

V-Model in Software Testing

V Model

V Model is a highly disciplined SDLC model in which there is a testing phase parallel to each development phase. The V model is an extension of the waterfall model in which testing is done on each stage parallel with development in a sequential way. It is known as the Validation or Verification Model.

Key Software Engineering

Terms:

SDLC: SDLC is Software Development Life Cycle. It is the sequence of activities carried out by Developers to design and develop high-quality software.

STLC: STLC is Software Testing Life Cycle. It consists of a series of activities carried out by Testers methodologically to test your software product.

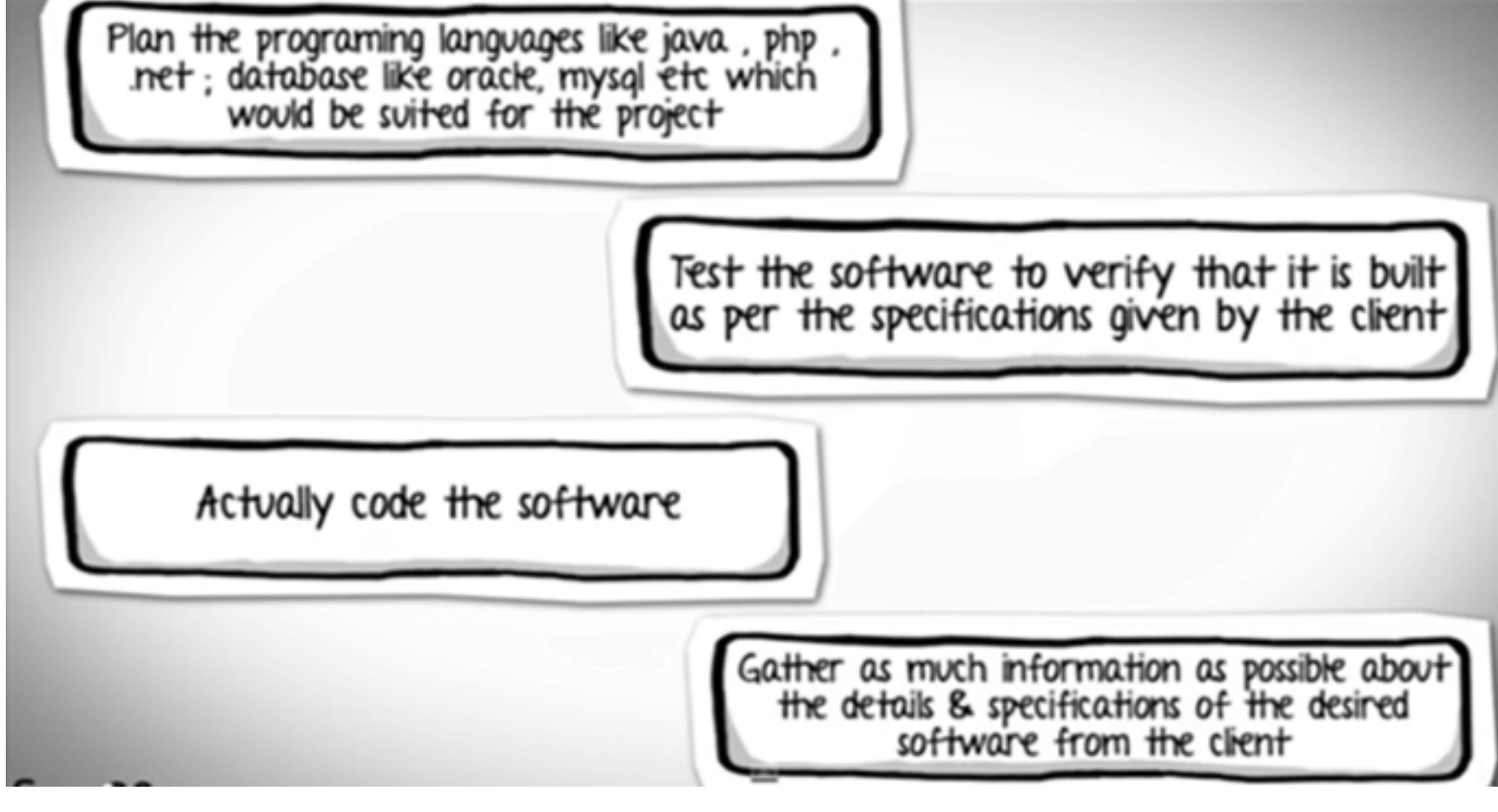
Waterfall Model: Waterfall model is a sequential model divided into different phases of software development activity. Each stage is designed for performing the specific activity. Testing phase in waterfall model starts only after implementation of the system is done.



Click [here](#) if the video is not accessible

EXAMPLE To Understand the V Model

Suppose, you are assigned a task, to develop a custom software for a client. Now, irrespective of your technical background, try and make an educated guess about the sequence of steps you will follow, to achieve the task.



The correct sequence would be.

Different phases of the Software Development Cycle	Activities performed in each stage
Requirement Gathering stage	<ul style="list-style-type: none">Gather as much information as possible about the details & specifications of the desired software from the client. This is nothing but the Requirements gathering stage.
Design Stage	<ul style="list-style-type: none">Plan the programming language like Java, PHP, .net; database like Oracle, MySQL, etc. Which would be suited for the project, also some high-level functions & architecture.
Build Stage	<ul style="list-style-type: none">After the design stage, it is build stage, that is nothing but actually code the software
Test Stage	<ul style="list-style-type: none">Next, you test the software to verify that it is built as per the specifications are given by the client.
Deployment stage	<ul style="list-style-type: none">Deploy the application in the respective environment
Maintenance stage	<ul style="list-style-type: none">Once your system is ready to use, you may require to change the code later on as per customer request

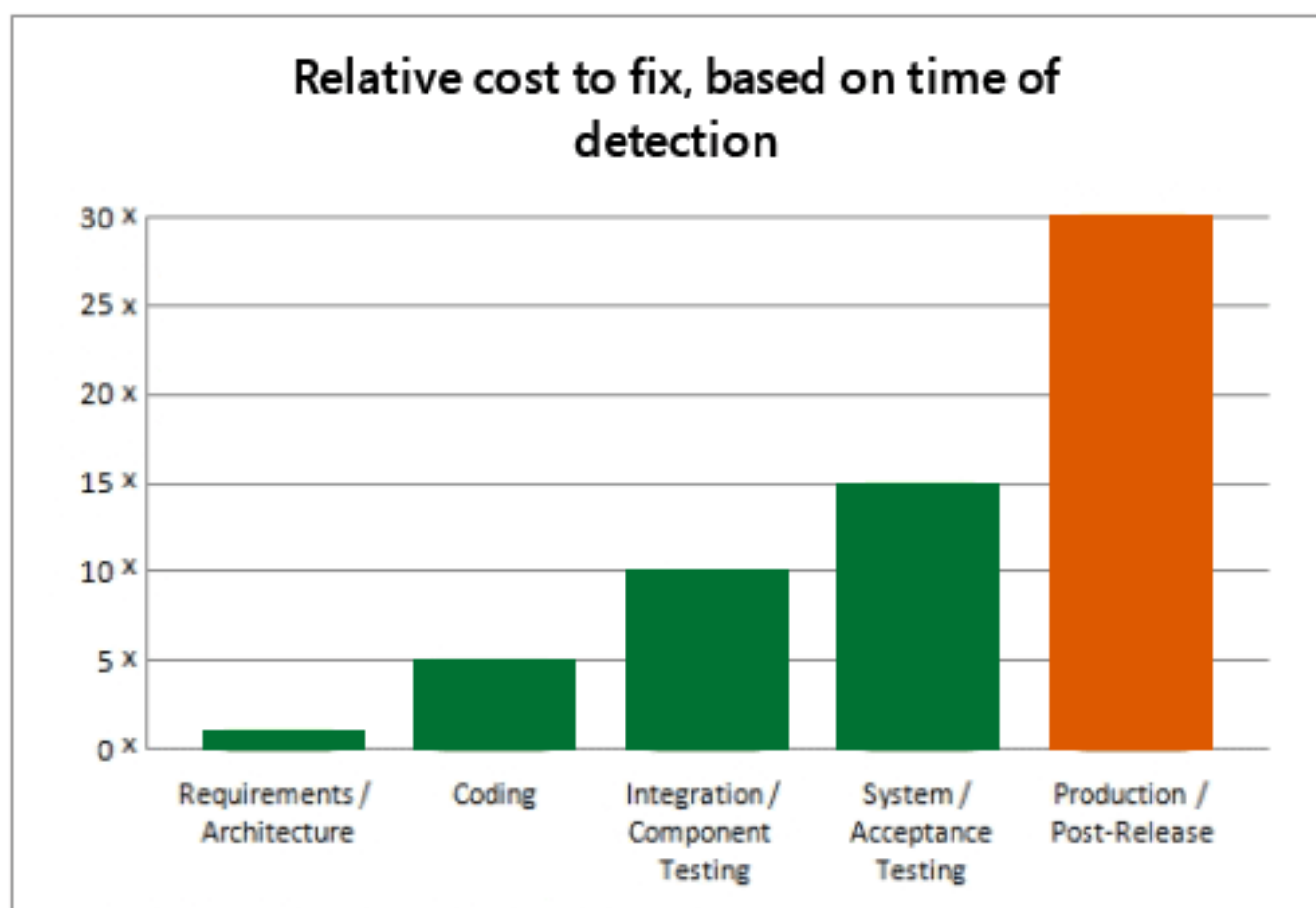
All these levels constitute the **waterfall method** of the software development lifecycle.

Problem with the Waterfall Model

As you may observe, that **testing in the model starts only after implementation is done**.

But if you are working in the large project, where the systems are complex, it's easy to miss out the key details in the requirements phase itself. In such cases, an entirely wrong product will be delivered to the client and you might have to start afresh with the project OR if you manage to note the requirements correctly but make serious mistakes in design and architecture of your software you will have to redesign the entire software to correct the error.

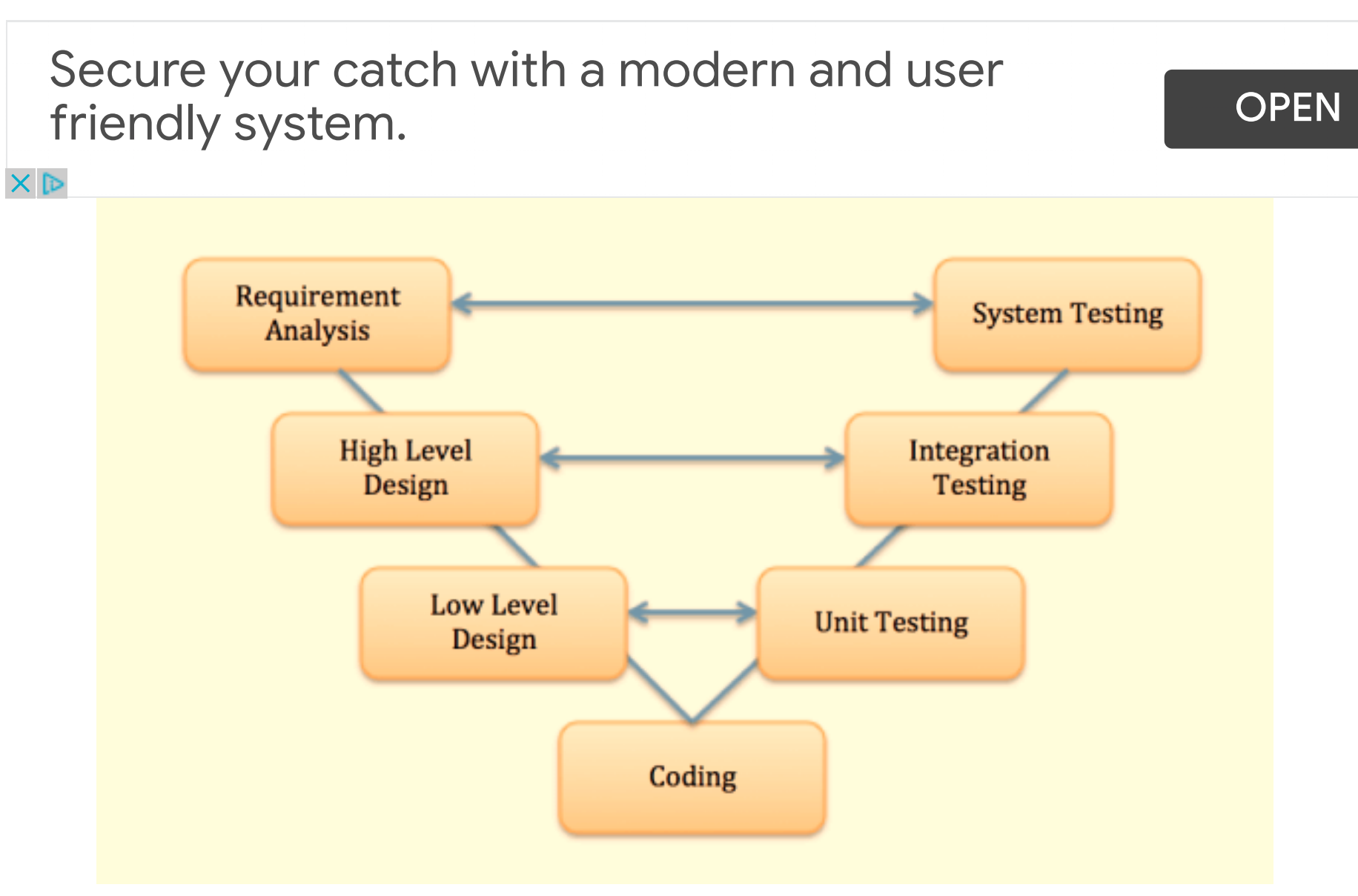
Assessments of thousands of projects have shown that **defects introduced during requirements & design make up close to half of the total number of defects**.



Also, the **costs of fixing a defect increase across the development lifecycle**. The **earlier in life cycle a defect is detected, the cheaper it is to fix it**. As they say, "A stitch in time saves nine."

Solution: The V Model

To address this concern, the **V model of testing** was developed where **for every phase, in the Development life cycle there is a corresponding Testing phase**



- The left side of the model is Software Development Life Cycle - **SDLC**
- The right side of the model is Software Test Life Cycle - **STLC**
- The entire figure looks like a V, hence the name **V - model**

Apart from the V model, there are iterative development models, where development is carried in phases, with each phase adding a functionality to the software. Each phase comprises its independent set of development and testing activities.

Good examples of Development lifecycles following iterative method are Rapid Application Development, Agile Development

Conclusion

There are numerous development life cycle models. **Development model selected for a project depends on the aims and goals of that project**.

- Testing is not a stand-alone activity, and it has to adapt the development model chosen for the project.
- In any model, testing should be performed at all levels i.e. right from requirements until maintenance.

[Prev](#) [Report a Bug](#) [Next](#)

YOU MIGHT LIKE:

AGILE TESTING

Best Agile and Scrum

Certifications in 2021

To cope up with rapidly changing IT market, and its growing demands and expectations,...

[Read more »](#)

SOFTWARE TESTING

Test Plan Template: Sample Document with Web

Application Example

What is test plan template?

TEST PLAN TEMPLATE is a detailed document that describes the test...

[Read more »](#)

SOFTWARE TESTING

Business

Intelligence (BI) Testing: Sample Test Cases

What is BI Testing?

BUSINESS INTELLIGENCE (BI) is the process of gathering, cleansing, analyzing,...

[Read more »](#)

AGILE TESTING

Agile Vs Waterfall: Know the Difference Between

Methodologies

What is Waterfall methodology? Waterfall Model methodology which is also known as Liner Sequential Life...

[Read more »](#)

SOFTWARE TESTING

What is Cloud Testing? SaaS Testing

Tutorial

Cloud Testing Cloud Testing is a type of software testing in which the software application is tested...

[Read more »](#)

COURSE

TEST

Management Tutorials: Complete Training Course

Course Summary Test Management is a series of planning, execution, monitoring and control...

[Read more »](#)

About
[About Us](#)
[Advertise with 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

SAP

Jmeter

Testing

Java

Informatica

Hacking

Python

JIRA