Debre Markos University
Institute of Technology
School of Computing, Software Eng A/program

Software Quality Assurance and Testing (SEng4112)

Chapter Four

Types of Software Testing

Objective

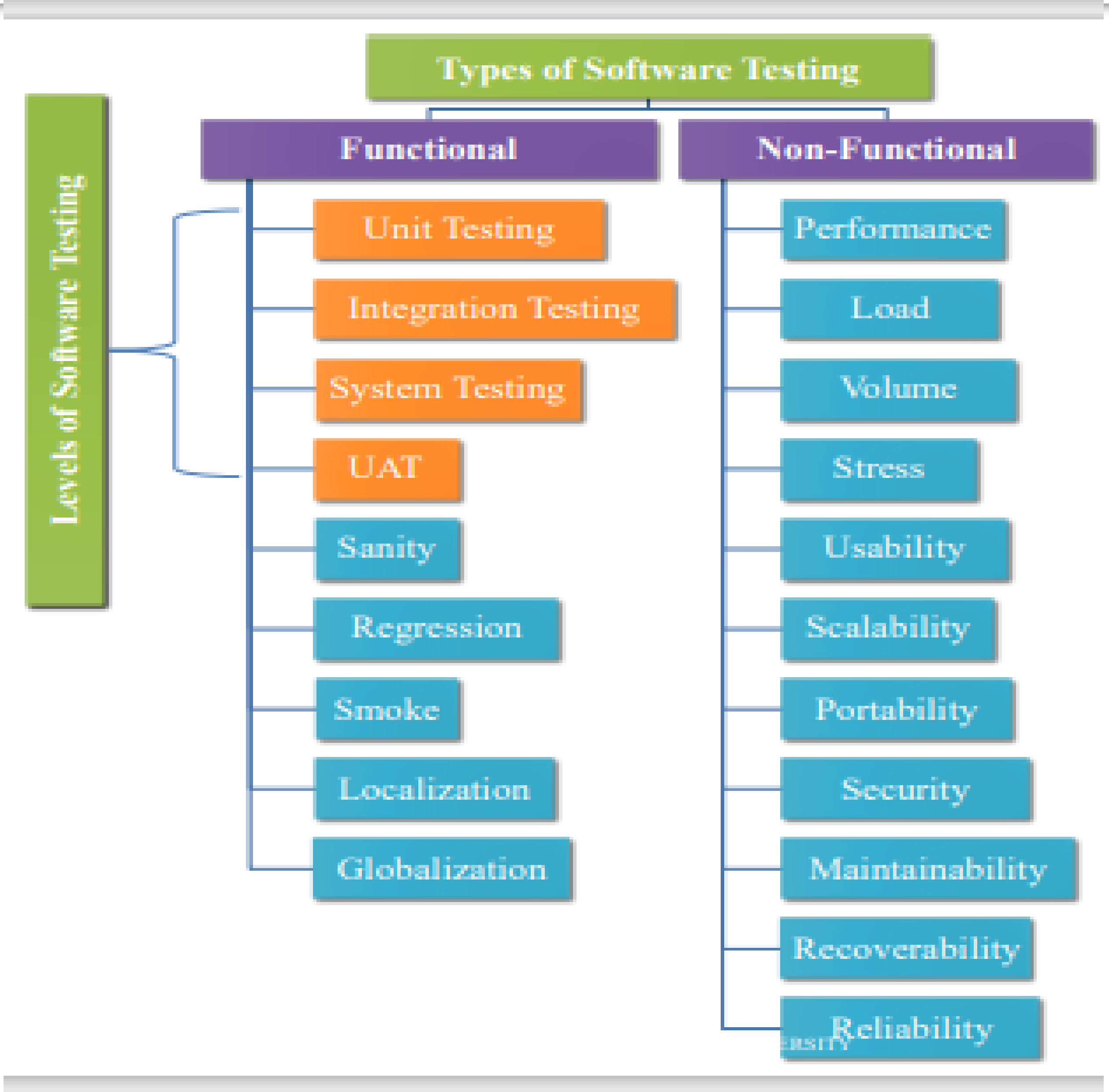
- **After successful completion of this chapter, students will be able to explain**
  - ✓ Software testing types
    - Functional software testing
    - Non-functional software testing

Software Testing Types

- The software testing is generally divided into two major categories, which further consist of numerous testing methods within them

  1. Functional testing
  2. Non-functional testing

Cont...



Types of Software Testing

Levels of Software Testing

Functional

- Unit Testing
- Integration Testing
- System Testing
- UAT
- Sanity
- Regression
- Smoke
- Localization
- Globalization

Non-Functional

- Performance
- Load
- Volume
- Stress
- Usability
- Scalability
- Portability
- Security
- Maintainability
- Recoverability
- Reliability

Functional Testing

- Functional testing is a software testing procedure used within software development wherein the software is tested to check that a software has all the desired functionality specific to its functional requirements.
- This type of testing is carried out based on the product’s functional specification document.
- Functional testing is always done against the application’s business requirement that means it is a black box testing type.
- Moreover, functional testing is conducted on a complete, integrated system to evaluate the system's compliance with its specified requirements.

Cont...

- Functional testing includes any user commands, data manipulation, searches, business processes, user screens, and integrations.
- This testing is done using the functional specifications provided by the client or by using the design specifications like use cases provided by the design team. So, this type of testing is strictly specific.
- For functional testing either manual testing or automation tools can be used but functionality testing would be easier using manual testing only.
- Functional tests tend to answer the question of
  - ➢Can the user do this? or
  - ➢Does this particular feature work?

Functional testing includes:

- Unit testing
- Integrity testing
- System testing
- User Acceptance Testing (UAT)
- Regression testing
- Sanity testing
- Smoke testing
- Localization testing
- Globalization testing

Regression Testing

- Whenever a change in a software application is made, it is quite possible that other areas within the application will be affected by the change.
- It is performed to verify that a fixed bug hasn't resulted in another functionality or business rule violation.
- The intent of regression testing is to ensure that a change, such as a bug fix should not result in another fault being uncovered in the application.

Cont…

- Regression testing is done due to the following reasons:
  - Minimizing the gaps in testing when an application with changes made has to be tested
  - Testing the new changes to verify that the changes made did not affect any other area of the application
  - Mitigates risks
  - Increasing test coverage without compromising timelines
  - Increasing speed to market the product

Mutation testing

- Mutation testing is a type of software testing where we mutate (alter/change) certain statements in the source code intentionally and check if the test cases are able to find the errors.
- It is a type of White Box Testing which is mainly used for unit testing.
- To determine if a set of test data or test cases are useful by deliberately introducing various bugs.
- Re-testing with the original test data/cases to determine if the bugs are detected.

Smoke Testing

- Smoke testing is a kind of software testing performed after software build to ascertain that the critical functionalities of the program are working fine.
- It is executed "before" any detailed functional or regression tests are executed on the software build.
- The purpose is to reject a badly broken application, so that the QA team does not waste time installing and testing the software application.
- In smoke testing, the test cases chosen cover the most important functionality or component of the system.
  - ➢ For example, a typical smoke test would be - check that the application launches successfully, Check that the GUI is responsive … etc.

Cont…

- Smoke testing makes sure that build received from the development team is testable or not.
- It is also named as the "Day 0" check and performed on the building level.
- It saves time as you don't have to test the whole application if the core functionalities of the program are not working.

Sanity Testing

- Sanity testing is a kind of software testing performed after receiving a software build, with minor changes in code, or functionality, to ascertain that the bugs have been fixed and no further issues are introduced due to these changes.
- The goal is to determine that the proposed functionality works roughly as expected.
- If sanity test fails, the build is rejected to save the time and costs involved in a more rigorous testing.
- Ensure that introduced changes doesn’t affect existing functionalities.

Localization (L10N) Testing

- Localization refers to the adaptation of a product, application or document content to meet the language, cultural and other requirements of a specific target market (a locale).
- Localization is sometimes written as L10N, where 10 is the number of letters between L and N.
- Often thought of only as a synonym for translation of the user interface and documentation, localization is often a substantially more complex issue.

Cont…

- **It can entail customization related to:**
  - Numeric, date and time formats
  - Use of currency
  - Keyboard usage
  - Collation and sorting
  - Symbols, icons and colors
  - Text and graphics containing references to objects, actions or ideas which, in a given culture, may be subject to misinterpretation or viewed as insensitive.
  - Varying legal requirements

Globalization (G11N) Testing

- Globalization is the design and development of a product, application or document content that enables easy localization for target audiences that vary in culture, region, or language.
- Globalization is often written G11N, where 11 is the number of letters between G and N in the English word.
- Globalization testing is to ensure that application can function in any culture or locale (language, territory and code page)
- It is also called internationalization testing.

Cont...

- **Globalization testing typically entails:**
  - Designing and developing in a way that removes barriers to localization or international deployment.
  - Providing support for features that may not be used until localization occurs.
  - Enabling code to support local, regional, language, or culturally related preferences.
  - Separating localizable elements from source code or content, such that localized alternatives can be loaded or selected based on the user's international preferences as needed

2. Non-Functional

- This section is based upon testing an application from its non-functional attributes.
- Non-functional testing involves testing a software from the requirements which are non-functional in nature but important such as performance, security, user interface, etc.
- Some of the important and commonly used non-functional testing types are discussed below.

Performance Testing

- It is mostly used to identify any bottlenecks or performance issues rather than finding bugs in a software.
- There are different causes that contribute in lowering the performance of a software:
  - Network delay, Client-side processing, Database transaction processing, Load balancing between servers, Data rendering, etc.
- Performance testing is considered as one of the mandatory testing types in terms of the following aspects:
  - Speed (i.e. response time, data rendering and accessing)
  - Capacity – number of users or operations that your application can handle
  - Stability - ability to remain stable after modification process
  - Scalability - the capacity to be changed in size or scale
- Performance testing can be either qualitative or quantitative and can be divided into different sub-types such as Load testing, Stress testing & Volume.

Load Testing

- It is a process of testing the behavior of a software by applying maximum load in terms of software accessing and manipulating large input data.
- It can be done at both normal and peak load conditions.
- This type of testing identifies the maximum capacity of software and its behavior at peak time.
- Virtual users (VUsers) are defined in the automated testing tool and the script is executed to verify the load testing for the software.
- The number of users can be increased or decreased concurrently based upon the requirements.
- Most of the time, load testing is performed with the help of automated tools such as Load Runner, AppLoader, IBM Rational Performance Tester, Apache JMeter, Silk Performer, Visual Studio Load Test, etc.

Volume Testing

- Volume testing is a non-functional performance testing where the software is subjected to a huge volume of data. It is also referred as flood testing.
- Volume testing is done to analyze the system performance by increasing the volume of data in the database.
- With the help of volume testing, the impact on response time and system behavior can be studied when exposed to a high volume of data.
  - For example, testing a music site behavior when there are millions of user to download the song.

What to test in volume testing?

- **In volume testing, the following things need to be tested**
  - Check if there is any data loss
  - Check the system's response time
  - Check if the data is stored correctly or not
  - Verify if the data is overwritten without any notification
  - Check for warning and error messages, whether it comes at all for volume problems
  - Check whether high volume data affects the speed of processing

Stress Testing

- Stress testing (sometimes called torture testing) is a form of deliberately intense or thorough testing used to determine the stability of a given system or entity.
- This testing is done beyond normal operational capacity; often to the breaking point, in order to observe the results.
- Stress testing includes testing the behavior of a software under abnormal conditions. **For example:**
  - Shutdown or restart of network ports randomly
  - Turning the database on or off
  - Taking away some resources
  - Applying a load beyond the actual load limit
  - Running different processes that consume resources such as CPU, memory, server, etc.

Recoverability Testing

- Recovery testing is the activity of testing how well an application is able to recover from crashes, hardware failures and other similar problems.
- Recovery testing is basically done in order to check how fast and better the application can recover from any type of crash or hardware failure or other catastrophic problems

Examples:

- This can be understood with a simple example of a browser. Say we have opened a browser with 6 sessions on it. Close it abruptly and relaunch it and check whether it is able to recover or not.

Usability Testing

- It is a black box testing technique, which is used to identify errors and defects as well as the improvements in the software by observing the users through their usage and operation.
- This type of testing assists the tester in finding any undetected defect before the release of an application.
- According to Nielsen, usability can be defined in terms of five factors, i.e.
  - Efficiency of use, learnability, memorability, safety and satisfaction.
- Check accessibility
- Check language and understandability
- Check the proper help text message
- Check the navigation

Security Testing

- **Security Testing** is a type of Software Testing that uncovers vulnerabilities, threats, risks in a software application and prevents malicious attacks from intruders.
- Through this type of testing the tester verifies system security, whether it can protect data and maintain functionality as intended.

Cont…

- **Some of the most important concepts covered by security testing are:**
  - **Confidentiality:** protecting against the disclosure of information to parties other than the intended recipient
  - **Integrity:** protecting information from being modified by unauthorized parties
  - **Authentication:** confirming the identity of a person, artifact or product
  - **Authorization:** determining that a requester is allowed to receive a service or perform an operation.
  - **Non-repudiation**: ensuring that a transferred message has been sent and received by the parties claiming to have sent and received the message. Guarantee no denial from both parties.
  - URL Manipulation
  - Session Expiry
  - Privilege Elevation
  - Cross Site Request Forgery

Maintainability Testing

- It basically defines that how easy it is to maintain the system. This means that how easy it is to analyze, change and test the application
- Maintainability testing shall use a model of the maintainability requirements of the software/system.
- It is the degree of measuring the software or system potential to undergo changes, to meet the requirements.
- These requirements may include
  - Correction of errors or faults
  - Additional functionality
  - Adapting the changing environment,
  - Prevention of unexpected failures,
  - Future maintenance, etc.

Scalability Testing

- Scalability testing, is the testing of a software application to measure its capability to scale up or scale down
- Scalability testing is the ability of a network, a system, a database or a process to continue to function well when changes are done in size or volume of the system to meet growing need.
- Scalability testing ensures that an application can handle the projected increase in user traffic, data volume, transaction counts frequency, etc.

Reliability Testing

- Reliability testing is a type of testing to verify that software is capable of performing a failure-free operation for a specified period of time in a specified environment.
- Reliability means "yielding the same," in other terms, the word "reliable" mean something is dependable and that it will give the same outcome every time.
- The same is true for reliability testing. Reliability testing in software assures that the product is fault free and is reliable for its intended purpose.
  - Example:- The probability that a PC in a store is up and running for eight hours without crashing is 99%; this is referred as reliability.

Portability Testing

- Portability testing is the process of determining the degree of ease or difficulty to which a software component or application can be effectively and efficiently transferred from one hardware, software or other operational or usage environment to another.
- The following are the strategies that can be used for portability testing:
  - Building executable (.exe) file to run the software on different platforms.
  - Transferring a developed software from one computer to another.
- This testing type includes overall testing of a software with respect to its usage over different environments.
- Computer hardware, operating systems, and browsers are the major focus of portability testing.

Cont...

Examples

- Software designed to run on Macintosh OS and Microsoft Windows operating systems.
- Applications developed to be compatible with Google Android and Apple iOS phones.
- Video Games or other graphic intensive software intended to work with OpenGL and DirectX API's.
- Software that should be compatible with Google Chrome and Mozilla Firefox browsers.

Cont…

- The ISO 29119 (2013) standard describes Portability with the attributes of

  1. Install-ability
  2. Compatibility
  3. Interoperability and
  4. Adaptability

A. Install-ability testing

- Install-ability of a software is tested on its ability to effectively install the target software in the intended environment.
- Install-ability may include tests for:
  - Space demand,
  - Checking prerequisites,
  - Installation procedures,
  - Completeness,
  - Installation interruption,
  - Customization,
  - Uninstallation

B. Compatibility/ Co-existence- Testing

- It tests the compatibility of multiple, unrelated software systems to co-exist in the same environment, without effecting each other's behavior.
- This is a growing issue with advanced systems, increased functionality and interconnections between systems and subsystems that share components.
- Components that fail this requirement could have profound effects on a system.

For example, if 2 sub-systems share memory or a stack, an error in one could propagate to the other and in some cases cause complete failure of the entire system.

C. Adaptability testing

- Adaptability testing is done to verify that the software can perform all of its intended behaviors in each of the target environments.
- Using communication standards, such as HTML can help with adaptability.
- Adaptability may include testing in the following areas:
  - Hardware dependency
  - Software dependency
  - Representation dependency
  - Standard language conformance
  - Dependency encapsulation and/or
  - Text convertibility.

D. Interoperability testing

- This is testing the capability to communicate, execute programs, or transfer data among various functional units in a manner that requires the user to have little or no knowledge of the unique characteristics of those units.

THE END
THANK YOU!!!