**Software Project Management Methodologies: Scrum vs. Waterfall**

# Introduction

The selection of suitable methodology for software projects determines the achievement of efficiency as well as quality and delivery timeliness. The widespread use of Scrum as an Agile methodology has coexisted with traditional Waterfall approaches in particular project situations. The report presents an overview of Agile thinking followed by an analysis of Scrum and Waterfall and it recommends evaluation factors to match projects with appropriate methodologies.

# Overview of Agile Mindset

Agile mindset places flexible operations alongside customer feedback loops alongside the importance of cooperation and progressive iterations. Agile approaches emphasize quantity over strict adherence to plans in order to implement needed changes in project requirements. Agile methodologies follow all principles mentioned in the Agile Manifesto (Manifesto, 2001) which prioritizes:

* Individuals and interactions over processes and tools.
* Working software over comprehensive documentation.
* Customer collaboration over contract negotiation.
* Responding to change over following a plan.

Teams under the Agile system develop continuous improvement practices while delivering software in smaller stages rather than as a complete whole and learning to adapt their work through ongoing feedback.



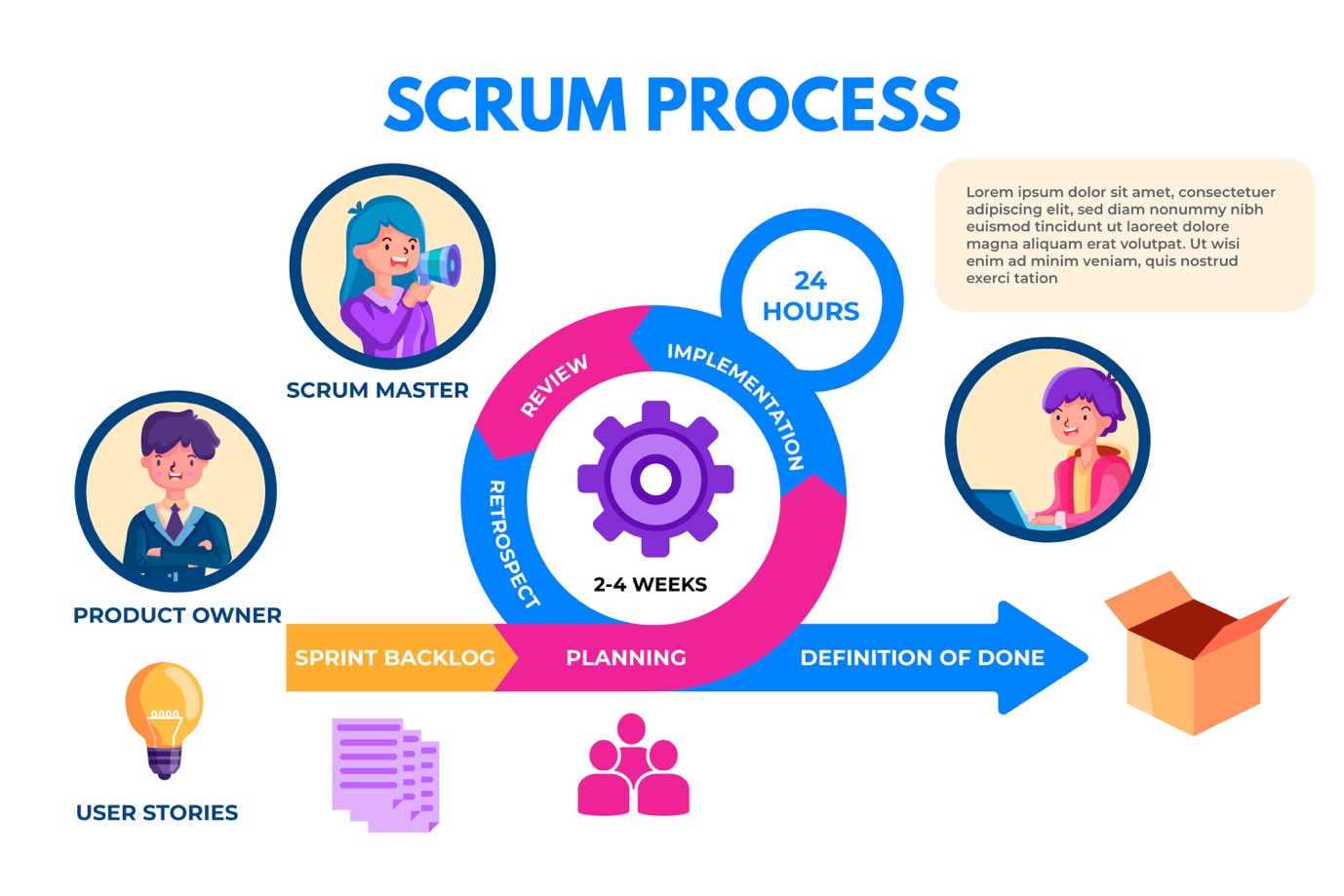
# Scrum Process Vs Waterfall

## Scrum Process

The agile proposal continues with the Scrum method which aims to define according to (Schwaber & Beedle, 2001) the project process that focuses primarily on individuals.

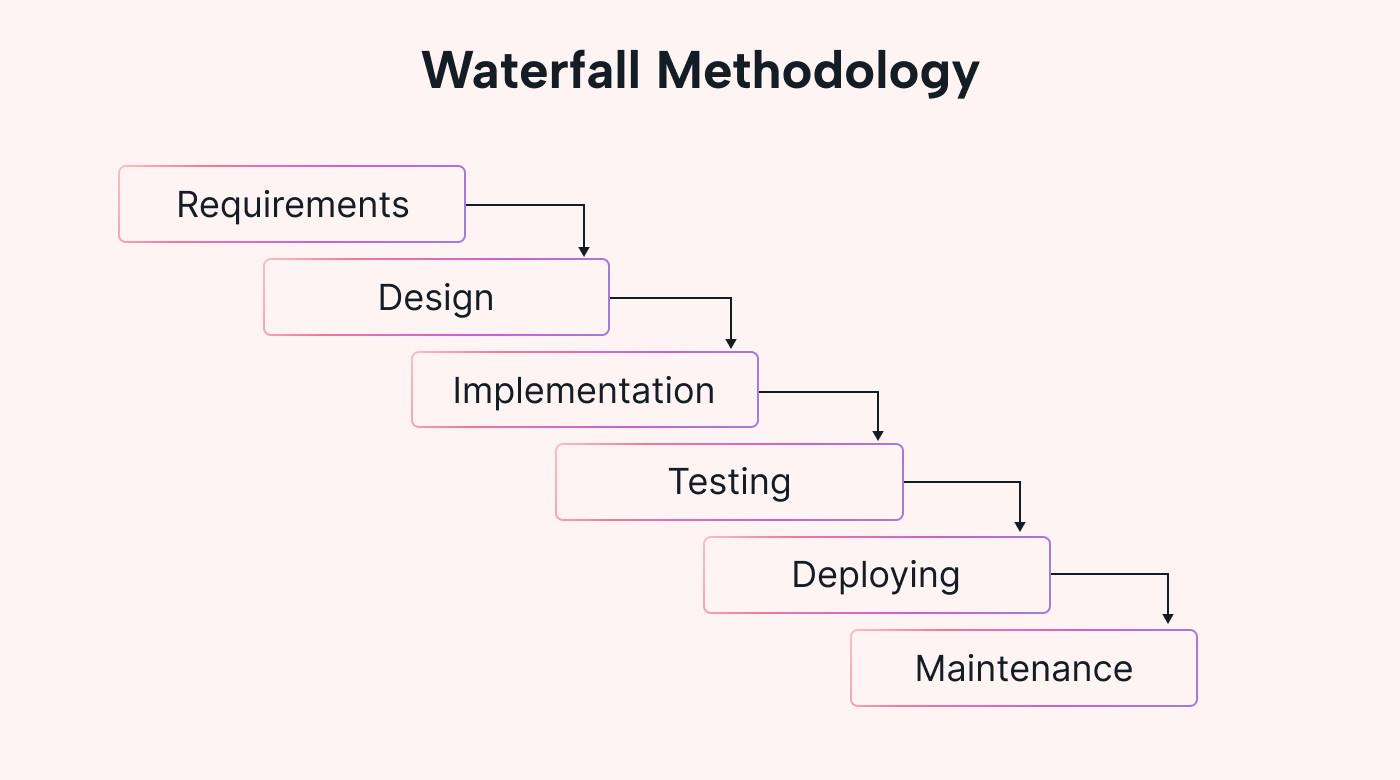
The principles of Scrum originated from Jeff Sutherland Ken Schwaber and Mike Beedle through six foundational characteristics (De Carvalho & Mello, 2011).

* Result flexibility
* Deadline flexibility
* Small teams
* Frequent reviews
* Cooperation
* Object orientation



## Waterfall

The System Development Life Cycle (SDLC) contains Waterfall method as one of its techniques which follows sequential workflow with no exception to complete the present stage before starting the next step. Each phase can receive maximum focus because there is nonexisting parallel work during the process (Adenowo & Adenowo, 2013).



# Scrum Vs Waterfall

## When to use Scrum

(Rodríguez et al., 2019) explains that Scrum contains events and artifacts which form the base structure of its methodology while integrating fundamental elements from the agile methodology theoretical foundation about communication and flexibility and continuous checking.

The development methodologies of larger entities ensure quality throughout the development project. The selection of development methodology depends on how well it will integrate into the system which programmers create. The restriction hampers the development team from selecting suitable development approaches that could further affect the overall project duration. The software development team stays focused on their schedule to deliver software on time while achieving all product requirements. The development of high-quality products is an extensive challenge to organizations (Beck et al., 2001).

## When to use Waterfall

This approach puts severe emphasis on meeting both requirements analysis and designing needs. Many believe that water-fall development as a rigid approach since project lifecycle design demands early specification of requirements and solutions together with specific project aims (Lei et al., 2017).

The Waterfall methodology is a way of handling project management by a team with perfect knowledge of the early requirements that have to be fulfilled and with a low need for requirement modifications (Lei et al., 2017). Quality achievements become possible through this model's phased implementation of projects that follow specific requirements for the creation of efficient results.

# Conclusion

Scrum and Waterfall serve different project needs. The Scrum method works best for feedback-based environments yet Waterfall provides the best solution for projects that have defined requirements and specific deliverables. Selecting between Scrum and Waterfall methodologies should depend on project difficulty as well as stakeholder needs and regulatory requirements.

# References:

Adenowo, A. A., & Adenowo, B. A. (2013). Software engineering methodologies: a review of the waterfall model and object-oriented approach. *International Journal of Scientific & Engineering Research*, *4*(7), 427-434.

Beck, K., Beedle, M., Van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., & Jeffries, R. (2001). Manifesto for agile software development.

De Carvalho, B. V., & Mello, C. H. P. (2011). Scrum agile product development method-literature review, analysis and classification. *Product: Management and Development*, *9*(1), 39-49.

Lei, H., Ganjeizadeh, F., Jayachandran, P. K., & Ozcan, P. (2017). A statistical analysis of the effects of Scrum and Kanban on software development projects. *Robotics and Computer-Integrated Manufacturing*, *43*, 59-67.

Manifesto, A. (2001). Manifesto for agile software development.

Rodríguez, P., Mäntylä, M., Oivo, M., Lwakatare, L. E., Seppänen, P., & Kuvaja, P. (2019). Advances in using agile and lean processes for software development. In *Advances in computers* (Vol. 113, pp. 135-224). Elsevier.

Schwaber, K., & Beedle, M. (2001). *Agile software development with Scrum*. Prentice Hall PTR.