Michael Ladderbush

Module 2 Journal

Static testing is when a programs code is reviewed without execution. It evaluates whether the code meets requirements and planned design. By statically testing code developers can find where possible issues are before ever compiling the code. Dynamic testing is when developers actually execute code and see how it performs in relation to system usage and whether it can accomplish the required tasks originally planned.

Static and dynamic testing differed in multiple ways, when they are performed, what they seek to test and what they hope to find. Static testing is usually performed before code is run and is used to gauge whether the program will fulfill its intended purpose, whereas dynamic testing is performed by manipulating the code and executing it in order to see if that confirmed purpose will execute without any issues. By statically testing code developers can confirm whether the program will contain any defects and if it can be optimized before execution, dynamic testing will test the performance and validate the programs output.

It is important to use both static and dynamic testing because they both aim to improve and verify the effectiveness of programs holistically. By first statically testing code a developer will be able to see possible defects in a program, and then subsequently they can confirm those defects by dynamically testing. By understanding defects before they occur and when they do programmers will be able to better assess potential corrections. Statically and dynamically testing alone won’t allow a developer to plan and organize optimizations or ensure that code not only executes correctly but also meets requirements given by the client.