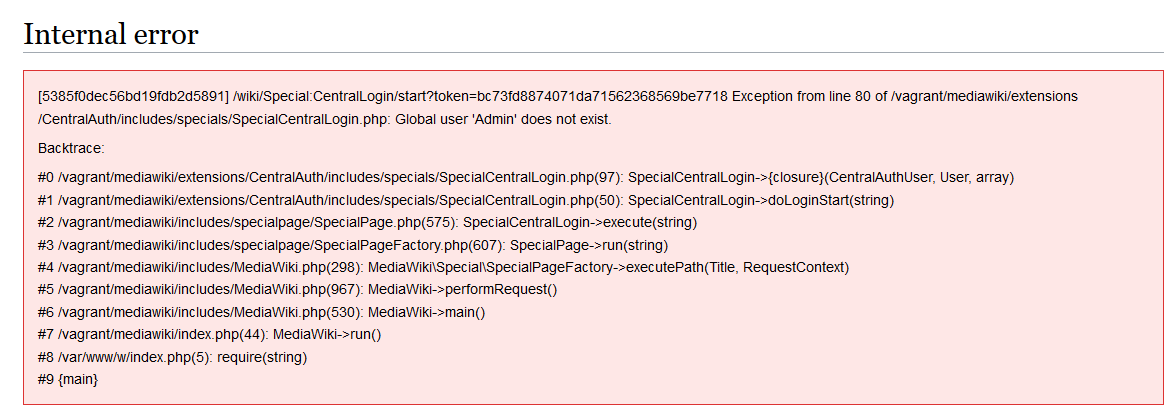
To create automation test, at first I need to understand functionality and work flow of Newcomer Homepage page. Based on the documents from this link <https://www.mediawiki.org/wiki/Growth/Personalized_first_day/Newcomer_homepage> and using the real website, I categorized the page into 5 parts:

* General
* Start module
* Help module
* Impact module
* Mentorship module

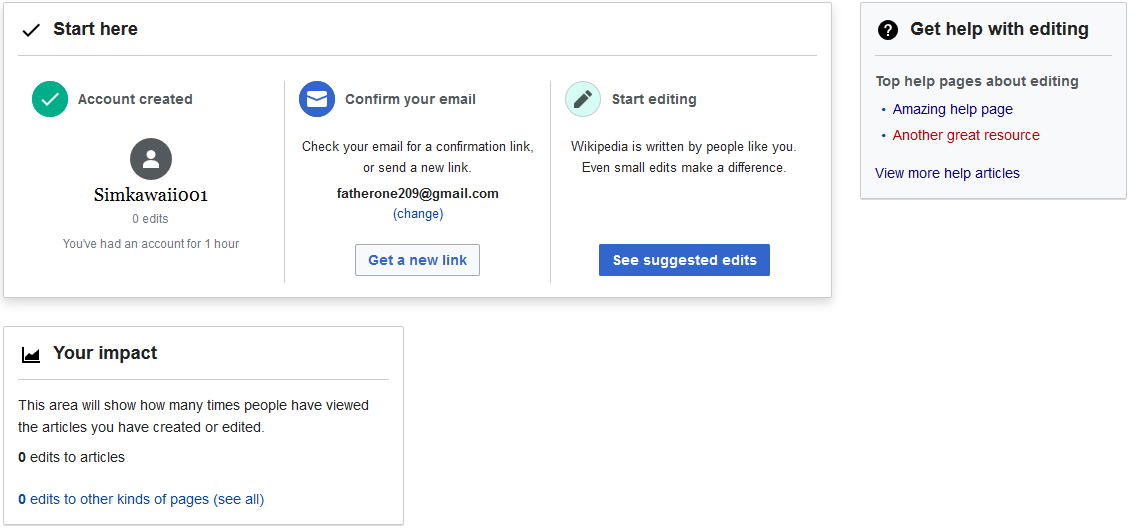
Test cases were written in excel file first so that I could easily to follow their steps to create automation tests. Besides, it made the tests more manageable and we could not miss any test case. See details in HomePage Test Cases.xlsx file.

Some issues that I encountered during this phase:

* I’ve got an internal error when tried to log in Admin account

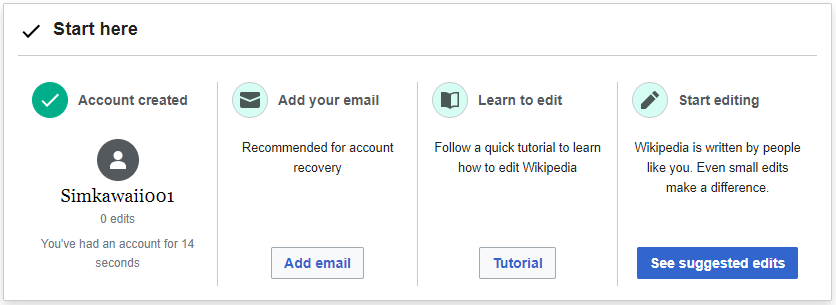


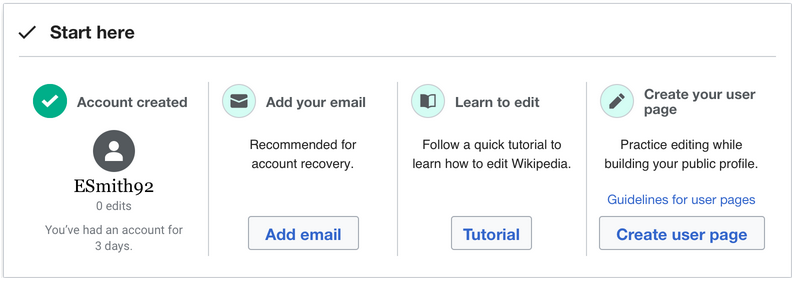
* Sometimes, some modules were disappeared on the web page such as “Learn to edit” in Start module, Mentor module, etc...



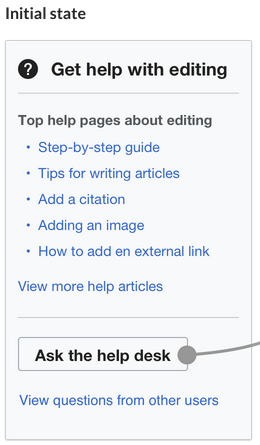
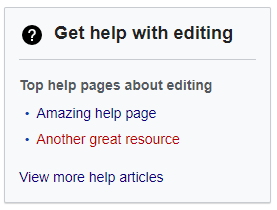
* The mockups and documents did not match with the real website, such as Start module, Help module. That was hard to determine if the documents were not updated or the website was built incorrectly, so it was difficult to decide where should be based on to create test cases. Finally, I decided to use this link <https://en.wikipedia.beta.wmflabs.org/wiki/> as the reference.

These are Start modules on real website and from the mockup





And these are Help modules on the real website and from the mockup



**1.3. Creating and running automation tests**

From test cases which were created in previous phase, I built automation tests for 2 parts: General and Start Module.

Some issues that I encountered during creating automation tests:

* I tried to save testing time by using the result of the previous test case as input for the next test case. Problem occurred when previous test case was failed so there was no correct input for the next test case, therefore, next test case could not run. So, a lesson learned: Test cases should be independent of each other
* In some test cases, after executing some steps to reach the destination screen, I wanted to check different parts of this screen. For example, after logging in an account the open Homepage page, I wanted to check the homepage title, username, email address, the button and so on… At first, I used one assert to check for each part and then put all these asserts in the same it(). But in this case, if one assert was failed, the remaining ones would not be executed. So my solution was I separated each assert into a separate it() with same steps. But with this way, it took more time to run tests because same steps were repeated again and again in different test cases.

Run automation tests of General and Start module for http://dev.wiki.local.wmftest.net:8080

**2. The result**

90 test cases in total for 5 parts: General, Start module, Help module, Impact module, Mentorship module are created. Among them, 35 test cases of 2 parts: General and Start module are automated and run and one issue was found from it: [Start-033] Start module title is "Start here" instead of "First steps complete."

