



CS250 FINAL PROJECT

PRESENTATION



Roles of the Agile

There are three main roles in the Agile Environment. These roles are extremely important in an project that needs to be completed.

Scrum Master: This role is akin to a coach for the Scrum Team, focusing on supporting and facilitating the processes according to the Scrum framework. The Scrum Master ensures that the team follows agile practices, removes impediments, and acts as a buffer between the team and any distracting influences.

Importance: Ensures the team operates efficiently and adheres to agile principles.

Product Owner (PO): The PO is responsible for maximizing the value of the product resulting from the work of the Development Team. This is achieved by managing the product backlog and ensuring that it is visible, transparent, and clear to all. **Importance:** Acts as the liaison between the stakeholders and the team, ensuring the team is working on the most valuable tasks first.

Development Team: A cross-functional group of professionals who do the actual work of delivering potentially shippable product increments at the end of each sprint. This includes developers, designers, and testers. **Importance:** They are the engine of the Scrum Team, turning ideas into valuable product features.



● Phases of the SDLC in an Agile Approach

- **Concept and Initiation:** An idea for a project is explored and its feasibility is determined. **Importance:** It sets the foundation for what the project is supposed to achieve and assesses whether it's worth pursuing.
- **Planning:** Unlike in Waterfall, planning in Agile is iterative and involves setting up the product backlog, sprint backlogs, and deciding on the scope of the initial sprint. **Importance:** Provides a flexible roadmap that guides the project but can adapt to changes.
- **Execution and Iteration:** Work is performed in short cycles called sprints, focusing on delivering small, incremental changes to the product. **Importance:** Allows for regular feedback and adjustments, ensuring the product meets user needs.
- **Testing and Quality Assurance:** Testing in Agile is integrated into each sprint, allowing for immediate feedback and adjustments. **Importance:** Ensures that the product is of high quality and meets customer expectations throughout its development.
- **Deployment:** Product increments are delivered to customers frequently, at the end of each sprint. **Importance:** Allows for quick market entry and continuous improvement based on user feedback.
- **Review and Retrospective:** At the end of each sprint, the team reviews the work done and discusses ways to improve future sprints. **Importance:** Promotes continuous improvement and team cohesion.



● Agile Vs. Waterfall Development

- In **Waterfall**, development follows a linear, sequential approach where each phase must be completed before the next begins, and there is no going back. If a problem is discovered late in the development cycle, it can be costly and time-consuming to go back and make changes.
- **Example:** If during the testing phase of a Waterfall project, it's discovered that a feature doesn't meet the user's needs, making adjustments would require revisiting previous stages, leading to delays and increased costs.
- In contrast, **Agile** allows for more flexibility and adaptability. The iterative process means that feedback is incorporated continuously, allowing for adjustments without significant setbacks.
- If we assume our instance is the example, we can see that through every step of the process we would have had a significant differences. During our process we had two times where the client gave us more feedback. In a waterfall approach, we would not have had the chance to adjust our goals as significantly.



● Choosing Between Waterfall and Agile

- **Project Complexity and Size:** Agile is well-suited for complex projects with uncertain requirements, allowing for flexibility. Waterfall might be more appropriate for simpler projects with well-defined requirements. Our project that we did this semester was not a large one. A waterfall method would have been sufficient, but agile was better.
- **Customer Involvement:** Agile requires and benefits from continuous customer feedback. If customer involvement is limited, Waterfall might be more practical. Due to customer feedback in our project, an agile method made more sense.
- **Risk Management:** Agile allows for early discovery of issues and continuous improvement, making it better for projects where risk management is crucial.
- **Timeline and Budget Constraints:** If the project has a strict timeline and budget, the structured nature of Waterfall can offer predictability, whereas Agile's flexibility might lead to scope creep.

