Reliability testing

This testing is done in order to make the system bug free. This type of testing is done whether a system can operate for a longer duration without developing failures.

Documentation testing

This type of testing is done to test whether the system's user guide and other help topics documents are correct and usable.

Security testing

This type of testing is done to ensure that the programs can be accessed by authorized personnel and that the authorized personnel can access the functions available to their security levels. The system should not allow unauthorized access to data and resources. The goal of security testing is to test how well a system protects against unauthorized internal or external access or damage. Security testing can be done in areas like authentication, authorization, data validation, transport security, data protection, session management, etc.

Usability testing

This type of testing is done to test whether the system is easy to use, learn and operate.

Requirements testing

This type of testing is done to check how the real user will work in the environment. Actual users try to test goals and tasks associated with a product under controlled conditions. It is mainly used for the design of applications. This testing is done by direct observations of people using the system, usability surveys, etc. User tests under this testing is also known as beta testing.

Load testing

This type of testing is done to test how the application behaves when multiple users try to access the system simultaneously across multiple locations. This testing ensures if the performance of the system is acceptable at a pre-determined load level. It tests the normal and predefined conditions of the system.

- **Define scope of testing:** The "in scope" and "out scope" of the testing should be clearly defined. The components of the system that will be tested are known as "in scope" and those components that won't be tested are known as "out scope". A precise scope will give everyone confidence and accurate information of the testing that is being done. All the project members will understand what is being tested and what not.
- **Identify testing type:** It describes the type of tests to be used in the project. Each testing type identifies a special type of product bug. The common goal of all testing types is early detection of all defects before the product is released to the customer. The commonly used testing types are unit test, API testing, integration test, system test, install/uninstall testing and agile testing.
- **Document risks and issues:** It describes all possible risks that may occur during testing phase like insufficient management, tight deadlines, inadequate or erroneous budget estimates. It also describes the effects of these risks on the product.
- **Create test logistics:** In test logistics, the Test Manager mentions the names of the testers and the tests to be run by them and when the test will be run. A tester having the following skills is considered to be most ideal for performing software testing:
  - Understanding customer point of view.
  - Strong desire for quality.
  - Good cooperation.
  - Giving attention to details.

Define test objectives

This phase defines the ultimate goals and the results that is expected from the test execution. The main goal of test objectives is to find as many software bugs as possible so that after release the product is bugs free. Following steps should be considered to define the test objectives—

- All the software features that may be tested should be listed.
- Target or goal based on the above features should be defined.

Define test criteria

The standard or rule on which a test procedure can be based is known as test criteria. There are two types of test criteria—

- **Suspension criteria**: The critical suspension criteria for a test should be specified. During testing if suspension criteria are met, the active test cycle should be suspended until the criteria are resolved.
- **Exit criteria**: The successful completion of a test phase is denoted by the exit criteria. It is the targeted result of the test and is necessary before proceeding to the next development phase. For example, 95% of all test cases must pass.

Resource planning

The detailed summary of all the types of resources required to complete the project task is known as resource planning. Resources can be human, equipment, materials, etc. needed to complete a project. Resource planning helps to determine the number of resources to be used in the project. It helps the Test Manager to make correct schedule and estimation of the project. Human resource can be test manager, tester, developer in test, test administrator, etc. System resource can be server, test tool, network, computer, etc.

Plan test environment

Set up of the software and hardware on which the testing team is going to execute test cases is called the test environment. For example, business environment, user environment, physical environments such as server, front and running environment. To set up the test environment, a strong cooperation is needed between the test team and the development team.

Schedule and estimation

For test estimation, the project is broken down into smaller tasks and time and effort required for each task are allocated. Then, a schedule is created to complete these tasks in the designated time

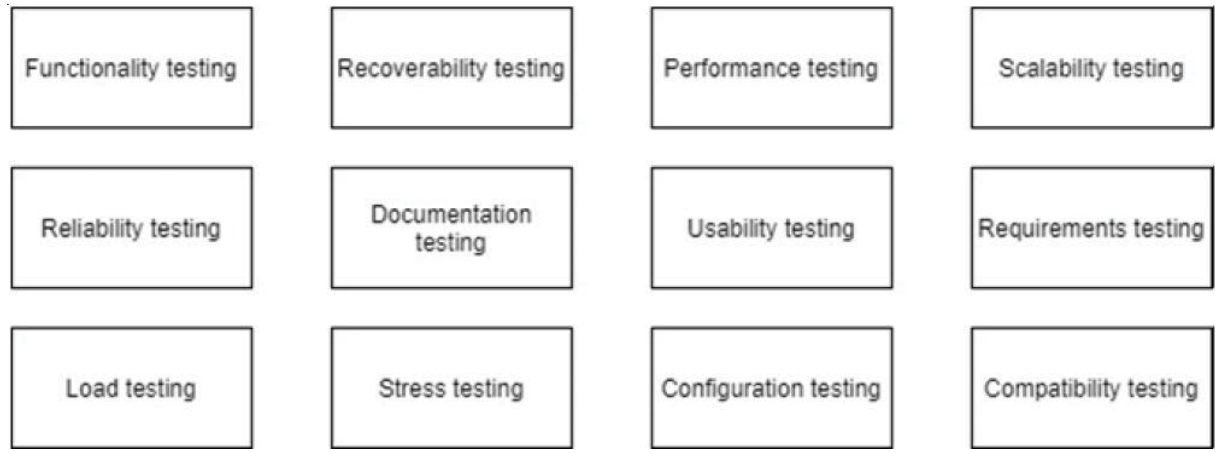


Figure 12.7: Types of System Testing

Functionality testing

This testing helps to test whether the functionality of a product is working as per the requirements specifications and it should be within the capabilities of the system. Functionality testing can be done manually or with automated tools.

Recoverability testing

This type of testing helps to determine whether operations can be continued after a disaster or after the integrity of the system is lost. For example, we are downloading a file and suddenly connection goes off. After connection resumes, downloading starts from where we have left. It does not start from the beginning.

Performance testing

This type of testing is done to test the performance of the system under various conditions in terms of performance characteristics. This testing is also known as compliance testing with respect to performance. This testing ensures that the system requirements are met. Performance testing is done based on three categories: speed, scalability and stability.

Scalability testing

This type of testing is done to test the system's scaling abilities in various terms such as geographic scaling, user scaling and resource scaling.

Prepare job performance aids

In this activity, specification and schedule of the materials to be used by the personnel to run the system are done. This includes activities like display of materials such as program codes, a list of input codes attached to CRT terminals and an instruction schedule posted to load the disk drive. These aids help in the reduction of training and employment of personnel at lower positions.

Prepare operational documents

During the test plan stage, it is important to finalize all the operational documents. Documentation is also done on the experience, training and educational qualifications of personnel in order to ensure proper operation of the system.

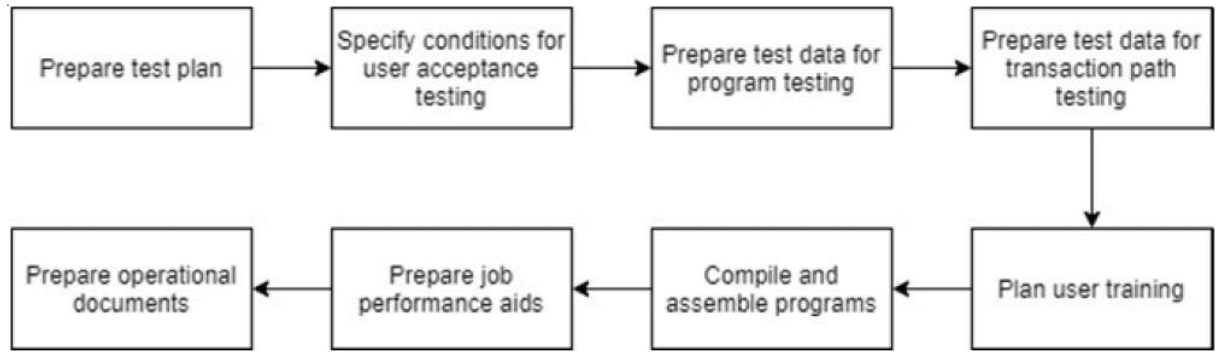


Figure 12.6: Process of Activity Network for System Testing

All the steps involved in activity network for system testing are shown in figure 12.6.

12.7 TYPES OF SYSTEM TESTING

The different types of system testing which are presented in figure 12.7 are explained in the following.

and using specified amount of effort. Inputs from multiple perspectives are required in order to create the schedule. For example,

- Number of working days, project deadlines, employee availability, daily resource availability.
- Risks associated with the project.

Establish test deliverables

The list of all documents, tools and other components that needs to be developed and maintained in order to conduct the testing is known as test deliverables. A different set of test deliverables is required before, during and after testing.

Test deliverables required before testing are—

- Test plan documents.
- Test cases documents.
- Test design specifications.

Test deliverables required during testing are—

- Test data.
- Simulators.
- Test scripts.
- Error logs and execution logs.
- Test traceability matrix.

Test deliverables required after testing are—

- Defect reports.
- Release notes.
- Test results/reports.
- Installation/test procedures guidelines.

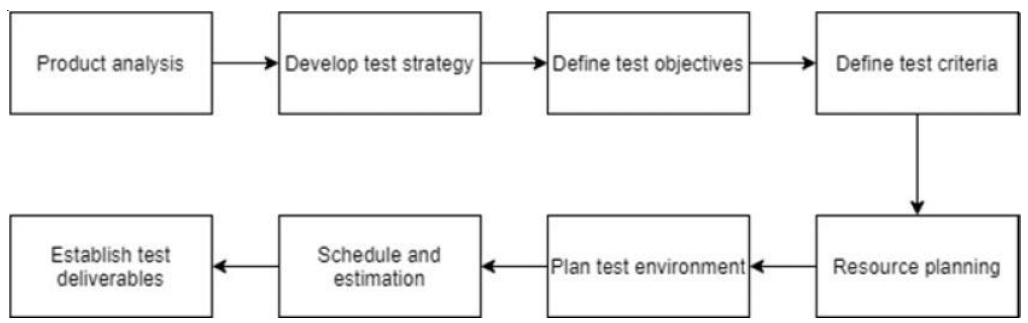


Figure 12.5: Process of Creation of Test Plan

The process of creating a test plan is depicted in figure 12.5.

12.6 ACTIVITY NETWORK FOR SYSTEM TESTING

Activity network for system testing consists of the following activities—

Prepare test plan

Following items are included in a test plan—

- Expected outputs from the system.
- Criteria for evaluating outputs.
- Volume of test data.
- Procedure for using test data.
- Personnel and training requirements.

Specify conditions for user acceptance testing

In order to plan for user acceptance testing, the user and the analyst must agree on the conditions for the test. Most of these conditions can be derived from the test plan. There should be agreement on the test schedules, test durations and the persons designated for the test. It is very important to specify in advance the starting dates and the termination dates for the tests.

Prepare test data for program testing

As the coding of the program begins, the preparation of the test data and documentation is done to ensure that all aspects of the program are properly tested. The data is filled up for future reference.

Prepare test data for transaction path testing

This activity is responsible for the development of data required for testing every condition and transactions that are to be introduced into the system. In order to obtain reliable results, the paths for each transaction from origin to destination should be carefully tested.

Plan user training

The design of the user planning is done to prepare the user for testing and converting the system. User involvement and training is done parallel with programming for the following reasons—

- System group has time for training.
- System gets a clearer image of the user's interest in the new system.
- Trained user participates more effectively in system testing.

Preparation of a checklist is important for user training. The checklist includes provisions for developing training materials like training manuals and other documents that are necessary to complete the training activity. It is important to specify and document the facility requirements and the necessary hardware. Supervisors are trained using a common procedure because of the following reasons—

- User supervisors have knowledge about the capabilities of their staff.
- Staff members respond and accept instructions better from supervisors as compared to outsiders.
- Users are better candidates for handling user training than the system analyst because the users are more familiar with their particular problems.

Compile and assemble programs

All the programs must be compiled and assembled for testing. Before doing this, a complete program description should be available. The description should include purpose of the program, its use, the programmers who wrote the programs and the amount of time taken by the computer to run it. For further reference, program and system flowcharts of the project should be available. Three types of activities are done in this phase—

- **Desk checking**: Programming errors are uncovered in the source code.
- **A run order schedule**: The transactions to be tested and the order in which they should be tested are specified.
- **A test scheme**: How the program should be debugged is specified in the test scheme. It follows two approaches—
  - Bottom-up approach which is responsible for testing small-scale program modules which are linked to a higher-level module. This is done until the program is completed.
  - Top-down approach in which the general program is tested first. It is followed by adding program modules one level at a time to the lowest level.