Testing

Several testing methods have been and continue to be used to enforce and measure the functional qualities of the system.

**System Integration Testing:**

This black-box testing technique tests the system as an integrated whole, evaluating the system’s compliance against specified requirements (outlined as user stories in the specifications). It tests the required interactions and verifies that all related systems can operate in coordination with other systems in the same environment. This was especially important to test once the php front-end of the application and its sql back-end were brought together.

**User acceptance testing:**

In Scrum, UAT refers to the functional testing of a user story by the development team during a sprint. Actual software users test the software to make sure it can handle required tasks in real-world scenarios, according to user story specifications.

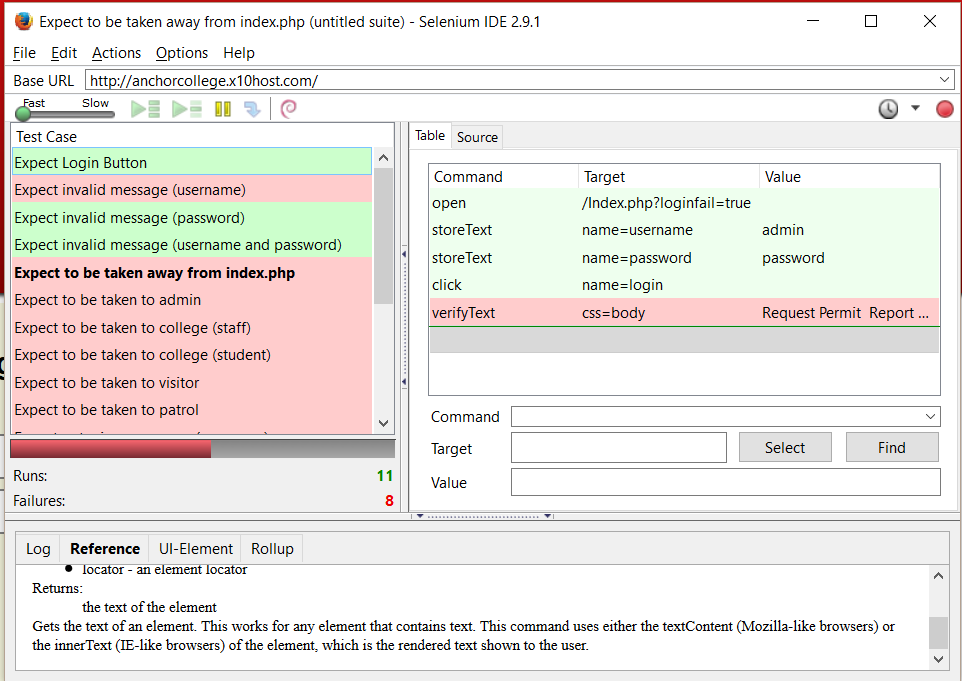
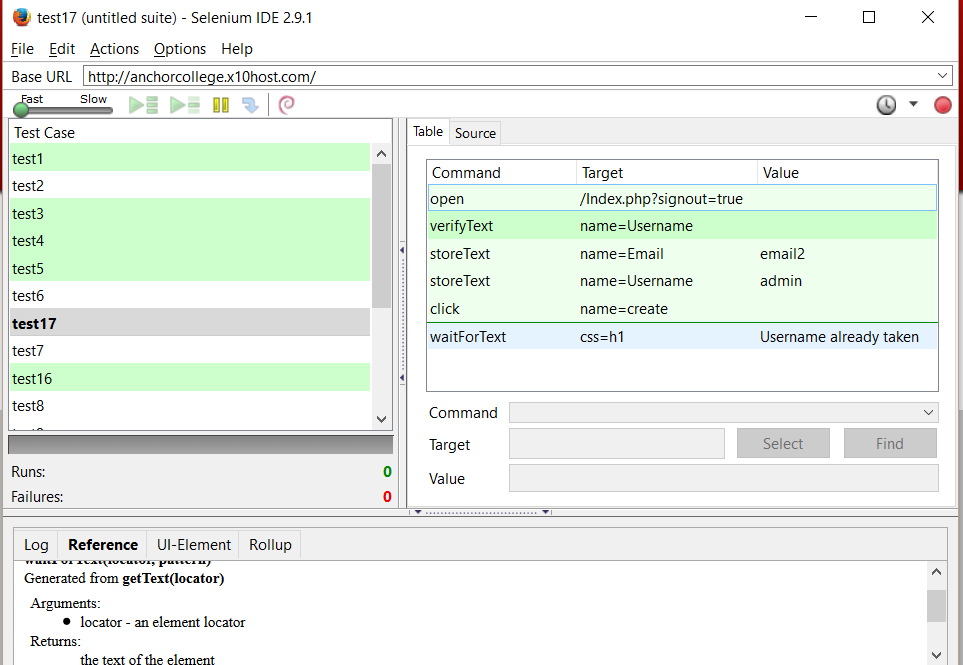
We integrated SIT and UAT by taking on a role of a user to test user stories in the system as far as it had been developed to date, in order to ensure that the system could meet the requirements set by the specification document and maintain its data integrity. It through this testing that we confirmed our completion of each user story outlined in sprint plans 1 and 2.

**Continuous Integration:**

CI is a development practice that requires integrating one’s new or changed code into the master code repository at least once a day. We used the X10hosting service to support this process as all developers had remote access in order to upload code to the shared domain.

**GUI Testing**

Selenium was used for GUI Testing.

*Figure 1: A suite of test cases. Figure 2: Refined test cases*

As the project is ongoing, it is not expected at this stage that all test cases will pass for release 1. Test cases for user stories check things such as restricted permissions and access to links for form submission.