

SOFTWARE FOR THE ALGORITHM OF EFFICIENT METRIC ELECTION

Lyudmyla Honchar¹⁾, Iryna Danylyuk²⁾, Ivan Korniychuk³⁾

Ternopil National Economic University

^{1)PhD., associate professor; ^{2)Master's Degree student}}

I. Statement for the task

Flexible testing methods for software, which are now the main market share, do not always allow unambiguously and fully detect all defects and establish the correctness of the software product, therefore, all existing testing methods operate within the formal software verification process

II. The purpose of the work

Purpose: software implementation and research of optimized algorithm of efficient metric election in the process of software testing.

III. Use JIRA error tracking system to support metrics

The project on which the developed algorithm was developed, the process of implementing JIRA and metrics, consisted of the following steps:

1. All information related to the project (documents, files, circuits, notes, online documents, etc.) was transferred to JIRA.
2. Creation of epic, where the requirements for the project, test cases and test collections will be kept. All of them create an interconnected flow, which allows you to track the metric's claims coverage.

In JIRA, for each test case, different statuses are assigned, which depend on the state of the test case:

- in Creating when the test case is added to the collection;
- in Testing, when the test case is filled with data;
- and after passing the test, the test case acquires one of two statuses: Pass (the test case has passed successfully) or Failed (the test case has not been passed).

The project that was used in this study, - is a web-based application for creating postcards, where users can order individual designs, use already created templates, order delivery and post your photo reviews. Project due date - from 9 to 12 months. The team consisted of five developers, three testers and a project manager. Scrum was selected as the development methodology, with iterations duration in 2 weeks. At the planning stage, 4 releases were added to the schedule, after the 1st, 3rd, 6th months of the development and final release had to take place within 9-11 months, and another one month has been added for risk taking. All information, required to calculate the selected metrics was obtained from JIRA. In order to do this fast, the language of JQL queries was used. With its functions you can find tasks by status, ties, etc. In order to calculate the number of test cases that have not passed, you should run the following command:

project = TES AND issuetype = "Test Case" AND status = Fail

You can also add a filter to other folders and team members that created or started test cases. This allows you to perform an analysis of the effectiveness of individual specialists.

Conclusion

The use of unified metrics allowed to establish communication between team members, as well as create feedback with the customer. A clear process of quality measuring has enabled you to always know the current state of the product and provide relevant information to interested parties.

References

1. Basili V. GQM+ Strategies in a Nutshell //Aligning Organizations Through Measurement. Springer International Publishing, 2014. – P. 9-17.
2. Rosenberg, Linda H, Theodore F. Hammer, and Lenore L. Huffman. CiteSeerX Requirements, Testing and Metrics. Requirements, Testing and Metrics. 15th Annual Pacific Northwest Software Quality Conference, Oct. 1998. Web. 17 Mar. 2014
3. GitBook: Selenium WebDriver [Electronic resource]: – Access mode: <https://kreisfahrer.gitbooks.io/selenium-webdriver/content/GLOSSARY.html#selenium>