SE 3501

SOFTWARE VALIDATION AND VERIFICATION

M. Mangong Clement

SE 3501
SOFTWARE VALIDATION AND VERIFICATION

Welcome!

This course is design to provide a comprehensive check of a software system against its specification and to ensure you understand the process in verifying and validating a software produced. **M. Mangong Clement**

I. Basic introduction to Software V&V

❖ **Summary**

✓ V & V is the process of investigating that a software system **satisfies specifications** and standards and it fulfills the required purpose.

➢ *Satisfies specifications and standards (verification)*

➢ *Fulfills the required purpose (validation)*

I. Basic introduction to Software V&V

❖ **Verification**

✓ Are you building the product right?

✓ It verifies whether the developed product fulfills the requirements that we have without bugs.

✓ Verification is **static testing**

✓ Activities involved : *Inspections, Reviews, walkthroughs , Desk-checking*

I. Basic introduction to Software V&V

❖ **Verification :** Methods of verification

✓ Peer-reviews

- Easiest and an informal way of reviewing the document or program/ software for the purpose of finding faults.

- Give document or program/software to others to review it so as to give their views about the quality of the product and also expect them to find the fault in the program/software.

I. Basic introduction to Software V&V

❖ **Verification :** Methods of verification

✓ Peer-reviews

▪ Activities include the SRS document verification, SDD verification ,and program verification

▪ The reviewer may prepare a short report on their observation or findings.

I. Basic introduction to Software V&V

❖ **Verification :** Methods of verification

✓ Peer-reviews - Example

➢ Do a peer-reviews of your program of a previously written code.

I. Basic introduction to Software V&V

❖ **Verification :** Methods of verification

✓ Peer-reviews : Advantages

▪ You can expect good results without spending any significant resources.

▪ It is very efficient and significant in its nature.

I. Basic introduction to Software V&V

❖ **Verification :** Methods of verification

✓ Peer-reviews : disadvantages

▪ It leads to bad results if the reviewer doesn't have sufficient knowledge.

I. Basic introduction to Software V&V

❖ **Verification :** Methods of verification

✓ Walk-through Method

- More formal and systematic type of verification method
- The author of the software document presents the document to other persons which can range from 2 to 7.
- Participants are not expected to prepare anything
- The author is responsible for preparing the meeting.

I. Basic introduction to Software V&V

❖ **Verification :** Methods of verification

✓ Walk-through Method

▪ Document is distributed to all participants

▪ At the time of the meeting of the walk-through

▪ The author introduces the content in order to make them familiar with it and all the participants are free to ask their doubts.

I. Basic introduction to Software V&V

❖ **Verification :** Methods of verification

✓ Walk-through Method - example

▪ Do a walk-through demonstration of a program develop.

I. Basic introduction to Software V&V

❖ **Verification :** Methods of verification

✓ Walk-through Method: Advantages

- It may help us to find potential faults

- It may also be used for sharing documents with others.

I. Basic introduction to Software V&V

❖ **Verification :** Methods of verification

✓ Walk-through Method: disadvantages

▪ The author may hide some critical areas and unnecessarily emphasize some specific areas of his/her interest.

I. Basic introduction to Software V&V

❖ **Verification :** Methods of verification

✓ Walk-through of a mobile App

▪ There are three phases in verification testing

1. Requirement verification

2. Design verification

3. Code verification

I. Basic introduction to Software V&V

❖ **Verification :** Methods of verification

✓ Walk-through of a mobile App

1. *Requirement verification*

- The process of verifying and confirming that the requirements are <u>complete</u>, <u>clear</u> and <u>*correct*</u>.
- Before the mobile app goes for design, the testing team verifies business requirements or customer requirements for their correctness and completeness.

I. Basic introduction to Software V&V

❖ **Verification :** Methods of verification

✓ Walk-through of a mobile App

1. *Requirement verification*

Assignment (1)

▪ *How can you test the completeness , clarity, and correctness of a requirement?*

I. Basic introduction to Software V&V

❖ **Verification :** Methods of verification

✓ Walk-through of a mobile App

2. Design verification

▪ The process of checking if the design of the software meets the design specification by **providing evidence**.

▪ The test team checks if layouts , prototypes, navigational charts, architectural designs and database logical models meet the requirements.

I. Basic introduction to Software V&V

❖ **Verification :** Methods of verification

✓ Walk-through of a mobile App

3. Code verification

- The process of checking the code for its <u>completeness</u>, <u>correctness</u> and <u>consistency</u>.

- The testing team checks if construction artifacts such as a source code , user interfaces, and databases physical model of the mobile app meet the design specification.

I. Basic introduction to Software V&V

❖ **Verification :** Methods of verification

✓ Walk-through of a mobile App

3. Code verification

▪ Assignment (2)

How can you test the completeness, correctness and consistency of a source code?

I. Basic introduction to Software V&V

❖ **Verification :** Methods of verification

✓ Inspections

- The most structured and most formal method type

- A team of 3 to 6 participants is constituted which is led by a moderator.

- Group members participate openly,

- After the meeting, a **<u>final report</u>** is prepared after incorporating necessary suggestions by the moderator.

I. Basic introduction to Software V&V

❖ **Verification :** Methods of verification

✓ Inspections

▪ Assignment(3)

How do you prepare a **<u>final report</u>** after a software inspection meeting?

I. Basic introduction to Software V&V

❖ **Verification :** Methods of verification

✓ Inspections : Advantages

- Very effective for finding potential faults or problems in the documents like SRS, SDD,…

- The critical inspections may also help in finding fault and improve the documents which can or in preventing the propagation of fault in the software development life cycle process.

I. Basic introduction to Software V&V

❖ **Verification :** Methods of verification

✓ Inspections : disadvantages

▪ They take time and require discipline

▪ It requires more cost and also needs skilled testers.

I. Basic introduction to Software V&V

❖ **Verification :** Other verification methods

✓ Formal verification

▪ It involves mathematically proving that the requirements are complete and consistent, and that the system will meet the requirements.

I. Basic introduction to Software V&V

❖ **Verification :** Other verification methods

✓ Formal verification

■ Assignment (4)

How can we mathematically proof that requirements are complete and consistent, and that the system will meet the functional and non-functional requirements?

I. Basic introduction to Software V&V

❖ **Verification :** Other verification methods

✓ Prototyping verification

▪ In involves creating a working prototype of the system and testing it to see if it meets the requirements.

I. Basic introduction to Software V&V

❖ **Verification :** Other verification methods

✓ Acceptance Testing

▪ It involves testing the system with real users to see if it meets their needs and requirements.

I. Basic introduction to Software V&V

❖ **Verification :** Other verification methods

✓ User feedback

▪ It involves gathering feedback from the users and incorporating their suggestions and feedback into the requirements.

I. Basic introduction to Software V&V

❖ **Verification :** Other verification methods

✓ Black-box-testing

▪ It involves testing the system without any knowledge of its internal structure or implementation, to see if it meets the requirements.

I. Basic introduction to Software V&V

❖ **Verification :** Other verification methods

✓ Model-based verification

▪ It involves creating a model of the system and simulating it to see if it meets the requirements

I. Basic introduction to Software V&V

❖ **Validation**

✓ Are you building the right product?

✓ A process of checking whether the software satisfies the customer.

✓ Validation is dynamic testing

✓ Activities involved : Black box testing, white box testing, unit testing, integration testing.

I. Basic introduction to Software V&V

❖ **Requirement Validation**

✓ A process of checking the requirement defined for development, define the system that the customer really wants.

✓ We do requirement validation to check issues related to the requirements.

I. Basic introduction to Software V&V

❖ **Requirement Validation**

✓ We perform the following checks

- Completeness checks
- Consistency checks
- Validity checks
- Realism checks
- Ambiguity checks
- verifiability

I. Basic introduction to Software V&V

❖ **Validation of a mobile application**

✓ It checks the functionality, usability and performance of the mobile application.

I. Basic introduction to Software V&V

❖ **Requirement Validation**

✓ Output of req. validation is the list of problems and agreed on actions of detected problems

II. Software Testing

❖ **What is Software Testing?**

✓ A process that consists of all test life cycle activities.

✓ Life cycle activities like static and dynamic testing, planning, preparation and evaluation of software products.

✓ To determine that the software products satisfy customers requirements and are fit for customers use.

✓ To find software defects or failure in advance.

II. Software Testing

❖ **What is Software Testing?**

✓ It can be divided into two parts.

1. Testing as a process

2. Objectives of testing

II. Software Testing

❖ What is Software Testing?

1. Testing as a process
   - ✓ Testing is a process
   - ✓ Testing is both static and dynamic
   - ✓ Planning
   - ✓ Preparation
   - ✓ Evaluation
   - ✓ Software products and related work products

II. Software Testing

❖ **What is Software Testing?**

2. Objectives of testing

✓ Determine that software meets product

requirements

✓ Determine that software is fit for use

✓ Find defects/bugs in software

II. Software Testing

❖ **Testing is a process**

✓ Testing is not a standalone activity

✓ It's a series of activities

II. Software Testing

❖ **Testing is a process - Example**

✓ Software testing is done in all phases of SDLC

✓ Testers start with static testing which includes reviewing the documents like software requirements Specification (SRS) , High level Design, and in the later phases of SDLC when testers gets the working build he installs the software and validates that the software functions as per the end user requirements.

II. Software Testing

❖ **Testing is both Static and Dynamic**

✓ Testing both static and dynamic verification

■ In initial phases of SDLC static testing is done

• Technical Reviews

• Walkthrough

• Static code analysis

II. Software Testing

❖ **Testing is both Static and Dynamic**

✓ Testing both static and dynamic verification

▪ Once the build is available to tester, he starts dynamic testing and validates that software meets customer requirements.

▪ Unit, Integration, system and acceptance testing

II. Software Testing

❖ **Planning**

✓ Test planning is the most important part of testing

▪ You need to plan for:

➢ What you want to achieve?

➢ Who will do what?

➢ Time frame of testing

➢ Control the test process

➢ Prepare test summary reports

II. Software Testing

❖ **Planning**

✓ Once the plan has been finalized you need to do test preparation

➢ Prepare test cases

➢ Prepare test environment

➢ Prepare test data

II. Software Testing

❖ **Evaluation**

✓ While test execution you also need to evaluate the

   software and make sure that:

➢ It meets the exit criteria

➢ It is easy to use (Usability testing)

➢ Meets end user requirements

II. Software Testing

❖ **Software products and related work products**

✓ Software testing is not just about testing the software code. Its requires testing all the related documents like

➢ Software requirements document(SRS)

➢ Design document

➢ Quick reference guide

➢ Training materials

➢ User guides , installation guide

II. Software Testing

❖ **Objectives of Testing**

1. Determine that software meets end user requirements

➢ Software Testing also checks products against

requirements.

➢ The design document is reviewed to make sure that it meets requirements and validation of software is done to ensure it meets the design and requirements.

➢ Testing ensures that product meets its specification and helps stakeholders to make release decisions.

II. Software Testing

❖ **Objectives of Testing**

2. Determine that software is fit for use

➢ Software Testing also demonstrates that the software is fit for use

➢ Testing is done to ensure that software is fit for use for the end users who will be using the software

➢ It ensures that the software does what end users expect it to do

II. Software Testing

❖ **Objectives of Testing**

3. Find Defects/Bugs in software

➢ Software testing detects the defects/bugs in software

➢ Fixing those defects improves the quality of software

➢ By doing root-cause analysis of defects found by testing, improves the software development process.

II. Software Testing

❖ **Why is testing necessary?**

Software Testing is necessary because

➤ A defect in software can cause harm to person, environment or company

➤ A defect can cause loss of money, time or business

➤ Testing improves software quality

➤ Testing reduces the risk

II. Software Testing

❖ **Why do we test something?**

➢ To ensure that it is ok

➢ Testing is necessary because we are all human beings and human beings make mistakes during development.

➢ Some human errors do not impact much on our day to day life and can be ignored, however some errors are so severe that they can break the whole system or software.

➢ In such situations you need to take care that such errors are caught well in advance before deploying the system/software in production environment

II. Software Testing

❖ **Software systems context**

➢ An error in your persona blog does not impact any one else

➢ An error in a business website may put oof the company as it looks unprofessional

➢ Net banking websites or ATMs should be thoroughly tested to maintain bank credibility

➢ Air traffic control system is also very critical for testing

II. Software Testing

❖ **Causes of software defects**

➢ People may make mistake during requirement gathering

➢ People may make mistake during design

➢ People may make mistake during coding.

Due to these mistakes they can be flaws in software and these flaws are known as defects or bugs.
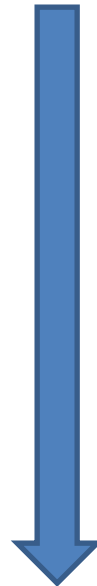
II. Software Testing

❖ **What is cost of defects?**

➢ **Requirements**

➢ **Design**

➢ **Coding**

➢ **Testing**

➢ **Production**

Cost of finding defects fixing the defects increase over time

II. Software Testing

❖ **Role of testing in software development, maintenance and operations**

➢ Testing is important in development and maintenance to identify defects/bugs

➢ Reduces risk of software failure in operational environment

➢ Improves quality of software

➢ Testing is also required as part of contractual agreement or legal requirements (for software which has high risk associated)

II. Software Testing

❖ **Testing and Quality**

➢ Testing helps to measure software quality

➢ Testing provides confidence in software based on number of defects found

➢ Well designed tests uncover most of the defects in software and if the test pass, it gives more confidence in software quality

➢ Testing helps to find defects and software quality improves when those defects are fixed.

II. Software Testing

❖ **What is software quality?**

➢ Quality : The degree to which a component, system or process meets specified requirements and/or user/customers needs and expectations

1. Software quality for developers and testers is that it meets specifications ,is technically good and has few defects.

2. Software quality for other stakeholders may be different- They also need value for money.

II. Software Testing

❖ **What is software quality?**

➢ Different viewpoints for software quality can be

▪ Attributes of products

▪ Fit for use

▪ Good development processes

▪ Value for money

II. Software Testing

❖ **What is root cause analysis?**

1. The finding of real reason for the failure

2. If the software you are testing fails then you do root cause analysis to find the actual cause of that failure.

II. Software Testing

❖ **What is root cause analysis?**

➢ **Ways of carrying out root cause analysis**

1. Brainstorming all the possible causes

2. Grouping them into categories

Welcome!

This course is design to provide a comprehensive check of a software system against its specification and to ensure you understand the process in verifying and validating a software produced.

QUESTIONS