# Chapter 3 : Verification and Validation

Prepared by Roshan Chitrakar

The terms '**Verification**' and '**Validation**' are frequently used in the software testing world but the meaning of these terms are mostly vague and debatable. Verification and validation are often used interchangeably but have different definitions. Generally speaking, Verification is the process confirming that something—software—meets its specification; and Validation is the process confirming that it meets the user's requirements.

Some distinctions between the two are given below: -

| Criteria | Verification | Validation |
|---|---|---|
| *Definition* | The process of evaluating work-products (not the actual final product) of a development phase to determine whether they meet the specified requirements for that phase. | The process of evaluating software during or at the end of the development process to determine whether it satisfies specified business requirements. |
| *Objective* | To ensure that the product is being built according to the requirements and design specifications. In other words, to ensure that work products meet their specified requirements. | To ensure that the product actually meets the user's needs, and that the specifications were correct in the first place. In other words, to demonstrate that the product fulfills its intended use when placed in its intended environment. |
| *Question* | Are we building the product *right*? | Are we building the *right* product? |
| *Evaluation Items* | Plans, Requirement Specs, Design Specs, Code, Test Cases | The actual product/software. |
| *Activities* | • Reviews<br>• Walkthroughs<br>• Inspections | • Testing |

**Software Walkthrough**

The software walkthrough is organized to serve the needs of the producer or author of the software artifact in acquiring superior knowledge of all aspects of the software artifact. It is a learning experience. A desirable side effect of the software walkthrough is the forging of a shared vision among the reviewers and consensus among participants on the approaches taken, product and engineering practices applied, completeness and correctness of capabilities and features, and rules of construction for the domain product. Since the software walkthrough caters to the needs of the author, it is the author who initiates the session. Consequently, there may be several walkthroughs in each life-cycle activity. Software walkthroughs yield open issues and action items. While these issues and action items may be tracked to closure, the only measurement taken is a count of the software walkthroughs held.

**Software Inspection**

The software inspection is structured to serve the needs of quality management in verifying that the software artifact complies with the standard of excellence for software engineering artifacts. The focus is one of verification, on doing the job right. The software inspection is a formal review held at the

conclusion of a life-cycle activity and serves as a quality gate with an exit criteria for moving on to subsequent activities.

The software inspection utilizes a structured review process of planning, preparation, entry criteria, conduct, exit criteria, report out, and follow-up. It ensures that a close and strict examination of the product artifact is conducted according to the standard of excellence criteria, which spans completeness, correctness, style, rules of construction, and multiple views and may also include technology and metrics. This close and strict examination results in the early detection of defects. The software inspection is led by a moderator and assisted by other role players including recorder, reviewer, reader, and producer. The software inspection is initiated as an exit criteria for each activity in the life cycle. Product and process measurements are recorded during the software inspection session and recorded on specially formatted forms and reports. These issues and defects are tracked to closure.

**Reviews**

The activities of the structured review process are organized for software inspections. Software walkthroughs may employ variations for planning, conduct, and follow-up.

Steps in Review Process are : -

1. Planning
2. Preparation
3. Conduct
4. Reporting
5. Follow-Up

**Elements of Review and Walkthrough (Inspection)**

| *Elements* | *Software Inspection* | *Software Walkthrough* |
|---|---|---|
| Structured review process | Planning | Planning: optional |
| | Preparation | |
| | Conduct | Conduct |
| | Report out | |
| | Follow up | Follow up |
| Standard of excellence | Completeness | Completeness |
| | Correctness | Correctness |
| | Style | |
| | Rules of construction | Rules of construction |
| | Multiple views | |
| | | Product and engineering practice |
| Defined roles of participants | Moderator | Moderator: optional |
| | Recorder | |
| | Producer | Producer |
| | Reviewer | Reviewer |
| | Reader | |
| Forms and reports | Inspection record | |
| | Inspection reporting form | |
| | Report summary form | |
| | | Open issues |
| | | Action items |

**Sources:**


- Don O'Neill, Inspection as an Up-Front Quality Technique, *Handbook of Software Quality Assurance*, Fourth Edition
- Verification vs Validation *downloaded from http://softwaretestingfundamentals.com/verification-vs-validation/*