

Software Testing Methodologies: Learn QA Models

What is Software Testing Methodology?

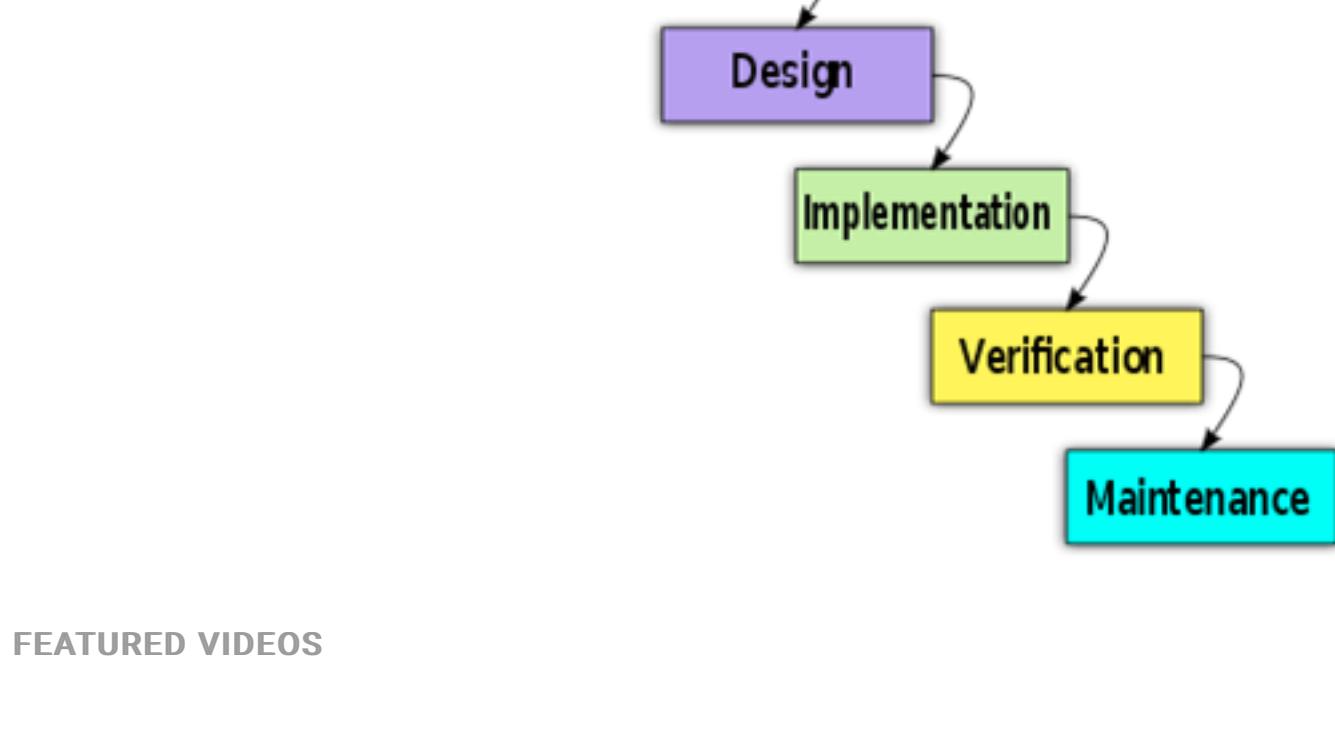
Software Testing Methodology is defined as strategies and testing types used to certify that the Application Under Test meets client expectations. Test Methodologies include functional and non-functional testing to validate the AUT. Examples of Testing Methodologies are [Unit Testing](#), [Integration Testing](#), [System Testing](#), [Performance Testing](#) etc. Each testing methodology has a defined test objective, test strategy, and deliverables.

Note: Since Software Testing is an integral part of any Development Methodology, many companies use the term Development Methodologies & Testing Methodologies colloquially. Hence Testing Methodologies could also refer to Waterfall, Agile and other QA models as against the above definition of Testing Methodologies. Discussion on various testing types does not add value to the readers. Hence, we will discuss the different development models.

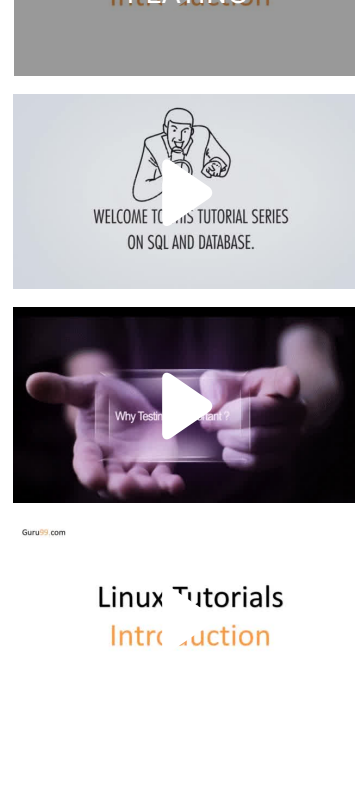
In this tutorial, you will learn-

- [Waterfall model](#)
- [Iterative development](#)
- [Agile methodology](#)
- [Extreme programming](#)
- [Which Software Methodology to choose?](#)
- [How to setup software testing methodologies?](#)

Waterfall Model



FEATURED VIDEOS



What is it?

In the [waterfall model](#), software development progress through various phases like Requirements Analysis, Design etc - **sequentially**.

In this model, the next phase begins only when the earlier phase is completed.

What Is The Testing Approach?

The first phase in the waterfall model is the requirements phase in which all the project requirements are completely defined before starting the testing. During this phase, the test team brainstorms the scope of testing, test strategy and drafts a detailed test plan.

Only once the design of software is complete, the team will move on to execution of the test cases to ensure that the developed software behaves as it expected.

In this methodology, the testing team proceeds to the next phase only when the previous phase is completed.

Advantages

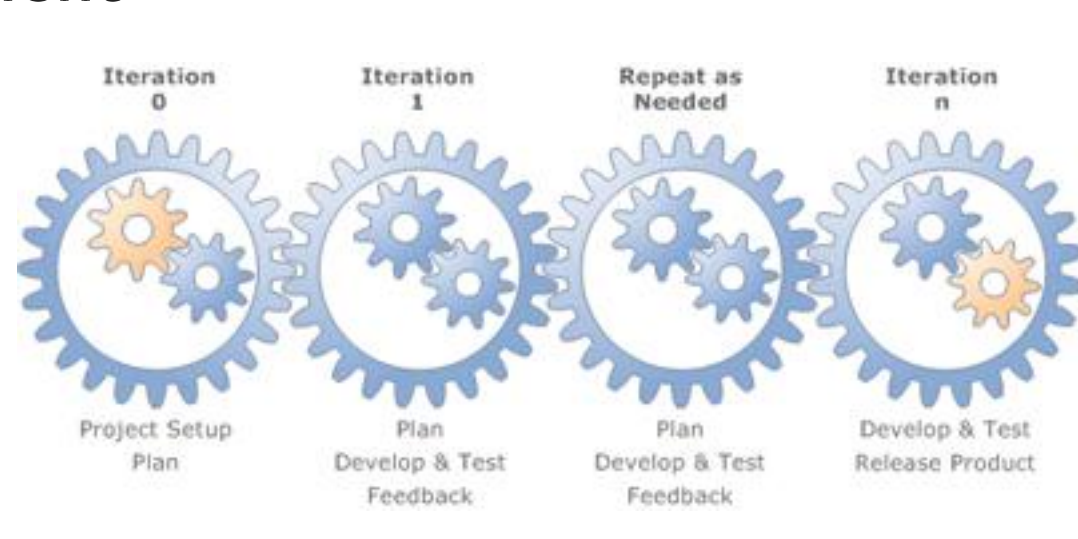
This software Engineering model is very simple to plan and manage. Hence, projects, where requirements are clearly defined and stated beforehand, can be easily tested using a waterfall model.

Disadvantages

In the waterfall model, you can begin with the next phase only once the previous phase is completed. Hence, this model cannot accommodate unplanned events and uncertainty.

This methodology is not suitable for projects where the requirements change frequently.

Iterative development



What is it?

In this model, a big project is divided into small parts, and each part is subjected to multiple iterations of the waterfall model. At the end of an iteration, a new module is developed or an existing module is enhanced. This module is integrated into the software architecture and the entire system is tested all together

What is the testing Approach?

As soon as iteration is completed, the entire system is subjected to testing. Feedback from testing is immediately available and is incorporated in the next cycle. The testing time required in successive iteration can be reduced based on the experience gained from past iterations.

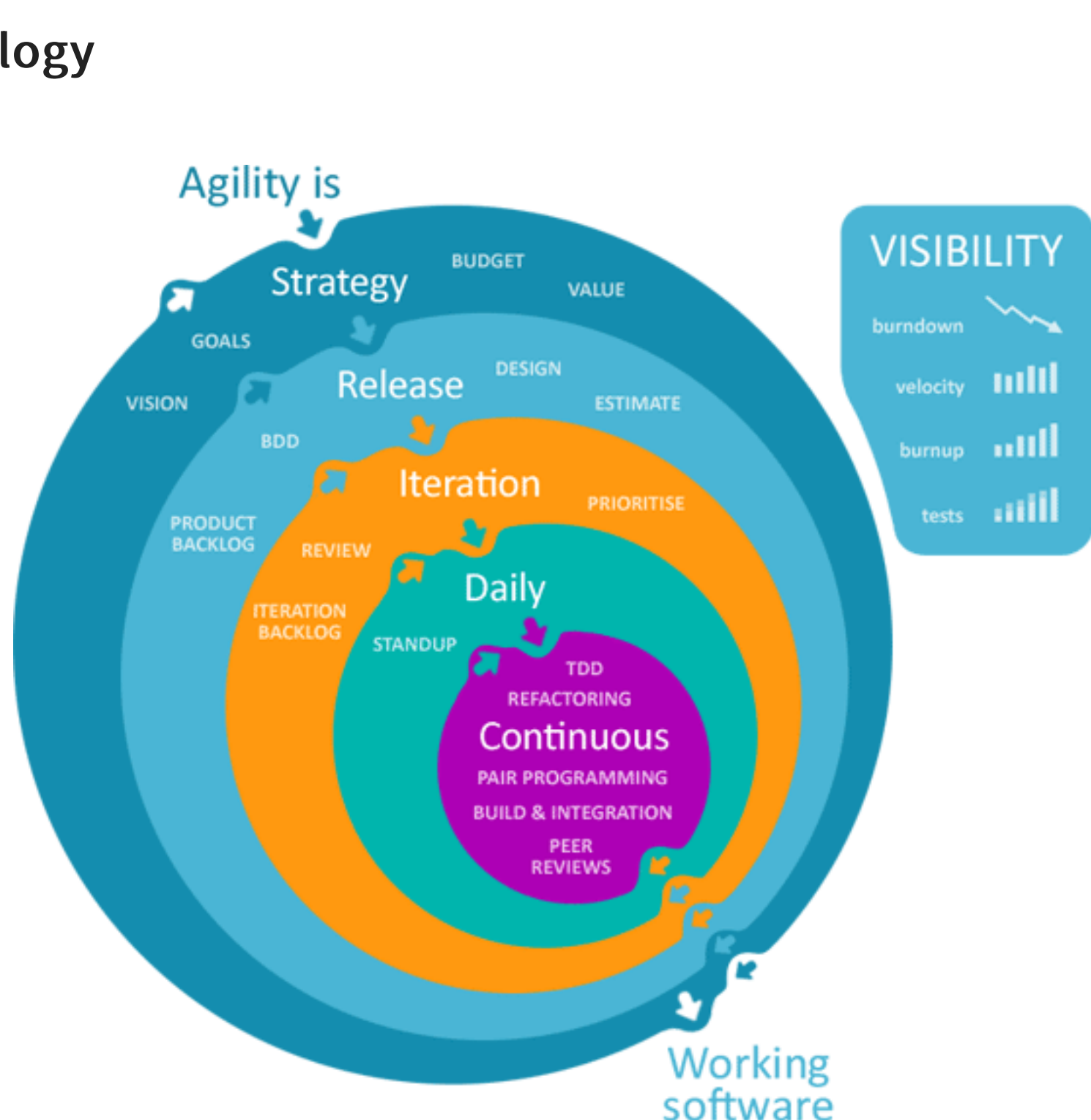
Advantages

The main advantage of iterative development is the test feedback is immediately available at the end of each cycle.

Disadvantages

This model increases communication overheads significantly since, at the end of each cycle, feedback about deliverables, effort etc must be given.

Agile methodology



What is it?

Traditional software development methodologies work on the premise that software requirements remain constant throughout the project. But with an increase in complexity, the requirements undergo numerous changes and continuously evolve. At times, the customer himself is not sure what he wants. Though the iterative model addresses this issue, it's still based on the waterfall model.

In Agile methodology, software is developed in incremental, rapid cycles. Interactions amongst customers, developers and client are emphasized rather than processes and tools. The agile methodology focuses on responding to change rather than extensive planning.

What Is The Testing Approach?

Incremental testing is used in agile development methods and hence, every release of the project is tested thoroughly. This ensures that any bugs in the system are fixed before the next release.

Advantages

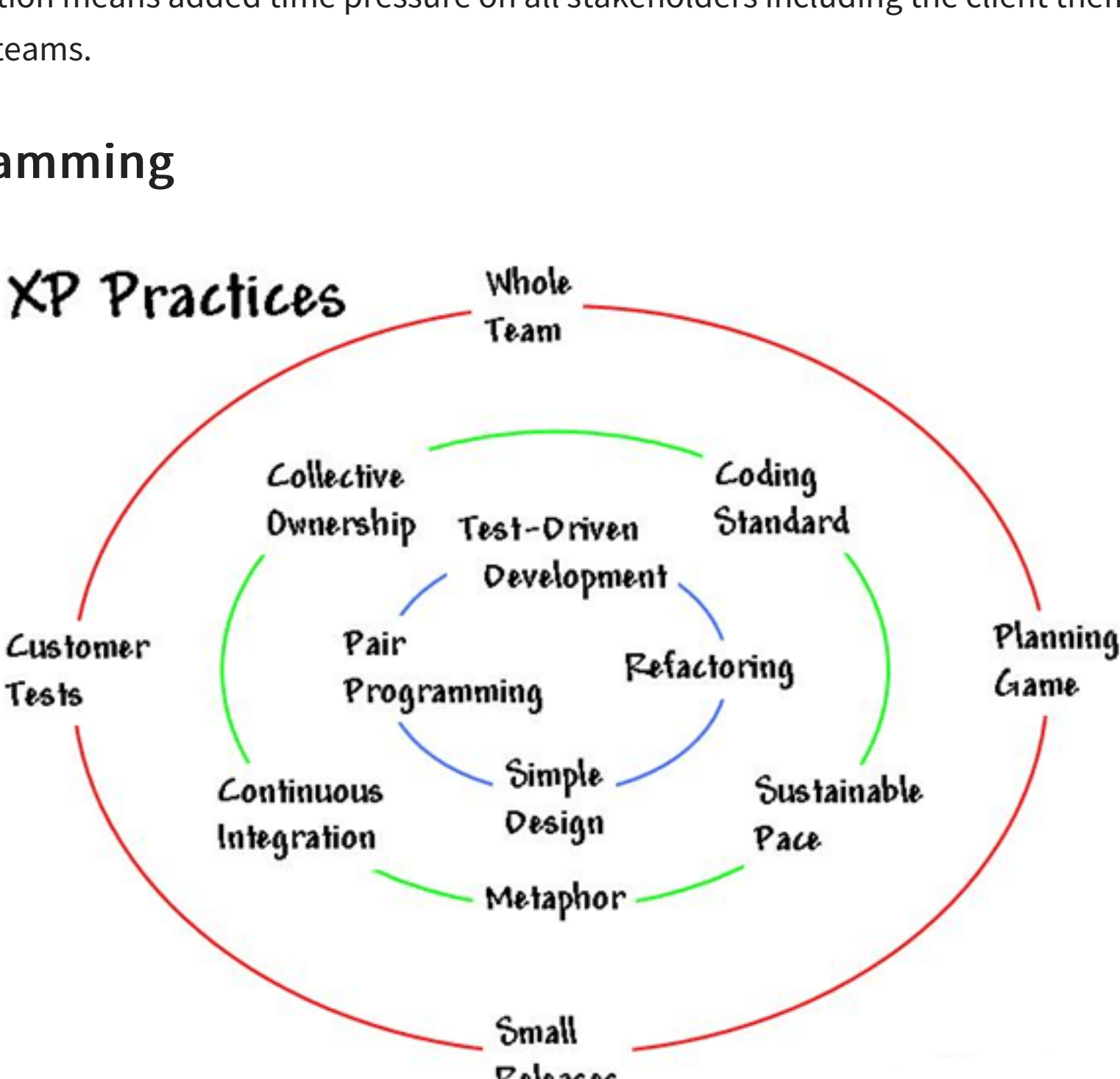
It is possible to make changes in the project at any time to comply with the requirements.

This incremental testing minimizes risks.

Disadvantages

Constant client interaction means added time pressure on all stakeholders including the client themselves, software development and test teams.

Extreme programming



What is it?

Extreme programming is a type of agile methodology which beliefs in short development cycles. A project is divided into simple engineering tasks. Programmers code a simple piece of software and get back to the customer for feedback. Review points from the customer are incorporated and the developers proceed with the next task.

In extreme programming developers usually, work in pairs.

[Extreme Programming](#) is used in places where customer requirements are constantly changing.

What Is The Testing Approach?

Extreme programming follows a Test-driven development which is described as follows -

1. Add a [Test Case](#) to the test suite to verify the new functionality which is yet to be developed
2. Run all the tests and obviously the new test case added must fail since the functionality is not coded yet
3. Write some code to implement the feature/functionality
4. Run the test suite again. This time, the new test case should pass since the functionality has been coded

Advantages

Customers having a vague software design in mind could use extreme programming

Udemy™ Official Site - Join Now & Get 20% Off

Join Millions of Learners From Around The World Already Learning On Udemy!

udemy.com

OPEN

Continuous testing and continuous integration of small releases ensure software code is delivered is of high quality

Disadvantages

Meetings amongst the software development team and clients add to time requirements.

Which Software Methodology to choose?

There are tons of methodologies available for software development and its corresponding testing. Each testing technique and methodology is designed for a specific purpose and has its relative merits and demerits.

Selection of a particular methodology depends on many factors such as the nature of a project, client requirement, project schedule, etc.

From a testing perspective, some methodologies push for testing input early in the development life cycle, while others wait until a working model of the system is ready.

How to setup software testing methodologies?

Software testing methodologies should not be set up just for the sake of testing software code. The big picture should be considered and the prime goal of the project should be satisfied with the testing methodology.

Scheduling

Realistic scheduling is the key to the implementation of successful testing methodology and the schedule should meet the needs of every member of the team.

Defined deliverables

In order to keep all the members of the team on the same page, well-defined deliverables should be provided. The deliverables should contain direct content without any ambiguity.

Test approach

Once scheduling is complete and defined deliverables are made available, the testing team should be able to formulate the right test approach. Definition documents and developer meetings should indicate the team about the best test approach that can be used for the project.

Reporting

Transparent reporting is very difficult to achieve, but this step determines the effectiveness of the testing approach used in the project.

[Prev](#)

[Report a Bug](#)

[Next](#)

YOU MIGHT LIKE:

<p>CSTE Certification Guide with Study Material</p> <p>What is CSTE? Full Form of CSTE is "Certified Software Test Engineer." The Certified Software...</p> <p>Read more ></p>	<p>What is Interface Testing? Types & Example</p> <p>What is Interface Testing? Interface Testing is defined as a software testing type which verifies...</p> <p>Read more ></p>	<p>Frontend Testing Vs. Backend Testing: What's the Difference?</p> <p>What is Frontend Testing? Frontend Testing is a type of testing that checks the Presentation layer of a 3...</p> <p>Read more ></p>
<p>Protocol Testing Tutorial: L2 & L3</p> <p>Before we learn about protocol testing, let's understand: What is Protocol? When a computer...</p> <p>Read more ></p>	<p>Best Agile and Scrum Certifications in 2021</p> <p>To cope up with rapidly changing IT market, and its growing demands and expectations,...</p> <p>Read more ></p>	<p>What is Test Analysis (Test Basis) in Software Testing?</p> <p>Test Analysis is Test Analysis in software testing is a process of checking and analysing the test...</p> <p>Read more ></p>

Top Tutorials

[f](#) [t](#) [i](#) [n](#) [l](#) [k](#) [e](#)

About

[About Us](#)

[Write For Us](#)

[Contact Us](#)

Career Suggestion

[SAP Career Suggestion Tool](#)

[Software Testing as a Career](#)

Interesting

[eBook](#)

[Blog](#)

[Quiz](#)

[SAP eBook](#)

Execute online

[Execute Java Online](#)

[Execute Javascript](#)

[Execute HTML](#)

[Execute Python](#)

Selenium

Testing

Hacking

SAP

Java

Python

Jmeter

Informatica

JIRA