The Scrum Framework

A.Calvagna

Scrum Principles

- Scrum focuses on **teamwork**
  - It has rules for iteration planning and progress reviews
- The Framework does not tell developers how to do their work
  - XP focuses on how to do the work
- Scrum è basato **sull'empirismo** e sul concetto di **lean (snello)**
- Recommendation: Use Scrum together with XP
  - Within the planning and review structure of Scrum, use XP for the work

81% of Agile Projects Use Scrum or a Scrum Hybrid

- 66% use Scrum by itself
  - How developers do their work is not specified
- 15% use a Scrum hybrid with another method
  - 9% ScrumBan (Scrum+Kanban)
  - 6% Scrum+XP

Structure Provided by Scrum

- Development is keyed to iterations, called **sprints**
- planning and review **events** during a sprint
  - **Daily Scrum** and **Sprint Review**
  - Sprints and events are **"time boxed"**; that is, they are time limited
- Scrum also specifies **roles** and **artifacts**
- Scrum pillars: **Transparency, Inspection, Adaptation**

The Scrum Team: Roles

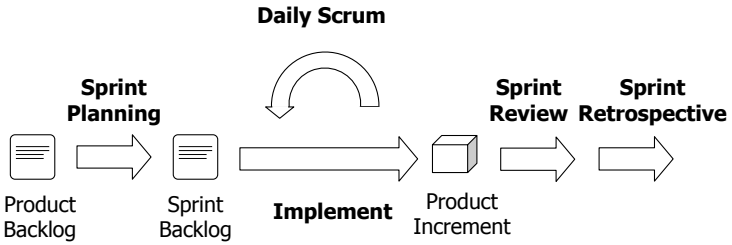
The **product owner** represents the voice of the customer
- Manages the **product backlog** of work items (requirements)
- Sets the **goal** for what to accomplish during a **sprint**
- The product owner is an individual, not a team
- In other contexts, the corresponding role is product manager

Developers choose how they implement work items
- Review events hold them accountable
- **deliver** potentially shippable product "**increments**" per sprint
- No one questions their **effort estimates** for work items

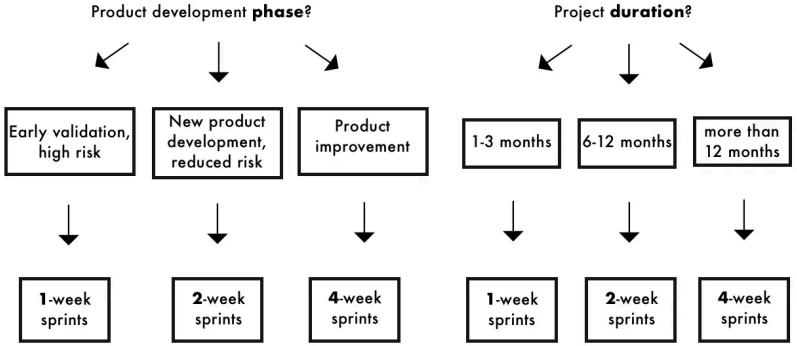
Scrum Master is a coach (servant-leader): un moderatore e aiuto
- **Organizes** and moderates all **events**
- Respectfully keeps events on track
- Additional responsibility during Daily Scrums
  - Takes responsibility for **removing external impediments**
- The Scrum Master is an individual, not a team

Overview of a Sprint: Events and Artifacts

Arrows show events, with one exception: Implement is not a Scrum event

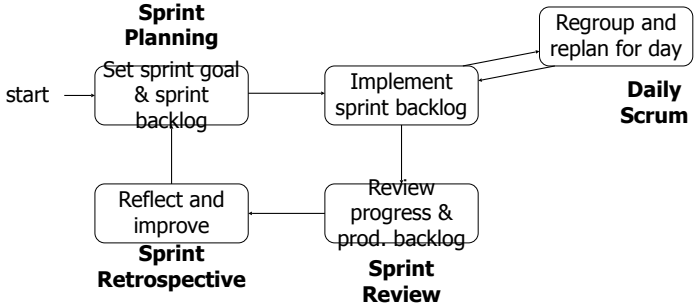


Optimal Sprint Duration



Activities Associated with Scrum Events

Activities are represented by boxes; events appear as labels next to activities



Events: Sprint Planning

- Attendees: The Scrum Team and users
- Max two hours/week * sprint weeks
- Led by product owner, **set a Sprint Goal**, which then cannot be changed
- Select a **Sprint Backlog** of work items to meet the goal
  - Users define the value/**priority** of backlog items
  - Developers estimate **effort** for work items and **what** will fit in a sprint

Events: Daily Scrum

- Attendees: Developers, with Scrum Master as moderator
- Max 15 min.
- Each developer answers 3 questions:
  - What did I do yesterday?
  - What will I do today?
  - Any obstacles?
- The Scrum Master takes responsibility for removing any external obstacles

Example: Removing External Obstacle

- Obstacle: Developers find a bug in a video subsystem from a supplier
  - The bug could delay the project
- The Scrum Master takes responsibility for removing the obstacle
  - The Scrum master works with the purchasing manager
  - Together, they contact the supplier
  - The contract is large, so the supplier expedites a fix

Events: Sprint Review

- Attendees: Scrum Team and users
- Review product increment for this sprint
- Update Product Backlog based on user feedback
- Max 4 hours for a four months project

Events: Sprint Retrospective

- Attendees: Scrum Team
  - Plan **process** improvements for the next Sprint
  - This event is not about the product
  - Max 3 hours for 4 months project

Scrum Artifacts

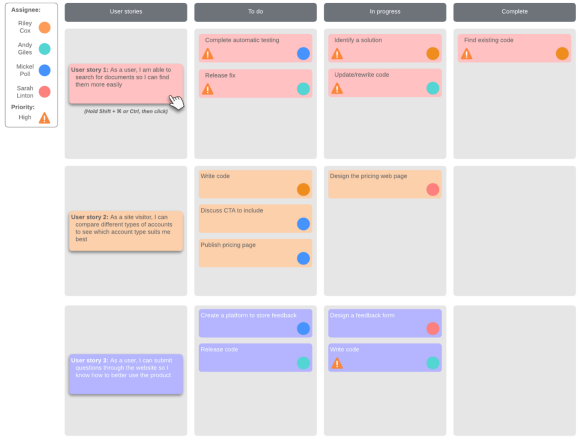
- **Product Backlog**: Ordered list of **all work items** that might be needed for a product
- **Sprint Backlog**: Items **selected** for meeting the Sprint Goal that is set during planning
- **Product Increment**: Sum of all Product Backlog items that are **completed** during a Sprint

For transparency of information, artifacts are open to all team members

Example Scrum Board

Shows current Sprint goals (user stories) and respective sprint backlog

Task move from left to right



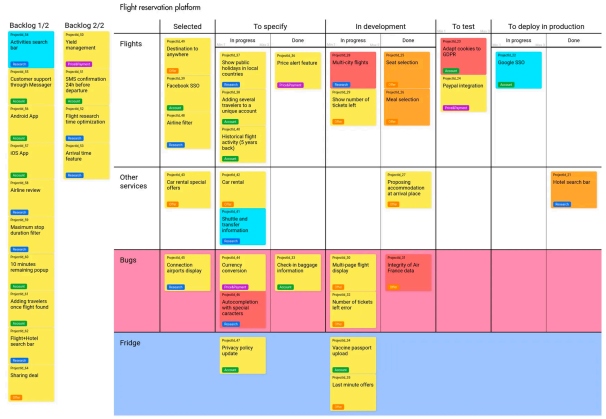
Example Kanban Board

visualize and track all work **in bird view**

rappresenta lo stato di avanzamento complessivo

Kanban boards are much more fluid and can be more easily adapted.

Kanban has no practice "rules"



What if the Sprint Backlog is Too Optimistic?

- Situation:
  - Developers overestimate what they can accomplish during a sprint
  - That is, they realize that they cannot implement the full Sprint Backlog
- Response:
  - Developers can **renegotiate** the Sprint Backlog with the Product Owner
  - The Sprint Goal cannot be changed

How Does Scrum Scale?

- Situation:
  - The development team is too large for a 15-minute Daily Scrum
  - With more than 7 developers, 15 minutes means less than 2 per person
- Response:
  - On approach is to **split** the team into subteams
  - Each subteam has its own Scrum Masters
  - The Scrum Masters from the subteams then have a Scrum of Scrums

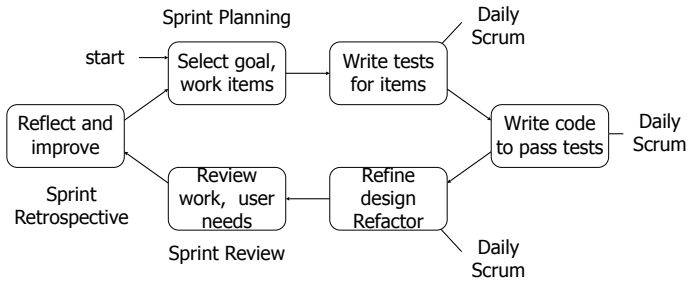
What if Some Features Don't Fit?

- Situation:
  - The entire Product Backlog cannot be done during the project schedule
- Response:
  - With any iterative project, the **highest priority** features during a sprint
  - **Inessential features** that don't fit within the schedule **can be dropped**
  - If essential features don't fit, **renegotiate** with users as early as possible
  - Avoid surprises by involving users in planning, from the start

Recommendations

- Use Scrum
- Adopt practices from XP – they fit seamlessly with Scrum
- Values drive XP's development practices
  - Kent Beck also created xUnit so there would be no excuse for not testing
- Early XP teams faced design risk
  - They did too little design too late

A Scrum/XP Hybrid Process



Recap

XP vs Scrum

- Shorter iterations
- Flexible with the changes
- Focus on technical practices
- Customer determines the order of feature development

- Longer sprints
- No changes within sprints
- Focus on managerial aspects
- Self-organized teams that decide what features to work on first