Department of
Computer Science and Engineering

Title: Product operation factors Correctness, Integrity, Usability.

Software Testing & Quality Assurance Lab

CSE 434

Green University of Bangladesh

1 Objective(s)

- To be familiar with software Operation factor.
- To be familiar with software Correctness, Integrity, Usability testing.

2 Problem analysis: Software Operation Factor

According to McCall's model, five software quality factors are included in the product operation category, all of which deal with requirements that directly affect the daily operation of the software. In this lab we will learn about three most important operation factors that ensure the quality of the software. These factors are as follows.

2.1 Software Correctness

One approach for testing a software is software correctness testing. Software correctness testing is the process of executing a program with the intent of finding errors. Although software correctness testing is aimed primarily at improving quality assurance, verifying and validating described functionality, or estimating reliability, it can also help to reveal potential security vulnerabilities since such vulnerabilities often have a negative effect on the quality, functionality, and reliability of the software.

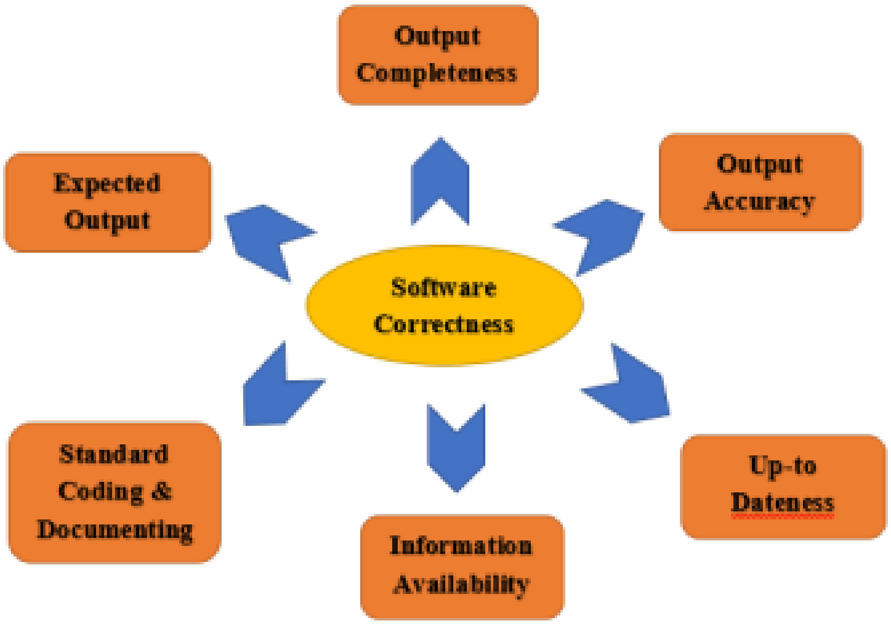

Figure 1: Factors for ensuring software correctness.

Correctness requirements are defined in a list of the software system's required outputs, such as a query display of a customer's balance in the sales accounting information system, or the air supply as a function of temperature specified by the firmware of an industrial control unit. Output specifications are usually multidi mensional.

Percentage of correctness of a software system can be calculated as;
= {(Number of requirements fulfilled) / (Total Number of requirements) * 100%}

2.1.1 Examples

The correctness requirements of a club membership information system consisted of the following:

- The output mission: A defined list of 11 types of reports, four types of standard letters to members and eight types of queries, which were to be displayed on the monitor on request.
- The required accuracy of the outputs: The probability for a non-accurate output, containing one or more mistakes, will not exceed 1
- The completeness of the output information: The probability of missing data about a member, his atten dance at club events, and his payments will not exceed 1
- The up-to-dateness of the information: Not more than two working days for information about participa tion in events and not more than one working day for information about entry of member payments and personal data.

- The availability of information: Reaction time for queries will be less than two seconds on average; the reaction time for reports will be less than four hours.
- The required standards and guidelines: The software and its documentation are required to comply with the client’s guidelines.

2.2 Software Integrity

Integrity requirements deal with the software system security, that is, requirements to prevent access to unau thorized persons, to distinguish between the majority of personnel allowed to see the information (“read permit”) and a 403 Software quality factors limited group who will be allowed to add and change data (“write permit”), and so forth.

Example: The Engineering Department of a local municipality operates a GIS (Geographic Information System). The Department is planning to allow citizens access to its GIS files through the Internet. The software requirements include the possibility of viewing and copying but not inserting changes in the maps of their assets as well as any other asset in the municipality’s area (“read only” permit). Access will be denied to plans in progress and to those maps defined by the Department’s head as limited access documents.

To assure software quality software vendors produce software that free from vulnerabilities.

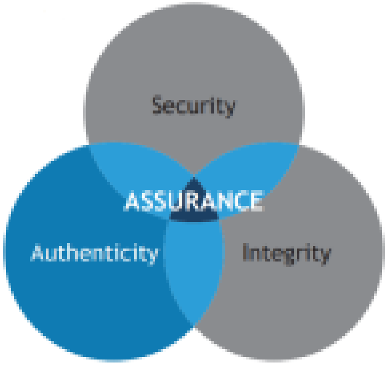

Figure 2: Vulnerability free software characteristics.

2.2.1 Characteristics of software Integrating Testing

- Data compatibility with older version of OS is ensured.
- It checks while verifying data in data tables, it is altered or not.
- It examines all data whether it is successfully save, delete, modify to Database or not.
- It checks whether individual user has access to other users details or not.
- Also helps in analyzing blank value or default value whether it can be retrieved from Database or not.
- Any user can access their account using wrong password/ID or not.
- It also includes different types of security test.

2.3 Usability

Usability Testing also known as User Experience(UX) Testing, is a testing method for measuring how easy and user-friendly a software application is. A small set of target end-users, use software application to expose usability defects. Usability testing mainly focuses on user’s ease of using application, flexibility of application to handle controls and ability of application to meet its objectives. This testing is recommended during the initial design phase of SDLC, which gives more visibility on the expectations of the users.

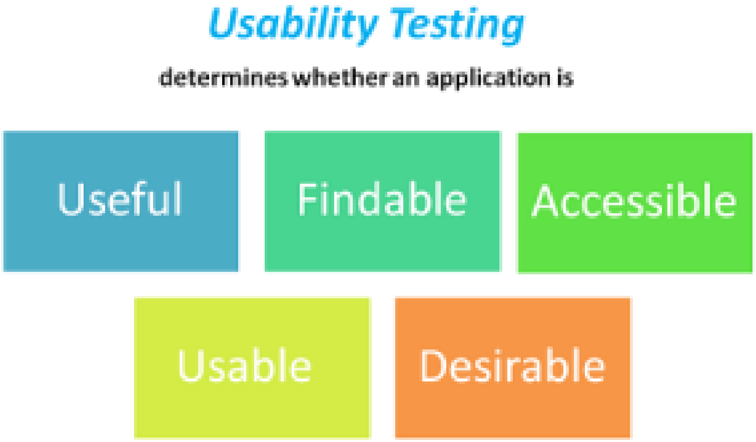

Figure 3: Example Usability Testing Test Cases.

The goal of this testing is to satisfy users and it mainly concentrates on the following parameters of a system:

- Is the system is easy to learn?
- Is the system useful and adds value to the target audience?
- Are Content, Color, Icons, Images used are aesthetically pleasing?

2.3.1 How to do Usability Testing

Usability testing process consists of the following phases;

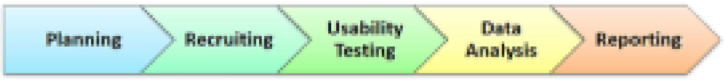

Figure 4: Usability testing process.

1. Planning:- During this phase the goals of usability test are determined. Having volunteers sit in front of your application and recording their actions is not a goal. You need to determine critical functionalities and objectives of the system. You need to assign tasks to your testers, which exercise these critical functionalities. During this phase, the usability testing method, number & demographics of usability testers, test report formats are also determined
2. Recruiting: During this phase, you recruit the desired number of testers as per your usability test plan. Finding testers who match your demographic (age, sex etc.) and professional (education, job etc.) profile can take time.
3. Usability Testing: During this phase, usability tests are actually executed.
4. Data Analysis: Data from usability tests is thoroughly analyzed to derive meaningful inferences and give actionable recommendations to improve the overall usability of your product.
5. Reporting: Findings of the usability test is shared with all concerned stakeholders which can include designer, developer, client, and CEO

3 Lab Task (Please implement yourself and show the output to the instructor)

1. Suppose you want to do a usability test for your proposed system, now write a test case for your software for doing a usability testing.

4 Lab Exercise (Submit as a report)

1. Design a test case for your software for doing a usability and integrity testing.

5 Policy

Copying from internet, classmate, seniors, or from any other source is strongly prohibited. 100% marks will be *deducted* if any such copying is detected.

6 Resources

1. Software Quality Assurance From theory to implementation by DANIEL GALIN
2. https://www.guru99.com/usability-testing-tutorial.html
3. https://www.invespcro.com/blog/usability-metrics/
4. https://www.geeksforgeeks.org/types-software-testing/?ref=leftbar-rightbar