

Software Testing Help

[Home](#)[FREE Updates](#)[Resources](#)[Write And Earn](#)[FREE EBooks](#)[Tutorials](#) ▾[Courses](#) ▾[Manual Testing](#)

Software Development and Testing Methodologies (with Pros and Cons)

Last Updated: **August 31, 2018**

What are the Software Development and Testing Methodologies?

Testing is an essential part of the Software Development Process. A robust and stable software product can be delivered with the use of standard testing methodologies that will help to predict the timeline of the software system.

A software application may turn even more complex with a large number of platforms and devices. More importantly, it is required to ensure whether they meet the specified requirements and can be efficiently installed and operated on the user's machine or not.

[Test Automation](#)[Database Testing](#)[Selenium Tutorials](#)[QTP/UFT](#)[ALM QC](#)[JMeter](#)[LoadRunner](#)[JIRA](#)[Mobile Testing](#)[Agile Testing](#)[SoapUI](#)[TestLink Tutorials](#)[ETL Testing](#)[Agile Scrum Tutorial](#)

for **200,000+ Testers**

Free ebooks and testing tips.

SCRIBE NOW!

Training Tutorials

[Tutorials](#)[Guide](#)[Free Study Guide](#)[HP LoadRunner Tutorials](#)[JMeter Tutorials](#)[JIRA Tutorials](#)



With the means of **security**, **compatibility**, and usability, a software product should be tested by using the proper testing methodology.

In this article, we will discuss on what is meant by Testing Methodologies, how it differs from testing strategies, and types of Software Testing Methods in detail.

What You Will Learn: [\[show\]](#)

Meaning of Testing Methodologies

Methodologies can be considered as the set of testing mechanisms used in software development lifecycle from Unit Testing to System Testing. Selecting an appropriate testing methodology is considered to be the core of the testing process.

[VBScript Tutorials](#)
[Best Test Management Tools](#)
[Appium Studio Tutorials](#)
[Unix Tutorials](#)
[Free DevOps Training](#)
[Free JAVA Tutorials](#)
[Free Python Tutorials](#)
[101+ Interview Questions](#)

Adv



Testing Techniques

Basically, there are 3 testing methodologies which are used for testing. They are White Box Testing, Black Box Testing, and Grey Box Testing. These are also called as **Testing Techniques**. Each of the testing technique is briefed below for your better understanding.

#1) White Box Testing:

White box testing technique is used to examine the program structure and business logic, it validates the code or program of an application. It is also called as *Clear Box Testing*, *Glass Box Testing* or *Open Box Testing*.

White Box Testing Techniques include:

- **Statement Coverage:** Examines all the programming statements.
- **Branch Coverage:** Series of running tests to ensure if all the branches are tested.
- **Path Coverage:** Tests all the possible paths to cover each statement and branch.

#2) Black Box Testing:

Black Box testing method is used to test the functionality of an application based on the requirement specification. Unlike White Box Testing it does not focus on internal structure/code of the application.

Black Box Techniques include:

- **Boundary Value analysis**
- Equivalence Partitioning(Equivalence Class Partitioning)
- Decision Tables
- Domain Tests
- State Models
- Exploratory Testing(Requires less preparation and also helps to find the defects quickly).

#3) Grey Box Testing:



This method of testing is performed with less information about the internal structure of an application. Generally, this is performed like Black Box Testing only but for some critical areas of application, White Box Testing is used.

Models in SDLC

Selecting proper testing methodologies also incorporate with choosing a proper model in SDLC.

The models include:

- Waterfall model
- V model
- Agile model
- Spiral model
- RAD

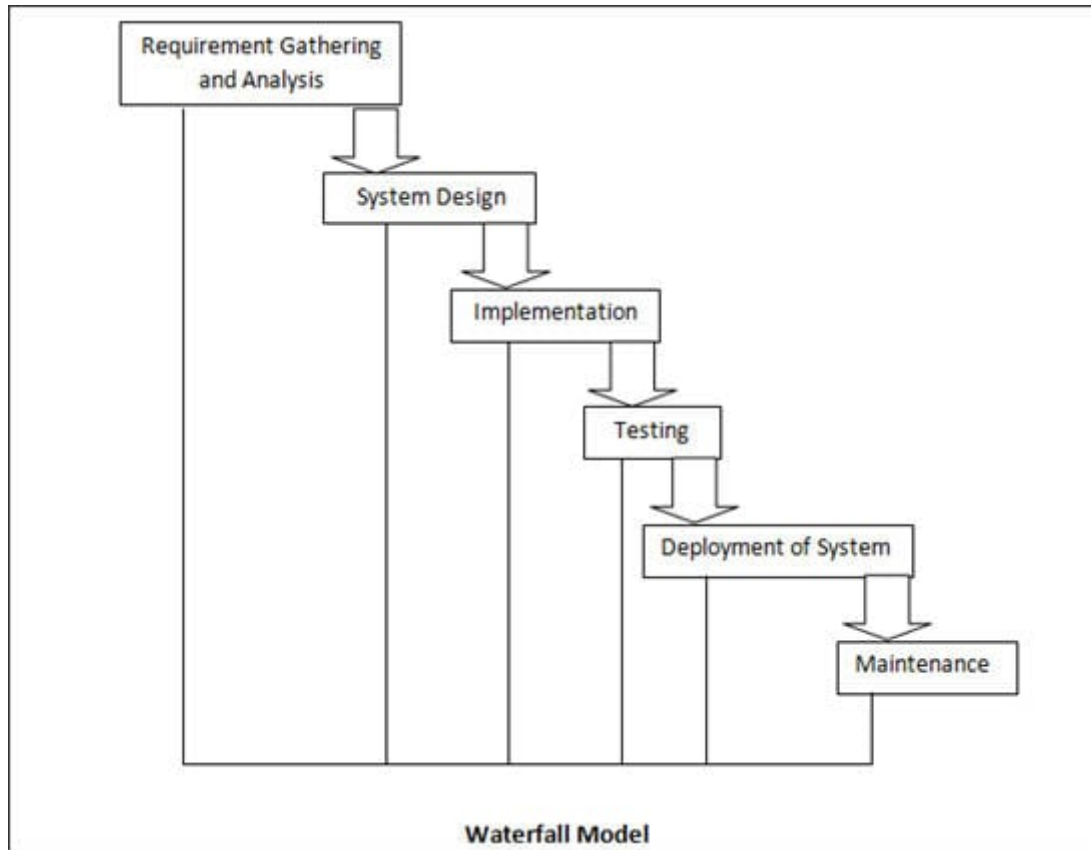
Let's have closer look at each Software Development Methodologies with a brief explanation.

#1) Waterfall Model

Waterfall model is the basic life cycle model which was developed by Winston Royce in 1970. This model represents multiple stages or processes in a sequential manner that flows progressively downward.

This approach is useful when requirements are well known, technology is understood and the resources with required expertise are available.

Waterfall model is defined by the following stages:



- **Requirement Gathering and Analysis:** Capture and analyze all the requirements and make sure whether they are testable or not.
- **System Design:** Create and document design based on requirement analysis. Define the hardware and software requirements.
- **Implementation:** Create robust code for components as per the design and integrate them.
- **System Testing:** Integrated components form a whole system, this phase is performed to ensure whether the system is working as per the requirements, tracking and reporting the testing progress.
- **System Deployment:** Make sure if system is stable with zero bugs, all test criteria had been met, ensure Environment Setup etc.
- **System Maintenance:** Makes sure if the application is working efficiently as per the requirement with the suitable environment. In case a defect is found then that should be fixed and deployed

(updated) in the environment.

Advantages of Waterfall Model:

- Simple and easy to understand.
- Easy to manage as each phase has its own specific deliverables.
- Overlapping of stages is avoided.
- Good for small projects.

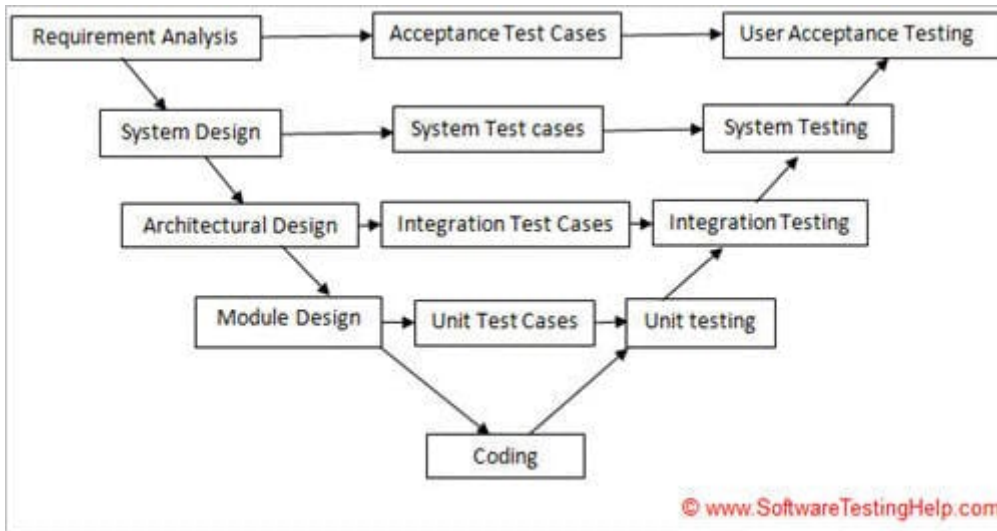
Disadvantages of Waterfall Model:

- Increase in the amount of risk and uncertainty.
- Once entered into Testing phase, cannot change anything in the previous stages **e.g** Design and Coding etc.
- Not good for complex and large projects.
- Not suitable where the requirements keep changing.

#2) V Model

V Model is an *extension of Waterfall Model* where the process execution takes place in a sequential style in V-Shape and is also known as Verification and Validation Model. In this approach, there exists a directly associated testing phase in every single phase of the development cycle.

It has been proven beneficial and cost-efficient than the waterfall model as the testing is performed at each development phase rather than at the end of the development cycle.



V Model is classified into 3 Phases.

- Verification Phase
- Coding Phase
- Validation Phase

a) Verification Phase:

- **Business Requirement Analysis:** Communicate with the customer to understand their expectations and requirements.
- **System Design:** Design complete system and its components along with the hardware and software requirements.
- **Architectural Design:** In this phase architectural specifications are captured. This also known as high-level Design.
- **Module Design:** This is also known as Low-Level Design, Detailed internal design for all the specified system modules.

b) Coding Phase:

This phase contains actual coding phase in the development lifecycle. Programming languages should be chosen based on the system and architectural design specified in the previous phase technology platform. Coding is performed according to the standards and guidelines that are pre-defined.

c) Validation Phase:

- **Unit Testing:** Performed on an individual module to eliminate the bugs at the early stage.
- **Integration Testing:** Performed to test the communication between different modules in the system.
- **System Testing:** **System Testing** is performed on a system as a whole.
- **Acceptance Testing:** This is associated with the business requirements. It is performed in a user environment from the user's point of view.

Advantages of V model

- Simple, easy to use and understand.
- Overlapping is avoided as phases are executed one at a time.
- Easy to manage and suitable for small projects.

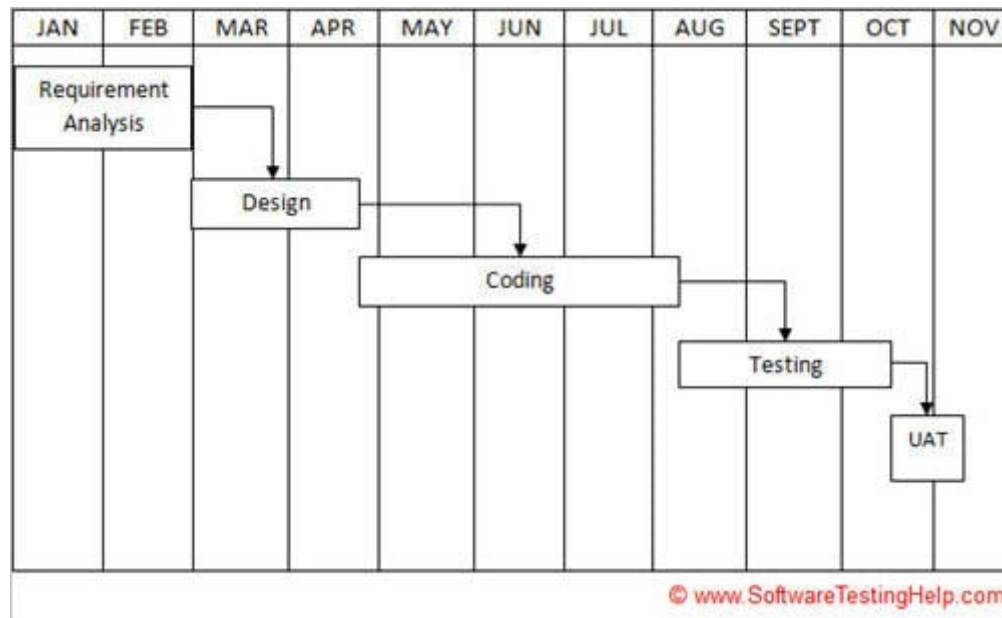
Disadvantages of V Model are more or less similar to the disadvantages of Waterfall model.

#3) Agile Model

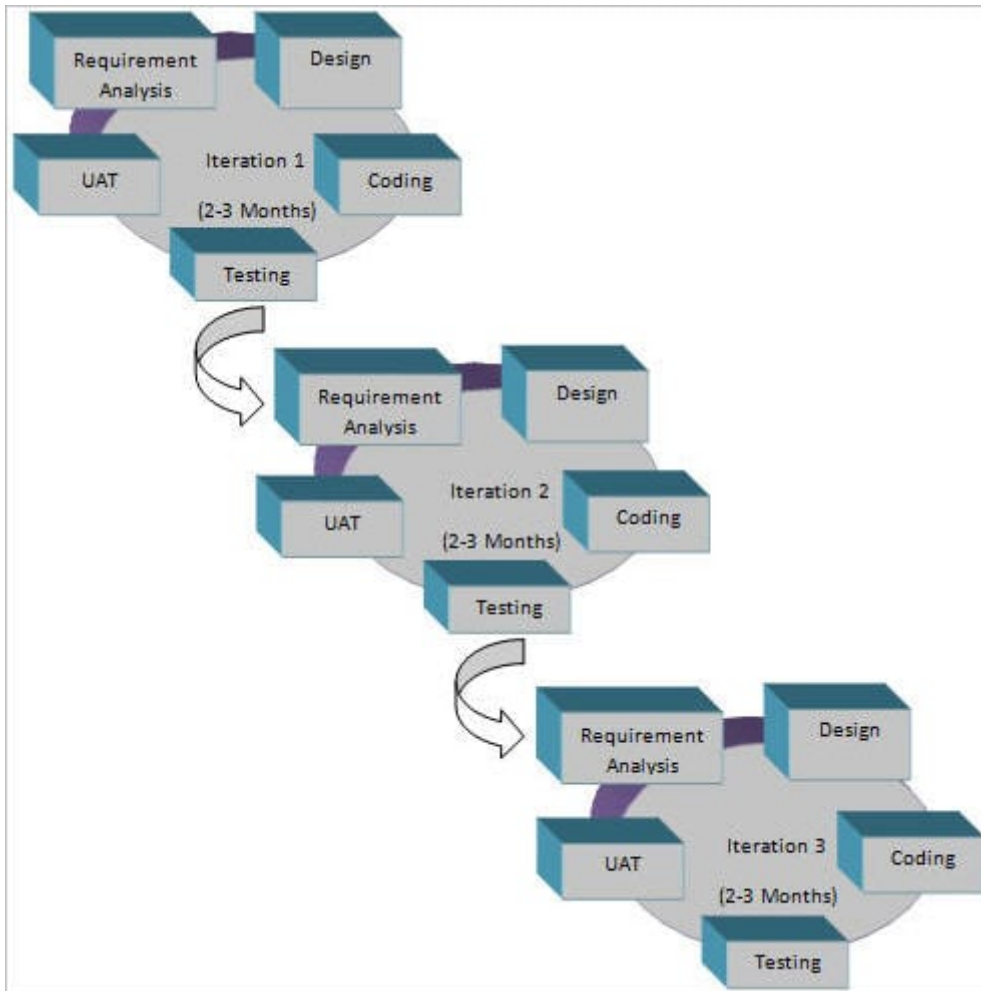
Agile Model shows an iterative and incremental approach. This approach breaks the product into small incremental units to provide iterations. Then each iteration involves steps like Planning, Requirement Analysis, Design, Coding, Unit Testing, Acceptance Testing etc.

This approach also allows continuous interaction with the customer for their feedback and corrections in the requirements at regular intervals.

Following diagram will help you to understand Agile Model approach more precisely:



Following image will show the iteration cycle in Agile Model:



Advantages of Agile model:

- A realistic approach to software development.
- Promotes teamwork.
- Eliminates mismatch between requirements and test cases.
- Rapid and requires minimum amount of resources.
- Suitable for large and long-term projects.
- Good for changing requirements.

- Easy to manage.

Disadvantages of Agile model:

- Not suitable for complex projects.
- Requires heavy amount of interaction with the customer which may cause delay.
- Misguidance of requirements may cause the incorrect development of the software product.
- Increased maintainability risk.
- Handover to another team may be quite challenging.

#4) Spiral Model

The spiral model incorporates iterative development approach along with the systematic approach of the waterfall model. It is similar to the incremental model and emphasis on Risk Analysis.

Spiral Model has four stages:

- Planning Phase
- Risk Analysis
- Engineering Phase
- Evaluation Phase

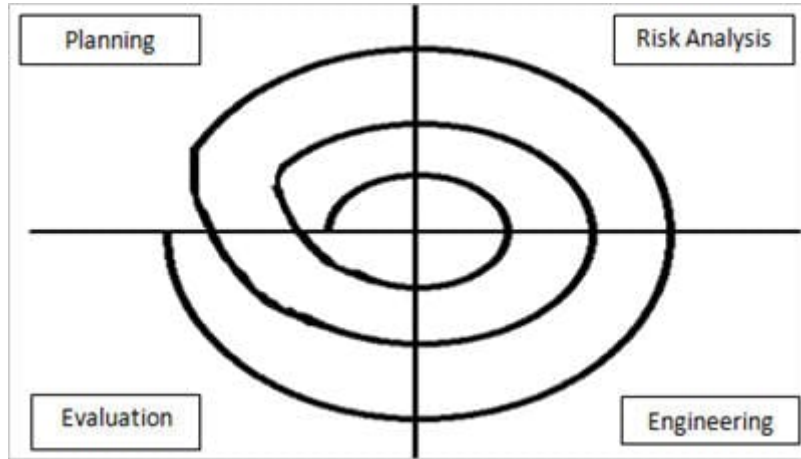
1) Planning Phase: In this phase, the requirements are gathered and reviewed to finalize the test case.

2) Risk Analysis: This stage includes identifying, monitoring and estimating management risks. Requirements are analyzed to identify the risks using techniques like brainstorming, walkthrough etc.

3) Engineering Phase: In this phase, the software is developed and tested at the end.

4) Evaluation Phase: This is the last stage where a customer evaluates the output of a project and gives their feedback for either next spiral or approval.

Pictorial representation of Spiral Model:



When to use Spiral model:

- For high-risk projects.
- When the requirements are complex.
- If a project is large.
- Have sufficient amount of time for getting user's feedback for the next spiral.
- Requires significant changes due to research and exploration.
- Users are not sure of their needs.

Advantages of Spiral Model:

- Avoidance of risk as it involves a high amount of risk analysis.
- Rapid Development.
- Changes in requirements are accommodated easily.
- Requirements can be acquired more accurately.

Disadvantages of Spiral model:

- Complex management.

- Not suitable for small projects.
- May involve no. of spirals(indefinite).
- Costly.
- Requires high amount of risk analysis and expertise for their project's success.

#5) RAD Model

Rapid Application Development(RAD) is a type of incremental model. In this approach, components are developed in parallel.

This is a rapid approach and it can give a fast product to the customer to provide feedback.

Phases in RAD are as follows:

- **Business Modeling:** Identifies vital information and its flow between various business channels.
- **Data Modeling:** Information gathered in the previous stage is used to define data objects required for the business.
- **Process modeling:** Data objects are converted to get business objective and flow of information.
- **Application Generation:** In this phase, automation tools are used to convert the process model into actual code.
- **Testing and Turnover:** Tests all the components of a system, hence overall testing time is reduced.

Advantages of RAD Model:

- Progress can be measured.
- Reduces development time.
- Increased reusability.
- Quick initial reviews.
- Enhances customer feedback.

Disadvantages of RAD Model:

- Requires high skilled resources.
- High-cost estimation.
- Not applicable for cheaper projects.
- High dependency on modeling skills.
- Only a modularized system can be built using RAD.

Difference between Testing Methodologies & Testing Strategies

The answer to this is not much complex as there is a simple difference between both.

Testing Methodologies are the methods or approaches to testing that includes from Unit testing through System Testing.

Testing Strategies is an overview of the key issues that occur in the testing process and is to be taken into consideration by the project manager, a team of developers and testers.

The above-discussed Software Testing Methods are used to implement n number of testing strategies.

Some of them are listed below:

1) Unit Testing:

- Focuses on very small functional units.
- The simplest way to check smallest units for isolation.
- Generally performed by developers.

2) Integration Testing:

- This is the next step to be performed on the developer's side.

- Provide mechanism to test interaction, inter-operation, and communication between the different modules of software

3) Functional Testing:

It is used to check the functionalities of a software system i.e. output to the given input.

4) Regression Testing:

Checks if the bug fixing has happened at one place so that the complex functionalities should not cause any change in another core area.

5) System Testing:

- Testing all the integrated modules as a collective system.
- Combines multiple features into end-to-end scenarios.

6) Performance Testing:

Tests the performance of the application in critical situations like transferring big sized file, concurrent users access to the system, configuration failure etc.

7) Acceptance Testing:

- Generally Final level of testing where software product is examined as users perspective by testers
- The result of this step is subjective and takes a little to find exact issue

Conclusion:

Choosing a proper Testing Methodology is the action or set of actions that lies at the core of the testing process. This may even be a versatile activity that changes according to the business requirements

and timeline of the software product.

However, one can choose single or even multiple Software Development and Testing Methodologies to have a more flexible and efficient end product that satisfies the customer's needs and expectation in the desired or less time limit.

Let us know your thoughts/suggestions in the comments section below.

[ABOUT US](#) | [CONTACT US](#) | [ADVERTISE](#) | [TESTING SERVICES](#)

ALL ARTICLES ARE COPYRIGHTED AND CAN NOT BE REPRODUCED WITHOUT PERMISSION.

© 2006 - 2018 SOFTWARE TESTING HELP — READ OUR **[COPYRIGHT POLICY](#) | [PRIVACY POLICY](#) | [TERMS](#) | [COOKIE POLICY](#) | [LINK TO US](#)**

Thanks For Reading! 50 Million Visits and Counting!