Selecting an Agile Model

for CGHM’s new e-commerce website:

A Recommendation Report

Introduction

Software development requires the performance of certain activities in an organised manner to mould the customer’s need into a powerful software solution. (Leon, 2015).

Agile software development is proven to effectively meet the dynamic needs of the present-day software user (Mohapatra, 2010,). This is echoed by Cooke (2016) who states that the “common-sense approaches” of agile methods are capable of delivering “high business-value software solutions” while considering a company’s resources.

It is in this context that an agile software development model is considered the best approach for the development of an e-commerce website for CGHM. A number of methodologies fall under the agile umbrella and the purpose of this report is to evaluate and compare the Scrum and Extreme Programming (XP) methodologies to determine which approach will produce maximum positive results for CGHM.

Discussion

Online platforms have 24/7 requirements and face unpredictable challenges. Speed and flexibility, comprehending market interest, a quality product, and happy customers are key (Biz4Group, no date). With these in mind, Scrum and XP are evaluated below for suitability for CGHM’s needs.

Extreme Programming (XP)

XP is a strategy for honing software development (Cooke 2016, p. 17). It is based on core values including communication, simplicity, feedback, and courage, which when used together, result in effective products (Singh 2021).

Every contributor is part of the whole team. The customer is part of the team and works with the team day-to-day (Lindstrom and Jeffries, 2004).

The most simple solution is produced to meet requirements, the customer gives feedback and the requirements are updated. The software is improved continually.

Key stages:

In order to fully appreciate XP and its potential contribution to CGHM's project, an overview of the mandatory practices involved are detailed below:

Customer On-Site: There must be regular engagement between the customer and team, with the customer on-site for feedback (Singh 2021).

Customer tests - CGHM will be expected to define acceptance tests, these will be used to demonstrate the feature is working.

Test-Driven Development: Every time code is released, unit tests (programmer tests), are run and must work. Furthermore the end of each iteration results in fully developed and tested software. Beck (1999) states, "if there is a technique at the heart of XP, it is unit testing".

Refactoring: Refactoring means continuous design improvement. Striving towards optimum design means business value with each iteration.

Pair programming: Two of the development team work closely together at one computer. One Developer is the Driver and writes the code, the other takes the role of Observer and reviews every line of code as it is written. The Observer has the potential to quickly spot where code can be improved (Singh 2021).

Iterative Development: Goes hand in hand with small releases, this provides business value for the customer as iterations can be released (Lindstrom and Jeffries, 2004).

Simple Design: It is key to keep the design simple, this allows early delivery (Singh, 2021)

Planning Game: Requirements are compiled in the form of User Stories. Values are assigned to each story, indicative of the amount of time a story will take to implement (Singh 2021). There is a focus on steering rather than predicting (Lindstrom and Jeffries, 2004).

Coding Standards: Code appears as if one person wrote it.

Continuous Integration: The software is produced by the team in a set of rigorously tested releases that are fully integrated. The design is continuously improved and therefore consistently suitable for the customer's current needs. This results in the system running with reduced bugs (Lindstrom and Jeffries, 2004). Bugs ultimately weaken position in the market.

40-Hour Week: The team works at a pace that can be sustained and avoids burnout

Collective Code Ownership: Code can be examined by many experienced developers.

Metaphor: One single vision for the program (Lindstrom and Jeffries, 2004).

Pros

- Timely: there is a strong focus on prompt delivery of the product Rising and Janoff (2000)
- Cost-effective: the methodology lends itself to less documentation (Singh, 2021)
- Highly adaptive (Beck, 1999)
- Simple code
- High Visibility: as each iteration is given to the customer (Lindstrom and Jeffries, 2004), this lends itself to accountability and progress
- Practices such as Pair Programming means if a developer leaves, others on the team will have knowledge of the code (Beck, 1999).
- Retrospective review - after each iteration the team reflects on what went well and areas for improvement in the following iteration
- XP provides an opportunity for less experienced developers to learn from their colleagues (Shahzad, 2009)

Cons

- Underestimation can be a problem, particularly "if you commit to more than you can accomplish”
- The customer must be committed or this methodology will be problematic.
- Everyone on the development team must want to work together (Beck, 1999)
- Singh (2021) states a 10 min build may not be possible on a multi-platform project. This project falls under that category.
- Immediate productivity can be lower with pair programming than two programmers working individually
- Kenirons (2021) states “Non-functional requirements cannot be expressed by stories but need to be considered early” e.g. product stability is hard to capture in a story

Figure 1 below details a selection of XP's pros and cons.

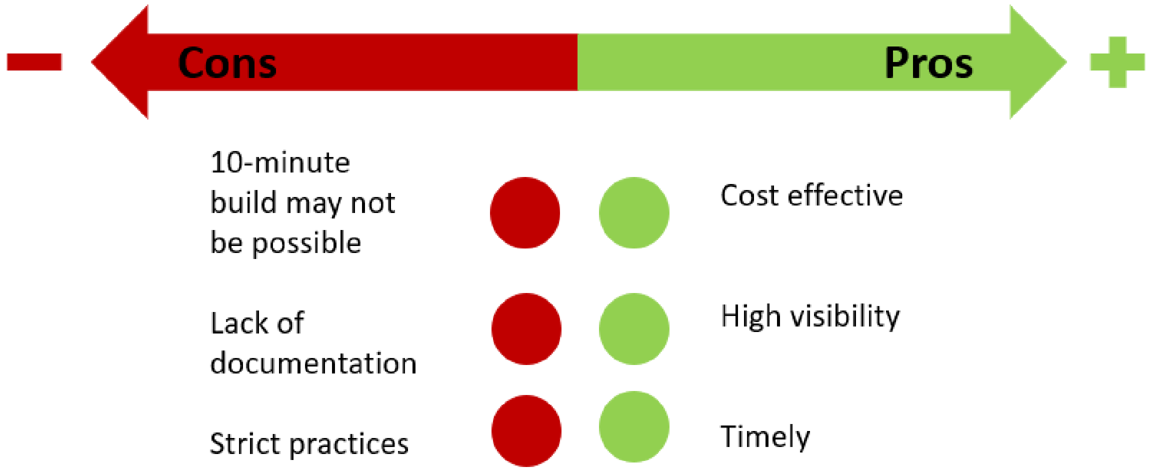


Figure 1 XP pros and cons

Scrum

Scrum is an iterative strategy. It equips teams with a framework to determine the necessary work to achieve the end goal and also to prioritise that work. The work is prioritised to ensure features of the greatest business-value are given utmost attention. There is constant interaction with the customer. (Cooke, 2016).

Key stages:

Lei et al (2015) describes the Scrum Team: the team comprises of the Product Owner, a Scrum Master, and Development Team.

The Product Owner will meet with CGHM to compile a list of requirements, the product backlog. The Scrum Master's role involves managing the product backlog and instructs the team on it, ensures the team fully understand the goals of the project. The Development Team implements and delivers the product at the end of each "Sprint" (Lei et al, 2015). An iteration is known as a sprint in this methodology and teams complete a sprint in 2-4 weeks.

The product backlog is given to the Sprint Team and the product owner explains the product backlog and goals of the project to the team, which is made up of the Scrum Master and developers and testers. This takes place at a Sprint Planning meeting where the team decides which of the items in the product backlog will go into the first sprint, these items become the sprint backlog. (Singh, 2021),(Lei et al, 2015).

A Sprint review meeting occurs at the conclusion of each sprint to discuss the sprint and may include a product demonstration to the product owner and/or the customer. A sprint retrospective is also scheduled to assess the sprint for plusses and minuses (Lei et al, 2015).

Pros

- Scrum considers what's required to beat competitors (Kenirons, 2021)
- Probability of success is high with scrum (Kenirons, 2021)
- Feedback is given
- The process is transparent

Cons

- Meetings can be lengthy
- Team cooperation is necessary
- Team experience is essential (Singh, 2021)

Figure 2 below details a selection of Scrum's pros and cons.

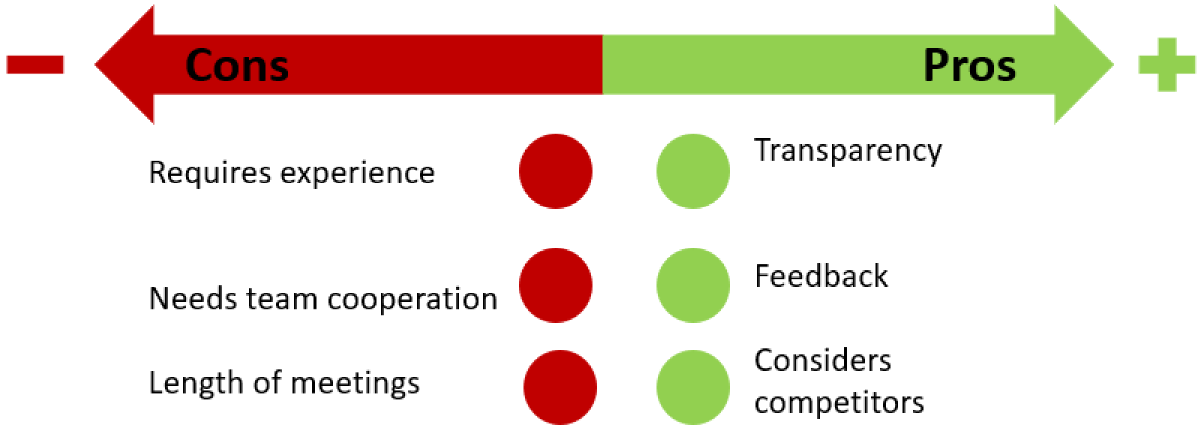
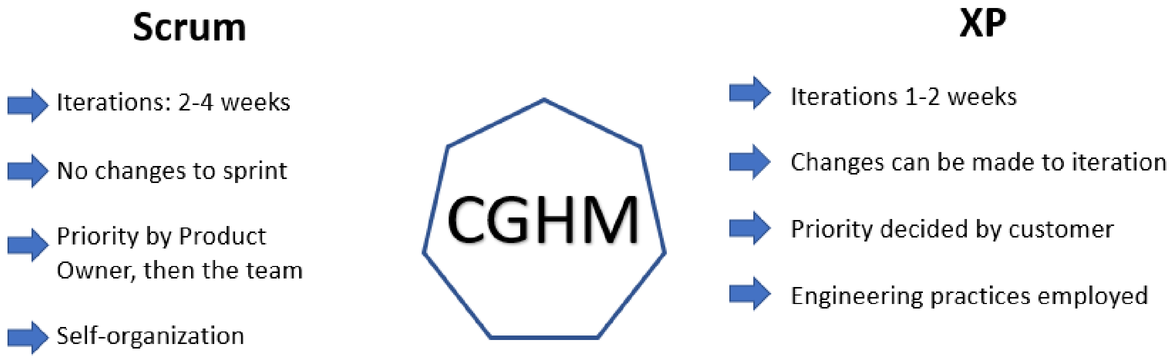


Figure 2 Scrum pros and cons

Key Differences

The key differences are summarised in figure 3 below



Scrum

- Iterations: 2-4 weeks
- No changes to sprint
- Priority by Product Owner, then the team
- Self-organization

CGHM

XP

- Iterations 1-2 weeks
- Changes can be made to iteration
- Priority decided by customer
- Engineering practices employed

Figure 3 Scrum vs XP

Scrum or XP for CGHM?

There are advantages and drawbacks to both.

Scrum employs an incremental and iterative approach which is highly advantageous in e-commerce as project complexities mean not everything is known at the start but scrum has the potential to evolve (Singh, 2021).

Teams are self-organised. This may mean the demands on CGHM’s time are less (Lei et al, 2015).

Mann and Maurer (2005) found that Scrum resulted in increased customer satisfaction, with customers citing reasons such as improved relationships with developers and feeling they were up-to-date with daily meetings.

For Scrum an experienced team is essential, the high level of expertise available in the development team fits well with this methodology Morandini et al (2021).

Altvater (2017). Using a Scrum approach can often see a higher return on investment due to a number of factors including a decrease in the time to get to market, due to regular feedback and review problems are identified and remedied promptly which leads to less associated costs. The Scrum methodology has a "proven productivity rate".

Lindstrom and Jeffries, (2004) advise to determine if XP is for your project, consider the characteristics of the project, the people and the company involved.

With XP the team sit together in an open workspace. If the team is too big this may not be possible. Lack of documentation may also present challenges for the company(Holcombe, 2008). Paulk (2001) states “XP’s lack of design, documentation and de-emphasis on architecture is risky”.

Visual-Paradigm say "There are no reasons why a team should choose between Scrum and XP." Scrum represents the framework for an agile development process and XP is a practice that can be used within the Scrum framework. (Visual-paradigm.com, no date).

Holcombe (2008) states "there have been successful approaches that combine Scrum with other agile methodologies", Scrum could be used for the general management of a project and another agile method employed for production

Conclusion and Recommendations

Scrum and XP are two agile methodologies which will provide the stepping stones for the production of an excellent e-commerce website for CGHM. They share common practices core to the proven success of agile methodologies. They both have many benefits and also a minor number of potential drawbacks. Taking this into account a framework of Scrum with elements of XP incorporated is our recommendation.

References

Altvater, A. (2017) *What is Scrum? How Does Rugby Help Software Developers Create Quality Products?* Available at: https://stackify.com/what-is-scrum/ (Accessed: 12 November 2021).

Beck, K. (1999) 'Embracing Change with Extreme Programming', *Computer*, 32(10), pp. 70-77. doi:10.1109/2.796139.

Biz4Group (no date) *6 Reasons Why Agile Methodology is Perfect for Developing eCommerce Projects.* Available at: https://www.biz4group.com/blog/6-reasons-why-agile-methodology-is-perfect-for-developing-ecommerce-projects (Accessed: 9 November 2021).

Cooke, J. L. (2016) *Agile: An Executive Guide : Real results from IT budgets*. Ely, Cambridgeshire, United Kingdom: IT Governance Publishing.

Holcombe, M. (2008) *Running an Agile Software Development Project.* Hoboken, N.J. :Wiley.

Kenirons, M. (2021) 'Extreme Programming' [Lecture]. *51863: Agile Software Development*. GMIT. 1 November.

Kenirons, M. (2021) 'Scrum' [Lecture]. *51863: Agile Software Development*. GMIT. 1 November.

Lei H. et al (2015) 'A statistical analysis of the effects of Scrum and Kanban on software development projects', *Robotics and Computer-Integrated Manufacturing*, 43, pp. 59-67. doi:10.1016.

Leon, A. (2015) *Software Configuration Management Handbook.* Boston: Artech House.

Lindstrom, L. and Jeffries, R. (2004) 'Extreme Programming And Agile Software Development Methodologies', *Information Systems Management*, 21(3), pp. 41-52. doi:10.1201/1078/44432.21.3.20040601/82476.7.

Mann, C. and Maurer, F. (2005) 'A Case Study on the Impact of Scrum on Overtime and Customer Satisfaction', *Agile Development Conference*. Denver, USA, 2005. IEEE, pp. 70-79. doi:10.1109/ADC.2005.1.

Mohapatra, P.K. (2010) *Software Engineering : A Lifecycle Approach.* New Delhi: New Age International.

Morandini, M. et al (2021) 'Considerations about the efficiency and sufficiency of the utilization of the Scrum methodology: A survey for analyzing results for development teams', *Computer Science Review*, 39. doi:10.1016/j.cosrev.2020.100314.

Paulk, M. (2001) 'Extreme Programming from a CMM Perspective', *IEEE Software*, 18(6), pp. 19-26. doi:10.1109/52.965798.

Rising, L. and Janoff, N.S. (2000) 'The Scrum Software Development Process for Small Teams', *IEEE Software*, 17(4), pp. 26-32. doi: 10.1109/52.854065.

Shahzad, S. (2009) 'Learning From Experience: The Analysis of an Extreme Programming Process' *Sixth International Conference on Information Technology: New Generations.* Graz University of Technology, Graz, Austria, 2009. IEEE, pp. 1405-1410.

Singh, V. (2021) *Agile Methodology.* Available at: https://www.toolsqa.com/agile/agile-methodology/ (Accessed: 10 November 2021).

Visual Paradigm (no date) *Extreme Programming (XP) vs Scrum.* Available at: https://www.visual-paradigm.com/scrum/extreme-programming-vs-scrum/ (Accessed: 10 November 2021).