MAWLANA BHASHANI SCIENCE AND TECHNOLOGY UNIVERSITY

Santosh, Tangail-1902

Assignment

Department of : Information and Communication Technology

Assignment No : 02

Name of the assignment : Different Agile Approaches

Course Title : Software Engineering

Course Code : ICT-3209

Submitted by

Name : Shetu Saha

ID : IT-21009 Session : 2020-21

Year : 3rd Semester : 2nd

Dept. of ICT MBSTU

Submitted to

Dr. Ziaur Rahman

Associate Professor

Dept. of ICT

MBSTU

Date of Performence : 30-11-24 Date of Submission : 06-12-24

Shetu Saha
IT-21009
MBSTU

Name of Assignment: Different Agile Approaches and their comparative Analysis.

1) How it works?

2) Applicability?

3) Effectiveness in terms of costs? Include example and diagrams in necessary cases.

Answer:

Agile is a set of principles used for project management and software development that emphasizes flexibility, collaboration, and customer-centricity. There are several Agile approaches, each with its own processes, roles, and strengths. In this analysis. I'll compare some of the most popular Agile approaches: Scrum, Kanban, Extreme Programming (XP), and Lean Agile. I'll discuss their working methodology, applicability and effectiveness in terms of costs.

Shetu Saha
IT-21009
MBSTU

Scrum

How it Works

Scrum is based on iterative development with fixed-length cycles called sprints (typically 2-4 weeks). Each sprint includes:

- Sprint Planning: Teams decide what work to complete in the sprint.
- Daily Scrum (Standup): A brief daily meeting to review progress and remove obstacles.
- Sprint Review: At the end of each sprint, the team reviews the completed work.
- Sprint Retrospective: Reflects on the sprint to identify improvements for the next one.

Key roles in Scrum include:

- Product Owner: Defines the product backlog
- Scrum Master: Facilitates Scrum processes and removes impediments.
- Development Team: Executes the work.

Applicability

Scrum is ideal for projects where requirements are expected to evolve. It's especially useful in software development where iterative progress with frequent feedback is important.

Effectiveness in Terms of Costs

- Costs: Scrum can reduce costs by delivering working features early, enabling early feedback and minimizing the need for large rework.
- Challenges: It requires constant interaction and frequent meetings, which can lead to overhead costs if not well managed.

Example:

In software development, a company developing a mobile application may use Scrum to iterate through features, testing, and refining with each sprint to ensure they meet customer expectations.

Shetu Saha
IT-21009
MBSTU

Kanban

How it works:

Kanban is a visual workflow management system that focuses on continuous delivery and process improvement. Teams use boards (Physical or digital) to represent the workflow stages and tasks (cards.). Work items are pulled from one stage to another when there is capacity. Kanban focuses on:

- Limiting Work in Progress (WIP): To ensure team capacity is not overburdened.
- Continuous flow: There is no fixed iteration or sprint.
- Focus on cycle Time: Reducing the time it takes for work items to move through the process.

Applicability:

Kanban is best for projects that require continuous delivery with flexible priorities. It works well for ongoing support, operations and maintenance team where work is unpredictable and needs to flow without the need for iteration cycles.

Effectiveness in Terms of Costs :

- Costs: Kanban's ability to limit work in progress helps prerent wasted resources and overwork, leading to cost savings. Since there is no sprint planning or overhead, costs are lower than in scrum.

- Challenges : Requires strict discipline in managing workflows and limiting WIP, which can be difficult to implement without commitment from the team.

Example:

A support team in a tech company may use Kanban to handle incoming customer requests, where each request moves through stages like "To Do","In Progress", and "Done" until the issue is resolved.

Shetu Saha
IT-21009
MBSTU

Extreme Programming (XP)

How it Works

Extreme Programming (XP) is an Agile methodology aimed at improving software quality and responsiveness to changing customer requirements.

XP emphasizes:

- Pair Programming : Two developers work together on the same task.
- Test-Driven Development (TDD) : Writing tests before coding to ensure functionality.
- Continuous Integration (CI) : Integrating code into a shared repository multiple times a day.
- Frequent Releases : Delivering small, frequent releases with high-quality code.

XP focuses heavily on technical practices to ensure high-quality, maintainable software.

Applicability :

XP is suitable for complex and high-risk software projects, particularly those requiring fast adaptation to changes, rapid delivery and high levels of collaboration. It's best for small to medium-sized teams working on high-quality software.

Effectiveness in Terms of Costs

- Costs: The emphasis on early testing, continuous integration and pair programming can initially increase costs, but these practices reduce defects and technical debt over time, leading to lower maintenance and rework costs.
- Challenges: Pair programming and continuous integration require highly skilled developers which can increase the upfront cost.

Example:

A software development company building a new online banking system might use XP practices to ensure high security and functionality, with constant testing and collaboration.

Shetu Saha
IT-21009
MBSTU

Lean Agile

How it Works

Lean Agile is based on principles from Lean manufacturing and Agile development. It focuses on:

- Eliminating Waste: Streamlining processes and eliminating non-value-adding activities.
- Continuous Improvement (Kaizen): Always looking for ways to improve processes.
- Delivering Small Batches: Breaking down work into smaller chunks to deliver value quicker.

Lean Agile doesn't rely on specific roles or iterations like Scrum. It focuses on principles and flow to improve efficiency.

Applicability

Lean Agile works well for teams looking to optimize processes, reduce inefficiencies, and deliver value quickly, especially in environments where quick turnarounds are required.

Effectiveness in Terms of Costs

- Costs : Lean minimizes waste and emphasizes efficiency, reducing the cost of delivering value. By cutting out unnecessary processes and focusing on high-priority items, it can lead to significant cost savings over time.
- Challenges : Requires a culture of continuous improvement and commitment to waste elimination, which can be difficult to sustain.

Example

A manufacturing company implementing software solutions for inventory management might use Lean Agile principles to continuously optimize its development cycle and eliminate bottlenecks in the workflow.

Shetu Saha
IT-21009
MBSTU

Comparative Analysis

Approach	How It Works	Applicability	Effectiveness in Terms of Costs
Scrum	Fixed-length sprints roles, regular feedback and iteration.	Suitable for projects with evolving requirements	Cost-effective in the long run by delivering value early and reducing rework.
Kanban	Continuous flow with visual boards and WIP limits	Best for continuous delivery and support operations	Lower Overhead cost but needs discipline to avoid bottlenecks
Extreme Programming (XP)	Emphasizes technical excellence, pair programming and continuous integration.	Complex and high-risk software projects	Higher upfront cost but reduces technical debt and long term maintenanance costs
Lean Agile	Focuses on eliminating waste improving processes, and delivering in small batches.	Works well for optimizing processes and delivering fast.	Very cost-effective by minimizing waste and streaming processes.