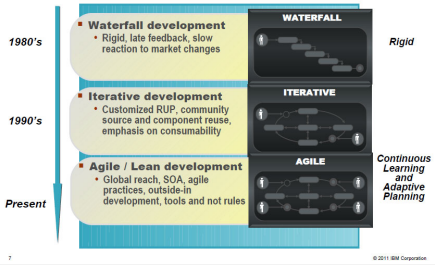
The Agile Approach

ISQA 8210

The Need for Agility

- Software developers realized that creating new software is a **high-risk** project when requirements **can not be fully known** when the project begins.
- A new approach was needed!
  - "It is not the strongest of the species that survive, nor the most intelligent, but the ones most responsive to change."
    - Charles Darwin – The Origin of Species
  - "Agility is the ability to both create and respond to change in order to profit in a turbulent business environment."
    - Jim Highsmith, Agile Software Development Ecosystems

IBM's Methodology Evolution



The Agile Manifesto

- Published in 2001 by an illustrious group of software developers, the Agile Manifesto stated 12 principles for better development practices
- Provided an alternative to documentation-driven, heavyweight software development processes
- Presented an approach that was more collaborative and responsive to a rapidly changing environment
- Agile software development practices were quickly adopted by organizations and developers

The Agile Manifesto

"We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:

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

That is, while there is value in the items on the right, we value the items on the left more."

The Agile Manifesto, 2001

Agile Development

- Agile development approach:
  - Collaborative approach
  - Multi-disciplinary development teams
  - Iterative application development
  - Short development cycles
  - Early customer feedback
  - Continuous improvement
- The objectives:
  - Reduce project risk
  - Reduce development time of overall product
  - Include adaptability throughout development process

Scrum Basics

- **Scrum** comes from a rugby strategy in which an out-of-play ball goes back into play quickly.
- In a Scrum, the team works together to focus on business priorities in a short "time box."
- The team engages in short sprints which provide incremental improvement of the software with each iteration.
- Scrum is **iterative development** – a continual process of evaluation, planning, requirements determination, analysis, design, coding, testing, and implementation.

Scrum Roles – The Stakeholders

- **Stakeholders:** Business individuals who will benefit from the product
  - Receive frequent feedback on product progress from the Product Owner
  - Control the product scope and may change requirements, add new requirements, or change priorities in the product backlog
  - Control the product schedule through the funding they provide

Scrum Roles – The Product Owner

- **Product Owner:** The one who fully understands the priorities of the business units – and how the agile team can address these
  - Liaison between the Stakeholders and the Scrum Master
  - Sets the software product requirements and priorities
    - What needs to be built now? What needs to be built next?
    - Balances risk and value of potential products
  - Prioritizes and manages user stories
  - Provides the team with unambiguous direction
  - Accepts or rejects work results
  - Ultimately responsible for the success of the project!

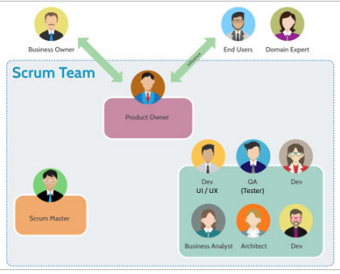
Scrum Roles – The Scrum Master

- **Scrum Master:** The protector of the team
  - Has an awesome ability to communicate and facilitate
  - Teaches the principles of scrum
  - Enforces the scrum rules, ensuring the team adheres to scrum processes
  - Removes obstacles and resolves dependencies among team members
  - Shields the team from outside influences, or non-essential information and meetings, which may inhibit progress

Scrum Roles – The Scrum/Agile Team

- **The Scrum Team:** Consists of a Scrum Master and team members from the IT and customer area
  - Is multi-disciplinary in order to complete the product
  - Is committed to developing a working system to solve a business need in a short period of time and incrementally adding functionality
  - Is known for accountability for their work, adaptability to changing requirements, and great teamwork, collaboration, and communication skills
  - Focus is on what the team as a whole can produce

Scrum Roles – The Scrum/Agile Team



Scrum Roles – The Scrum/Agile Team

- **The Team Room / Workspace**
  - All team members work in close proximity
  - All team members have access to whiteboards or walls on which ideas are gathered and progress is displayed
  - The work environment can be optimized to meet team needs
  - Includes access to individual privacy areas for "heads-down" work

The Agile Sequence

- Starting out: Discovery Session
  - Product Requirements Determination
- The Product Backlog
- The Sprint
  - Sprint Planning
- The Scrum
  - The Daily Stand-Up
- Measuring Sprint Progress
- Measuring Product Progress

Starting out: Discovery Session

- **Discovery Session:** Initial introduction to the project
  - Project Orientation
    - Business case overview
    - Business process analysis
    - Project scope & objectives
    - High-level Interface Ideas / Site Navigation
    - Initial product backlog
  - Process Orientation
    - Agile process training
    - Sprint/Release cycles
  - Team Orientation
    - Team room artifacts
    - Team norms
    - Technical environments (servers, lab equipment)
    - Work agreements

Sometimes called Sprint 0 or Setup Sprint

Product Requirements Determination

- Discovery Phase:
  - Understanding the Stakeholders and their business goals
    - Who are the users?
    - What are their needs?
    - How are we currently meeting those needs? How are we not?
  - Specify all product requirements and technical requirements
  - Determining the full scope of the project
  - Estimate the project timeline and development cost

Product Requirements Determination

- Discovery MVPs:

XB Software, 2021

Product Requirements Determination

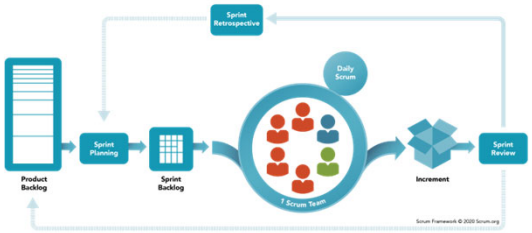
- Product Requirements Hierarchy:
  - **Theme:** a strategic initiative that connects development work to organizational goals
  - **Epic:** large quantities of work that are usually delivered across multiple releases
    - Major product versions or projects
  - **Features** are prominent services or functions of a system that deliver business value and fulfill customer needs
    - Shopping carts on e-commerce sites
    - Facial recognition on smartphones
  - **User Stories**

Product Requirements Determination

- Product requirements and features become **User Stories**
  - **User Stories:** functional units of work as defined by the users as informal, general explanations of a software function from an end user perspective. Also, the smallest unit of work in agile.
    - **Story Points** provide an estimate of the effort required to implement a user story, based on complexity, amount of work, and risk.
  - **Tasks** are specific pieces of technical work needed to complete a user story.

We'll look at building User Stories in detail soon!

The Project Takes Off

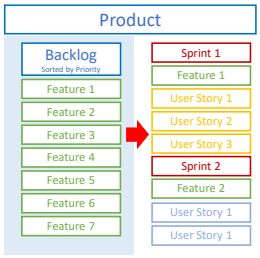


Product Backlog

- **Product Backlog:** list of all features and user stories for a product; all work to be done to create, maintain, and sustain a product.
  - Managed by the Product Owner
  - Tasks are sorted and prioritized based on business value
- **Product Goal:** The future state of the product which is the target for the Scrum Team. The Product Backlog moves toward the Product Goal

Product Backlog

- Following prioritization, the Product Owner assigns the product features to successive sprints (development iterations)
- Now the Agile Team can get to work!



The Sprint

- **Sprint:** a set period in which to complete a task, time-boxed to one month or less
  - **Sprint Goal:** the purpose of a sprint, usually a business problem to be dealt with
  - **Sprint Backlog:** list of user stories to be tackled in a specific sprint
  - **Sprint Planning:** one-day planning activity to start a sprint and prioritize the sprint backlog
    - What stories and tasks will be taken on?
    - What needs to be built first?
    - Is a buffer needed due to risk?
    - Results in **Sprint Plan**

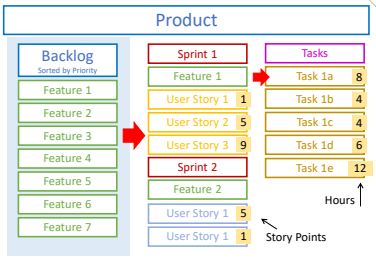
The Sprint

- Characteristics of successful sprints:
  - The goal is established and well understood by the team
  - Sprints are four weeks or fewer
  - All sprints must end on time
  - Sprint teams are not interrupted or controlled by other stakeholders
  - Sprints should complete everything they have committed to
  - Sprints may only be cancelled by the Product Owner or Scrum Master and only for valid business reasons

Sprint Planning

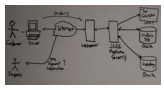
- Sizing Stories with Story Points into Tasks with Task Hours
- Team members commit to getting a set of tasks done!

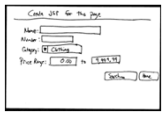
We'll look at estimating Story Points soon!



Sprint Planning

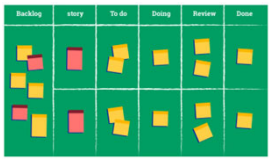
- Methods used to plan the sprint:
  - Discussions, interviews, brainstorming
  - User Stories & Use Cases
  - Prototyping/experimenting with a new process
  - Capturing current processes on video or using screen-capture software to avoid writing long descriptions
  - Design to-be processes iteratively
  - Use whiteboards and mock-ups rather than formal diagramming tools





The Scrum

- **Scrum:** A daily routine that focuses on product delivery
  - **Daily Scrum:** daily event during which developers review and plan
  - **Scrum Board:** physical board which visualizes team progress and shows the sprint backlog



The Daily Scrum / Daily Stand-Up

- Short daily morning meeting that discloses project work details and facilitates communication and teamwork – 15-minute limit!
- Often conducted standing up to speed up the meeting
- Each team member answers 3 questions:
  - What did I do (complete) yesterday?
  - What am I planning to do (complete) today?
  - What impediments may block my success today?

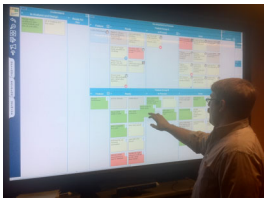
The Daily Scrum / Daily Stand-Up

- **Impediments**
  - Anything that is holding up a team member from completing a task
  - Impediments are exposed by team members during the daily stand-up
  - Impediments need to be identified and removed quickly
  - Scrum Master should take the lead on eliminating impediments
  - Agile puts a 24-hour rule on impediment removal; if not done, it escalates one level up

Team Member -> Scrum Master -> Product Owner -> Sponsor

The Scrum / Kanban Board

- Scrum Boards or Kanban Boards track the progress of the Scrum Team following each daily standup meeting



The "Definition of Done"

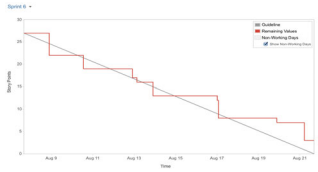
- A formal description of the state of the increment when it meets the quality measures required for the product
  - A list of acceptance criteria that must be met by a product increment
  - Included in user stories as conditions of satisfaction
  - Having explicit criteria reduces misunderstandings and conflicts with the product owner or stakeholders
  - When a Product Backlog item meets this definition, the increment is complete and can be released. If it does not meet this definition, it cannot be released or even presented at the Sprint Review.

Measuring Sprint Progress

- **Increment:** the complete and valuable work produced by the developers during a sprint. Multiple increments (over multiple sprints) form a complete product
- **Sprint Review/Demo:** demonstration of the product to the Product Owner at the end of a sprint and review of progress towards product goal
  - Early sprints show designs or artifacts
  - Product is matched to user story
  - Team provides feedback

Measuring Sprint Progress

- **Sprint Burndown:** measures the completion of story points during a sprint



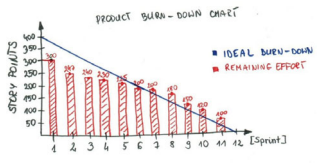
Sprint Burndown Chart

Measuring Sprint Progress

- **Sprint Retrospective/Heartbeat Retrospective:** review of completed sprint and planning of work for future sprints
  - All team members reflect on the past sprint
  - What do we want to start doing, stop doing, and continue doing?

Measuring Product Progress

- **Release Burndown:** measure of the entire product backlog of items over the course of the development process and release plan
- **Burndown Chart:** chart which shows the amount of work remaining in the backlog, often measured in user stories completed



stxnext.com

Agile Benefits

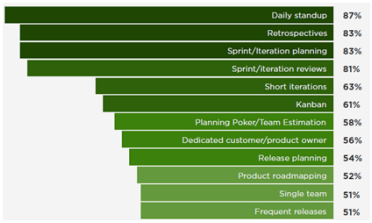
- Great for higher-risk projects where not all requirements are known at the start of the project
- Promotes rapid delivery of value to customers
- Provides timely and regular visibility of the solution to customers, product owners, and stakeholders
- Delivers increases in productivity, quality, and ROI for organizations

Why do Project Teams like Agile?

- Agile builds empowered, motivated, and self-organizing teams
- Teams focus on delivering measurable results
- Clear expectations are established and communicated
- Direct customer/user communication and feedback
- Clear definition of what "done" means for each deliverable
- Cost & time savings due to waste elimination and efficiency
- Feeling of accomplishment and recognition

Why do Project Teams like Agile?

- Agile techniques and practices used in organizations:



digital.ai, 2021

Why do Customers / Users like Agile?

- A piece of working software is delivered with each sprint
  - Features are moved into production incrementally
- Team works on highest-priority work first
  - Faster delivery of business value (time-to-market)
- Collaboration with team results in usable and valuable software
- Agile is more flexible and can better accommodate business need changes than the waterfall methodology
- Users are empowered to make decisions and influence outcomes
- Project progress is highly visible
- Risk is highlighted earlier in the project and better managed

Agile Challenges

- Challenges faced by organizations as they introduce agile techniques, practices, and tools:
  - Inconsistent processes and practices across teams
  - Organizational culture does not match agile values
  - General organizational resistance to change
  - Lack of skills and experience with agile methods
  - Insufficient leadership participation
  - Inadequate management support and sponsorship
- Many of these challenges have been experienced for years

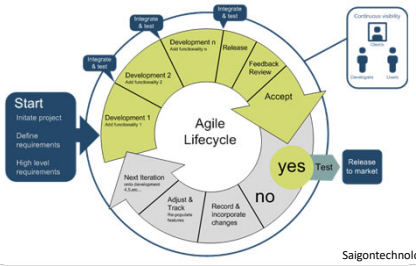
digital.ai, 2021

Agile Processes in Organizations

- Different organizations have implemented agile in different ways and have over time developed their own best practices
- Agile terminology may differ across organizations
- Agile principles and values remain in effect

Later on this semester, we'll examine the Scaled Agile Framework for use in larger organizations.

The Big Picture of Agile/Scrum



Saigontechnology.com, 2022