**Software testing (Chapter 4)**

\_\_\_\_\_\_\_\_\_\_

1 . System testing

SYSTEM TESTING is a level of software testing where a complete and integratedsoftware is tested. The purpose of thistest is to evaluate the system'scompliance with the specified requirements. system testing: The process of testing an integrated systemto verify that it meets specified requirements.

\_\_\_\_\_\_\_\_\_\_\_\_

2.Verification

Verification is the process of evaluating work-products of a development phase to determine whether they meet the specified requirements. verification ensures that the product is built according to the requirements and design specifications.

\_\_\_\_\_\_\_\_\_\_\_

3.Performances testing

Performance Testing is a type ofsoftware testing that ensures softwareapplications to perform properly under their expected workload. It is a testingtechnique carried out to determine system performance in terms of sensitivity, reactivity and stability under a particular workload.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

4.Load testing

Aload test is type ofsoftware testing which is conducted to understand the behavior of the application under a specific expected load. Load testing is performed to determine a system's behavior under both normal and at peak conditions.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

5.Regression Testing

REGRESSION TESTING is defined as a type ofsoftware testing to confirm that a recent program or code change has not adversely affected existing features. Regression Testing is nothing but a full or partial selection of already executed test cases which are re-executed to ensure existing

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

6.Agile Testing

Agile testing is a software testing process that follows the principles of agile softwaredevelopment. Agile testing aligns with iterative development methodology in which requirements develop gradually from customers and testing teams.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

7.Acceptances Testing

ACCEPTANCE TESTING is a level of softwaretesting where a system is tested for acceptability. The purpose of this test is to evaluate the system's compliance with the business requirements and assess whether it is acceptable for delivery.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

8.Type of Acceptance testing

.Alpha testing

.Beta testing

\_\_\_\_\_\_\_\_\_

9.alpha testing

testing performed to identify bugs before releasing the product to real users or the public. ... Alpha testing is typically performed by in-house softwareengineers or QA staff. It is the final testing stage before the software is released into the real world.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

10 . Smoke testing

SMOKE TESTING, also known as “Build Verification Testing”, is a type ofsoftware testing that comprises of a non-exhaustive set of tests that aim at ensuring that the most important functions work.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

11. Introduction to metrics

Software Testing Metric is be defined as a quantitative measure that helps to estimate the progress, quality, and health of a software testing effort. A Metricdefines in quantitative terms the degree to which a system, system component, or process possesses a given attribute.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_

12. Size Oriented Metrics

Size oriented software metrics are derived by normalizing quality and/or productivity measures by considering the size of the software that has been produced. If a softwareorganization maintains simple records, a table of size-oriented measures, such as shown in below fig.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

13.Basic Metrics

Base metrics is the raw data collected byTest Analyst during the test case development and execution (# of testcases executed, # of test cases). While calculated metrics are derived from the data collected in base metrics.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_

14.Complexity matrices

COMPLEXITY is asoftware metric used to measure the complexity of a program. It is a quantitative measure of independent paths in the source code of the program. ... Cyclomatic complexity can be calculated with respect to functions, modules, methods or classes within a program.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\*\*\*\*\*\*\*\*\*\*\*\*\*\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_