TenStep, Inc.

TenStep Supplemental Paper

9 November 2003

Testing

A project is performed to achieve a certain pre-defined objective or goal. Checking to see whether or not this goal is achieved forms the basis of testing. To ensure that things have not gone awry, a plethora of tests are conducted.

However, before tests are conducted, it is vital to define a test suite. A test suite is a set of machines configured as platforms for testing. They represent the end-users systems, which could be of various configurations. This is true of the latest developments in today's IT arena, where a multitude of platforms are available and used by the end-user. This is actually a downsized version of a test lab, which is a specially equipped and designed facility or space used for usability testing.

Once a test suite is defined, various kinds of tests are conducted. There are many different options for testing:

- **Validation Testing.** Input fields in a project should be verified. HTML code for websites should be verified to meet the Document Type Definition (DTD) for HTML.
- **White Box Testing.** This type of testing includes the testing of all paths, loops and branches. The intention is to go right into the program and test the programmer's code.
- **Black Box Testing.** Each output is tested and verified. Further, the outputs are verified to ensure that they conform to the inputs.
- **Usability Testing.** This includes testing software for understandable interface, ease of navigation and user-friendliness.
- **Static Analysis.** Here, the code itself is examined to ascertain if it matches the specifications that were been laid-down.
- **Compatibility Testing.** The internet needs to cater to an abundance of browsers, operating systems and machines. Advances in networking have made open systems essential. Testing, therefore, should be performed on various kinds of browsers, machines and operating systems.
- **Integration Testing**. While each module or unit of a system is tested at the time of development, the entire system will run as an integrated whole. The dynamics could easily change with various types of interactions. The system should be tested as a whole. Top-down (first the whole system and then each unit), bottom-up (first the sub-systems and then the entire system) or big bang (all units tested together) approaches could be used for integration testing.
- **System Testing.** Testing of non-functional areas like data security, user-friendliness, or the amount of load or stress a system can take. In a web-site, it would include testing the server in terms of download speed, server request handling and machine

resource usage. The system should also be tested in operational surroundings with interfaces to other systems and the real world.

- **Security Auditing**. Security breeches or loopholes, adequate usage of firewalls, machine safety and configuration all need to be tested.
- **Regression Testing.** This refers to testing new code to ensure that the old code is not affected.
- **Acceptance Testing.** Conflict between user acceptability and developer understanding is almost a foregone conclusion. To avoid this inherent area of controversy and dissonance, it is crucial to lay down the protocol of acceptance, including handling and resolving of problems during the acceptance stage at the very onset. These should be documented, and a sign-off must be obtained when the project commences.

Testing should make use of appropriate tools to eliminate unnecessary steps. It should end with a formal documentation of the tests themselves, the methods used and their results. This documentation will be helpful if the tests need to be redone because of some error found.