

# Agile Project Management with Scrum

---

## Resource links

---

- <http://www.agilealliance.org/>
- <http://www.agilemanifesto.org/>
- <http://www.scrum-master.com/>

## **Manifesto for Agile Software Development**

- Individuals and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

## **12 Principles behind the Agile Manifesto (1)**

- Our 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 to the shorter timescale.
- Business people and developers must work together daily throughout the project.

## 12 Principles behind the Agile Manifesto (2)

- Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.
- The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.
- Working software is the primary measure of progress.
- Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.

## 12 Principles behind the Agile Manifesto (3)

- Continuous attention to technical excellence and good design enhances agility.
- Simplicity--the art of maximizing the amount of work not done--is essential.
- The best architectures, requirements, and designs emerge from self-organizing teams.
- At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.



## Introduction of Scrum

---

- Scrum flow
- Scrum roles
- Scrum artifacts



## Sprint planning meeting

---

- Stake-holders to refine and re-prioritize the Product Backlog and Release Backlog and to choose the goals for the next iteration, usually droved by the highest business value and risk
- Scrum team and Product Owner meet to consider how to achieve the requests, and to create a sprint backlog of tasks to meet the goals

## Roles

---

- Product owner
- Scrum master
  - Responsible for the scrum process
    - Teaching
    - Implementing
    - Ensuring
- Scrum team

## Scrum Artifacts

---

- Product backlog
- Burndown chart
- Spring Backlog

## Product backlog

---

- Ever changing
- Prioritized list
- Owned by product owner
- Spread sheet example

## Spring Backlog

---

- The spring backlog defines the work, or tasks, that a team defines for turning the Product backlog it selects for that Spring into an increment of potentially shippable product functionality
- Task should be 4-16 hours each
- Highly visible, real-time picture of the work
- Owned by the team
- Maintained as spread sheet daily by a tracker or responsible individuals.

## Burndown chart

- Visualize the correlation between the amount of work remaining and the progress in reducing the work
  - X: date
  - Y: hours of work remaining
- Updated according the Sprint backlog

## What will we do?

- Assign the Role
  - Product owner
    - One from each team (pretending)
  - Scrum Master
    - Team coordinator
  - Scrum team
    - Everyone in your team
  - Others stakeholders
    - Instructor
    - End users

## What will we do?

- Artifacts
  - Product Backlog
    - Created and maintained by product owner
      - Do not change it if you are not the product owner
    - Available to the public to see
  - Spring backlog
    - Created and maintained by the team
    - Updated daily by a tractor or individuals
    - Available to the public
  - Burndown chart
    - Created every day and available to the public

## What will we do?

- Sprint Planning meeting
- Sprint time
  - 3 weeks for each sprint to fit our schedule (not 30 days)
  - 3 sprints
- Demo at the end of each sprint
- Daily standup meeting
  - What have you done since last meeting?
  - What will you do before the next meeting?
  - What is blocking you?

## What will we do?

- So what about requirement analysis and Design and deliverables/documentation?
  - Spend more time to do requirement analysis design in the first sprint, a little less in second one, and even maybe less in the third one.
  - At the end of each sprint, standardized documents should be created for requirement and design

## What will we do?

- So you still do them, but incrementally
  - Start from the big picture
    - Requirement analysis
    - architecture design
    - high level class design
    - storage design
    - interface design
  - Then focus on the requirements that will be implemented in each sprint
    - Use case
    - GUI design
    - Detailed class design
    - Collaboration Design (UML diagram)
  - Then implementation the feature
    - Write the code
    - Test
    - Write User documentation



## Reminder

---

- Do not forget the goal of this class
  - This class is not just about writing a program
    - You should know some programming language by now.
  - By the end of the term, each student should be able to:
    - To be able to work in a team on a large software project.
    - To understand the basic steps of large software project development.
    - To be able to effectively analyze a programming problem
    - To be able to effectively formulate use cases and scenarios
    - To be able to create class, object, use case, interface, and state machine diagrams in UML notation.
    - To be able to effectively design a solution to a programming problem.
    - To be able to assess risks of large software project.



## Heads-up

---

- You will get first hand experience about the complexity working a software project in a team
- Hopefully You will be convinced that software engineering is critical to your professional development
- You will enjoy the professional collaboration with your team mates
- You may feel some documentation are not necessary for the project that you are working on.
  - But it's better that you and your team faithfully to take the time to do the exercises.

## Important Dates

- The first sprint timed-box (3 weeks)
  - starts Monday 9/20
  - ends Friday 10/8
- 10/11 demo

## Resource and Tools

- All team member: Read Scrum FAQ and read online resource that FAQ referenced
- Scrum Basics
- The team coordinator is the scrum master
  - Design the sprint backlog spread sheet, burn down chart
  - Update the daily (every other day) burn down chart

