

Agile Methodology

- Agile is a group of software development methodologies or frameworks based on **iterative development**, where requirements and solutions evolve through collaboration between self-organizing cross functional teams.
- Agile methodologies emphasize on **frequent delivery of working software and face-to-face communication** over written documentation.
- Agile is ***adaptive* (adaptive to changes)**.
- Popular agile frameworks are: Scrum, XP and Crystal

Scrum

- Scrum is one of the **agile frameworks** for software development
- Scrum is an **iterative incremental framework** for managing complex work (such as new product development)
- Developed by Jeff Sutherland, Ken Schwaber and Mike Beedle.
- Project is divided into multiple short time boxes of 2-4 weeks duration called “**sprint**”.
- Focus is on delivering working software **fast and on team collaboration**

Scrum

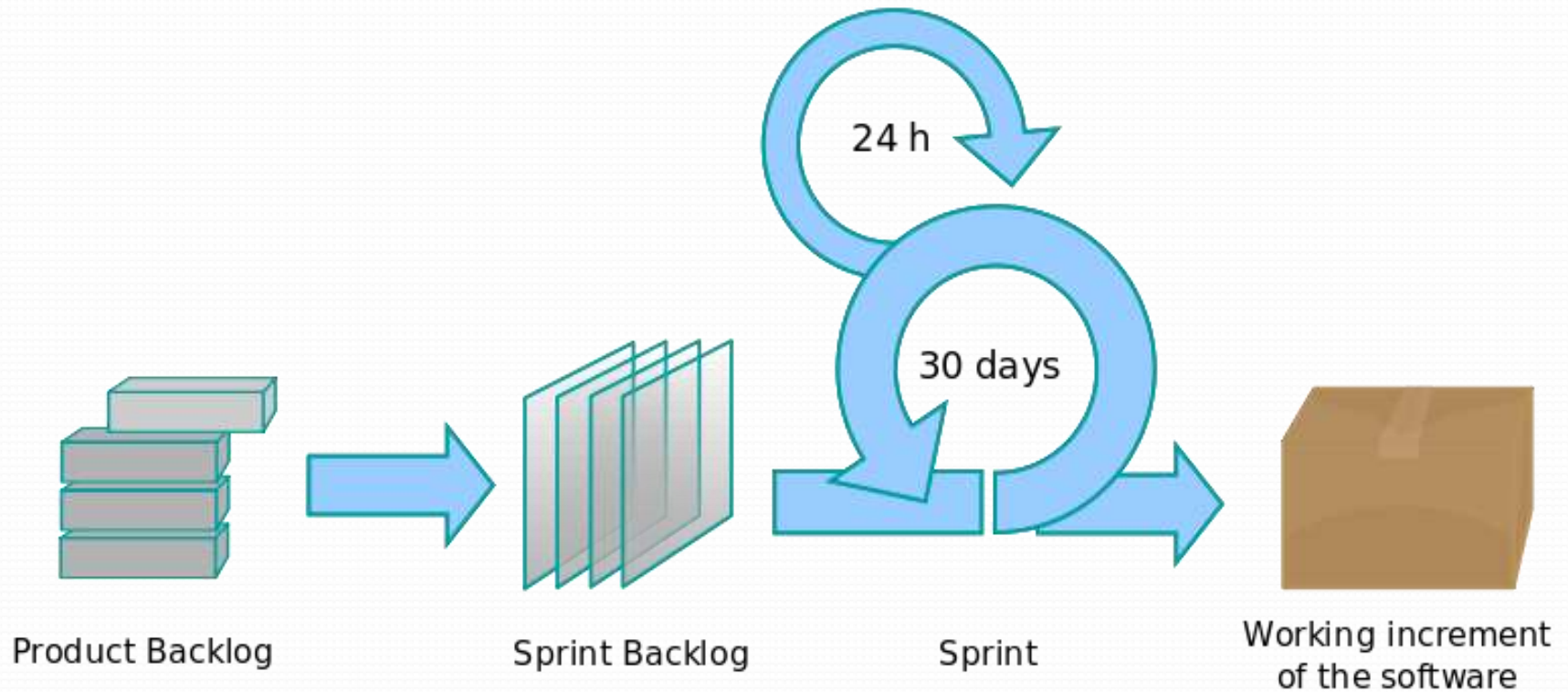
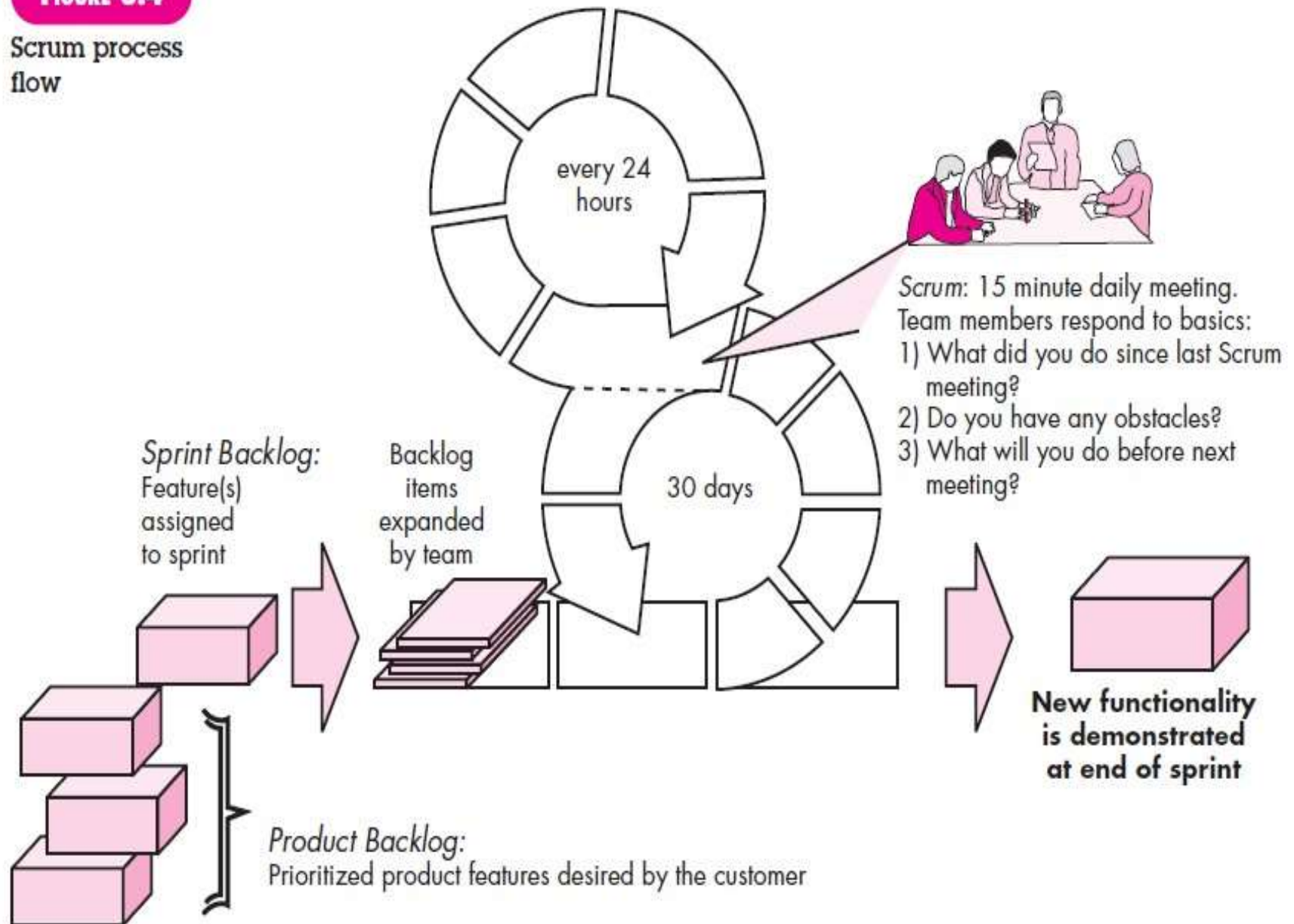


FIGURE 3.4

Scrum process
flow



Roles in Scrum

1. **Product Owner**-Represents the customers, users, and other stakeholders.

- Defines and documents the product features called “user stories”
- Prioritizes the features to be developed
- Controls the product backlog (prioritized list of product features)
- Provides a point of contact between the stakeholders and developers

2. **Scrum Master**

- Responsible for the success of scrum
- Scrum values, practices and rules are enacted and enforced
- Scrum Master isn't a project manager because the team is self-organizing, so it guides itself.
- Conducts all daily scrums
- Removes impediments

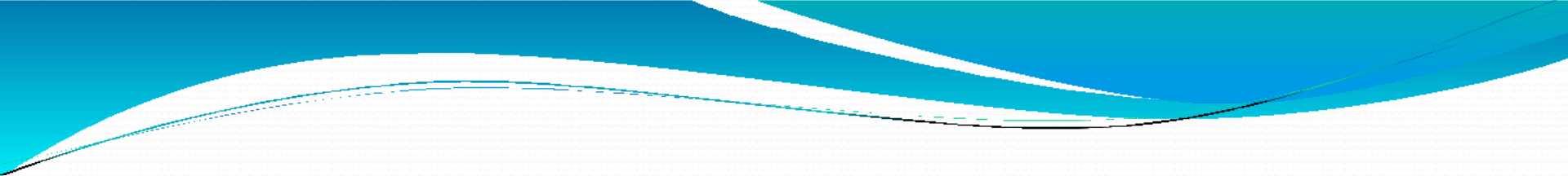


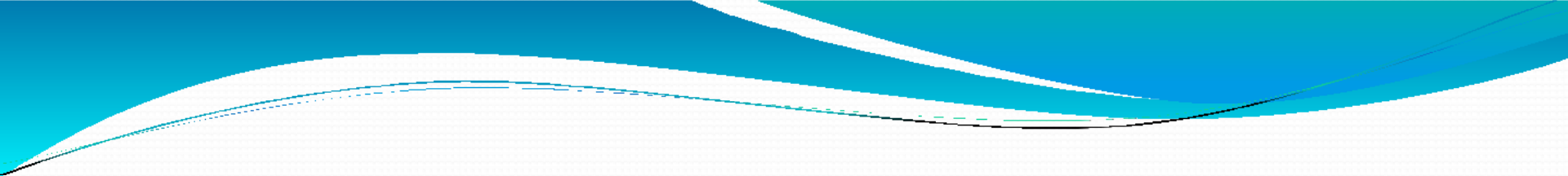
3. Team member(Scrum Team)

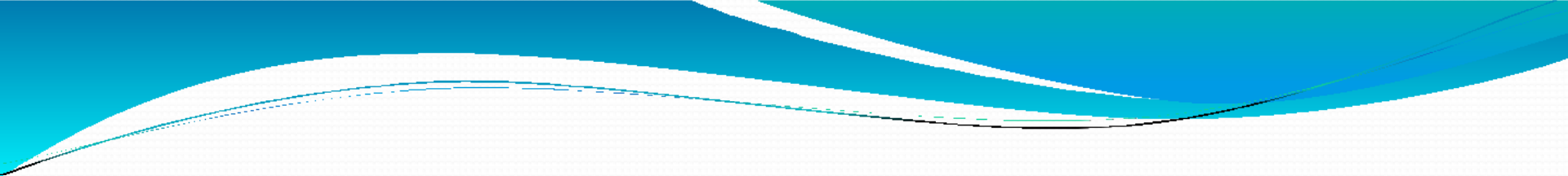
- Commits to achieving a sprint goal and have the full authority
- Seven +- 2 Cross Functional team members
- self-organized cross-functional team members who build the application.
- During each iteration, each team member handle all the typical tasks need to write a decent piece of software (analysis, design, program, test, document, and so forth).

Scrum Sprints

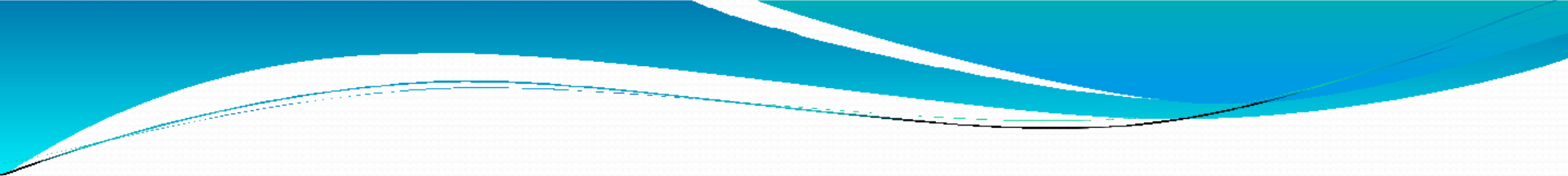
- A Scrum project creates a series of timeboxed incremental iterations, which are usually called *sprints*.
- *In traditional Scrum, a sprint is 30 days long; although some people prefer shorter sprints of one, two, or three weeks.*
- The result of each sprint is a fully tested and approved piece of software, which is sometimes called a *potentially shippable increment (PSI)*.
- Before each sprint begins, the team holds a *sprint planning meeting*-timeboxed to four hours

- 
- During that meeting, the product owner decides which user stories should be selected for the upcoming sprint.
 - Fixes for any outstanding bugs should also be included.
 - When the meeting is done, the selected items are moved from the product backlog into the *sprint backlog* .
 - Developers break the user stories into tasks. They analyze the tasks, design solutions, write code, and test.

- 
- During the sprint, the team holds a quick 10–15 minute ***daily scrum*** (sometimes called a “standup,”/“scrum”) where each developer answers the Three Questions of Agile Development:
 - What did you do since the last scrum?
 - What do you hope to accomplish before the next scrum?
 - What obstacles do you see in your way?

- 
- After all the work is done, the sprint ends with a *sprint review meeting*.
 - *The development team* presents the current PSI to the product owner, who checks the results against the items that were originally selected for the sprint to make sure the sprint's goals have been met.



- 
- After the sprint review meeting, the Scrum Master and the development team hold a **retrospective meeting** *where they discuss the recent sprint. Here they discuss the three big questions:*
 - What went well and how can we make it happen again?
 - What went poorly and how can we avoid that in the future?
 - How can we improve the next sprint?

Lifecycle of Scrum

