**Testing in the Software Development Lifcycle**

Matthew A Keaton

Southern New Hampshire University

CS 320: Software Test, Automation QA

Professor Federico Bermudez

March 9th, 2025

The testing stage is a crucial part in the software development lifecycle (SDLC) to ensure the software being created works as expected, meets user requirements, and has no defects. The testing stage involves different levels of testing, such as component (unit) testing, where individual parts of code are tested before being added all together. Another level in the testing stage is integration testing, where all the individual parts are put together to be test the interactions and interfaces between the integrated parts. The next level that occurs in the testing stage is system testing, that focuses on the whole system’s behavior in a live environment. The last level in the testing phase is acceptance testing, that focuses on the end user’s confidence by ensuring that the software meets expectations and functions properly. During the testing stage, when defects are fixed, the changed software should be retested to ensure the defect is gone. Also, regression testing should be performed on the unchanged software so no new defects are found. Regression testing “involves the creation of a set of tests, which serve to demonstrate that the system works as expected” (Hambling et al., 2019). This process continues until the software functions as expected and meets all requirements.

The testing stage is vital to a successful SDLC because it ensures that the software is functioning as expected, high quality, avoids delays, and meets the user’s requirements prior to deployment. Testing is the “systematic and methodical examination of a work product using a variety of techniques, with the express intention of attempting to show that it does not fulfil its desired or intended purpose” (Hambling et al., 2019). By testing software to show that it does not fulfill its intended purpose, it allows testers to fully test their product and dig deep to find any defects that may be missed. Without a successful testing stage, the software would be deployed with potential defects that could become very costly. The software would fail to meet user requirements, fail to function properly, create poor user experience, and have security issues. It is crucial to have a successful testing stage to promote functionality, security, usability, and reliability.

There are exceptions in which the testing stage would occur earlier or later than it typically does in the SDLC. Different SDLC methodologies implement the testing stage in different parts of the process from each other. For example, in the Waterfall model, each phase must be completed before the next phase can begin. This causes the testing phase to come later in the Waterfall model towards the end of the SDLC. Whereas an Agile model focuses on iterations throughout the entire process where the product is tested frequently. Agile is typically a better approach than Waterfall because testing happens early and throughout the process, which helps identify and fix issues sooner. However, the SDLC method to use depends on the specific needs and goals of the project which can determine when the testing stage occurs.

**Reference**

Hambling, B., Morgan, P., Samaroo, A., Thompson, G., & Williams, P. (2019). *Software testing: An ISTQB-BCS certified tester foundation guide* (4th ed.). BCS Learning & Development Limited. [https://ebookcentral.proquest.com/lib/snhu ebooks/detail.action?docID=5837074](https://ebookcentral.proquest.com/lib/snhu%09ebooks/detail.action?docID=5837074)