# SDLC – Functional and Technical Specifications

Zaid Fadel

Grand Canyon University



.

**Date: 26-Oct-2022**

This Paper prepared for SWE-530

Table of Contents

[SDLC – Functional and Technical Specifications 1](#_Toc117667520)

[Explanation of your plan 2](#_Toc117667521)

[Main Agile methodologies: 4](#_Toc117667522)

[1. Revision or addition of any specifications that you see should be completed at this time 4](#_Toc117667523)

[Description of how the planning phase of the SDLC has helped in the design of the specifications 4](#_Toc117667524)

[Disadvantages: 4](#_Toc117667525)

Agile methodology is a “step by step” dynamic focused on short-term visibility but never losing the long-term product goal.

There are 5 main Agile methodologies: Scrum, Kanban, Extreme Programming (XP), Lean Development e Crystal.

Explanation of your plan

Agile methodologies argue that, above all, **we should seek client satisfaction through the continuous delivery of value-adding software, by staying in constant communication with the client, and also by focusing on communication between team members.** Contrary to previous practices, Agile methodology is not characterised by the complete definition of a product, but rather “step by step” – a complete analysis or the definition of all categories/requirements, by dynamic interaction that allows constant delivery – Focused on “near-shore” visibility but never losing the long-term product goal.

**According to the Agile Manifesto and some of its 12 principles:**

👉Customer satisfaction is the top priority, demonstrated through continuous delivery and added value.

👉 Changes to requirements should be accepted – rather than pursuing “rigid” requirements, even if at a late stage in development: “Agile processes harness change for the customer’s competitive advantage”, as is stated in the Manifesto.

👉 The customer and the development team should work together on a daily basis, facilitating team and product synchronisation.

👉 It is vital to provide a pleasant environment and good support to development teams. Only in this way is it possible to keep them motivated.

👉 Agile processes promote sustainable development due to their constant rhythm and technical excellence, which in turn improves productivity.

👉 Retrospective moments inside a team are essential, allowing it to make the necessary adjustments and promote efficiency.

Basically, Agile development follows an incremental model, which develops collaboration within the team and continuous planning, as well as everlasting evolution and learning. Agile methodologies should respect the software development cycle – planning, execution and final delivery – therefore allowing software to be developed in stages; this makes it easier to identify and resolve bugs or new needs.

The main advantage of using Agile methodologies is not just the fast delivery of software, but also the constant delivery of “value” to the customer, since deliveries are incremental.

There are countless methodologies that follow this Agile mindset. In this blog post, we highlight the **five main Agile methodologies** and their advantages and disadvantages in the software development universe. But we cannot dive into the different methodologies without first referring to the growing popularity of the Agile methodologies (or some of them at least) with business management. This proves that is just not software development that can be enriched using these practices.

Business development is becoming an increasingly unpredictable playground, just like software development. So the challenge is now why not implement Agile methodologies in business process management? **Agile methods are adaptable**, allowing for rapid decision-making and instant influence on business development.

Main Agile methodologies:

1. Revision or addition of any specifications that you see should be completed at this time

Scrum is, undoubtedly, the most used of the many frameworks underpinning Agile methodology. **Scrum is characterised by cycles or stages of development, known as sprints**, and by the maximisation of development time for a software product towards a goal, the Product Goal. This Product Goal is a larger value objective, in which sprints bring the scrum team product a step closer.

It is usually used in the management of the development of software products but can be used successfully in a business-related context.

Every day starts with a small 15-minute meeting, the **daily Scrum**, which takes the role of synchronising activities and finding the best way to plan out the working day, allowing for a check on sprint “health” and product progress.

Description of how the planning phase of the SDLC has helped in the design of the specifications

* Team motivation is good because programmers want to meet the deadline for every sprint;
* Transparency allows the project to be followed by all the members in a team or even throughout the organisation;
* A simple “definition of done” is used for validating requirements
* Focus on quality is a constant with the scrum method, resulting in fewer mistakes;
* The dynamics of this method allow developers to reorganise priorities, ensuring that sprints that have not yet been completed get more attention;
* Good sprint planning is prioritised, so that the whole scrum team understands the “why, what and how” of allocated tasks.

Disadvantages:

* The segmentation of the project and the search for the agility of development can sometimes lead the team to lose track of the project as a whole, focusing on a single part;
* Every developer role may not be well defined, resulting in some confusion amongst team members.

