ABSTRACT

Developers and testers encounter with many challenges in software development lifecycles. During the testing processes, several bugs and defects of a product are detected by testers. Each bug is submitted to a bug tracking system with its relevant information such as summary, severity, and priority. This project aims to simplify and shorten bug triaging process by finding the duplicate bugs and grouping them together.

OBJECTIVES

• To implement a machine learning system that successfully identifies bugs as duplicate and attaches it to a previous bug, therefore saving development time
• To deploy this system in Siemens’ development environment

PROJECT DETAILS (AUTOMATED DUPLICATE BUG DETECTION)

We used Mozilla’s Bugzilla repository as our data source. All our approaches were done with Python, and we also used libraries such as Scikit-learn and NLTK for text processing and machine learning. We downloaded all bugs which were created in the last 10 years which in total accounted for 952,361 bugs. We decided to focus on top 5 products (Core, Firefox, Firefox OS, Toolkit, Thunderbird) because they contained almost half of the bugs. Our first step was to cluster the duplicates we had in those bugs product by product. For this we used bug_id and dupe_of fields, since if a bug had dupe_of field not empty, it contained the id of one of it’s duplicates. We used this approach for all bugs we had and the ones created in 2017.

After the clustering step, we did use bag of words approach and found cosine similarities of bugs in top 5 products. In this step we didn’t use any preprocessing methods. We also selected same amount of random bugs and also calculated their cosine similarities to see if we can use cosine similarities to identify duplicates.

REFERENCES