Implementing Scrum in Company A

Completion Date: 21st November 2024

Table of Contents

<u>Summary</u>

"Company A" faces multiple issues with the current development process of Waterfall due to numerous projects witnessing a drastic increase in project failure recently therefore the organisation desires to implement agile methods for software development.

This report covers critical evaluation on the theoretical transition process from Waterfall to an alternative mixed methodology that involves both traditional and agile techniques, specifically on the agile framework of Scrum. Additionally, developmental and business challenges combined with solutions are discussed relating to the implementation of the methodology and how it would operate in a project, including relevant skills required for career development in software development using Agile.

CHAPTER 1 – Introduction

Cohen, Lindvall and Costa (2004) mention the main components required for a business to implement agile techniques successfully. One of these components included project size stated by a notable Computer Scientist who initiated the Agile Movement named Alistair Cockburn. He states that "coordinating interfaces become a dominant issue" which implies that face-to-face communication becomes more difficult and significantly problematic as the project size grows. However, a solution was generated and successful as large teams were decomposed into small teams. These small teams were further decomposed. The teams were united, understood the overall goals of the project and collaborated well. [Cohen, Lindvall & Costa, 2004]. This is an example of mixing a scaled agile framework with large projects but Scrum will be the main agile framework described, showing success with small to medium project sizes. Scrum is a reliable methodology that can be feasible within Company A's current development process, offering better efficiency and common project success.

1.1. Overview of the Organisation: COMPANY A

Company A is a French-based independent software development organisation that produces software for smaller businesses and clients. The organisation contains 2 sites in Paris and Lyon combined with a total employee count around 150 and a revenue under 50 million euros. It has been five years since the official creation of the organisation, maintaining Waterfall as the main software development methodology. By definition, this marks the organisation as a medium sized business shown in Figure 1 [Eurostat, 2022].
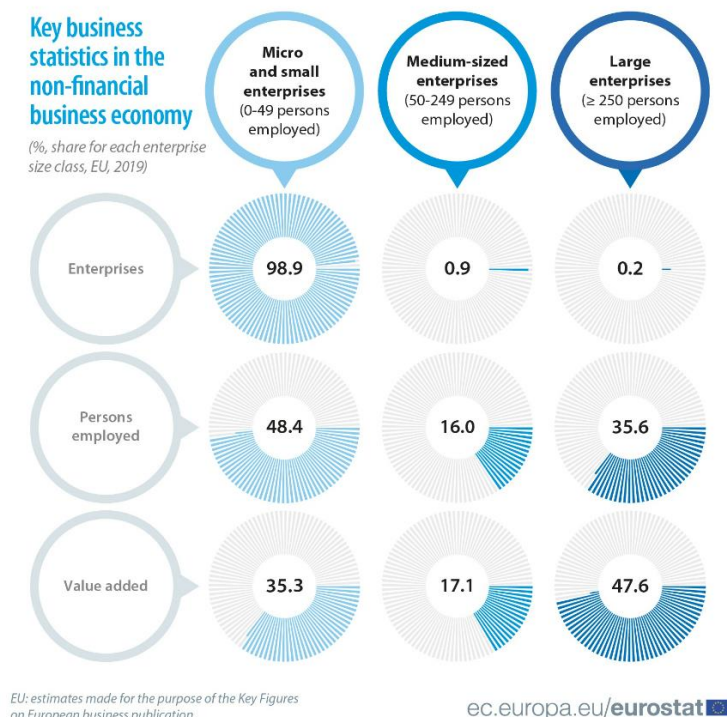

Figure 1 – shares for each business size classes

CHAPTER 2 – Discussion on Software Development Methodologies

Company A has consistently used Waterfall for the entirety of their projects since creation. The methodology offers an easy linear and sequential structure on the development phases within a project however it spawns a multitude of concerns such as overran projects leading to an increased risk of project failure. Figure 2 shows the cancellations, completions and the overruns of 214 projects during the Waterfall development stages. From the table, nearly a quarter of the projects were cancelled where 51 failed with 163 projects completed. Although completed, most of the stages were overran which indicated that the majority of the stages were completed behind schedule/late to complete (McManus & Wood-Harper, 2007). This is a common problem which Company A is facing as the rate of progression per development stage is slow and devastating while applying Waterfall as the main methodology. The methodology could be terminated completely and replaced entirely with Scrum however the quick transition is risky, leading to an exponential amount of severe business challenges therefore Scrum should be implemented with caution with the recommendation to combine it with existing Waterfall practices for reduced transitional challenges (Eason, 2016).

Waterfall method lifecycle stage	Number of projects cancelled[3]	Number of projects completed	Number of projects overrun (schedule and/or cost)
Feasibility	None	214	None
Requirements analysis	3	211	None
Design	28	183	32
Code	15	168	57
Testing	4	164	57
Implementation	1	163	69
Handover	None	163	69
Percentages	23.8%	76.2%	

Figure 2 – a table of 214 project cancellations, completions and overruns

2.1. The Agile Method: Scrum

Scrum is a progressive agile framework unique for its iterative and incremental quality, ensuring consistent and guaranteed progression of work. It includes using small self-organizing teams, committing short timed "sprints" - a 2 week-1 month development work where developers, designers and other relevant roles deliver essential features to the product while constantly communicating with stakeholders, reducing the chances of extended time during implementation and a guaranteed prototype of the product. Scrum handles the disadvantages that Waterfall casually possesses such as limited flexibility between stakeholders. It recognises commitment, courage, focus, openness and respect along with collaboration, prioritization and iteration with the incremental process to modify the product without overall hindering changes during every sprint (Sachdeva, 2016).

Large popular organisations have implemented Scrum and the variations into their work culture such as Spotify, showing successful growth in business (Kniberg, 2014).
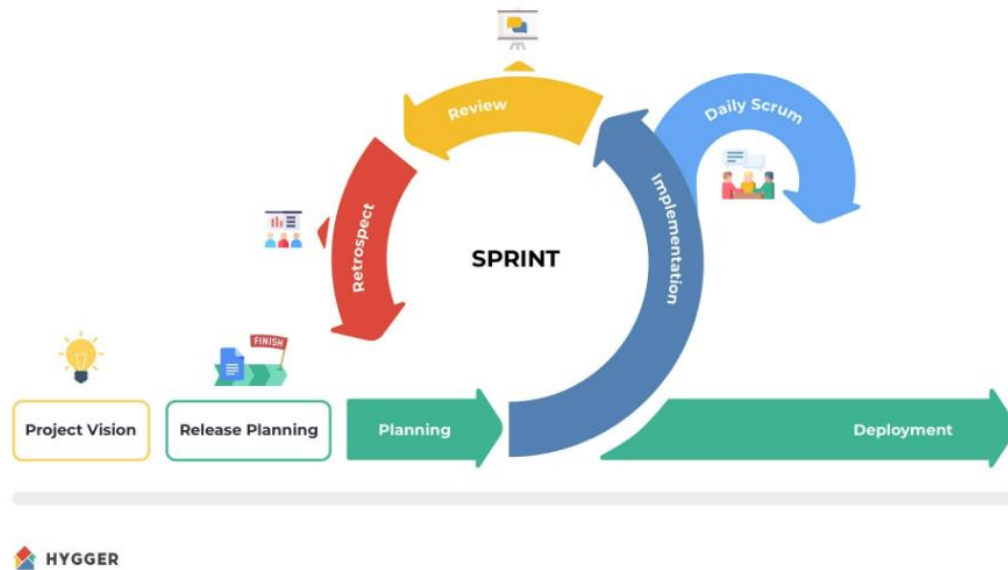

Figure 3 – the Scrum sprint model (Hygger, 2020)

For Scrum to operate successfully, there are three specific roles required:

Product Owner -> responsible for the project vision and dictates the validity of the product developed by the development team, making business decisions and managing the product backlog.

Scrum Master -> responsible for the team adhering to Scrum values and deals with organisational or collaboration issues during Scrum implementation.

Development Team -> self-organized team full of developers and designers responsible for the development of the product in the project.

Figure 3 provides a visual model of a Scrum sprint. Before a sprint begins, the Product Owner and the stakeholders meet and produce requirements with other project details such as scheduling, deadlines, budgeting, tools and constraints for the product. After this meeting, the Product Owner converts the requirements into tasks and a brief description called a "User Story" is aligned with the individual task. The user stories are organised and prioritised in a list known as the Product Backlog. The user stories at the top of the list are the most prioritised for the development team to concentrate on during the sprint. This is the Sprint Backlog. A project vision is clearly defined at the end of planning.

The Scrum Master plans the duration of the sprint (usually a month or 2 weeks) and sets the rules and regulations of the sprint. Each individual from the development team is provided a user story with a task required to be committed during the sprint selected from the Sprint Backlog. "Daily Scrum" begins as the sprint starts which is a short daily meeting held by the Sprint Master, usually 15-20 minutes for the team to communicate and reveal individual team member's progress and issues on their assigned task. At the end of the sprint, a Sprint Review meeting is held and a "deployable" product is shown to the Product Owner, who decides whether the product is compatible to the stakeholder's demands and reveals the prototype. A Retrospective meeting is held by the Scrum Master to reflect on the positives, negatives and potential improvements for the current and future sprints (Sachdeva, 2016).

2.2. Scrum Vs Waterfall in Industries

This table compares certain characteristics of the two methodologies within organisations (Palmquist et al, 2013):

Scrum	*Waterfall*
Iterative and incremental structure	Linear and sequential structure
Project scope constantly changes as it adapts to changing requirements	Fixed project scope as requirements is defined and immutable requirements
Quite flexible during development stages	Limited flexibility during development stages
Limited minimal documentation with the focus on working software	Detailed well defined completed documentation
Stakeholders are involved regularly during the entirety of the project	Stakeholders are only involved initially and their presence fade for the rest of the project
Small to medium self-organizing teams	Large teams in silos
Quick delivery of the product	Delivery of the product is time-consuming

Both methodologies offer different characteristics on how they function depending on the aspects of the project. It can show that one methodology does not present any superiority against the other rather they are both suitable and have the ability to spawn success for if used correctly. Scrum should merge at a steady pace by introducing a hybrid methodology

which can utilize the existing benefits of Waterfall and the new Agile framework. Businesses have implemented "hybrid" methodologies, combining the key values of Agile and Waterfall to obtain progress regardless of the project's conditions. Water-Scrum-Fall is a hybrid methodology that follows the linear and sequential structure of Waterfall however Scrum is applied completely during the implementation phase (West et al, 2011).

2.3. Water-SCRUM-Fall for Company A

The Water-Scrum-Fall methodology can be integrated into the work environment of Company A. Like many methodologies, Water-Scrum-Fall possesses several challenges as it is used as a development process, inherited from Waterfall and Scrum however effective solutions will mitigate or terminate those challenges. For example, Company A wishes to develop a database application using the hybrid methodology as followed in each stage:

Requirements Gathering & Design

- The Project Manager, the Product Owner and stakeholders meet and collect requirements for the database application such as data infrastructure, validation and memory management (Silberschatz et al, 2011)

- Additional details are documented such as budget and scheduling. A system design document is constructed with design frameworks and visual data diagrams.

Implementation

- A Product Backlog is produced by the Product Owner passed by the Project Manager and a Sprint Backlog is made for the development team to follow during the sprint.

- The Scrum Master determines the duration of the sprint with daily scrum. Individual members in the development team contributes to their task related to their assigned user story, developing a feature along with performing unit tests for the application.

- Once the sprint has been completed, a prototype of the database application should be "usable" but can incrementally improve and modified next sprint depending on the stakeholder's change in requirements. A Sprint review and retrospective is carried out and the next sprint occurs until the Product Owner is satisfied with the application, adhering to the requirements of the stakeholders where it can be approved by the Project Manager to begin the phases of testing and deployment.

Testing, Deployment & Maintenance

- Before the application is deployed to the stakeholder, it proceeds through multiple system test cases by quality assurance testers, database administrators and stakeholders. Any bugs and test case errors will be fixed with one sprint.

- Project Manager constructs a document to prepare the application for launch to the stakeholders officially where it is strictly orchestrated.

- There is moderation of feedback from the stakeholders on the application and improvements on features are made overtime behind current projects.

Certain challenges which could occur during development in this scenario:

- The development team may lack the expertise in specific sectors of producing databases, leading to time-consumption to train and difficulty in collaborating with other development teams who are more knowledgeable on the matter. Database administrators need to be included earlier to the implementation phase to assist (Highsmith, 2006).

- Due to Scrum's nature to increment change regularly, version control issues are prevalent with handling existing database schema, causing complex management. It is suggested to find specialized tools as version control for schema changes like Flyway (Ambler and Sadalage, 2006).

CHAPTER 3 – Transition & Implementation

3.1. The Transition Process

Transition will begin in the main site in Paris followed by the additional site in Lyon, starting with less prioritised projects. Roles such as a Scrum Master and a Product Owner will be added while maintaining the traditional Waterfall role of Project Manager along with other major changes that will affect work structure, culture, tools and skills. The biggest change would be more active collaboration between teams and stakeholders along with new additional tools. Company A will need to regulate and deal with multiple business and developmental challenges as agile principles are merged within each site (Eason, 2016).

1. The addition of the managerial scrum role of a Product Owner and a Project Manager are very similar in terms of responsibilities. It can cause a loose definition of responsibilities with no differentiation made, resulting in confusion quite quickly.

Boundaries and defined responsibilities must show when each role may actively be involved within each phase and the difference on how they operate (West et al, 2011).

2. Company employees and stakeholders would most likely resist against the characteristics of Scrum due to a significant shift in work culture, leading to slow project progression. The company needs to promote more transferrable skills useful for Scrum such as communication, interpersonal skills, adaptability, etc. It may host mock project workshops during inactive work hours using Scrum to teach employees these skills and hone them through experience (Eason, 2016).

3. Lack of specialized tools and techniques for Scrum causes difficulty to utilize the methodology during implementation. Tools and techniques which are compatible with both Scrum and Waterfall should be introduced. They should either be used separately or used at any phase. Examples include Jira and Kanban techniques for visually organising progress in project management (Wilhite, 2019).

4. Difference in the documentation process within Scrum and Waterfall poses a challenge for the quality of the product before, during and after releases. For example, in Waterfall the documentation must be completed with full definition on requirements and design before implementation however it does not fully represent the working product at the end due to frequent modifications of requirements throughout sprints. Documentation should contain the minimum description on project scope and allow updates during development (West et al, 2011).

Conclusion

In conclusion, the gradual transition from Waterfall to Water-Scrum-Fall is the onset of implementing Scrum with existing traditional practices with employees integrating with known processes yet with new agile methods of development. The limitations of the hybrid methodology must be understood as it becomes obsolete quickly due to the disadvantages from the two combined methodologies, especially Waterfall. Once Scrum is successfully implemented well with consistent project success, it can become the main software development methodology in the business where all aspects of Waterfall are removed.

5. Reference List

- Cohen, D., Lindvall, M. and Costa, P. (2004). An Introduction to Agile Methods. Advances in Computers, [online] 62, pp.1–66. doi:https://doi.org/10.1016/s0065-2458(03)62001-2. Accessed on: 1st October 2024

- Eurostat (2022). EU Small and Medium-Sized Enterprises: An Overview. [online] ec.europa.eu. Available at: https://ec.europa.eu/eurostat/web/products-eurostat-news/-/edn-20220627-1. Accessed on: 1st October 2024

- Eason, O.K. (2016). Information Systems Development Methodologies Transitions: An Analysis of Waterfall to Agile Methodology. Accessed on: 18th October 2024

- McManus, J. and Wood-Harper, T., 2007. Understanding the sources of information systems project failure. Accessed on: 19th October 2024

- Sachdeva, S. (2016). Scrum Methodology. International Journal Of Engineering And Computer Science, 5(6). doi:https://doi.org/10.18535/ijecs/v5i6.11. Accessed on: 19th October 2024

- Kniberg, H. (2014). Spotify engineering culture (part 1). [online] Spotify Engineering. Available at: https://engineering.atspotify.com/2014/03/spotify-engineering-culture-part-1/. Accessed on: 19th October 2024

- Palmquist, M.S., Lapham, M.A., Miller, S., Chick, T. and Ozkaya, I., (2013). Parallel worlds: Agile and waterfall differences and similarities. Software Engineering Institute, 1(1), pp.1-3. Accessed on: 20th October 2024

- West, D., Gilpin, M., Grant, T. and Anderson, A., (2011.) Water-scrum-fall is the reality of agile for most organizations today. Forrester Research, 26(2011), pp.1-17. Accessed on: 25th October 2024

- Silberschatz, A., Korth, H.F. and Sudarshan, S., (2011.) Database system concepts 6th Edition. Accessed on 20th November 2024

- Ambler, S.W. and Sadalage, P.J. (2006). Refactoring Databases. Pearson Education. Accessed on 20th November 2024

- Highsmith, J., 2009. Agile project management: creating innovative products. Pearson education. Accessed on 20th November 2024

- Wilhite, D. (2019). How to be Agile in a Waterfall Environment. [online] Scrum.org. Available at: https://www.scrum.org/forum/scrum-forum/6594/how-be-agile-waterfall-environment. Accessed 20th November 2024