

# te testing experience

The Magazine for Professional Testers



## Improving the Test Process

# Testing process evolution

by Antonio Robres Turon

The role of testing within the software development life cycle started in the 1950s, but was very different from today. In the last twenty years the software testing discipline has evolved and progressed inside several software models. In every model, the role of the testing or QA team has changed to work together with the latest software development approaches and adapt to project requirements.

This article describes the testing process and testing activities in several development models showing the evolution of the testing discipline. The models analyzed are:

- Waterfall
- V Model
- Agile methodologies

## Waterfall model

The waterfall model was one of the earliest models that included testing. In this model the software development is divided into several phases and the tasks executed sequentially. The model defines several stages that the team must undertake in order to ensure the implementation of user requirements.

The stages in the Waterfall model are:

- **Requirements:** The first phase consists of requirements definition. The end-user's needs and wishes are converted into functionalities for the software to be developed. The team analyzes, determines and prioritizes the requirements for the next stage.
- **Design:** Using the requirements analyzed in the first stage, the project team translates these requirements into software solutions. In this step, the technology and architecture to be implemented is decided.
- **Implementation:** In this step, the specifications are divided into several modules. The programmers implement the source code of the modules using the technology and the architecture defined in the design stage.
- **Verification:** After the source code is finished, the testing team verifies the software. In this phase the team matches the software developed with the requirements defined and looks for bugs inside the code.
- **Maintenance:** In this phase the software is released and delivered to the client.

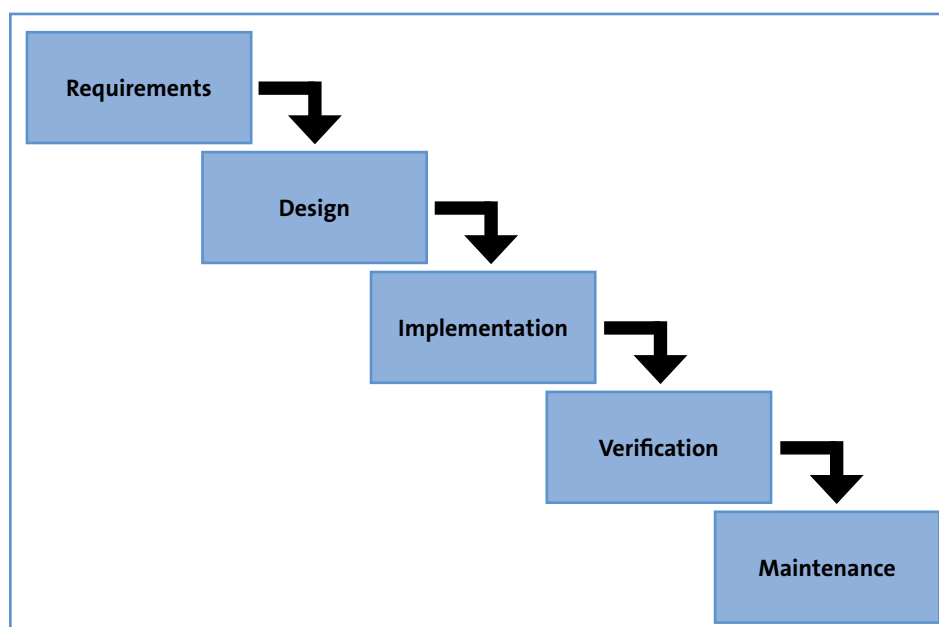


Figure 1: Waterfall model

In this model all testing takes place at the end of the software development process. During the implementation stage, the testing team can plan all the testing activities to be performed in the testing stage.

The principal testing activities in the Waterfall process are test design, test analysis, test specification and finally test execution. The main tests designed and executed are located at the system test level and focus on failures and mismatches between the software released and user requirements.

In the Waterfall process, contact between testing team and development team is not mandatory. Both teams work with the same requirements and specifications and communication between departments is not specifically required. The testing team does not take responsibility for the requirements and design stages; only for the verification stage.

One of the advantages of the Waterfall model for the testing team is the time allocated for every stage, including the verification stage. In this model the test team can execute all the tests with the same software version, and no changes in the source code are made during the execution. Another advantage is the time to prepare the testing stage. During the design and implementation stages the testing team can plan and design the test cases and can prepare the testing environment and the testing data.

The principal problem in this methodology is that errors are detected late in the project. When an error is found by the testing team, the problem can only be fixed by going back to the design of the system. The principal consequence of this is the high cost of defect fixing.

### V-model

The V model was developed in the 1980's to address some of the problems found using the Waterfall model.

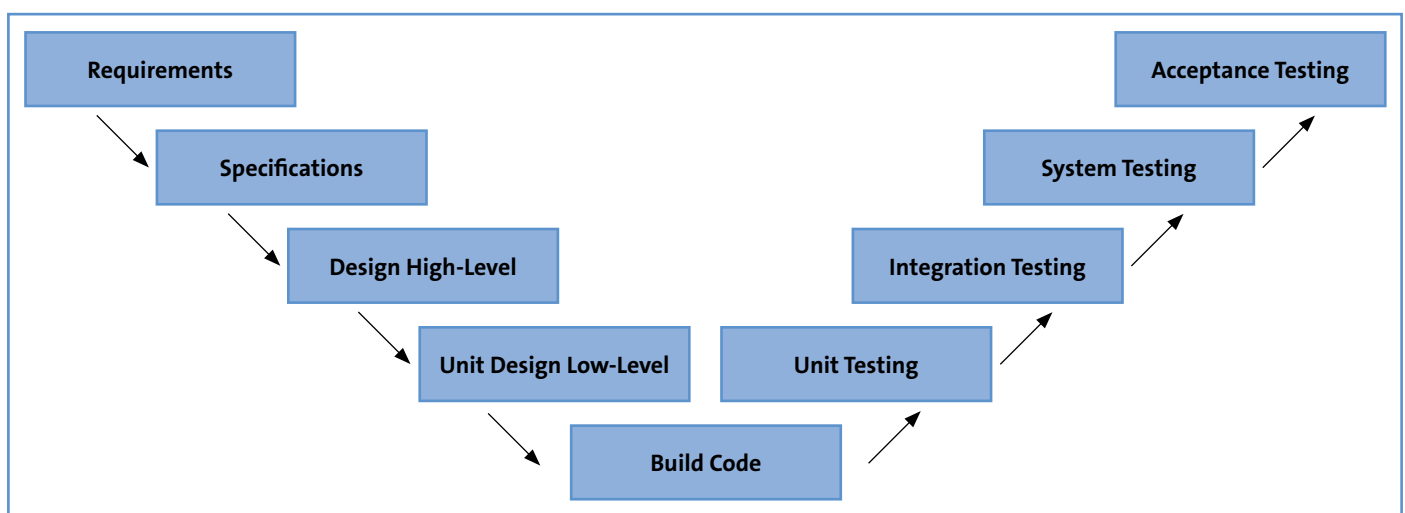


Figure 2: V-model

Other activities in the testing process performed according to the "V" model are the review of the test basis documents. For example, it is good practice to involve the testing team in the user requirements definition, software requirements or system design to find defects or problems in the early phases when they are cheaper and easier to fix.

As mentioned above, the main problem of the Waterfall model is the defect detection in the late phases of software development. The testing team is not involved in the project until the code is finished. The V-model provides a process to perform the testing activities in parallel with software development.

For every development stage there is a testing level to verify and validate the software development activities.

The testing levels in the V-model are:

- **Acceptance test:** The acceptance test is the last stage of software verification and validation and is performed when the software is released to the users. The goals of this level are to ensure that the software released matches with user requirements and to establish confidence in the system. Normally acceptance testing is performed by the user in a "production" environment.
- **System test:** In the system test stage, the testing activities focus on the behavior of the whole software product. This level uses the software specifications and business process to design and execute functional and non-functional tests. In this stage a controlled test environment is required.
- **Integration test:** In this level the interfaces between components and the interactions of the different modules are verified. Integration testing also tests the interaction between the software components and the external applications or hardware. The goal of integration testing is to verify the design and architecture implemented by the programming team.
- **Component test:** The goal of component testing is to find defects in an isolated module (class, objects or functions). Another goal is to verify the functionality separately from the rest of the components. Generally these tests are performed by the development team using "xUnit" frameworks or debugging tools.

With this model the testing team is involved in the software development process in the early stages reviewing the documents, and in the later stages performing system testing or acceptance testing.

The principal advantage of the V-model is the possibility of finding errors in all the software development stages and thereby de-

creasing the cost of the defect fixing. Another advantage is the inclusion of the testers in the software development activities, which makes it easier to test design and the testing levels.

One of the disadvantages of the V-model is the amount of documentation generated in the process. If an error is detected in the last stages of software development (for example in the system test), the team must change the source code to fix the error and all relevant documentation, including the design document, specification document and the test documents in all the testing levels. Another drawback of the V-model is the possibility of the re-testing in some components. Some functionalities or components can be verified in two or more different levels, duplicating the work and the documentation generated.

### Agile methodologies

One of the methodologies most adopted in the recent years has been “Agile” software development.

Agile software development is based in an iterative and incremental methodology where the requirements are developed and implemented through collaboration among all team members.

The software is developed in several iterations or Sprints which typically last 2-4 weeks and involve daily meetings to review the work of all team members and support the tasks or problems found by other team members.

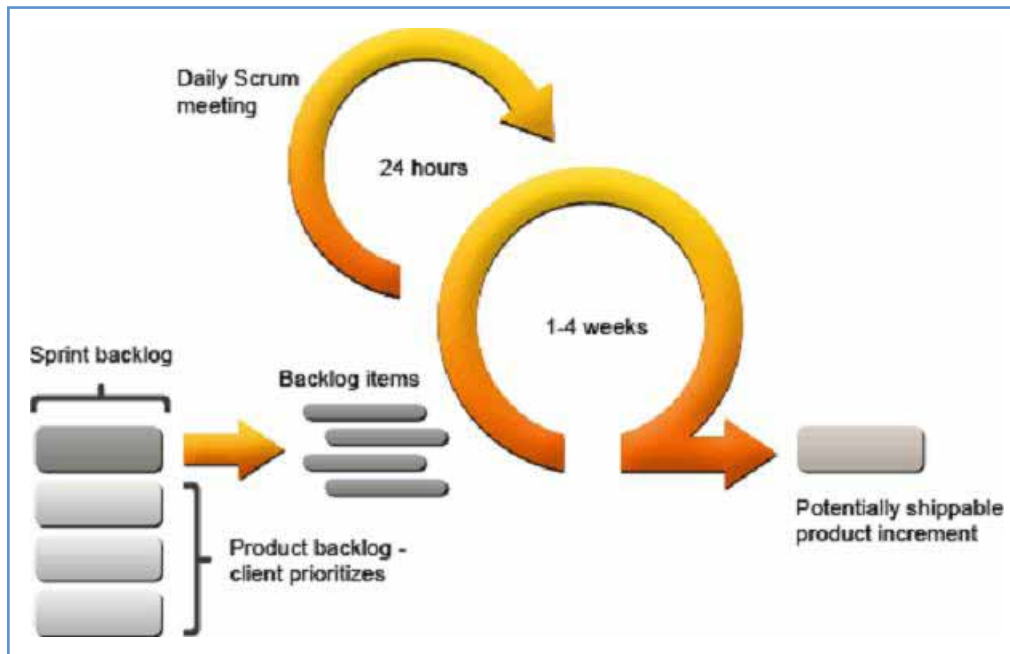


Figure 3: SCRUM methodology

At the beginning of each sprint, the team decides the functionalities to be developed and implemented in the next sprint, including testing tasks. This involves collaboration with all the people involved in the project, including business experts, stakeholders and the programming team. The requirements and specifications are defined by all team members to match with the user requirements.

The test team activities include all activities of the Waterfall-model (test planning, test design and test execution), but the activities are planned and developed in every sprint or iteration.

Manual testing requires a lot of time and effort for the testing

team. For this reason, testing must be automated to execute the tests in the next sprints to assure the quality of the functionalities developed in previous iterations. The automated testing can be performed in several levels, such as GUI test, acceptance test, or unit test, involving all the team (not only testers) in the automation activities.

Another important activity during the testing process using “Agile” methodologies is supporting the development process by creating examples or scenarios required to translate the user requirements into specifications for the programming team. To perform this activity, it is important to have collaboration between the stakeholders and the testing team to clarify the requirements, and also collaboration between the testing team and the programmers to help them design and implement the user requirements.

The principal advantage in the “Agile” methodologies is the flexibility to perform changes in the functionalities and the requirements of the software and adapt to user requirements. Another advantage is the possibility to obtain a shippable product in every sprint and show it to the stakeholders to obtain their feedback about it.

**Editorial remark: the waterfall and V-Modell have discussed advantages and disadvantages. The discussion on Agile only discusses advantages – there should be some coverage of disadvantages to be consistent.**

### Conclusions

Testing processes have changed in recent years by using different testing activities to adapt the test teams to software development models.

One of the trends in the software models is the increase of testing activities during the software development life cycle. Adding these activities, the complexity of the testing process is increased. For example, in the Waterfall model the testing team was only included in the verification stage of the project, whereas in the V-model or with Agile methodologies the testing team is involved in all stages of

the software development process.

Another trend is the collaboration and integration between the testing team and the rest of the team, including the programmers, stakeholders and business experts. In the Waterfall model the collaboration between testing team and the rest of the team members does not exist. In the Agile methodologies it is one of the most important characteristics.

In conclusion, it can be said that testing processes have evolved in software models, increasing testing activities and collaboration with other departments, such as programmers or stakeholders. This certainly helps in obtaining a final product of the required quality.

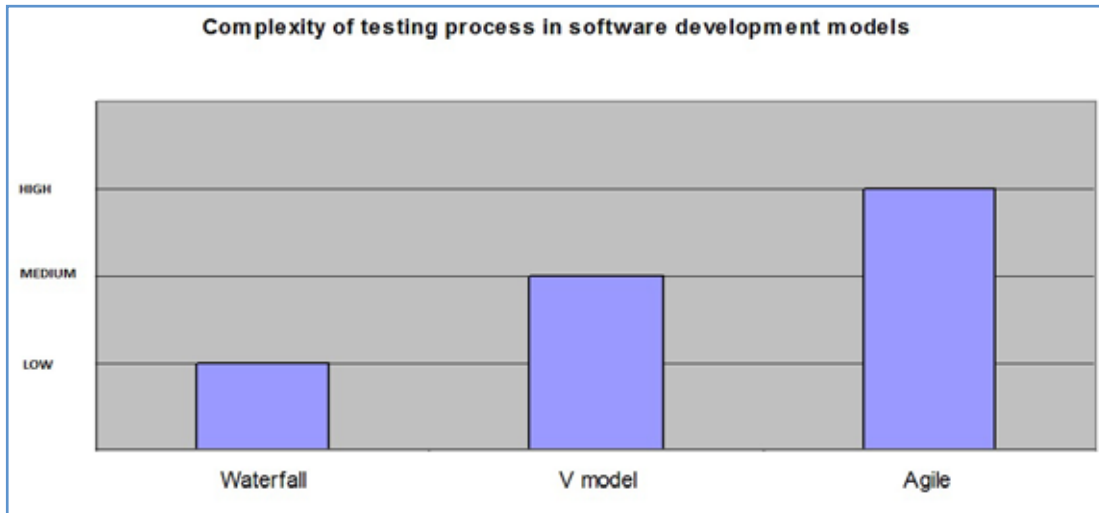


Figure 4: Complexity of testing process

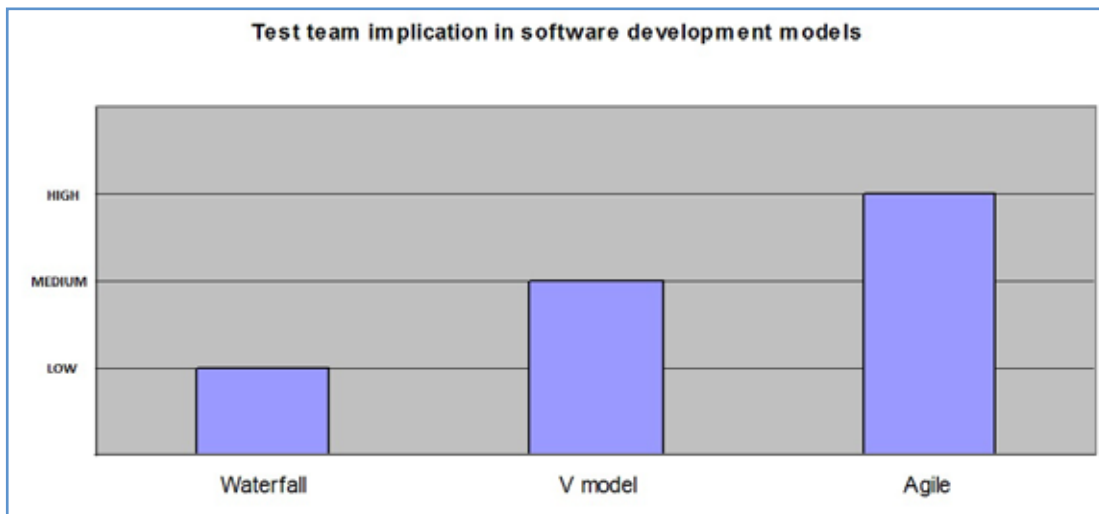


Figure 5: Test team involvement

### > biography



#### **Antonio Robres Turon**

is a Test Engineer at Diagnostic Grifols in Barcelona, Spain. He studied Telecommunications Science at the Polytechnical University of Catalonia in Spain. He holds a Master Degree in telecommunication administration and is an ISTQB® Certified Tester Foundation Level. He has been working for 5 years in the field of the software testing and quality engineering for companies including Telefonica, Gas Natural or Grifols. His work focuses on the design and execution of several testing projects, mainly for embedded systems and Web applications. He is also involved in the design and development of test automation projects with open source tools. He was a speaker at the last QA&TEST conference presenting in detail about the testing and QA structure of Diagnostic Grifols. He is a writer in the testing blog [www.softqatest.com](http://www.softqatest.com), and also a member of the TestQA association.