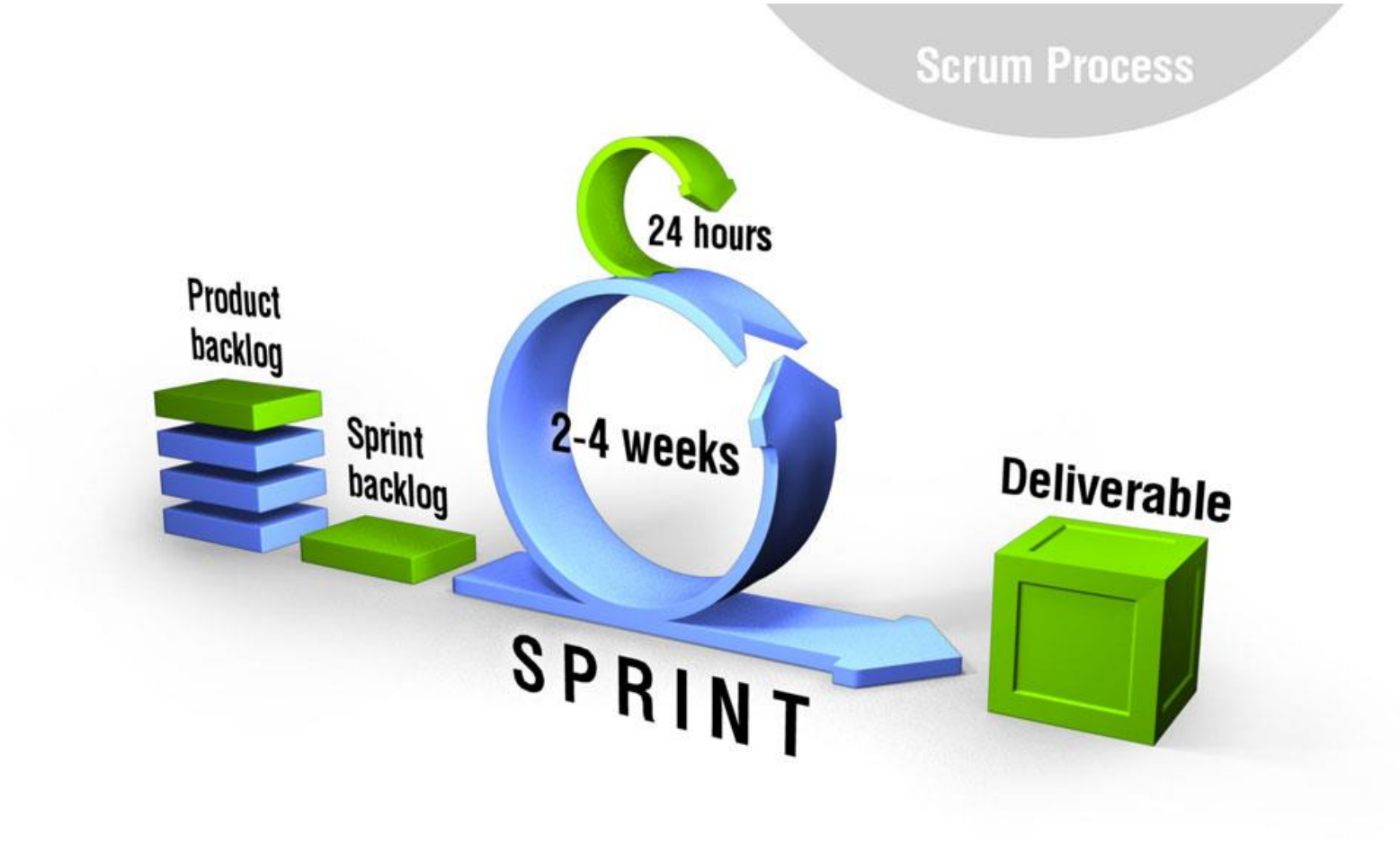
University of Kelaniya Sri Lanka

Software Engineering

COSC 21052

BECS 21213

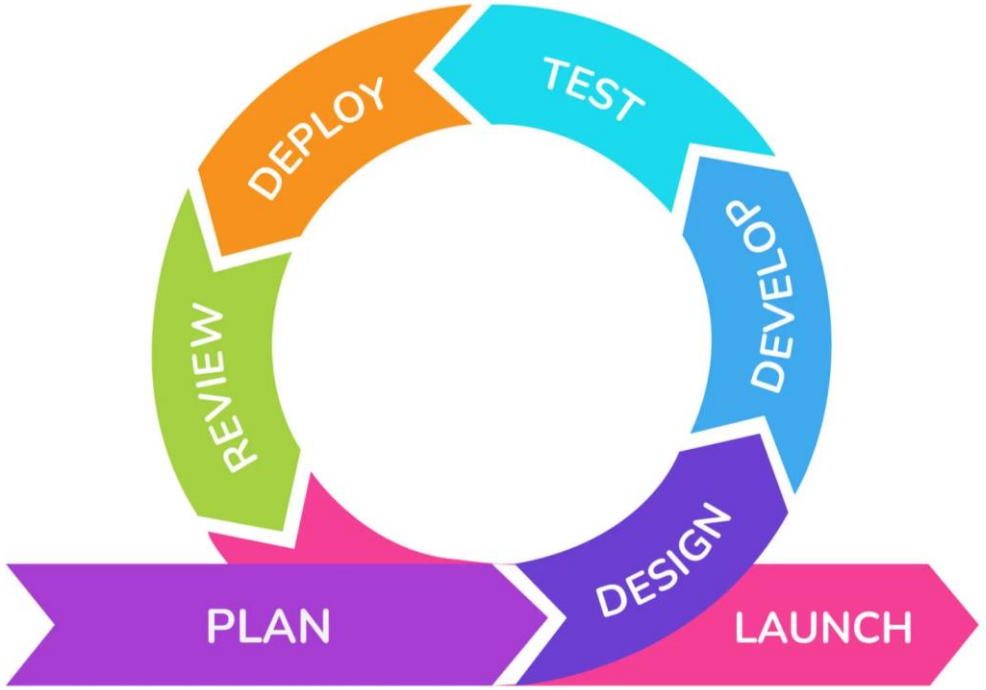
AGILE DEVELOPMENT



What is “Agility”?

- Effective (rapid and adaptive) response to change
- Effective communication among all stakeholders
- Drawing the customer onto the team

- Agile project management is based on an incremental, iterative approach.
- Instead of in-depth planning at the beginning of the project, Agile methodologies are open to changing requirements over time and encourage constant feedback from the end users.
- Agile refers to any process that aligns with the concepts of the Agile Manifesto.



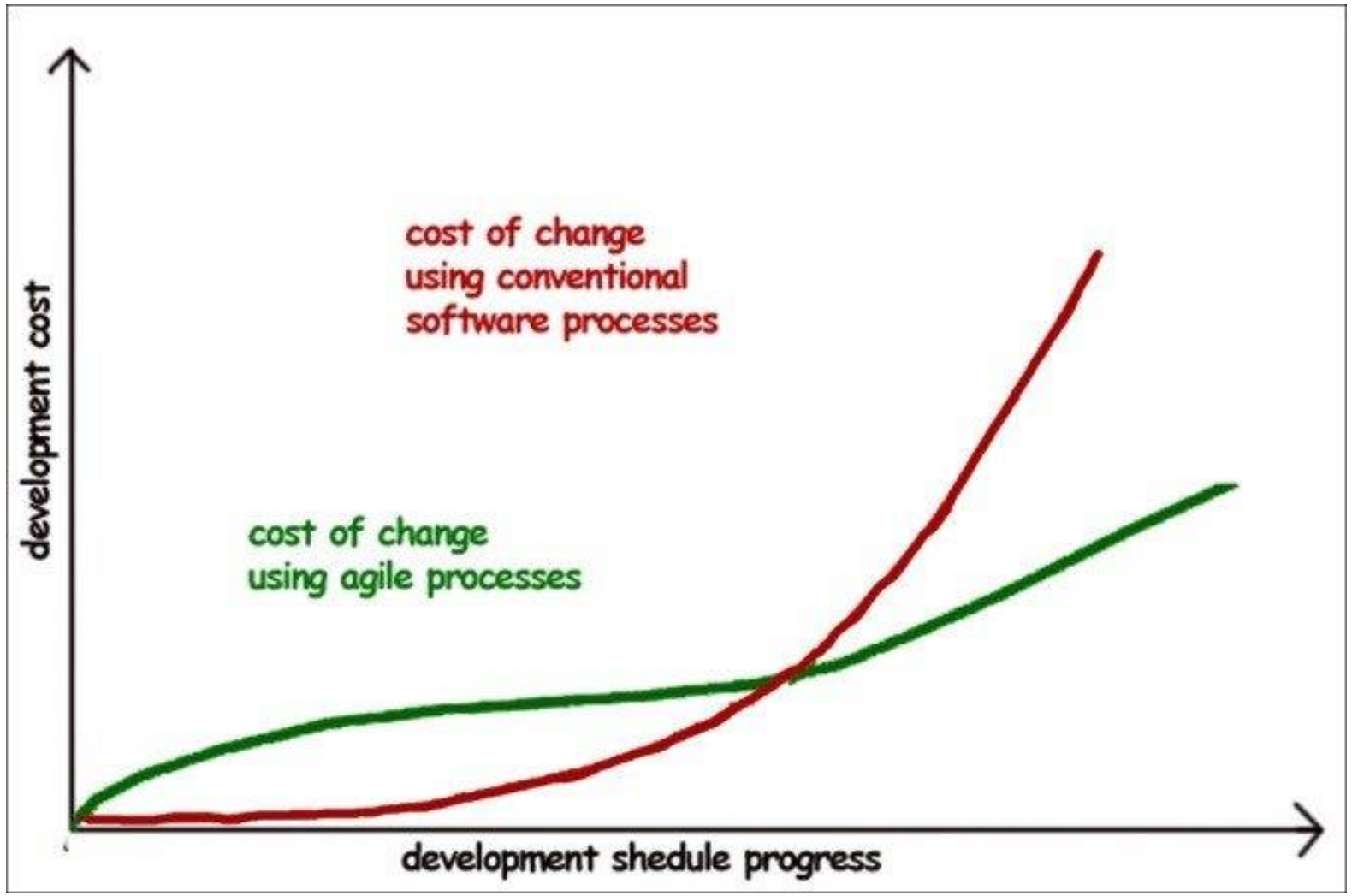
Agile Development Cycle

- Planning: project team comes together to identify features, prioritize each feature, and assign them to an iteration.
- Requirements analysis: Key stakeholders and users meet to identify business requirements.
- Design: The design is prepared from the requirements identified and the team considers what the product or solution will look like.
- Implementation, coding or development: Coding or developing features, and scheduling iterations for deployment.
- Testing: Test the code against the requirements to make sure the product is actually solving customer needs.
- Deployment: Deliver the product to customers. Once customers start using the product, they may run into new problems that the project team will need to address in future iterations.

Principles In Agile Manifesto

- **The Agile Manifesto lists some principles to guide teams on how to execute with agility.**

  - Highest priority is to satisfy the customer through early and continuous
  - delivery of valuable software.
  - Welcome changing requirements, even late in development. Agile processes harness change for the customer’s competitive advantage.
  - Deliver working software frequently, from a couple of weeks to a couple of months, with a preference for a shorter timescale.
  - Businesspeople and developers must work together daily throughout the project.
  - Continuous attention to technical excellence and good design enhances agility



- An agile process reduces the agility and cost of change because software is released in increments and change can be better controlled within an increment.

Human Factors

- Human factors in this context refer to considering human abilities, behaviors, and interactions within the Agile framework to create an environment that fosters effective teamwork and successful project outcomes.
- key traits must exist among the people on an agile team and the team itself:
  - Competence.
  - Common focus.
  - Collaboration.
  - Decision-making ability.
  - Fuzzy problem-solving ability.
  - Mutual trust and respect.
  - Self-organization.

Advantages of Agile

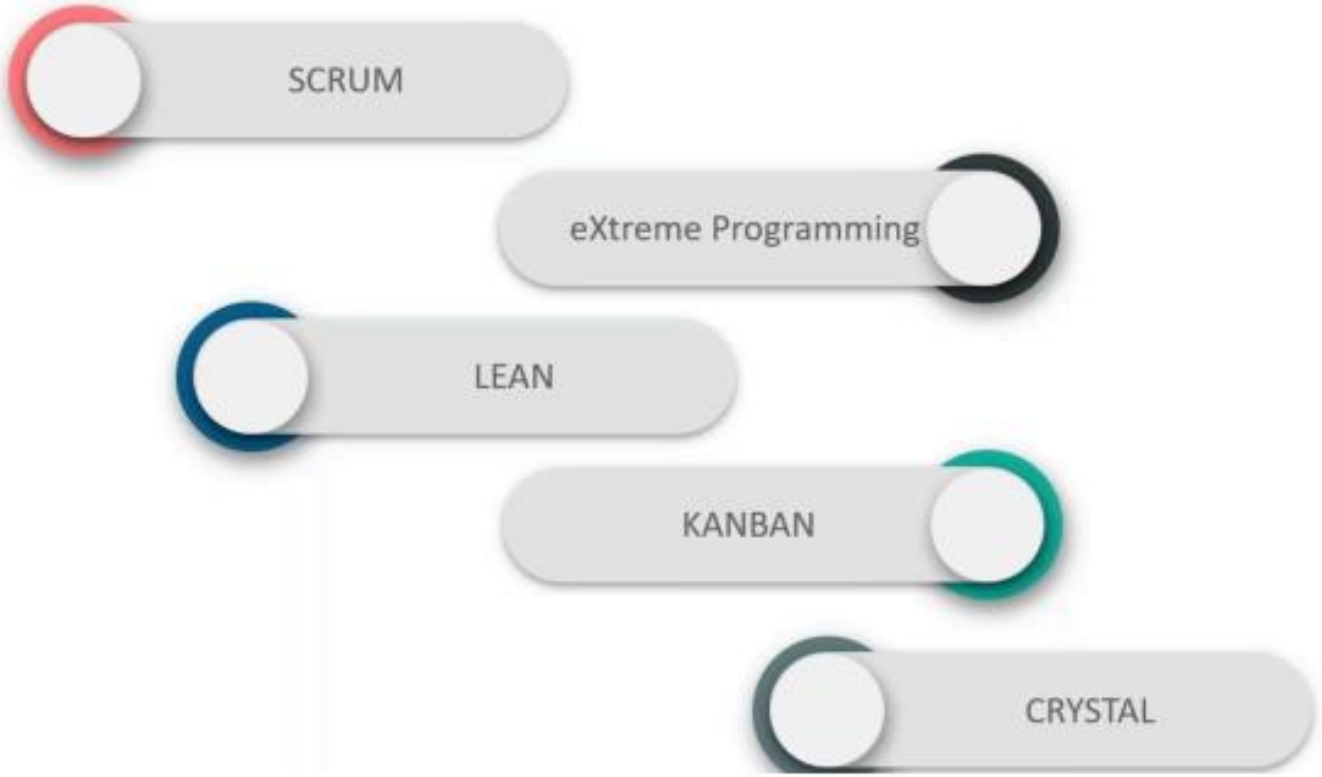
- Faster, high-quality delivery: Breaking down the project into iterations allows the team to focus on high-quality development, testing, and collaboration.
- Strong team interaction: Agile embraces frequent communication and face-to face interactions.
- Customers are heard: Customers have many opportunities to see the work being delivered, share their input, and have an impact on the end product.
- Continuous improvement: Feedback is encouraged from users and team members throughout the project, so lessons learned are used to improve future iterations.
- Welcome to Changes at Any Stage

Disadvantages of Agile

- Planning can be less concrete: Because project managers are often reprioritizing tasks, it’s possible that some items scheduled for delivery may not be completed in time.
- Team must be knowledgeable: Agile teams are usually small, so team members must be highly skilled in a variety of areas and understand Agile methodology.
- Documentation can be neglected: Agile prefers working deliverables over comprehensive documentation
- Time commitment from developers: Active involvement and collaboration is required throughout the Agile process.

Top Methodologies Used to Implement Agile

There are several specific methods within the Agile movement.

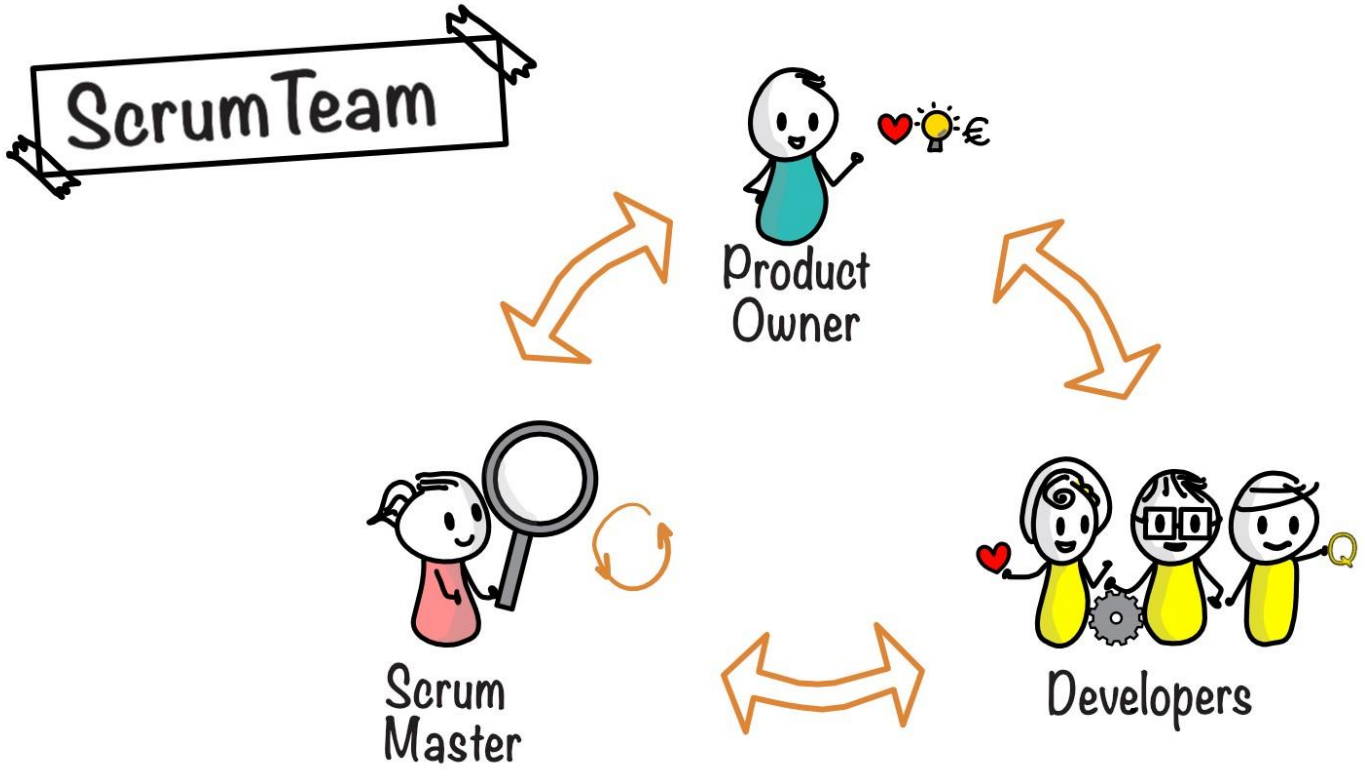


Scrum Methodology

- Scrum is a subset of Agile and one of the most popular process frameworks for implementing Agile.
- It is an iterative development model often used to manage complex software and product development.
- Fixed-length iterations, called sprints lasting one to two weeks long, allow the team to ship software on a regular cadence.
- At the end of each sprint, stakeholders and team members meet to plan the next steps.
- Sprints are usually limited to one calendar month.


SPRINT 1
TEST
DEVELOP
DESIGN
PLAN
LAUNCH
REVIEW
DEPLOY
SPRINT 2
TEST
DEVELOP
DESIGN
PLAN
LAUNCH
REVIEW
DEPLOY
SPRINT 3
TEST
DEVELOP
DESIGN
PLAN
LAUNCH
REVIEW
DEPLOY

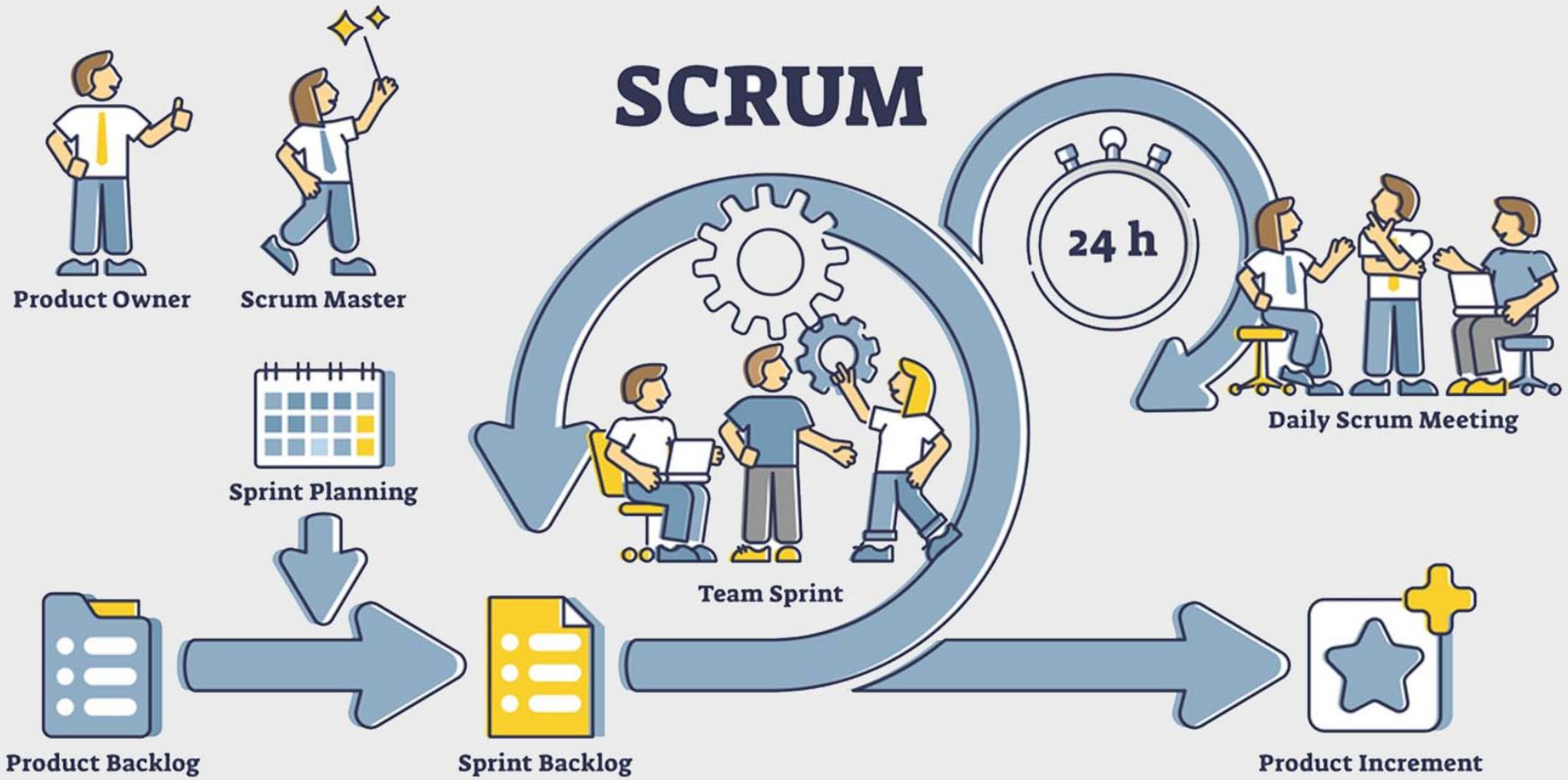
What are the roles in Scrum?



Roles in Scrum

- Product Owner: The Scrum Product Owner has the vision of what to build and conveys that to the team. He or she focuses on business and market requirements, prioritizing the work that needs to be done.
- Scrum Master: Often considered the coach for the team, the Scrum Master helps the team do their best possible work. This means organizing meetings, dealing with roadblocks and challenges, and working with the Product Owner to ensure the product backlog is ready for the next sprint.
- Scrum Team: The Scrum Team is comprised of five to seven members. Unlike traditional development teams, there are not distinct roles like programmer, designer, or tester. Everyone on the project completes the set of work together.

What is Scrum methodology?



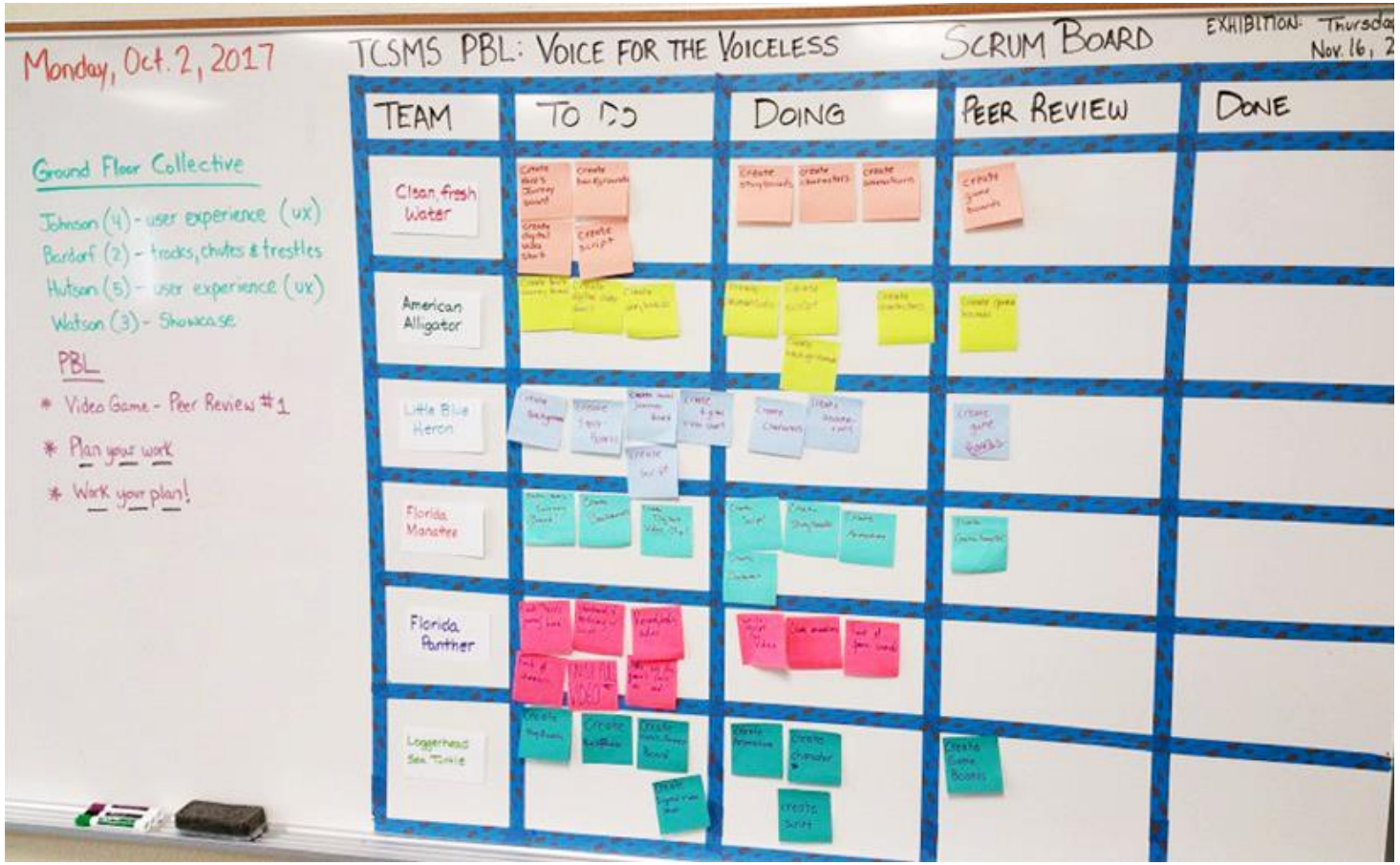
Steps in the Scrum Process

- Product backlog: The product backlog is not a list of things to be completed but rather a list of all the desired features for the product.
- Sprint planning: Before each sprint, the Product Owner presents the top items on the backlog in a sprint planning meeting. The team determines the work they can complete during the sprint and moves the work from the product backlog to the sprint backlog. (Time boxed at 2 hours a week)
- Backlog refinement: At the end of each sprint, the team and Product Owner meet to make sure the backlog is ready for the next sprint.
- Daily Scrum meetings: The Daily Scrum is a 15-minute stand-up meeting that happens at the same time and place every day during the sprint. During the meeting each team member talks about what they worked on the day before, what they'll work on today, and any roadblocks.(Time boxed at 15 minutes)

- Sprint review meeting: At the end of each sprint, the team presents the work they have completed as a live demo rather than a presentation.(Time boxed at 1 hour a week)
- Sprint retrospective meeting: Also at the end of each sprint, the team reflects on how well Scrum is working for them and talks about any changes that need to be made in the next sprint.(Time boxed at 45 minutes a week)

Tools in Scrum

- Scrum board: The Scrum board helps to visualize your sprint backlog and traditionally involves index cards.



Advantages of Scrum

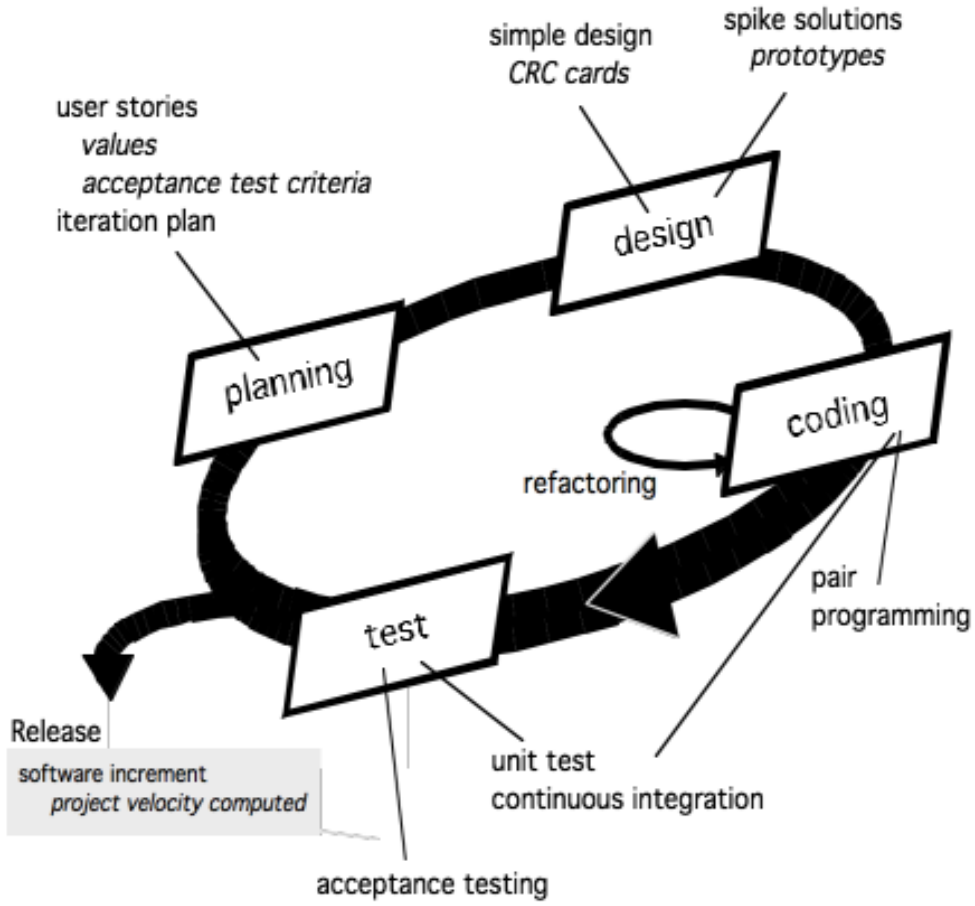
- More transparency and project visibility: With daily stand-up meetings, the whole team knows who is doing what and issues are identified in advance, improving communication and enabling the team to take care of issues right away.
- Increased team accountability: There is no project manager. Instead, the team collectively decides what work they can complete in each sprint.
- Easy to accommodate changes: With short sprints and constant feedback, it’s easier to accommodate changes.

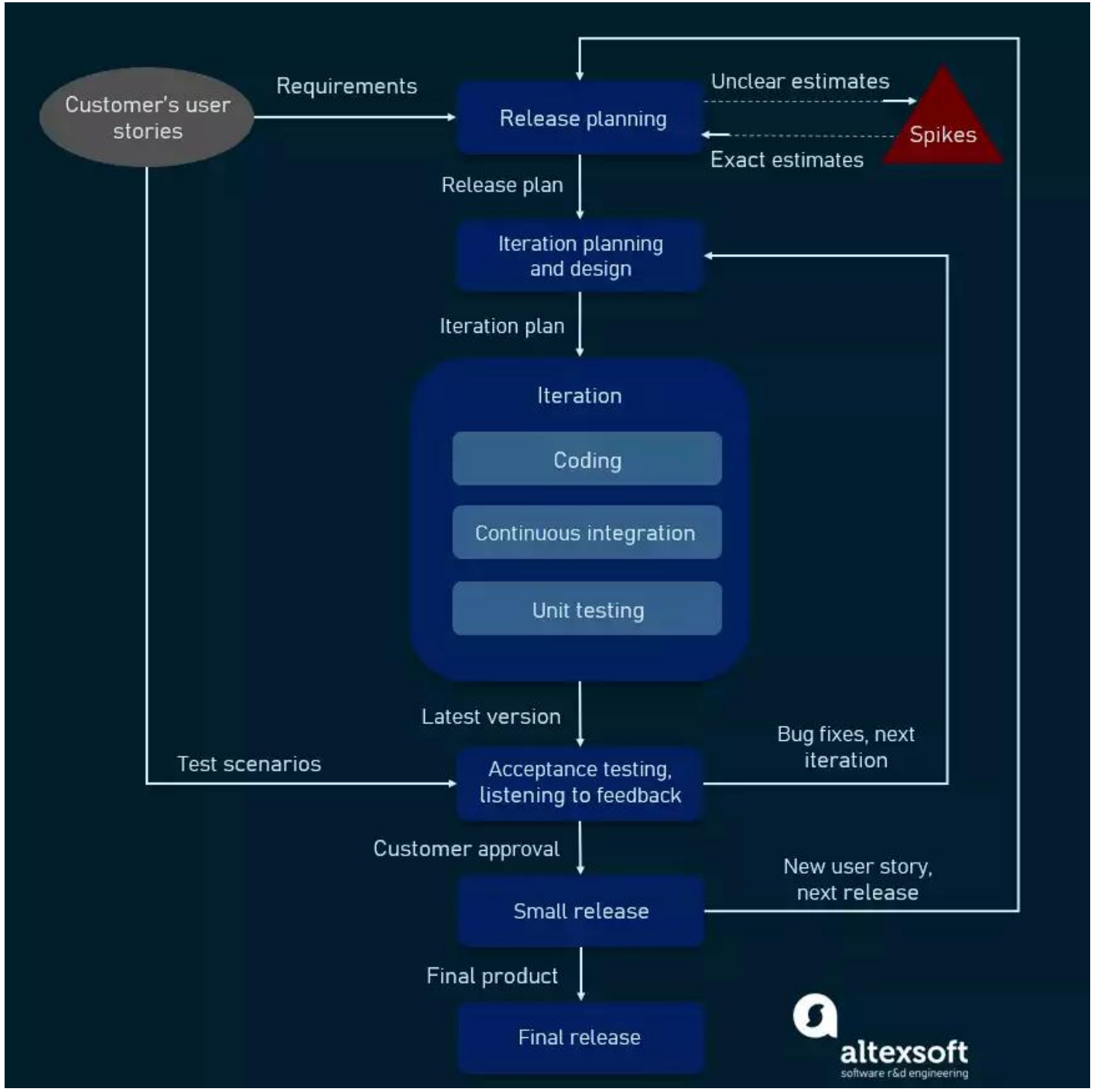
About Smartsheet

- By providing a broad range of smart views – Grid, Calendar, Gantt, Sights –Smartsheet works the way you want. Our newest view, Card View, gives teams a more highly-visual way to work, communicate, and collaborate in Smartsheet.
- Want to know how Smartsheet can help individuals and teams implement Agile practices? Click here

Extreme Programming (XP)

- It is used to improve software quality and responsiveness to customer requirements.



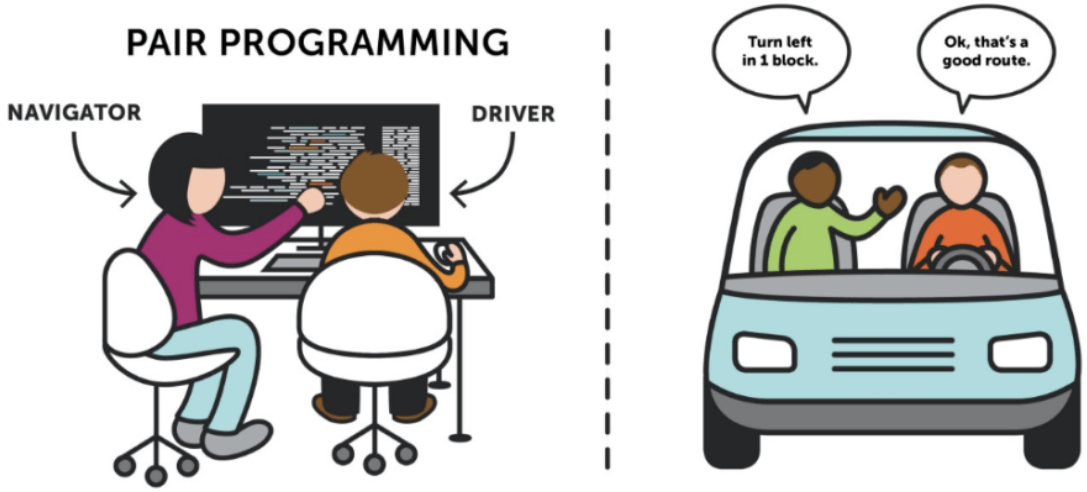

Customer's user stories
Requirements
Release planning
Unclear estimates
Spikes
Exact estimates
Release plan
Iteration planning and design
Iteration plan
Iteration
Coding
Continuous integration
Unit testing
Latest version
Test scenarios
Acceptance testing, listening to feedback
Bug fixes, next iteration
Customer approval
New user story, next release
Small release
Final product
Final release
altexsoft
software r&d engineering

The process and roles of extreme programming

- Planning, the first stage, is when the customer meets the development team and presents the requirements in the form of user stories to describe the desired result.
- The team then estimates the stories and creates a release plan broken down into iterations needed to cover the required functionality part after part.
- If one or more of the stories can’t be estimated, so-called spikes can be introduced which means that further research is needed.

- Designing is actually a part of the planning process, but can be set apart to emphasize its importance.
- It’s related to one of the main XP values discussed below simplicity.
- A good design brings logic and structure to the system and allows to avoid unnecessary complexities and redundancies.

- Coding is the phase during which the actual code is created by implementing specific XP practices such as coding standards, pair programming.



- Testing is the core of extreme programming. It is the regular activity that involves both **unit tests** (automated testing to determine if the developed feature works properly) and **acceptance tests** (customer testing to verify that the overall system is created according to the initial requirements).

- Listening is all about constant communication and feedback. The customers and project managers are involved to describe the business logic and value that is expected.

- **XP has simple rules that are based on 5 values to guide teamwork:**
  - Communication: Everyone on a team works jointly at every stage of the project.
  - Simplicity: Developers strive to write simple code bringing more value to a product, as it saves time and effort.
  - Feedback: Team members deliver software frequently, get feedback about it, and improve a product according to the new requirements.
  - Respect: Every person assigned to a project contributes to a common goal.
  - Courage: Programmers objectively evaluate their own results without making excuses and are always ready to respond to changes.

- Extreme Programming (XP) emphasizes technical excellence, customer involvement, and continuous development. Scrum focuses on delivering value through fixed-length sprints, with roles like Product Owner and Scrum Master. XP has more technical practices, while Scrum relies on Scrum events and prioritization by the Product Owner.
- Both XP and Scrum aim to enhance collaboration, quality, and customer satisfaction. The choice between them depends on the specific needs and preferences of the development team and the organization.

END.