**CS1699 – DELIVERABLE 3: Web Testing with BDD**

**Website:** Wikipedia (https://en.wikipedia.org/)

**Authors:** Sheridan Zivanovich & Ryan McDonald

Initially, we struggled to decide on a website that would provide sufficient functionality to test, while simultaneously not requiring unnecessary effort to properly access important features. We discussed testing the cs.pitt.edu website; however, the functionality of this website is quite limited, and as such there would be very little for us to test. We then moved to the opposite side of the spectrum, considering Twitter as our next candidate. However, we quickly realized that many of Twitter’s features required significant interaction between users with accounts. As such, it would have been necessary to make several fake Twitter accounts, with corresponding fake e-mails for these accounts, and when considering the difficulty of both making and managing these accounts, we concluded Twitter was not an appealing option. Eventually, we made the decision to test Wikipedia. Wikipedia has significant functionality for non-users, and unlike Twitter, it is very easy to create a dummy Wikipedia account to use for testing.

Despite the attractiveness of Wikipedia as a test subject, we came across several difficulties in developing scenarios that would be testable without access to the codebase or the underlying infrastructure. For example, we were unable to test features such as article creation, because the Wikipedia community in general frowns upon the creation of frivolous articles, and our test account and/or IP would be quickly blocked from creating further articles. Additionally, as the format of Wikipedia is fairly regular across the entire site, it can be difficult to determine whether or not you are simply testing the same functionality multiple times. For example, testing the ability to edit a user page is very similar to testing the ability to edit a regular article. However, since these two scenarios apply to two different user stories, we decided to keep overlapping tests in cases such as this, despite the possible redundancy.

Going forward, we expect that it would eventually become necessary to test certain scenarios on a non-production server, to allow for the testing of article creation, modification, deletion, and other similar features. Without the ability to conduct this type of testing, we certainly felt limited in our scenario and feature choice.