Week 4 Assignment—E-Commerce Website Test Management Strategy

Shaun Hoadley

CST313: Software Testing

Michael Hayden

06/28/2021

**Test Management Strategy Overview**

For the E-commerce website, we need to develop a test management strategy. The test management strategy will cover test teams, exit criteria, test estimated effort, test and risk, incident reporting, defect classification, and configuration management. Having a good test management strategy will aid in keeping costs and time in line.

**Test Teams**

The consideration of the test teams and the approach to each level of tests are vitally important in achieving efficient development of the E-commerce website. Spillner et al. state, developers should not test their own code, if possible, “because there is a tendency to be blind to our own errors, it is much more efficient to let different people perform testing and development and to organize testing as independently as possible from development.” (2014). Because of this reason, ideally designated testers on the development team should do component testing, a deignated team (ie. Business or IT) on the project should do integration testing, and a designated team of specialists for systems testing. “Especially in system testing, it is often necessary to extend the test team by adding IT specialists, at least temporarily, to perform work for the test team”, allowing the system to be tested and viewed from multiple perspectives (Spillner, Linz, & Schaefer, 2014).

**Test Roles**

To conduct testing of the E-Commerce, the following roles should be assigned: test manager, test designer, test automator, test administrator, and tester (Spillner, Linz, & Schaefer, 2014). The test manager should have experience with personnel management, project management, quality management, and software testing. The test designer should have a test methods and specifications, testing, and software engineering in their skillset. Testing, scripting, programming, test tools, and automation experience is what is desired in a test automator. The test administrator needs a skillset that includes the setup and support of test environments, networking, and system administration. Testers need to be able to follow procedures, execute tests, report failures, and use test objects and testing tools (Spillner, Linz, & Schaefer, 2014).

**Exit Criteria**

Every test case should have specific exit criteria to guide testers in determining when a test is considered to reach completion. The importance of exit criteria is “They prevent tests from ending too early, for example, because of time pressure or because of resource shortages” (Spillner, Linz, & Schaefer, 2014). This exit criteria helps in determining when tests should be started or stopped for the E-commerce site. “To make a right decision to stop testing is an arduous resolution, the pre-defined exit criteria can help simplify this process. It is a very important step where all test processes get stopped and this decision is either made by the tester or the whole team together” (Nidagundi & Novickis, 2016), and by identifying them in the beginning, the testers and developers can rely on the criteria to determine when to stop the tests.

**Test Estimated Effort**

Consideration for both the cost of testing and the costs of undetected defects is vital when planning software testing. Initiating the test effort estimation in the planning phase is the test managers responsibility to ensure the proper assignment and distribution of resources. To provide the most reliable test effort estimation, the E-commerce website test manager will use as a base the estimations on former or similar projects of the same scale. Spillner et al states, “task-driven test effort estimation tends to underestimate the testing effort. Estimating based on experience data of similar projects or typical values usually leads to better results” (2014).

**Test and Risk**

To ensure that the most significant defects are revealed as early as possible, the E-commerce website will implement a risk-based prioritization. “Risk based prioritization of the tests ensures that risky product parts are tested more intensively and earlier than parts with lower risk. Severe problems (causing much corrective work or serious delays) are found as early as possible.” (Spillner, Linz, & Schaefer, 2014). This approach accomplishes a couple of things. First, it will prevent critical errors from downstream effects. Second, it will reduce the costs and time spent handling critical defects that might otherwise make it to production.

Implementing risk management techniques (identifying, prioritizing, and mitigating risks) is important in reducing risks. According to Spillner et al, risk is “the mathematical product of the loss or damage due to failure and the probability (or frequency) of failure resulting in such damage” (2014). Testing, itself, can be used as a technique for risk management as it “provides information about existing problems and the success or failure of correction” (Spillner, Linz, & Schaefer, 2014).

**Incident Reporting**

Throughout the development and testing of the E-Commerce Website, incident reporting will be done for documenting and managing incidents. Every significant and legitimate defect, that is not a result of badly designed testing, should be documented. The following template from *Software testing foundations: A study guide for the certified tester exam* (4th ed.) will be used for documenting incidents (Spillner, Linz, & Schaefer, 2014).

Table

Description automatically generated

Using a template such as this allows incident communication to be consistent, in turn, allowing developers to easily reproduce the defects in order to fix them. “Determining the impact in an

understandable way for developers (data loss, functionality loss, software instability, etc.) and for customers and the hierarchy (impacts in financial terms or in usability terms, noncompliance to requirements, etc.) enables a quick recognition of the anomaly.” (Homès, 2012).

Spillner et al states that an incident report will contain the tested software, test environment, name of the tester, the class containing the defect, the defect prioritization, and information relevant to locating and reproducing the defect (2014). Incident reporting may be done by testers, developers, clients, and users, and the incident reports should be updated upon corrections, thus helping with tracking and managing incidents.

**Defect Classification**

Prioritizing incidents related to the E-commerce website will be implemented through defect classification. There are five levels of severity that reflect the level of impairment a defect causes: 1-Fatal, 2-Very Serious, 3-Serious, 4-Moderate, and 5-Mild. Following is a table from *Software testing foundations: A study guide for the certified tester exam* (4th ed.) that shows the criteria for determining the severity level.

Table

Description automatically generated

Additionally, four levels of prioritization are to be used to identify how quickly the problem should be resolved: 1-Immediate, 2-Next Release, 3-On Occasion, and 4-Open. The following table from *Software testing foundations: A study guide for the certified tester exam* (4th ed.) has the criteria for the prioritization levels.

Graphical user interface

Description automatically generated with medium confidence

**Configuration Management**

Throughout the development of the E-commerce site, configuration management will be used to track the version history of the project and allow multiple developers to contribute without interfering with each others work on the project. A number of problems can be avoided with good configuration management, such as the overwriting of each others code by developers, component integration problems due to unknown versions, and testing difficulties from changes that are untraceable or not knowing which version of the test object that test cases belong to (Spillner et al, 2014). By implementing version management, configuration identification, incident and change status control, and configuration audits, these circumstances can be avoided.

**References**

Homès, B. (2012). Fundamentals of Software Testing, John Wiley & Sons, Incorporated. ProQuest Ebook Central, http://ebookcentral.proquest.com/lib/ashford- ebooks/detail.action?docID=1120766. Created from ashford-ebooks on 2020-04-25 14:33:08.

Nidagundi, P., & Novickis, L. (2016). Introduction to Lean Canvas Transformation Models and Metrics in Software Testing. Applied Computer Systems, 19(1), 30.

Spillner, A., Linz, T., & Schaefer, H. (2014). [*Software testing foundations: A study guide for the certified tester exam* (4th ed.)](https://ashford.instructure.com/courses/86077/modules/items/4350686). Rocky Nook.