

# Building Software in an Agile Manner

## Abstract

The technology industry continues to evolve with new products and category innovations defining and then redefining this sector's shifting landscape. Over the last few years, there has been even greater volatility as a result of the advent of SaaS, cloud, and other related technologies and this has affected the makeup of hardware and software companies. Companies which not only aim to survive but want to prevail in customers' minds are reinventing traditional methodologies and turning to disruptive frameworks to stand apart from the competition. They are, in fact, redefining product development rules. These companies seek to provide their customers with greater performance, higher revenues and better features, all in rapid time.

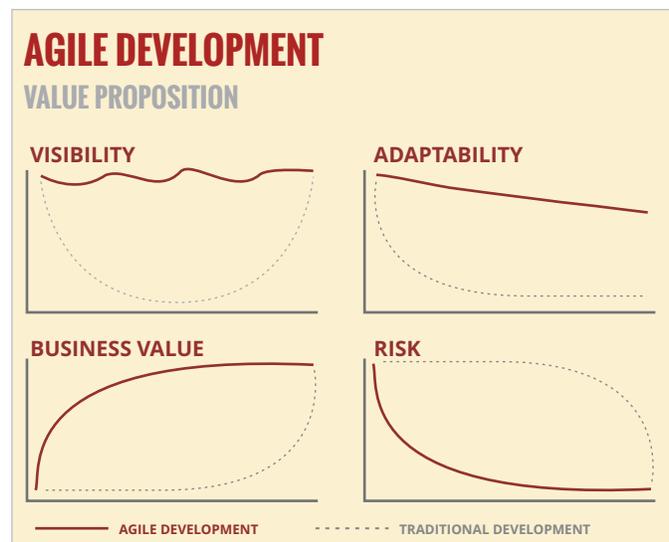
One method which ensures that software companies empower their customers to react to market needs is the Agile/Scrum Methodology. Trigent, for example, routinely uses Scrum project management framework with engineering best practices across several projects.

This paper describes Agile methodology, by detailing a case study, the best practices implemented and results discovered.

## Introduction

"Agile programming of everything from applications to basic infrastructure is essential to enable organizations to deliver the flexibility required to make the digital business work. Software-defined networking, storage, data centers and security are maturing. To deal with the rapidly changing demands of digital business and scale systems up - or down - rapidly, computing has to move away from static to dynamic models. Rules, models and code that can dynamically assemble and configure all of the elements needed from the network through the application are needed." **says Gartner in its report, "Top 10 Strategic Technology Trends for 2015"**.

The traditional development methodologies have proven challenging for software professionals since long. The problems faced and their real-life experiences have resulted in Agile methodology, which offers a lightweight framework. This method ensures that there is rapid value realization with significantly reduced risk.



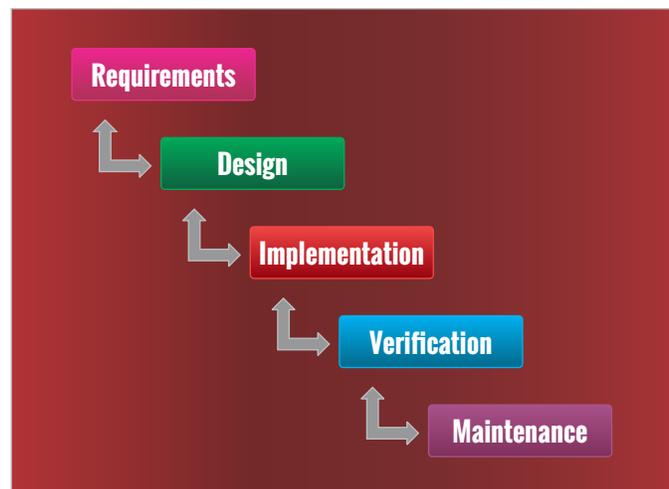
The Agile methodology starts off by accelerating the realization of initial business value and then, through a continuous process of planning and feedback it offers maximized value through the development cycle. The final product is no surprise for customers as its evolution is consistent with adaption to changing specifications and requirements of the customers. Thus, the final product is not a standalone software-offering but, on the other hand, a product aligned closely to customers' business requirements. Agile helps software companies to deliver to customers, better value, visibility and adaptability right from the onset of a project, thereby reducing risk considerably.

## Challenges faced with traditional methodologies

According to the popular CHAOS Report, nearly 25 percent of projects experience failure and software created never deployed. The significant reason for this failure is the traditional development methodologies adapted by product companies.

Traditional methodologies are based on certain presumptions that play a vital role in the software's evolution. The first one is that the end goal is thoroughly understood. Based on this understanding, the design is created and the software developed.

Unfortunately, this route is fraught with risks. Hard thinking at the initial stage leads to rigid design and a product which delivers on possibly outdated original goals. It is hierarchical and fixed, relying heavily on standardization. More often than not, customers participate only in the initial stages and then the development team continues the effort. With traditional



methodologies, there is an assumption that all the processes are predictable, can be measured and variations identified and controlled during the development life cycle.

### Some key disadvantages with these methodologies are:

- Products that do not evolve sequentially and fitting in iterations after completion can be expensive, time consuming and at times, may not work at all. Also, incorporating changes when the product is nearly mature can cause confusion to the development team, resulting in bugs and issues which in turn can lead to delayed time-lines and added costs.

- Customers are not expected to be explicit at the beginning of a project regarding what they want from the product. This is because a product is the result of an idea and only when the idea takes shape does its real identity emerge. Hard and fast rules do not hold good for product development and this is where Agile methodology works better than traditional ways.
- In traditional methodologies, hardware and software requirements are finalized at the outset. However, as the product evolves, the chosen tools and hardware may not be completely suitable, once again leading to cost escalation and slipped time-lines.
- Customers share their requirements, but this needs to be translated into IT language. There can be discrepancies if customer requirements are not clearly understood. All flaws come to the fore at the end and by then it may be too late. Incorporating changes can result in shifted deadlines, increased costs, and delayed releases – all of which can add to the overall cost of the project.

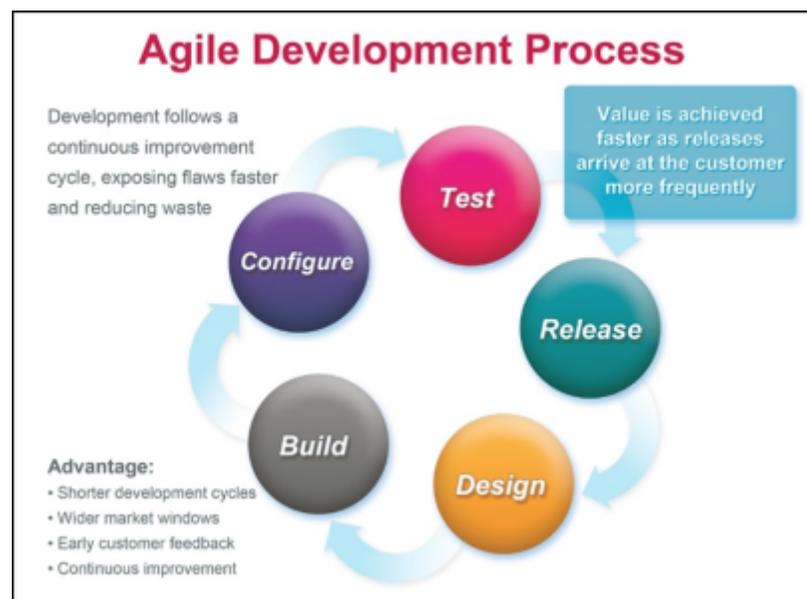
**Customers are not expected to worry about the development methodology. Their worries would be more on the lines of:**

- What if the development team has not understood my requirements clearly?
- What happens if the end product does not match specifications?
- Does the development team have a clear domain knowledge to really understand what I want my product to accomplish?

## Advantages of Agile/Scrum Methodology

Agile methodology is a conceptual framework with multiple methods for software development. One of the most popular methods is Scrum which minimizes risk and maximizes profits. As a lightweight framework, Scrum comes with broad applicability which empowers iterations and increments during the project's evolution. It also focuses on team and task management and this becomes crucial for projects which are constantly being viewed and reviewed by both project owners and end customers.

Given below is a case study which outlines the various steps in the



Scrum process and the advantages of the same.

One of Trigent's large, global customers, required a highly customized web-based content management system to be built on LAMP technology stack. During the discussion phase, Trigent suggested that this project be executed in its offshore facility to keep the development cost low. To overcome the challenges with the traditional waterfall development model with its drawback of sequential development lifecycle phases, difficulty in implementing changes in requirements, etc., Trigent offered to follow the Scrum development process.

### **Product Conceptualization**

Trigent worked with the customer to define the product idea. In parallel, Trigent's business analyst worked with the customer to understand the business goals and align the product features to achieve the same.

Trigent's domain experience combined with in-depth technology expertise helped the customer to look beyond expected functionalities to a flexible, ramp up/down end product with limitless possibilities.

### **Requirement Gathering**

The product requirements from the customer were captured as user stories and Trigent's analyst worked with the customer to develop the same. The complete set of product features formed the product backlog and the customer and Trigent's analyst became the Product Owners (PO).

When the PO wrote the user stories, Trigent made sure that the acceptance criteria for the features was defined and complete. This provided clarity to the development team on what was expected from them to meet the requirements.

### **Sprint Planning**

The product development phase was divided into four Sprints. Each Sprint spanned a period of three weeks with a defined set of features to be implemented in each Sprint, agreed by the Scrum team. Also, a designated Scrum master was allocated for the project.

Trigent identified critical features of the product and these features became a part of the earlier sprints. The initial design of the product was created to meet the critical product features. Each Sprint then enhanced the product design.

Sprint review meetings were conducted to find out what went wrong, lessons learned, changes required in the process.

## Prototyping

To validate the design to get an early feedback from the customer, a product prototype was implemented. The feedback received was reflected in the user stories and the design.

Since the team had better clarity on the features with the acceptance criteria, the effort estimation process improved a lot. The team was able to correctly estimate the time required to implement the user stories and stick to the time estimated. This reduced the pressure on the team and they were able to deliver what was committed for each Sprint.

## Development & Testing

The features to be implemented were maintained as Sprint backlog. The user stories were divided into smaller tasks. Tasks were assigned to the developers.

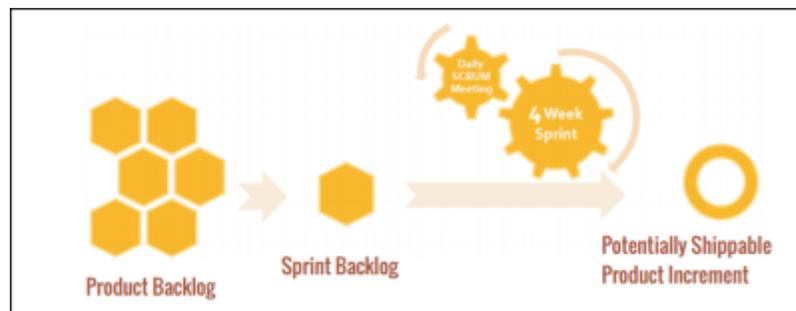
Trigent ensured that the product owners be part of the team. This provided better clarity and communication to the team. It also gave better visibility to the client on the progress. POs were also involved in the incremental review and their feedback incorporated, on-the-go.

A daily stand-up team meeting was conducted to address issues, track the progress of the team. Burn down chart was used to track the project progress. The scrum master maintained the burn-down chart and it was visibly displayed in the meeting rooms. The chart was a constant reminder to the team and if they were lagging behind, the schedule helped them to better manage their work.

Also, as part of the process, the POs were involved during the development phase to test the features against the acceptance criteria. This testing helped the team to fix issues during the development stage itself. Peer-to-peer testing was followed wherein each developer tested the feature implemented by his peer. This helped the team to get an overall working of the product.

## Delivery

The product was delivered to customers at the end of each iteration. This helped to get early feedback from the customer making it easy to implement changes or fix bugs.



## Summary

Agile/Scrum methodology serves very well in product development engineering. Working incremental product deliverables at the end of each sprint provides clarity to all the stakeholders on the progress. An emphasis on the purpose rather than documentation and process aids the team to get the product to market faster. It provides better communication between customer and development team enabling the customer to get what he wants rather than accepting what the development team delivers.

Trigent has a strong expertise of working with Scrum development models in delivering successful projects. Working in Scrum framework with Trigent helps customers to:

- ❑ Develop products and applications rapidly with minimal documentation
- ❑ Accommodate rapidly changing customers' needs and requirement specifications
- ❑ Involve customers' teams with Trigent consultants in development quite intensively
- ❑ Build a competent co-development team for building products and applications with high visibility, tight budgets, and least overheads.

### References:

[Gartner Identifies the Top 10 Strategic Technology Trends for 2015](#)  
[2000 CHAOS Report](#)

## About Trigent Software Inc.

Trigent is a privately held, professional IT services company and a Microsoft Gold Partner with its U.S. headquarters in the greater Boston area and its Indian headquarters in Bangalore. We provide consulting services in various technologies including Microsoft Solutions. Our operating model is to conduct sales, customer relationships and front-end consulting (e.g., business case, requirements, architecture) onsite with our clients and perform the detail design, development, integration, testing and quality assurance offshore at our world class development and support center in Bangalore. We are a SEI CMM Level 4 company and is ISO 9001:2000 TickIT certified organization.

For sales contact [sales@trigent.com](mailto:sales@trigent.com) or call 508-490-6000.



**Microsoft Partner**  
Gold Application Development  
Gold Collaboration and Content  
Silver Mobility