AGILE & SCRUM REVIEW

What is Scrum and Agile?

Agile refers to a set of "methods and practices based on the values and principles expressed in the Agile Manifesto," which includes things like collaboration, self-organization, and cross functionality of teams. *Scrum* is a framework that is used to implement Agile development.

History of Agile:

- Created in 2001
- 17 persons involved
- 4 values 12 principles

Values of Agile:

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

12 Principles of Agile:

- Highest priority is to satisfy the customer through early & continuous delivery of valuable software
- Welcome changing requirements even late in development. (Harnesses change for customer's competitive advantage)
- Deliver working software frequently, couple of weeks to couple of months
- Business team & developers must work together DAILY throughout the project
- Build projects around motivated individuals
- Most efficient & effective method of communication between development team is FACE-TO-FACE conversation
- Working software is the primary measure of progress
- Agile processes promote sustainable developments
- Continuous attention to technical excellence & good design enhances agility
- Simplicity – the art of maximizing the amount of work not done – is essential
- Best architectures, requirement design emerge from self-organising teams
- At regular intervals, the team reflects on how to become effective, then tunes & adjusts accordingly

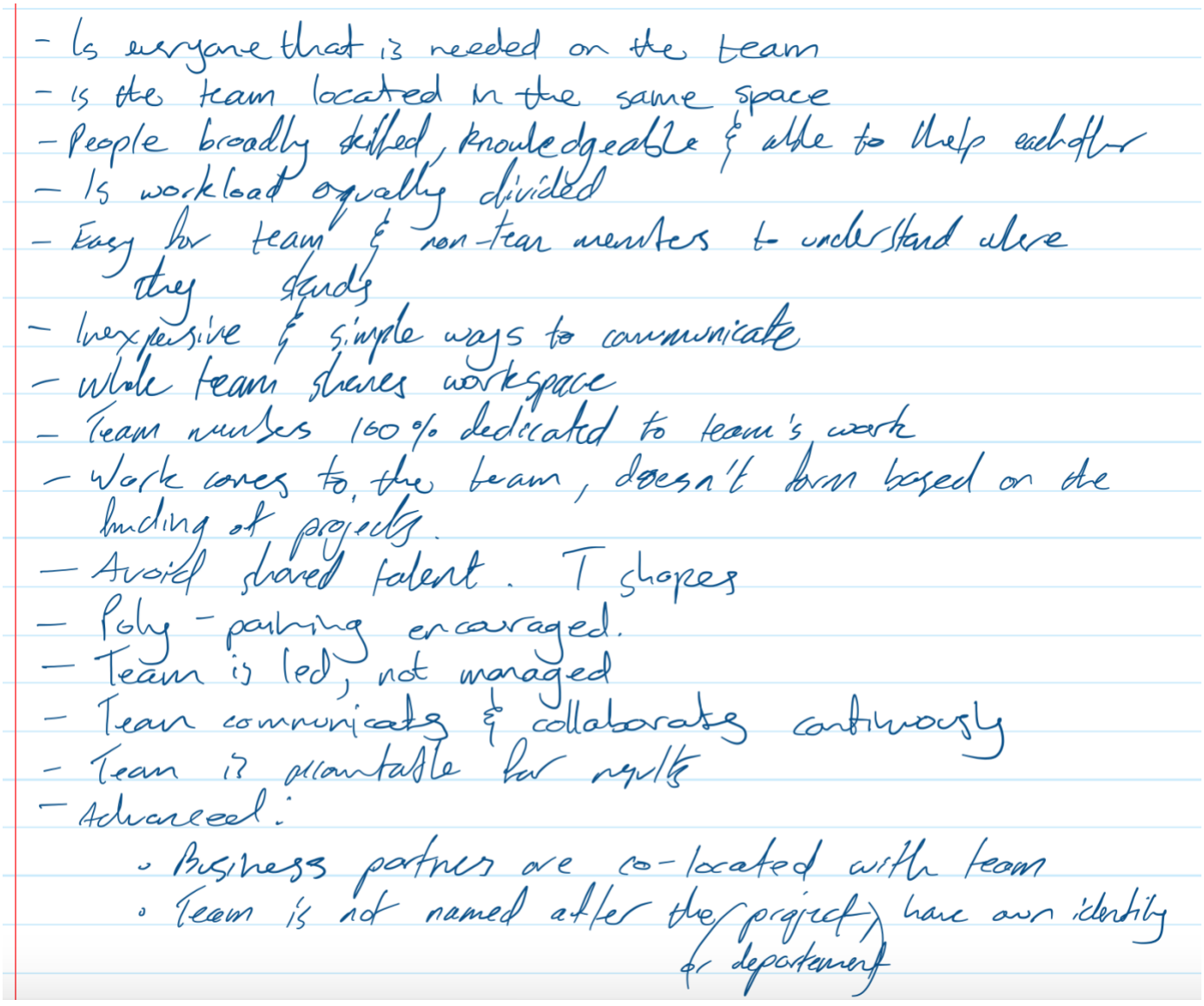
advantages and disadvantages of agile compared to the waterfall development method

- Advantages of Agile compared to Waterfall
    - Easier to change, extend and improve as the product is delivered in small iterations rather than the whole final product like in waterfall
    - Product is reliable & delivered on time as bugs and failures are caught early in process due to Agile's implementation for the Fail-Fast methodology. Unlike waterfall where all the testing is done towards the end of development where bugs and failures might cause the whole project to be redone from the start
    - Referring to the above point, Agile is then cost effective

- Final product fulfills the needs of the customer as changes can be made by the customer during the Agile process where waterfall does not allow for that.

- Disadvantages:
  - With a less successful project manager, the project can become a series of code sprints. If this happens, the project is likely to come in late and over budget.
  - As the initial project doesn't have a definitive plan, the final product can be grossly different than what was initially intended.

Essentials:

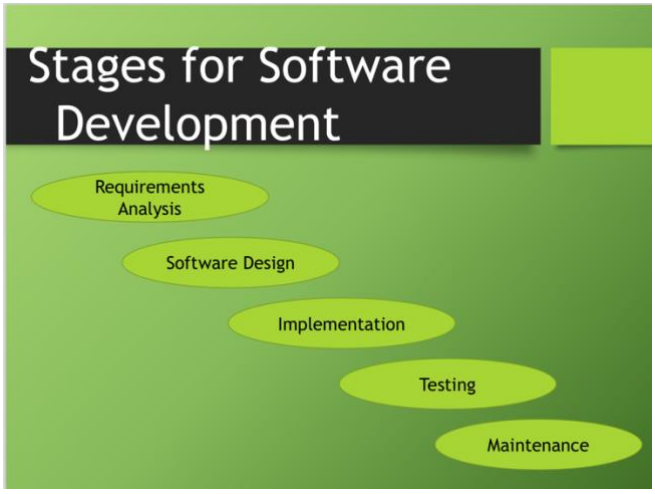


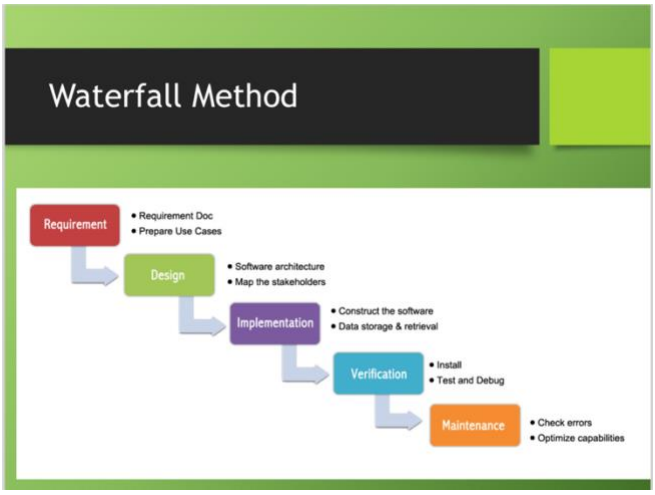
Commons Issues of Software:

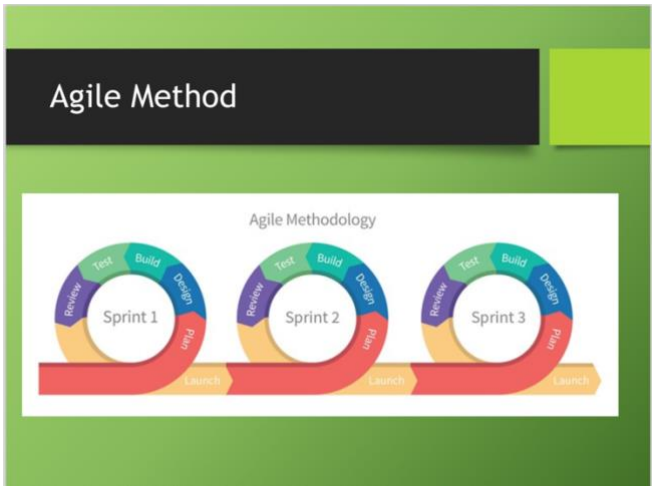
- Final software doesn't fulfill the needs of the customer
- Hard to extend & improve
- Bad documentation
- Bad quality: errors, bugs, difficult to use
- More time & costs than expected

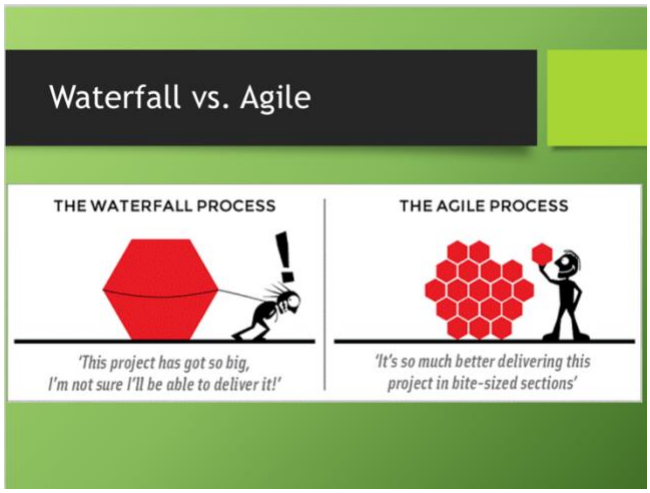
Solution: Software Engineering:

- The study & application of methodologies to develop quality software that fulfills customer needs
- Objective
    - On time
    - Reliable: doesn't crash
    - Complete: documentation & customer needs fulfilled



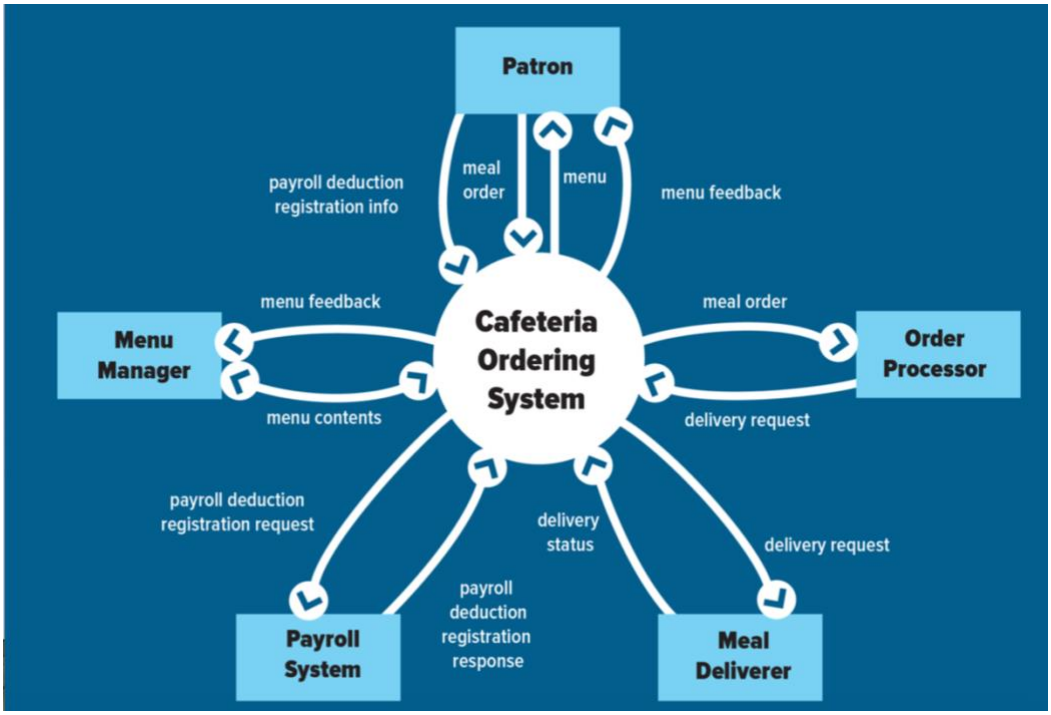


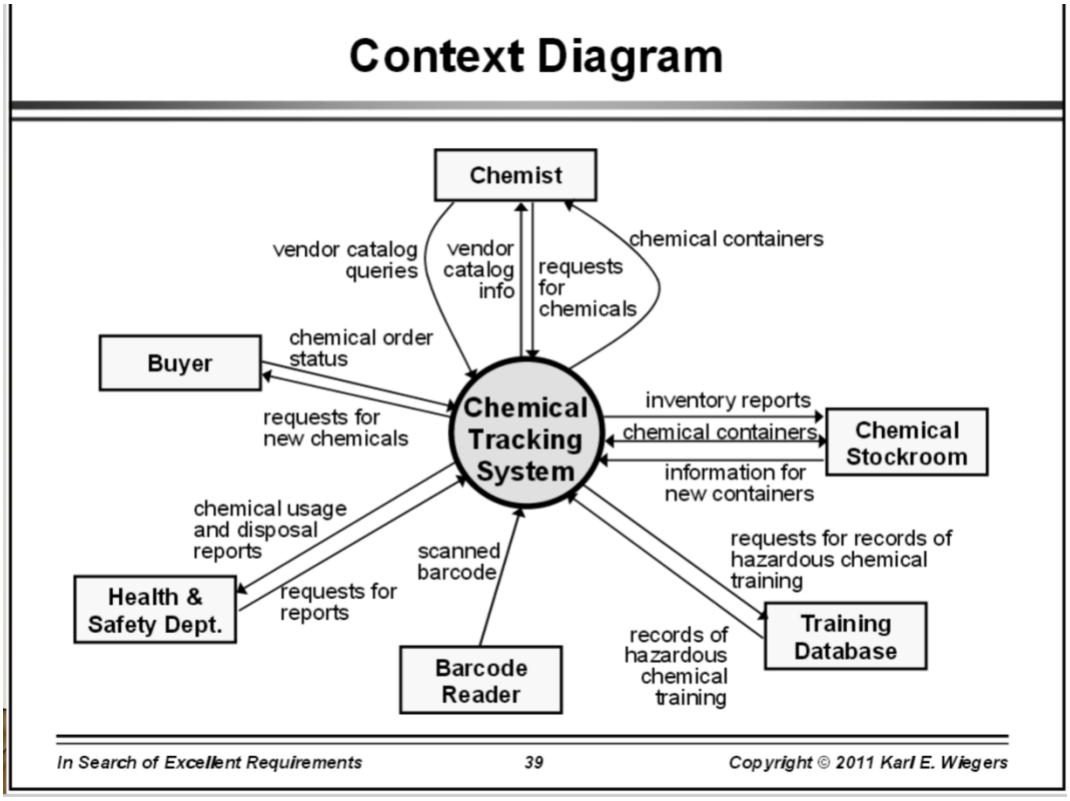




Requirments Gathering:

- A system Context diagram:
  - Is a diagram that defines the boundary between the system, or part of a system, & it's environment showing entities that interact with it.



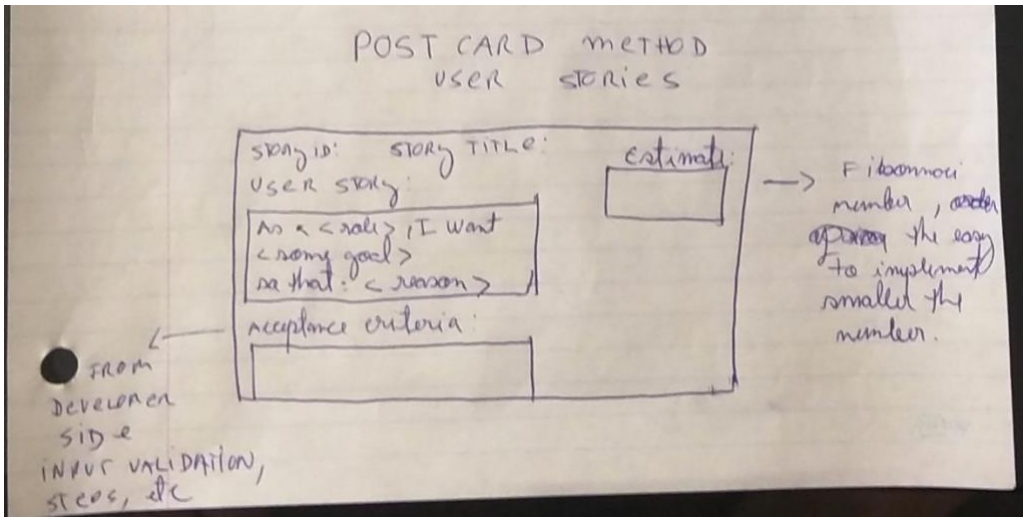


User Stories:

- A User Story is a tool used in Agile software development to capture a description of a software feature from an end-user perspective
- The User Story describes the type of user, what they want & why. A user story helps to create a simplified description of a requirement
- Combines strengths of verbal and written requirements
- Agile requirements are barely sufficient – don't get bogged down in the details.
- Eg.
    - I am a job seeker, I want to search for jobs
    - I am a customer, I want to order a coffee
    - I am an app user, I want to upload a photo
- Eg1.
    - As a
        - Registered user
    - I want to
        - Change my password
    - So I can
        - Keep my account secure
- Eg2.
    - As an
        - Admin user
    - I want to
        - Disable a user
    - So I can
        - Prevent unauthorized logins by past employees
- Eg3.
    - As a
        - Website visitor
    - I want to
        - Subscribe to the mailing list for a product
    - So I can
        - Get product updates through email
- Eg4.
    - As a
        - Mobile app user
    - I want to
        - Save all my data to the cloud
    - So I can
        - Access it from another device

- Advantages
    - Gives clarity as to why a feature is useful
    - Can influence how a feature should function
    - Can give you ideas for other useful feature that support user's goals
    - Can help to see simpler ways of achieving the same goal
- Who, what & why

User Story PostCard:



INVEST

- Independent – small as possible, not dependent on other stories
- Negotiable – not fixed in stone, not a contract
- Valuable – have value to the user or the owner of the system
- Estimable – know what it takes to build it
- Small – not too small but not big
- Testable – can be tested – tests should be developed before development of the prototype

SCRUM

Scrum Roles:

Product Owner:
- Responsible for prioritising work on the product backlog
  - Priority of list is the order of the list
- Know the requirements of the product
- Accountable for the product backlog

Scrum Master:
- Responsible for ensuring that the team follows the Scrum ceremonies and effectively use the Scrum artifacts
- Responsible for supporting the Scrum team by coaching & guiding them through the Scrum process and removing any impediments
- Protector of the team

Team Member:
- Responsible for building the actual product increment
- Are Cross-functional, self-organizing and accountable as a team for the product increment
- Have 3 – 9 members

Scrum Artifacts:

- Product Backlog
- Sprint Backlog
- Product Increment
- Goals:
  - Maximize transparency
  - Promote a shared understanding of the work
- **Product Backlog:**
  - Is an ordered list of everything that might be needed in the product
  - Makes priorities transparent to all stakeholders
  - Development team works on product backlog by descending order of priority (order of the list is the priority of the list)
  - Single PO is responsible for the product backlog –
    - Adding or removing items
    - Reordering in terms of priority
  - Items have a(n)
    - Description
    - Estimate
    - Value
  - Constant evolving to changing requirements, probably by:
    - Feedback on a completed product increment during a sprint review meeting
    - Feedback and ideas from current and potential users and customers
    - Ideas from internal stake holders
    - Emerging competitive and technical opportunities

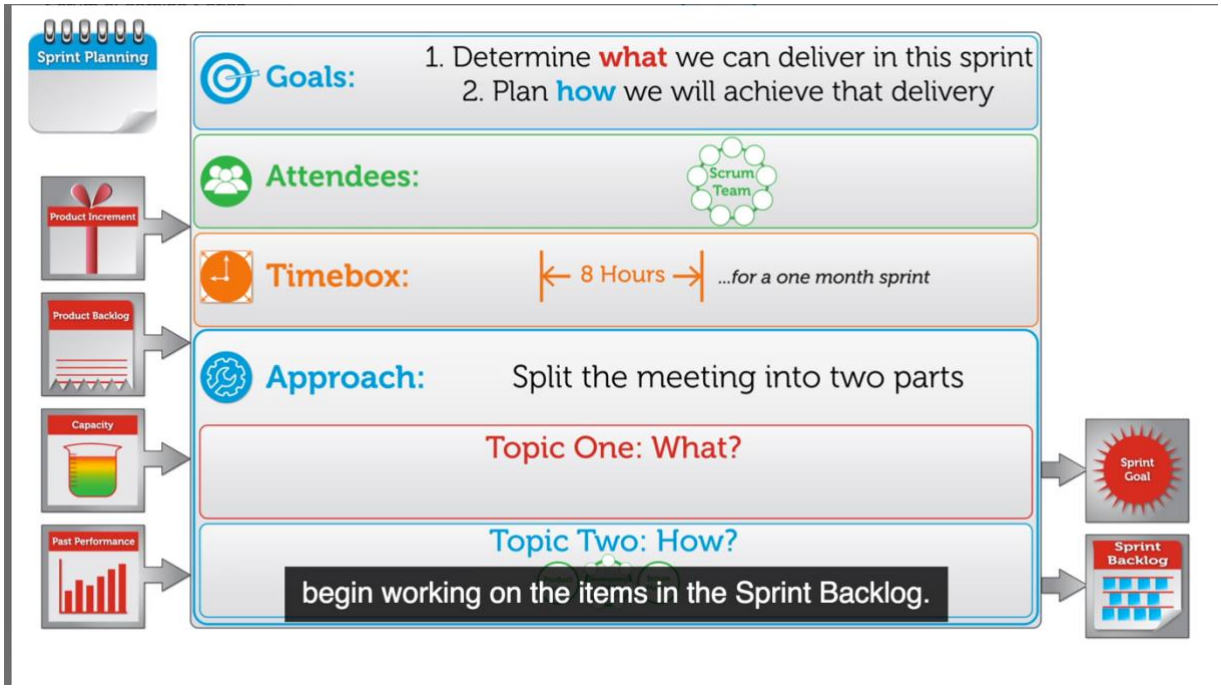
- **Sprint Backlog:**
    - The Sprint backlog is a highly visible, real-time picture of the work that the Development Team plans to accomplish during the sprint
    - Collaboratively created during Sprint Planning by the development team
        - Sprint goal
        - Sprint Backlog
    - During Sprint planning. The Development Team pulls the highest ordered item from the Product Backlog into the Sprint Backlog
    - Contains 2 components
        - The What
            - Pull as many items that can be done within the sprint timebox
        - The How
            - The plan, Daily scrums
    - Sprint modifications include:
        - Adding newly discovered work
        - Updating estimates
        - Removing unnecessary elements
        - Only Developments team can make changes to the Sprint Backlog

- **Product Increment**
    - In every Sprint and increment to product is delivered
    - Each increment is a new, updated usable version of the product
    - The increment is considered "Done" if it can be immediately released without any additional work
    - Each increment is thoroughly tested

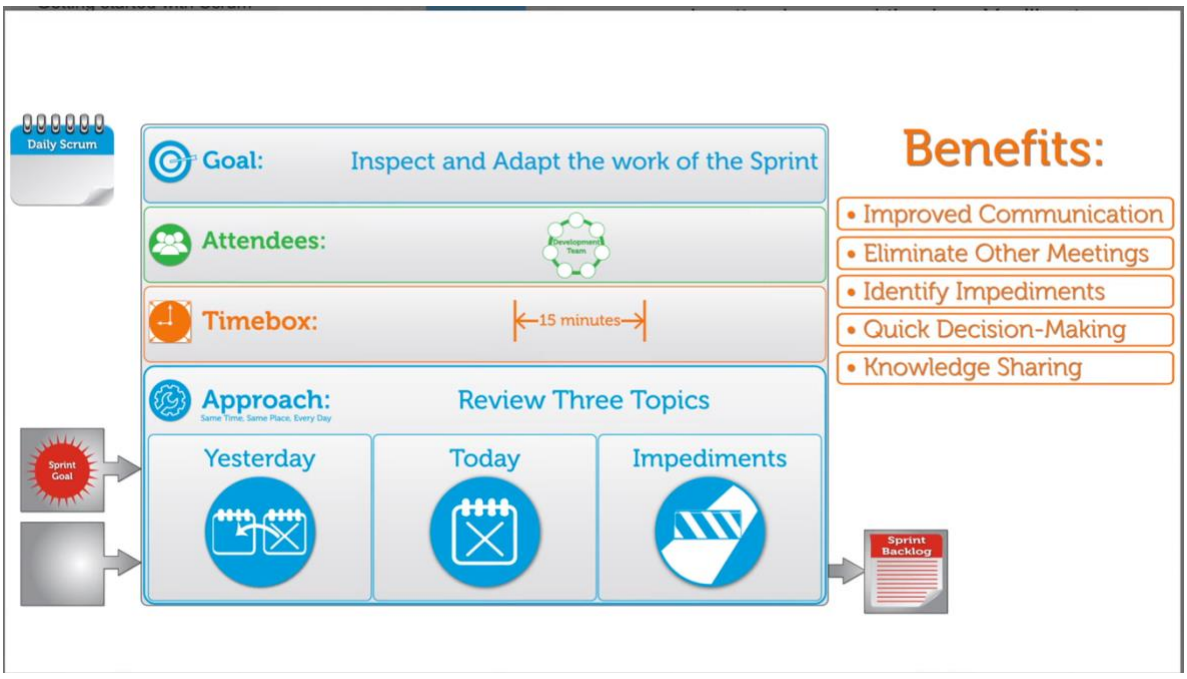
<u>Scrum Ceremonies</u>

- Sprint Planning
- Daily Scrum
- Sprint & Retrospective
- At the beginning of the sprint, the Scrum Team comes together in the Sprint Planning meeting to assess which items from the top of the Product Backlog they can pull into the sprint. The team crafts a shared Sprint goal. The Sprint Backlog is then created. Once Sprint Planning is complete, Development Team begins their development work which continues to end of the sprint. Each day the team meets together for a Daily Scrum Meeting to inspect Sprint Plan and make any adaptation necessary. When sprint is complete, a releasable product increment is delivered. The increment is inspected at the Sprint Review Meeting and adapt future plans for the product by updating the Product Backlog. Finally, the Scrum Team holds a Sprint Retrospective to inspect the system of work itself, and make adaptations to be more effective in future Sprints.
- Maximum Sprint length is one calendar month
- Usually 2 – week sprints are also common
- No new work can be pushed into the Sprint
- Product owner may choose to cancel a Sprint if
    - Sprint Goal becomes obsolete
    - Market conditions change
    - Company makes a major strategy change

- **Sprint Planning**
  - Used to plan the work of the Sprint
  - Is the first event within the Sprint
  - Goals:
    - Determine **what** can be delivered in the current sprint
    - Plan **how** it will achieve that delivery
  - Attendees:
    - Scrum Team
    - Product owner
  - Time box:
    - All scrum Events are timeboxed
    - Sense of Urgency
    - Focus on Highest Priority Discussions
    - 8 hrs. for a one month sprint
  - Approach
    - What
    - How
  - The 4 inputs are:
    - Latest Product Increment
    - Current Product Backlog
    - Projected capacity of the Development Team
    - Past Performance of the Development Team
  - The 2 outputs are:
    - Sprint Goal
    - Sprint Backlog

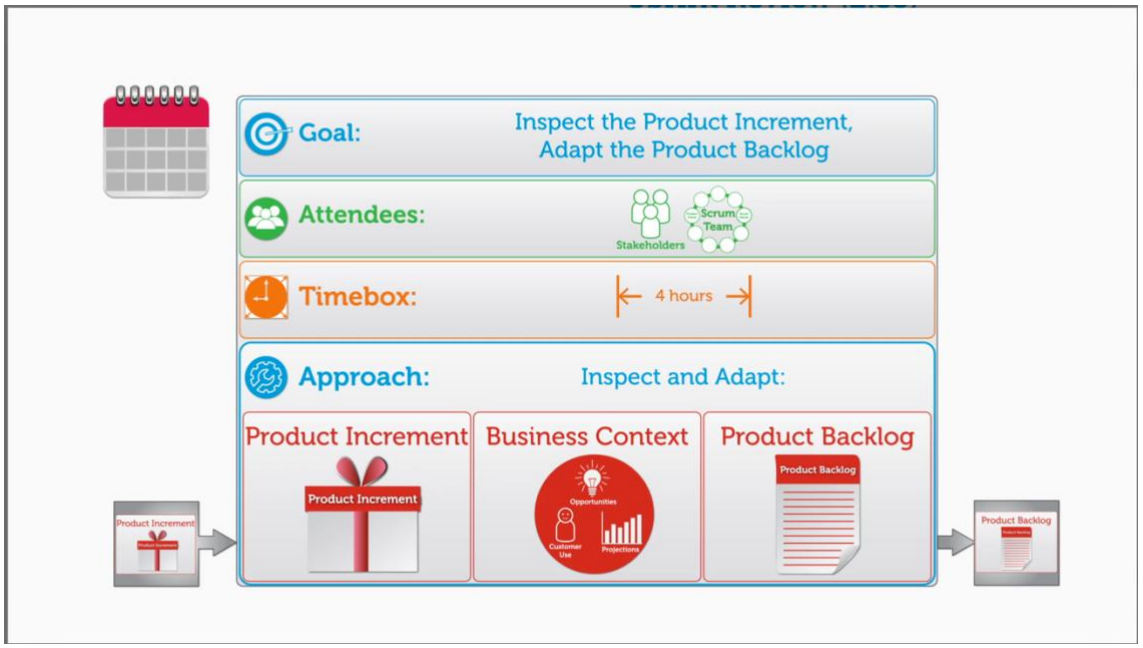


- **Daily Scrum**
  - Once a day the Development Team holds the Daily Scrum meeting
  - Goals:
    - To inspect and adapt the ongoing work of the sprint.
  - Attendees:
    - Development Team
    - Scrum Master
  - Time box:
    - 15 Mins
  - Approach
    - Same time, same place, Every day
    - Review Three Topics:
      - What was done yesterday to reach Sprint goal
      - What was done today to reach Sprint goal
      - Impediments in the way of sprint goal that effects a member or team
  - The 2 inputs are:
    - Sprint Goal
    - Sprint Backlog
  - The 1 outputs are:
    - Sprint Backlog
  - Benefits:
    - Improved Communication
    - Eliminate other meetings
    - Identify Impediments
    - Quick Decision Making
    - Knowledge Sharing



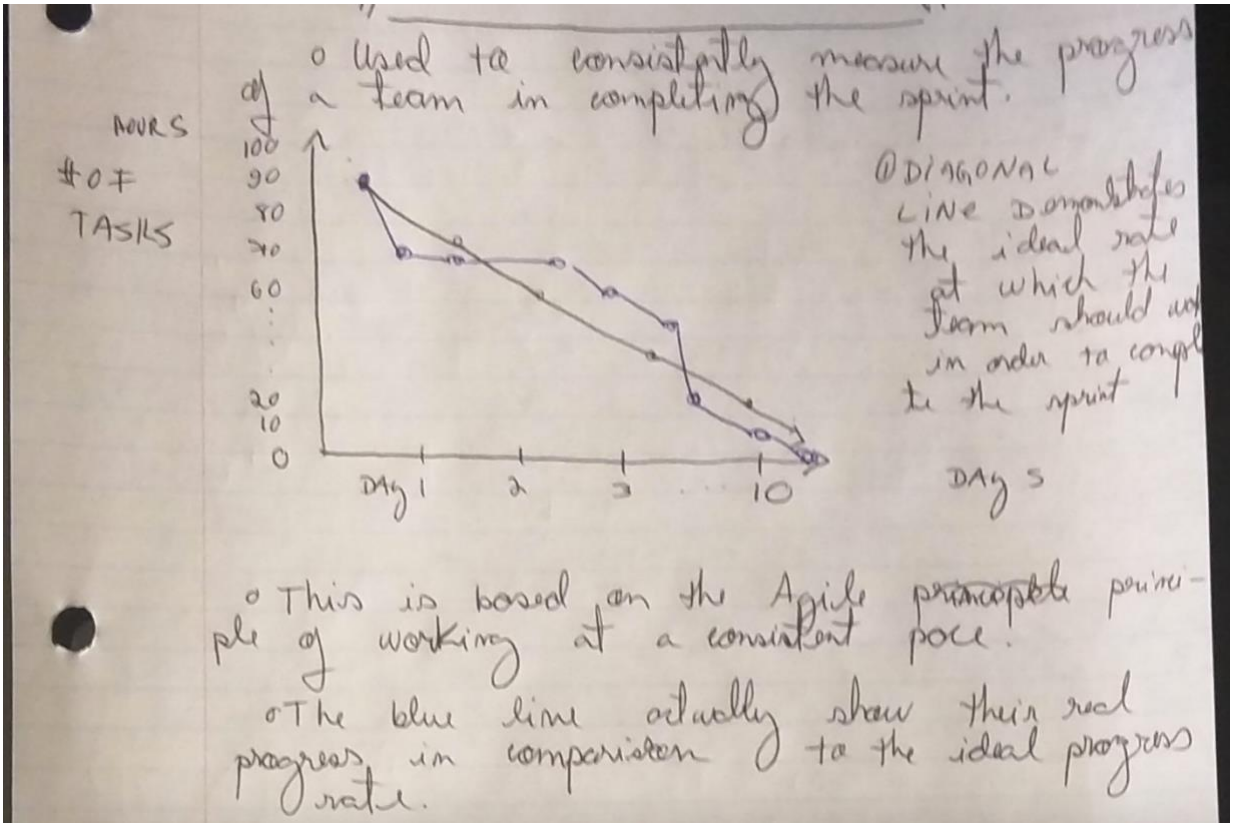
SPRINT

- **Sprint Review**
  - The Sprint Review meeting is held at the end of the Sprint
  - Goals:
    - To inspect the product Increment
    - Adapt Product Backlog as needed
  - Attendees:
    - Scrum Team(Scrum Master, PO, Development team)
    - Stakeholders
  - Time box:
    - 4 hrs. for a one-month Sprint
  - Approach
    - Product Increment is demonstrated
      - Q&A's
    - Business Context
      - New developments in market place
      - Feedback from users
      - Projected release dates
      - Other relevant details
    - Product Backlog is adapted
  - The 1 inputs are:
    - Increment developed during the Sprint
  - The 1 outputs are:
    - Revised Product Backlog



- **Sprint Retrospective**
    - Final event (ceremony) of the sprint
    - Enables Scrum Team to continuously improve their approach to their work Sprint after Sprint
    - Goals:
        - To inspect and adapt the Scrum Team's System of Work
    - Attendees:
        - All members of Scrum Team attend
            - Scrum master
            - Product Owner
            - Development Team
    - Time box:
        - 3 hrs. for a one-month Sprint
    - Approach
        - Inspect how the work was completed in the last Sprint
            - People
            - Relationships
            - Processes
            - Tools
        - Identify and order major items that went well, and opportunities for improvement
        - The team creates a plan for how they will roll out any chosen improvements

<u>Burndown Chart</u>



Planning Poker

- Use Fibonacci numbers to estimate the priorities of the task to be done with respect to difficulty
- Smaller the number means easiest and quick to implement, Higher is vice versa

Fail Fast Principle

- Works with the idea of failing quickly to fix early on in development, rather than later as to deliver a working product rather than a hindered on due to errors occurring near the end of development

Agile vs Waterfall

- Agile provided the flexibility for a changing product requirements as they occur
- Errors and bugs are handled during the development cycle rather than later on when delivry of the product is more important

Scrum vs XP

- Scrum
    - Agile management methodology
- XP
    - Agile engineering methodology

In Scrum, Things that are decided as a team

- Sprint planning
- Review & retrospective
- The sprint itself when the team is empowered to make decisions on it's own