**Group Members**

Simran Kaur

Harpreet Singh

Rajwant kaur Boparai

Contents

[PROG6001 Managing Software Development Projects – Assessment 2 1](#_Toc151966198)

[Task 1: 1](#_Toc151966199)

[Task 2: 3](#_Toc151966200)

[Task 3 4](#_Toc151966201)

[Overview of the Agile Mindset 4](#_Toc151966202)

[Description of the Scrum Process and the Waterfall Methodology 5](#_Toc151966203)

[Guidelines for Determining Whether Scrum or Waterfall Suits Specific Software Development Projects 5](#_Toc151966204)

[Differences Between Agile (Scrum) and Waterfall Methodologies 5](#_Toc151966205)

[Conclusion 5](#_Toc151966206)

[References 6](#_Toc151966207)

# PROG6001 Managing Software Development Projects – Assessment 2

# Task 1:

**Collaborative Project Using Git**

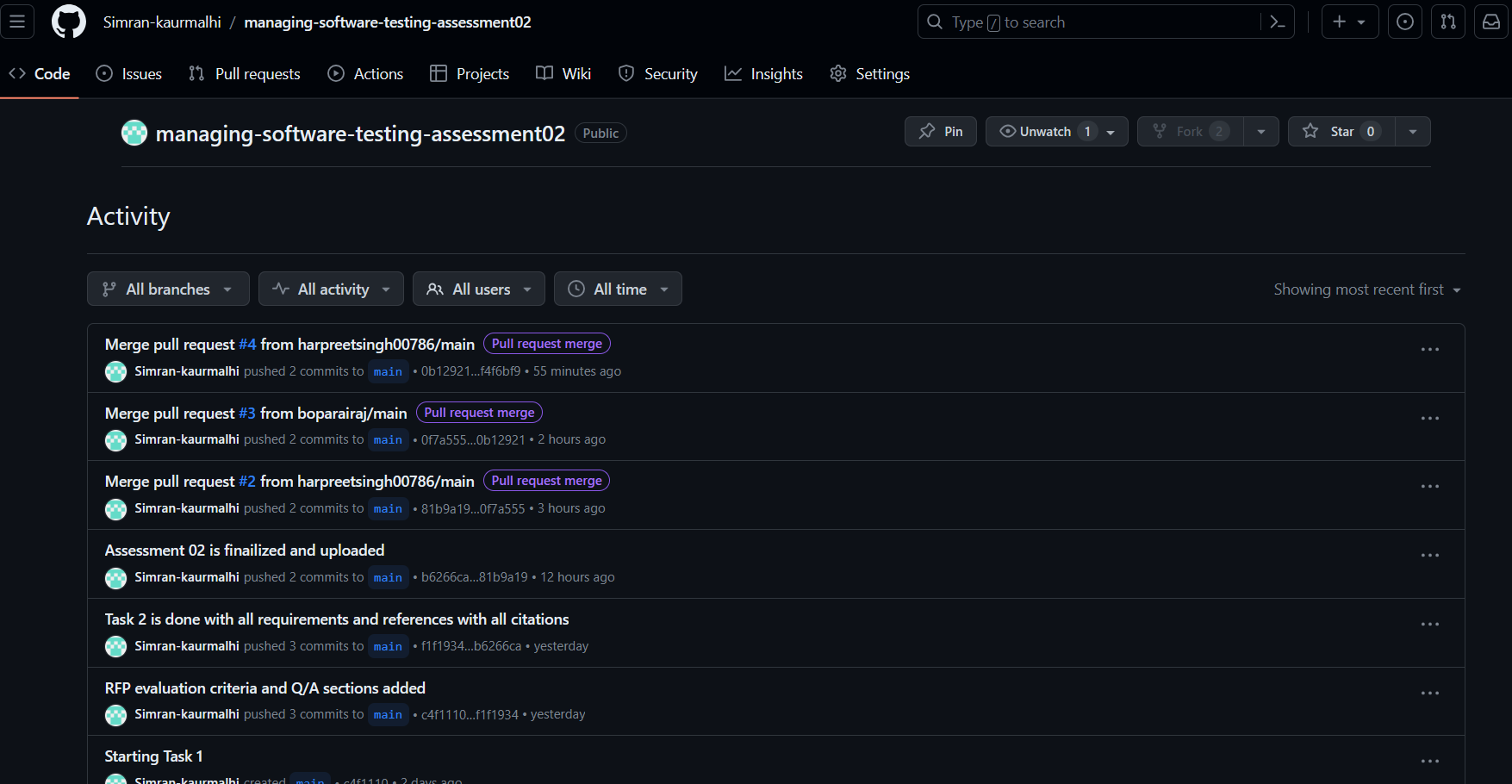
* **Simran Kaur**

**Git Repository Creation:**

**A screenshot of a computer

Description automatically generated**

**Commit History with Messages**

****

**Pull Request Acceptance and merged:**

**A screenshot of a computer

Description automatically generated**

* **Harpreet Singh:**

**Git Repository Fork**

**A screenshot of a computer

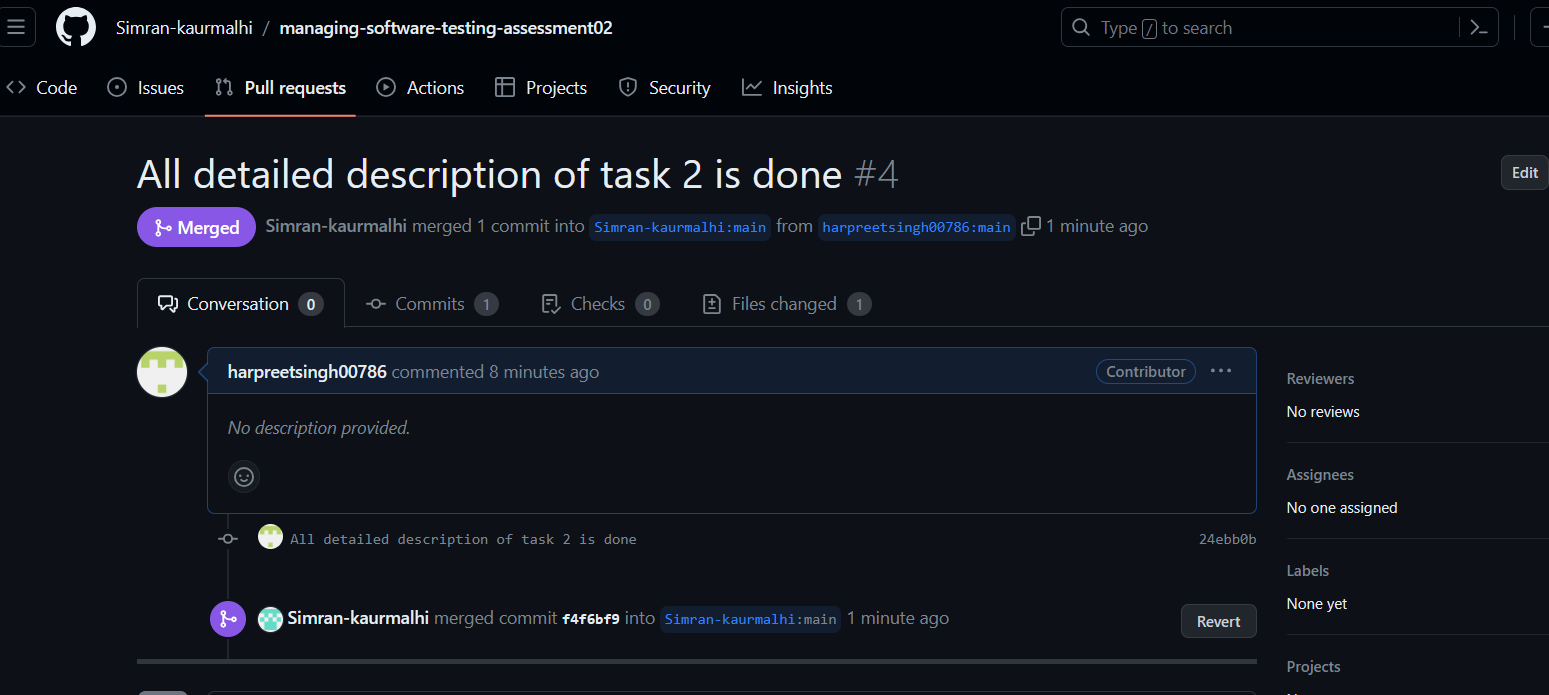
Description automatically generated**

**Commit History with Messages**

**A screenshot of a computer program

Description automatically generated**

**Pull Request**

****

* **Rajwant kaur Boparai**

**Git Repository Fork**

**A screenshot of a computer

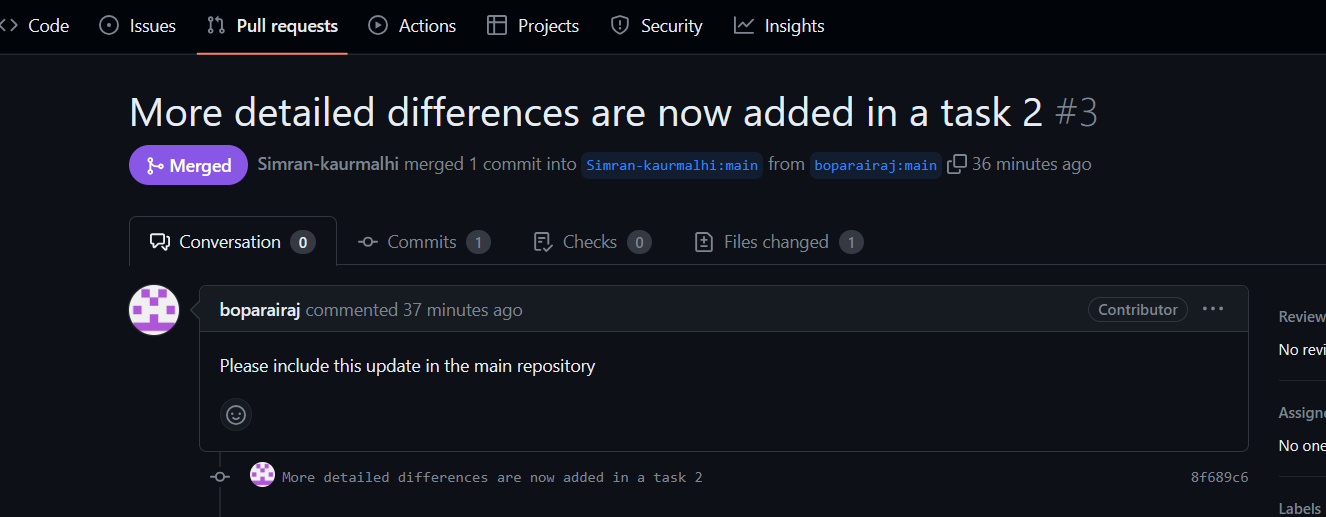
Description automatically generated**

**Commit History with Messages**

**A screenshot of a computer

Description automatically generated**

**Pull Request**

****

# Task 2:

**Aussie Business Buzz (ABB) Request for Proposal (RFP) for Integrated System Development**

Issued by Aussie Business Buzz (ABB)

Date of Issue: 20 November 2023

Proposal Submission Deadline: 30 November 2023

Aussie Business Buzz is a dynamic enterprise in the technology product sector that is excited to invite proposals for the development and implementation of an integrated system. Our vision encompasses the support of our current four branch shops with the flexibility to accommodate future expansion. The imagined system should be a confluence of a Customer Relations Database and must a Digital Marketing System along with a Stock Management System and Management Reporting tools.

**System Description:** The main objective of this RFP is the development of a system that integrates several key business functions. Predominantly, a robust Customer Relations Database is sought to manage a list of customer details including their purchase history and records of devices entrusted to us for repairs. Additionally, the system should incorporate a dynamic Digital Marketing System. This system should make use of the customer relations database and allow integration of prospective customers' details from our existing website. It is considered equally valuable for the implementation of a Stock Management System which can handle our diverse inventory including products for sale and parts for repairs. This system should facilitate automated ordering from wholesalers and allow inter-branch stock checks and transfers. Furthermore, the development of comprehensive Management Reporting tools is necessary. These tools should provide our management team, who may be located at any of our branches, with real-time updates on all aspects of our operations.

**Proposal Evaluation Criteria:** Proposals submitted will be evaluated on a scale of criteria. The ability of the proposal to comprehensively meet our system requirements will be vital. We are particularly interested in solutions that offer scalability and flexibility to adapt to our growing business needs. It must be evaluated based on cost-effectiveness and without compromising on the quality and efficiency of the solution, will be a key consideration. We also place high value on innovative solutions that bring creativity and modern techniques into production. The experience and reputation of the vendor, evidenced by a track record of successfully handling similar projects, will be considered.

**Process for Answering Questions:** To facilitate clear communication we will organize a dedicated question-and-answer session following the initial review of the proposals. Vendors are encouraged to submit their queries via email by a specified deadline. All inquiries and responses will be compiled and shared with all potential vendors to ensure a transparent and equitable information-sharing process.

**Additional Considerations:** In this RFP, we seek proposals that not only align with our current operational needs but also demonstrate an understanding of the potential integration with our existing website and systems. We encourage vendors to propose innovative solutions, which can include bespoke software development, leveraging existing applications, SaaS solutions, or a combination thereof. We have intentionally avoided overly technical specifications in this RFP to leave room for creative and potentially groundbreaking solutions. Vendors should also include detailed plans for support and maintenance post-implementation of the system.

**Submission Guidelines:** Interested parties are requested to submit their proposals by 30 Nov 2023, to [ceo@aussie.business.com](mailto:ceo@aussie.business.com). It is important to note that submissions received after the deadline will not be considered.

This RFP represents an exciting opportunity for innovative solution providers to collaborate with Aussie Business Buzz in creating a system that not only meets our current needs but also paves the way for future growth and success.

**Issued by:**

CEO,

ceo@aussie.business.com  
Aussie Business Buzz Management

# Task 3

## Overview of the Agile Mindset

The Agile in software development is an approach that places a strong emphasis on flexibility as well as teamwork and the ability to adapt quickly to changes. This method is different from traditional software development methods, which usually have a more rigid structure and process.

In Agile development the focus is on creating software in small as well as manageable sections which is well-known as iterations. Each iteration involves a cycle of planning including development as well as testing and review. This iterative process allows for continuous feedback and adjustments. As a result the software being developed can better meet the actual needs of its users as those needs may change or become clearer over the course of the project.

Agile relies heavily on collaboration with all team members. Teams are typically self-organizing to decide internally how to best accomplish their work rather than being directed by others outside the team. These teams are also cross-functional to possess all the skills necessary to complete the project without needing to rely on external resources. This setup encourages a high level of communication and collaboration within the team.

A key principle of Agile is adaptive planning. Unlike traditional methods, where the entire project is planned from the start and followed rigidly. Agile allows for plans to be adjusted as the project progresses. This adaptability makes Agile particularly suited for projects where requirements are expected to change or are not fully understood at the outset.

One standard Agile methodology is Scrum. Scrum promotes the idea of evolutionary development, to have early delivery and continuous improvement. It is a framework that helps teams work together to encourage teams to learn through experiences, self-organize while working on a problem, and reflect on their wins and losses to continuously improve (Kaur and Kumar, 2015).

## Description of the Scrum Process and the Waterfall Methodology

**Scrum** is known as an Agile framework that organizes work into small to manageable chunks known as sprints typically lasting two to four weeks. During these sprints, cross-functional teams work collaboratively to deliver specific features or product increments. Key roles in Scrum include the Product Owner who defines the product vision the Scrum Master who facilitates the process and the development team. Scrum emphasizes regular reflection and adaptation with frequent meetings and reviews to adjust to changing requirements and priorities (Ben-zahia et al., 2022).

Scrum is well-known framework in Agile project management that aims to improve how teams work together and respond to changes. It breaks down work into short phases known as sprints, which usually last about two to four weeks. In each sprint, diverse teams work together to build and deliver small parts of the product, known as increments.

In Scrum, there are several important roles. The Product Owner is responsible for setting the direction for the product, prioritizing the work that needs to be done, and making sure it meets the users' needs and the business goals. The Scrum Master helps the team follow the Scrum process and removes any obstacles that might slow down their work. The Development Team does the actual work of creating the product features, deciding as a group how to tackle each task.

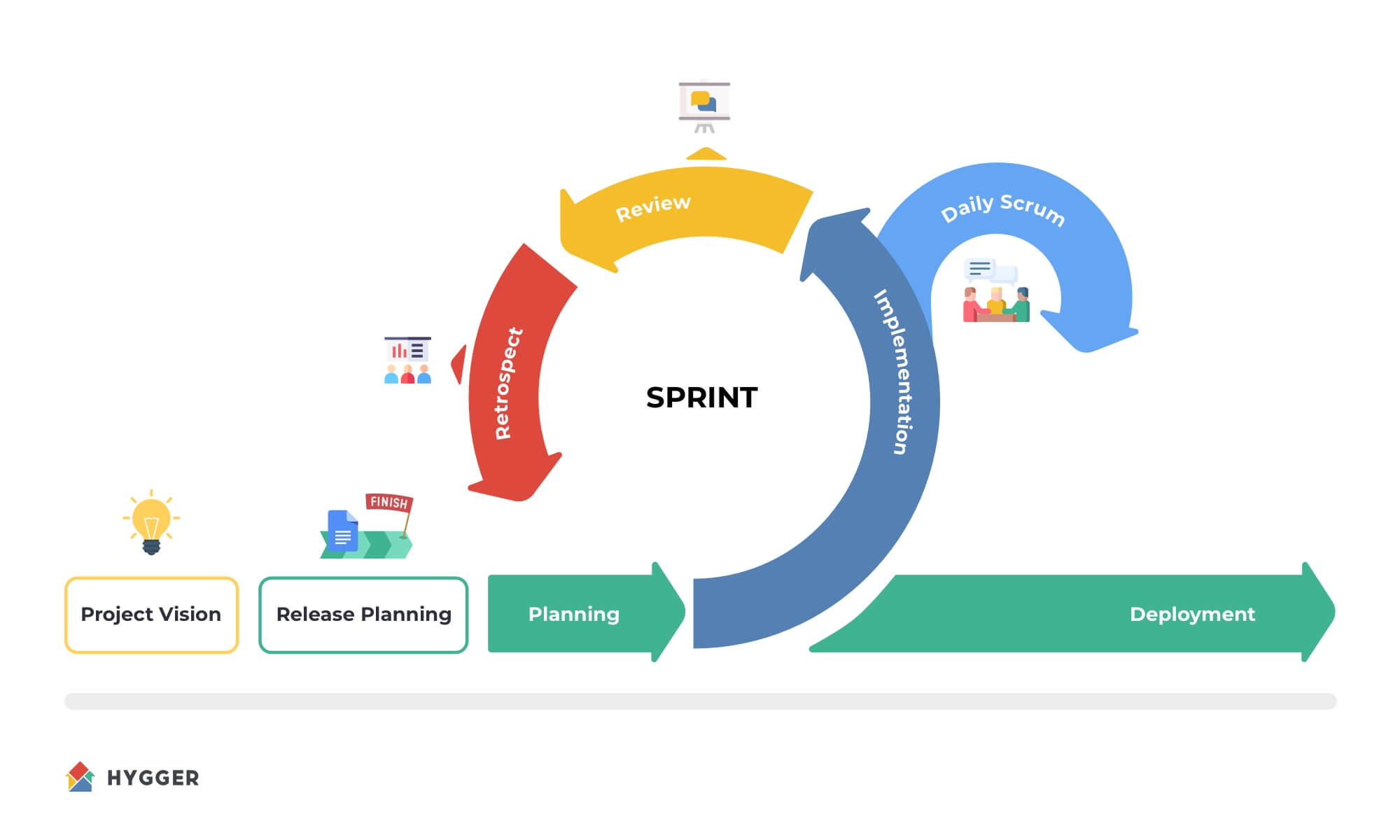
Scrum places a strong emphasis on regular check-ins and adjustments. Through meetings such as Sprint Planning, Daily Stand-ups, Sprint Reviews, and Sprint Retrospectives, the team keeps communication open, makes sure everyone is aligned, and looks for ways to improve their work process.

The following figure shows the cycle of the Scrum process. It starts with setting **Project Vision** where the main goals for the product are decided. This is followed by **Release Planning** which lays out what will be delivered to the users and when.

At the center of the cycle is the **Sprint** shown as a loop to indicate that this process repeats. It starts with **Planning** where the team picks tasks to work on. Then comes the **Implementation** where the team works on these tasks to have a meeting daily to talk about progress and any problems they face.

After the work of the Sprint is done so that the team holds a **Review** to show what they have completed and to get feedback. They then have a "Retrospect" meeting to think about what went well and what could be better next time.

The cycle ends with **Deployment** where the new part of the product is made available to users. This step completes the Sprint's efforts and the team is ready to start again, using what they learned to do better in the next Sprint. The image is a clear guide to the ongoing, step-by-step nature of Scrum, highlighting the importance of feedback and the ability to adapt at each step.



In contrast, the **Waterfall methodology** is a linear and sequential approach. It is characterized by a series of phases: i) requirements gathering ii) design iii) implementation iv) verification and maintenance with each phase completed before the next begins. This approach is often considered more rigid as it typically does not accommodate changes once a phase is completed. Waterfall is more appropriate for projects with exact results and outcomes are expected without changing requirements and a comprehensive plan can be executed without the need for significant changes.

The Waterfall model is a straightforward and orderly method used in software development. It is known for having distinct phases that follow one after the other. The process begins with gathering all the necessary information about what the software should do, which is called **requirements gathering**. Once this is evident, the team moves on to **design** where they create plans for how the software will work and look.

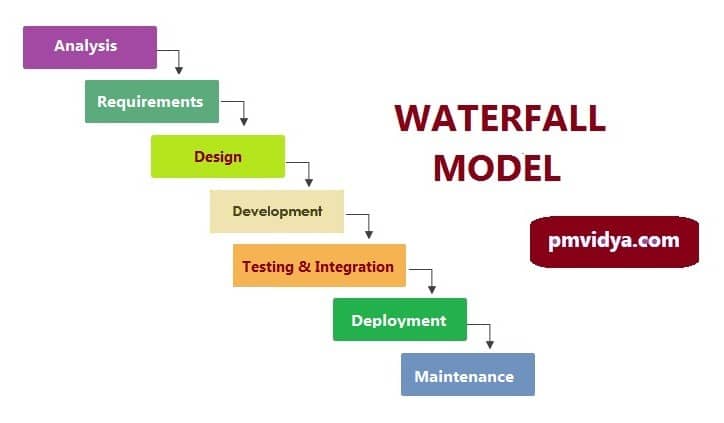
After the design is ready for **implementation** follows, which is when the actual building of the software takes place. By following the above, there's **verification** where the team checks and tests the software to make sure it does what it's supposed to do. Finally, after the software is released, there is **maintenance** to fix any problems that come up later.

This method works best for projects where the goals and requirements are not expected to change much as the project goes on. It's suited for situations where a clear plan can be made at the start and followed step by step without needing to go back and change things.

The following figure illustrates the stages of the Waterfall methodology in a step-like diagram, emphasizing its linear nature. Each box represents a different phase of the project, starting from the top left and moving downwards, signifying the completion of one phase before proceeding to the next.

The first box, labeled 'Analysis,' is where the team studies the problem they're solving. Next, the 'Requirements' box shows where the detailed needs of the software are collected. Following this is 'Design,' depicted as a phase where the software's architecture is planned out. Then, 'Development' is where the actual software coding takes place.

The 'Testing & Integration' phase is represented next, where the software is thoroughly tested and all its parts are brought together to work as a whole. After this, the 'Deployment' box indicates the release of the software to the users. Lastly, 'Maintenance' is shown as the final step, where ongoing support and updates are provided for the software.



## Guidelines for Determining Whether Scrum or Waterfall Suits Specific Software Development Projects

In choosing a software development method such as Scrum or Waterfall it's essential to consider the nature of the project the ability and skills of the teams, and the specific needs of the client. (McKinney, 2016). **Scrum** is particularly more appropiate for projects where requirements might change or evolve over time. This approach allows for a high degree of flexibility and the ability to make frequent adjustments. It's especially useful in fast-paced industries where anyone is adapting quickly to market changes or customer feedback is crucial. Scrum facilitates a dynamic work environment, enabling teams to pivot or alter their course with relative ease, ensuring that the project remains aligned with current demands and expectations. Alternatively, **Waterfall** the Waterfall methodology is ideal for projects with clearly defined requirements that are unlikely to change. This approach follows a linear as well as step-by-step process for making it straightforward and predictable. Waterfall is particularly effective in environments where changes can be expensive or complicated to implement. Projects in highly regulated industries or those requiring extensive planning benefit from this methodology, as it allows for a thorough initial planning phase. Each stage of the project is completed in turn, minimizing the risk of unexpected developments and ensuring a steady progression towards the project's goals. This method is characterized by its structured nature, offering a clear roadmap from the project's inception to its conclusion (Andrei et al., 2019).

## Differences Between Agile (Scrum) and Waterfall Methodologies

Agile methodologies considering the Scrum approach which are well-known for their quick adaptability and for including feedback from those involved in the project. In Scrum, the work is divided into short phases known as sprints. Each sprint is a focused effort where teams complete a portion of the project. At the end of each sprint the team presents their progress and gathers feedback. This approach is especially effective for projects where not all the details are known in advance or where the project may need to alteration based on new insights. It's a process that allows teams to stay in tune with what the customer wants and to make sure that the project remains on track and valuable from start to finish. Scrum's flexibility means it can better handle projects where the final goal might evolve as the work progresses (McKinney, 2016).

The Waterfall model is a methodical approach to project management, marked by its clarity and step-by-step progression. It begins by collecting all the necessary details about the project. Once this foundation is laid, the process moves into the design phase where the blueprint of the software is created. Following the design, the actual construction or coding of the software takes place. After the building phase thorough testing is conducted to ensure everything works correctly. The final steps are to release the software to the users and then enter a maintenance phase to resolve any issues that may arise after the launch.

Each of these stages happens in a fixed order, and you must complete one before moving on to the next. This linear approach is particularly well-suited for projects where the requirements are well-understood from the outset and are expected to stay the same. The Waterfall model offers a straightforward project plan, which can simplify the management of timelines along with mentioning the budgets and personnel. However, its structured nature means that it lacks the flexibility to adapt to new challenges or changes that weren't expected at the project's start. This makes it less ideal for projects where requirements are uncertain or may change during the development process (Kuhrmann et al., 2017).

|  |  |  |
| --- | --- | --- |
| **Criteria** | **Agile (Scrum)** | **Waterfall** |
| Project Flexibility | High flexibility to adapt to new information and changes | Low flexibility, with a focus on initial planning |
| Development Approach | Iterative, with regular revisions and frequent deliveries | Sequential, with each phase completed before moving to the next |
| Client Engagement | High level of client involvement throughout the project | Limited client engagement, primarily at the beginning and end |
| Adaptation to Change | Designed to embrace and incorporate changes readily | Changes are difficult to incorporate once the process has begun |
| Planning Process | Adaptive planning that evolves with the project’s needs | Detailed planning upfront with a focus on sticking to the plan |
| Risk Management | Continuous risk management and mitigation | Risk management is generally confined to the testing phase |
| Best Used For | Projects where requirements are expected to evolve | Projects with clear, detailed requirements that are unlikely to change |

## Conclusion

When choosing between Scrum and Waterfall methodologies for a project by consider the projects unique needs and circumstances. Considering the example of developing a Customer Relationship Management (CRM) system for global banking institutions. Scrum could be more supportive due to its flexible and iterative nature in such a case. This approach is particularly useful in environments where customer requirements and regulations are diverse and subject to change. Scrum allows the development process to adapt quickly to new trends and alterations would be required especially in the fast-evolving financial technology sector. However, if the project has clear and well-defined requirements that are unlikely to change then the Waterfall methodology might be a better fit. This method works well when a project follows a linear and sequential development path. It's ideal for situations where the project's direction is set from the start and major changes are not expected. It has been finally concluded that the choice between Scrum and Waterfall should be based on a balance between the need for adaptability and the specific demands of the project. Factors like the scope as well as complexity and the nature of the project's requirements should be carefully considered to make the best choice.

# References

ANDREI, B.-A., CASU-POP, A.-C., GHEORGHE, S.-C. & BOIANGIU, C.-A. 2019. A study on using waterfall and agile methods in software project management. *Journal of Information Systems & Operations Management***,** 125-135.

BEN-ZAHIA, M. A. O., ABURAS, A. & GHAWAR, M. The Challenges of Software Development: Waterfall and Agile. 2022. Libyan International Conference for Applied Science and Engineering.

KAUR, C. & KUMAR, V. 2015. Comparative Analysis of Iterative Waterfall Model and Scrum. *International Journal of Computer Science Research (IJCSR) sve,* 3**,** 11-14.

KUHRMANN, M., DIEBOLD, P., MÜNCH, J., TELL, P., GAROUSI, V., FELDERER, M., TREKTERE, K., MCCAFFERY, F., LINSSEN, O. & HANSER, E. Hybrid software and system development in practice: waterfall, scrum, and beyond. Proceedings of the 2017 international conference on software and system process, 2017. 30-39.

MCKINNEY, J. 2016. Optimal Software Development: Waterfall, Scrum & Kanban.