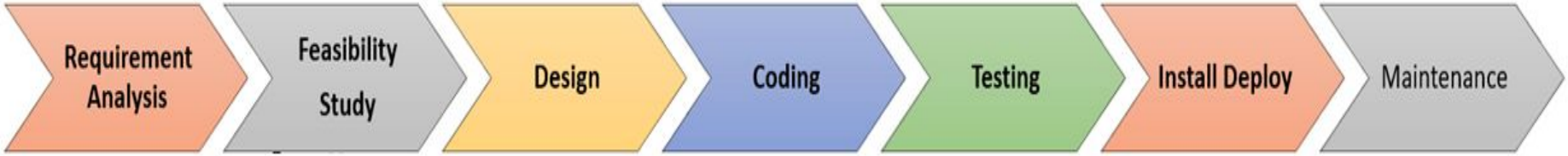
Agile Process & Methodologies

Talking Points

- SDLC
- Why do we need Agile?
- Introduction to Agile
- Scrum
- Q & A

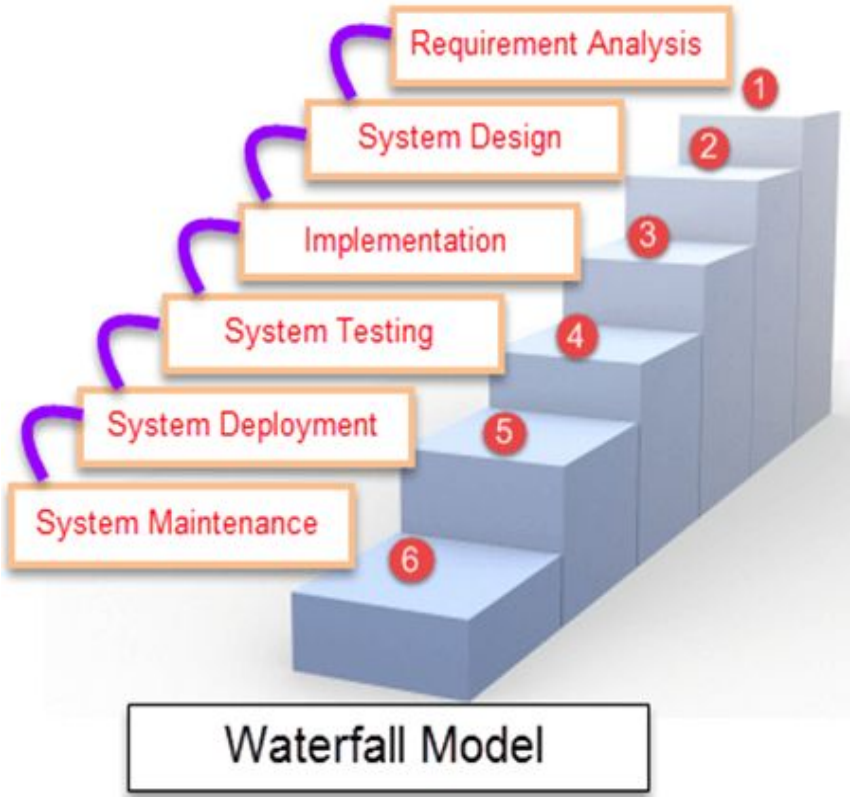
SDLC - Overview

The Software Development Life Cycle is a systematic process for building software that ensures the quality and correctness of the software built. SDLC process aims to produce high-quality software which meets customer expectations. The software development should be complete in the pre-defined time frame and cost.

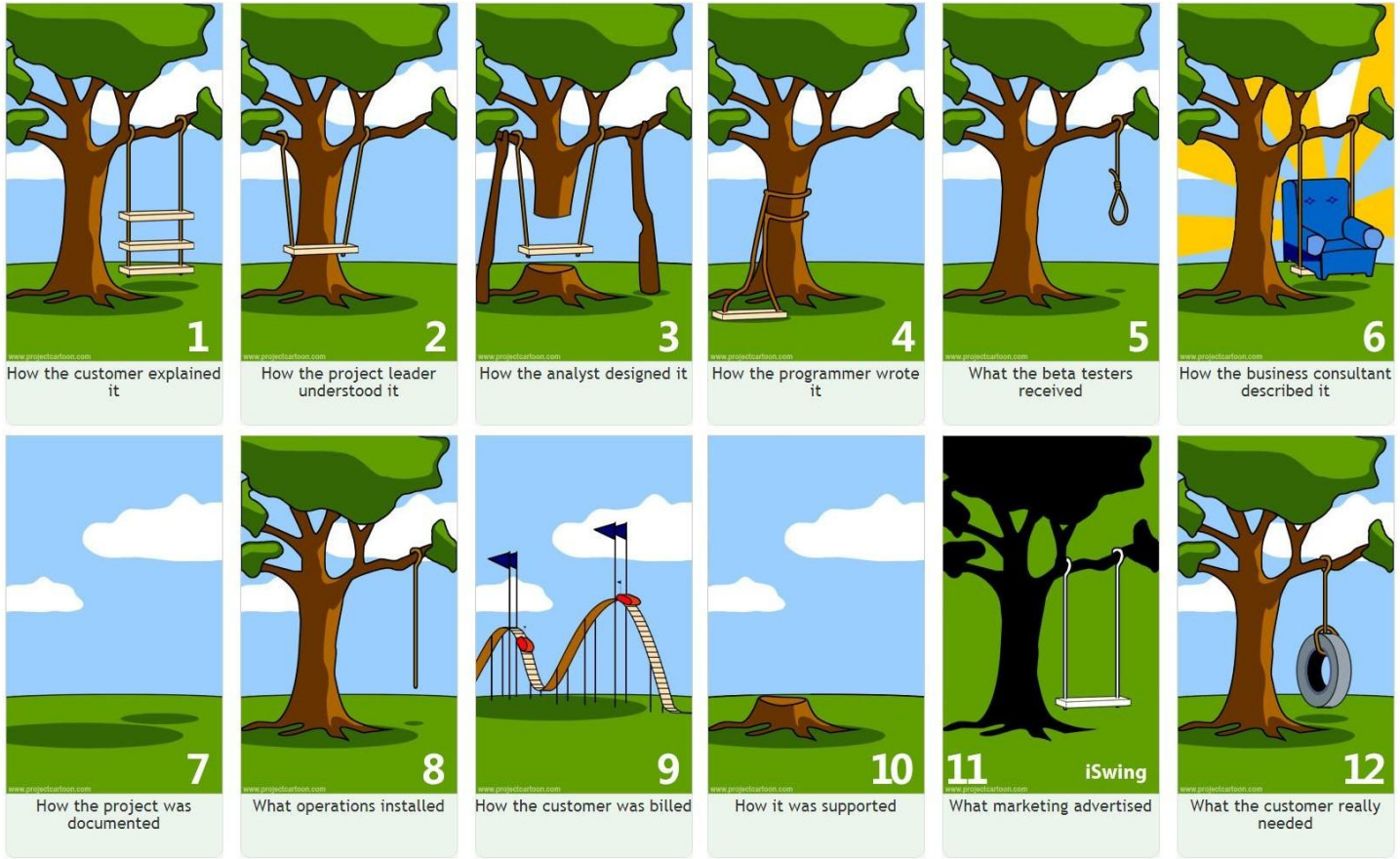


SDLC - Waterfall Model

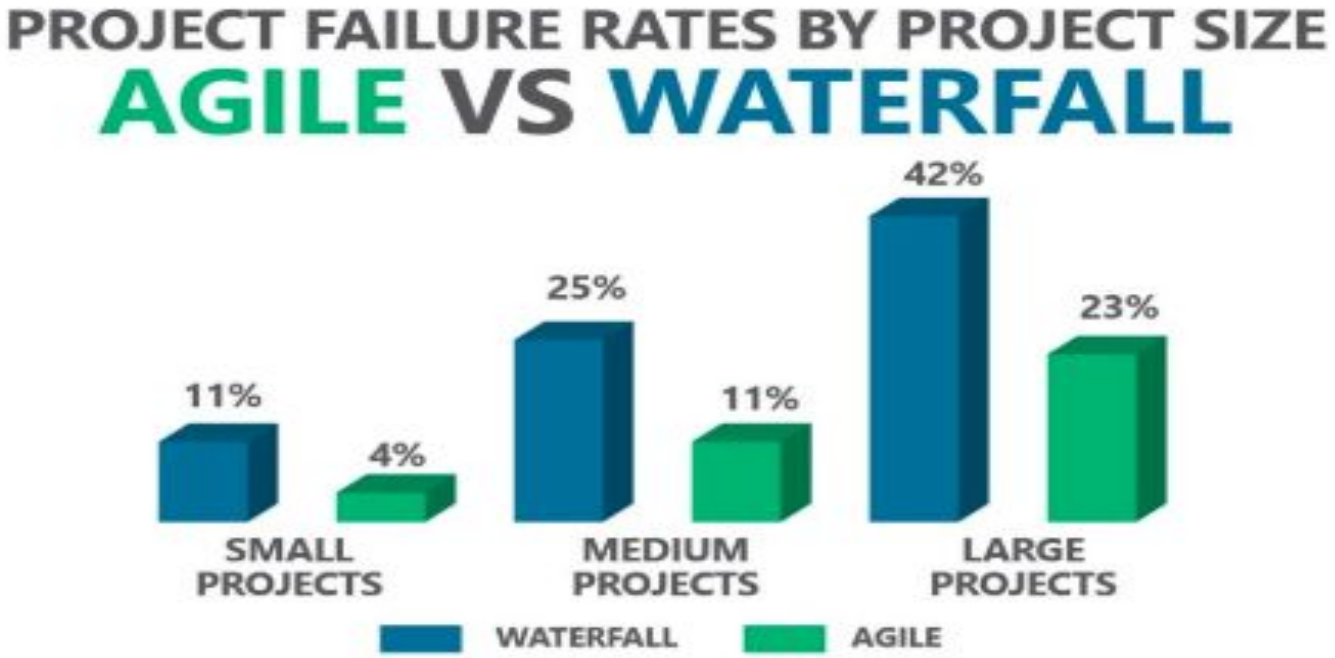
Waterfall Model is a sequential model that divides software development into different phases. Each phase is designed for performing specific activity during SDLC phase.



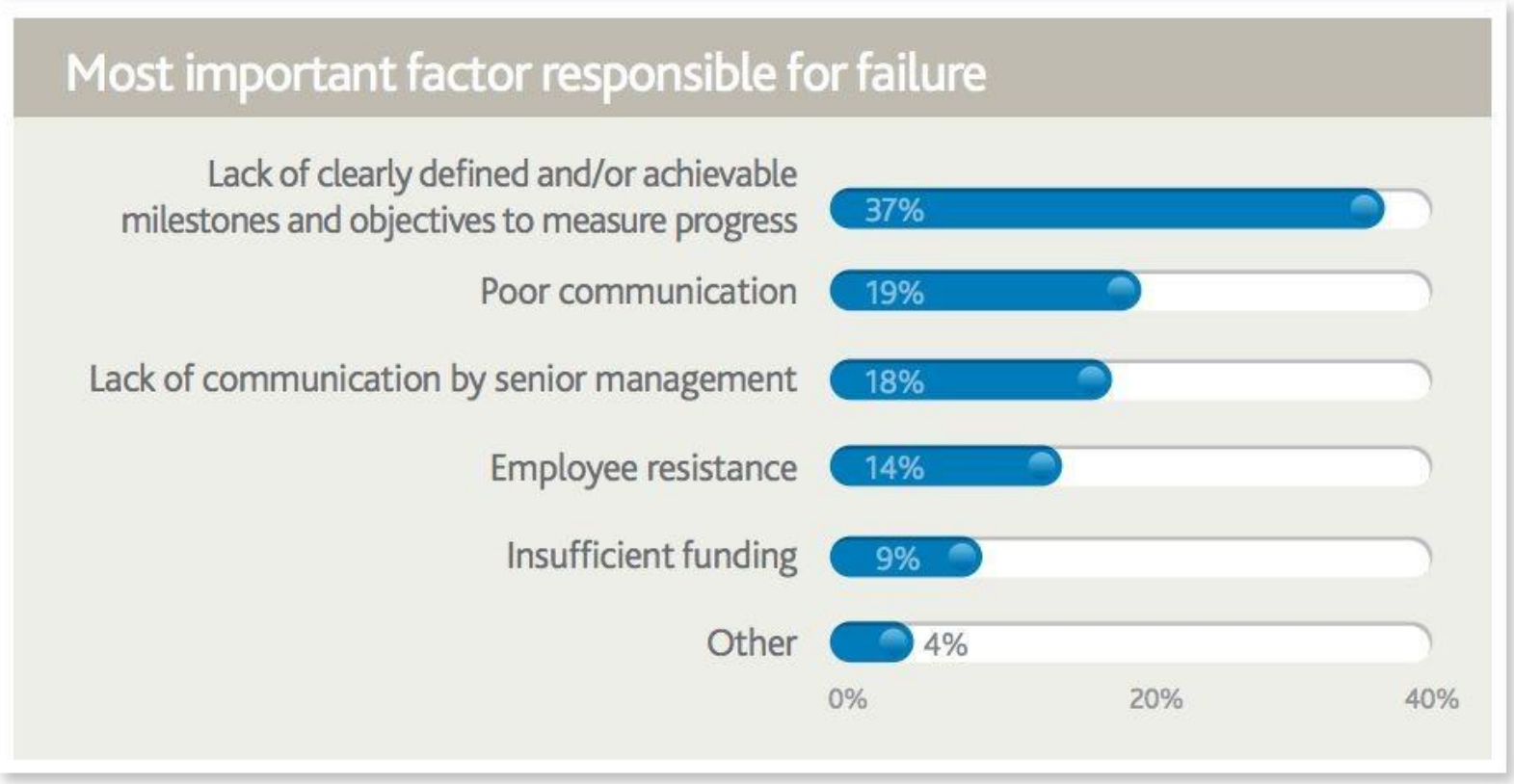
Why do we need Agile?



Why do we need Agile?



Why do we need Agile?



Source: Pulse of the Profession 2017

Software Development is Complex

Software Development is Complex

Evolving/Changing Business Requirements

- Multiple, inconsistent inputs
- Change is usually good though

Inherent R&D Nature of Software Development

- Software performance issues
- Changing/New technologies

Distributed Teams Needing to Collaborate

- Team communications
- Need for complementary skills

Technology Environment with Many Moving Parts

- Dependence on external technology components –HW/SW
- Constantly changing environment

- Unlike other engineering disciplines (e.g. manuf, bldg const), building software is harder and more complex
- If not managed properly, a lot can go wrong!
- Needs to be properly managed – with ***appropriate*** level of process discipline
- Agile approach works very well in most cases

Talking Points

- Why do we need Agile?
- Introduction to Agile
- Scrum
- Q & A

Introduction to Agile

Agile is a mindset that adheres to the principles of 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 processes and tools **Working software** 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.

Manifesto for Agile Software Development, www.agilemanifesto.org

Principles behind Agile Manifesto

1 Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.

2 Welcome changing requirements, even late in development.

3 Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.

4 Business people and developers must work together daily throughout the project.

5 Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.

6 The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.

Principles behind Agile Manifesto

7 Working software is the primary measure of progress.

8 Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.

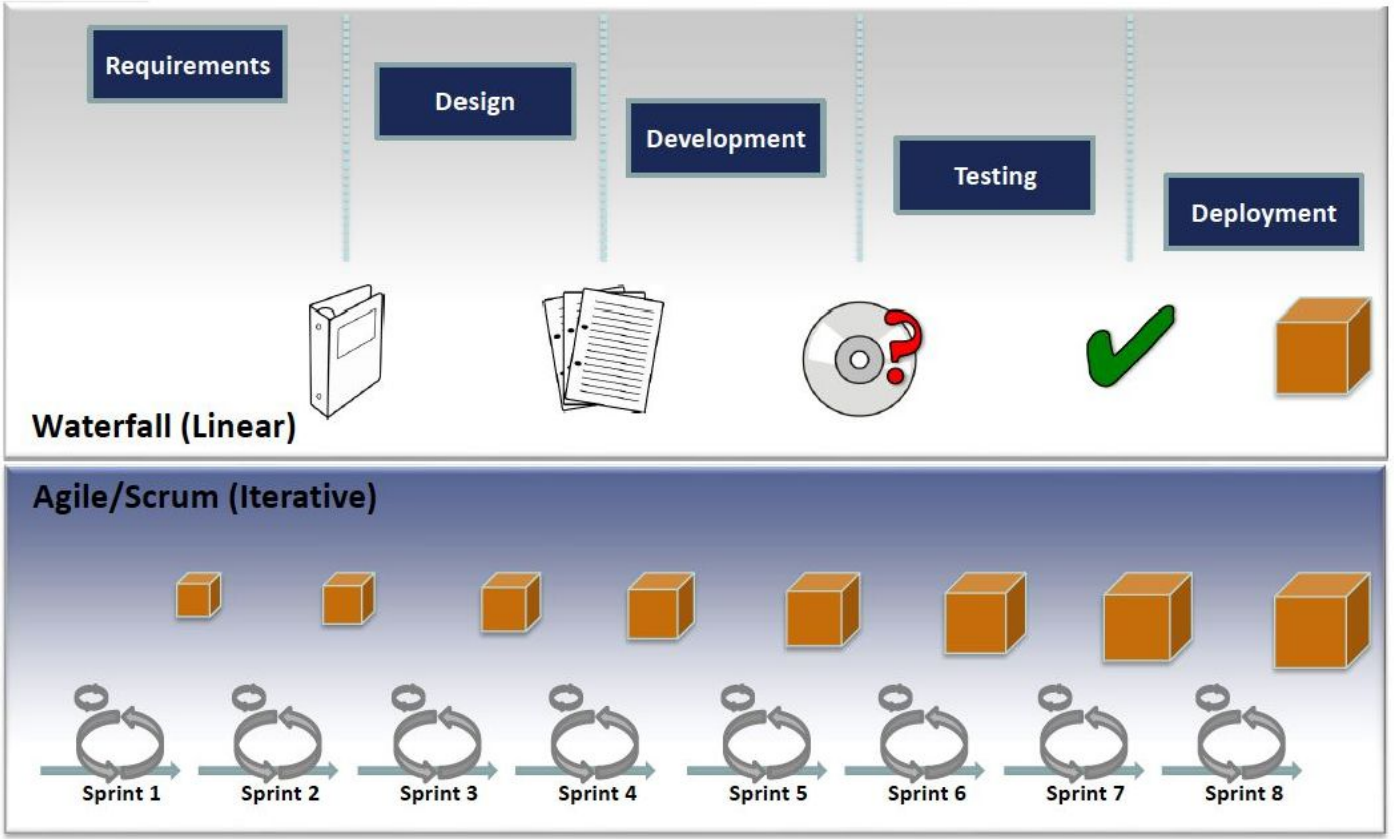
9 Continuous attention to technical excellence and good design enhances agility.

10 Simplicity--The art of maximizing the amount of work not done--is essential.

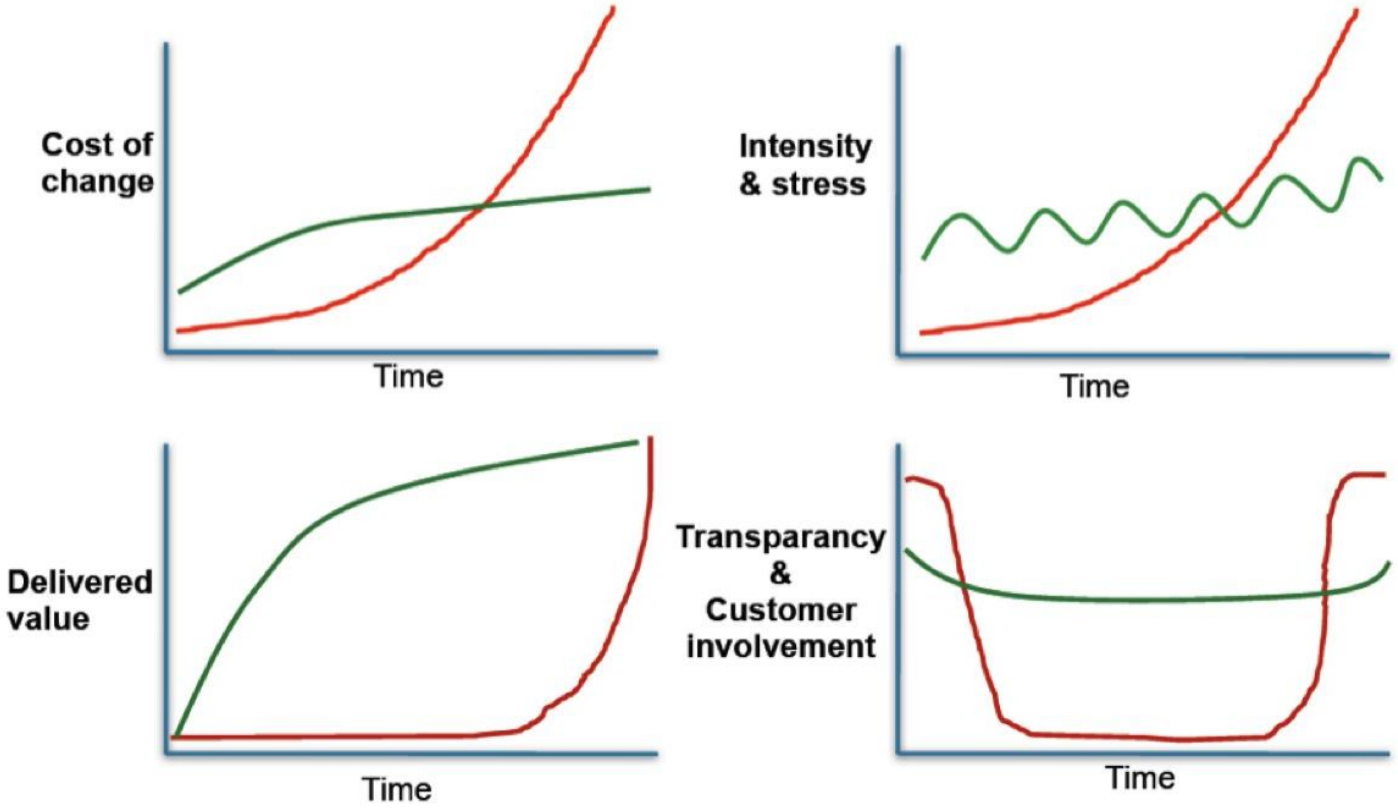
11 The best architectures, requirements, and designs emerge from self-organizing teams.

12 At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.

Linear vs. Iterative process



Characteristics of Agile vs. Waterfall



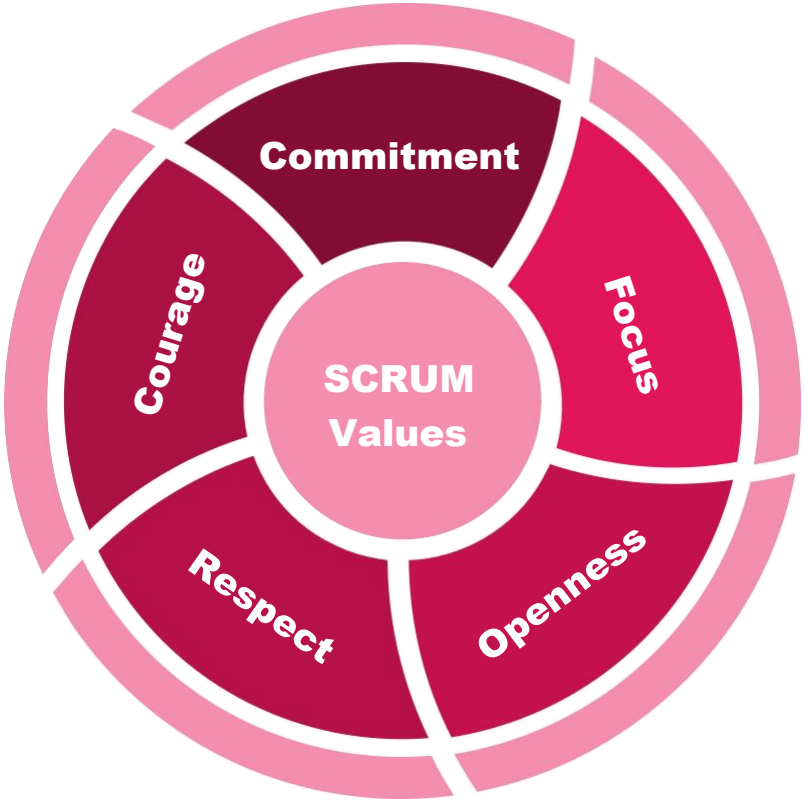
Talking Points

- **Why do we need Agile?**
- **Introduction to Agile**
- **Scrum**
- **Q & A**

What is Scrum?

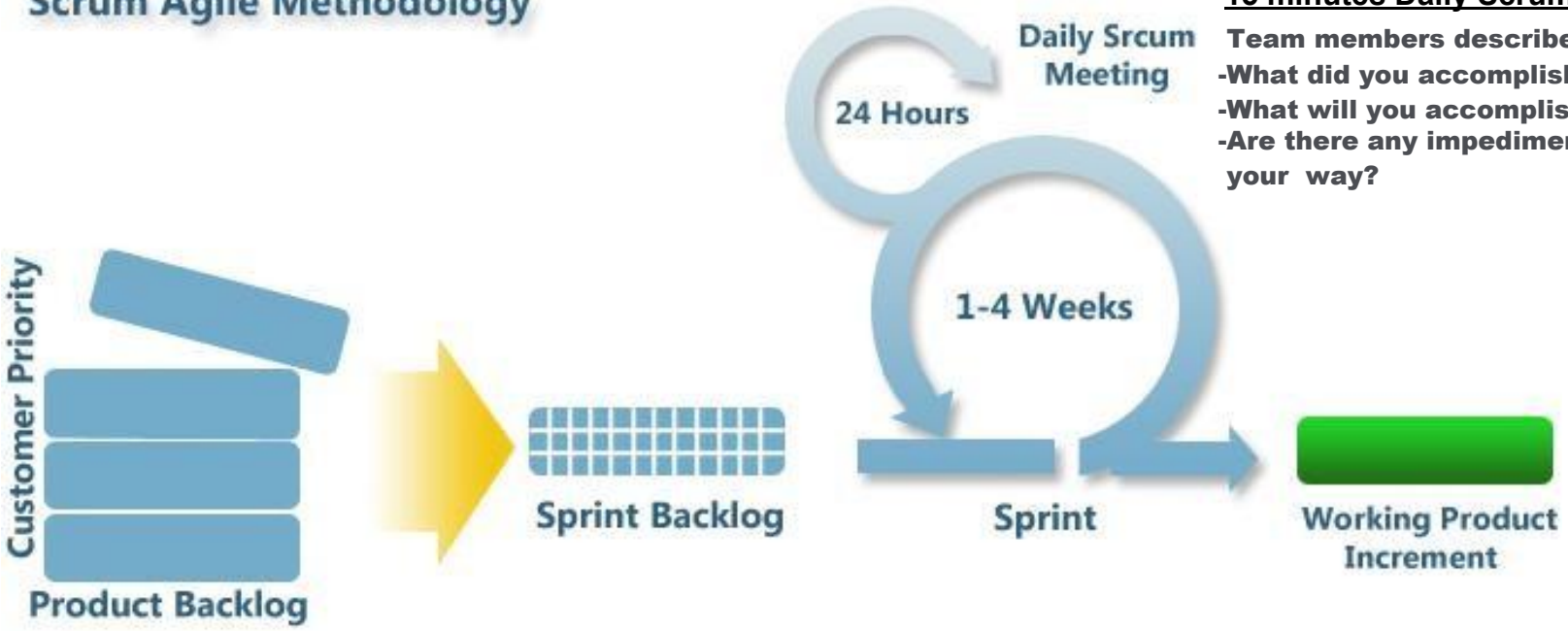
Scrum is a team-based empirical process to incrementally develop products.

Scrum Values



Scrum Process

Scrum Agile Methodology



15 minutes Daily Scrum Meeting

Team members describe:

-What did you accomplish yesterday?

-What will you accomplish today?

-Are there any impediments or risks in your way?

Scrum - Working Agreement - Possibilities

- **Show Respect**: Don't interrupt; let people finish what they're saying. It's OK to disagree with each other. No personal attacks, attack issues, we debate the merit of ideas, not people.
- **Contribution**: Everyone has equal voice and valuable contribution.
- **Meeting** - Be on time, end on time, have an agenda.
- **Decision Making** – We make decisions together.
- **Be transparent** - No hidden agendas. We will give feedback, we will receive feedback, and we will act on feedback.
- **Impediments** - Solve roadblocks within the team. If the impediment can't be solved within the team, give it to the Scrum master.
- **Commitment** - We make commitments as a team. We will be held accountable to our commitments. We work as a team to make a commitment and deliver on it.
- **Incomplete stories are not good** – it is better to help get an existing story to "done" than to start another story that can't be finished in the current sprint.
- **Communication** – We communicate using high-fidelity communication mediums.

Scrum - Roles and Artifacts

Roles

- Team
- Scrum Master
- Product Owner

Artifacts

- Product Backlog
- Sprint Backlog
- Product Increment

Events

- Sprint (1-4 weeks)
- Sprint Planning Meeting (before each sprint - 2hrs/week)
- Daily Scrum/Standup Meeting (15 mins)
- Sprint Review Meeting / Sprint Demo (Just before sprint end - 1hr/week)
- Sprint Retrospective Meeting (after each sprint - 1hr/week)
- Backlog Grooming/Refinement meeting (before each sprint - 1hr/week)

Scrum - Roles

Product Owner

- Is Team's sole source of truth for requirements and priorities
- Owns requirements (new features, bug fixes) and ranking
- Works with customers, stakeholders to define & rank user-facing features
- Collaborates with Team to ensure proper implementation
- Often a Product Manager, Business Analyst

Scrum Master

- Does whatever is needed to make Team as productive as possible
- Acts as Servant Leader
- Owns process (enforces, tracks, expedites problem resolution)
- Runs Daily Stand-Up, Sprint Planning, Retrospective Meetings
- Often a Project Manager

Team

- 5-9 people
- Self-organizes cross-functional members to implement, test features
- Software & test engineers, database architects, UI developers, etc.
- Owns estimates, tasks, assignments

Scrum - Artifacts

Product Backlog

- Simplest definition - “A list of all things that needs to be done within the project”. It replaces the traditional requirements specification artefacts. These items can have a technical nature or can be user-centric e.g. in the form of user stories.
- The Scrum Product Owner uses the Scrum Product Backlog during the Sprint Planning Meeting to describe the top entries to the team. The Scrum Team then determines which items they can complete during the coming sprint.

Sprint Backlog

- Sprint Backlog contains all the committed Stories for the current Sprint broken down into Tasks by the Team. All items on the Sprint Backlog should be developed, tested, and integrated as per “Definition of Done” to fulfil the commitment.

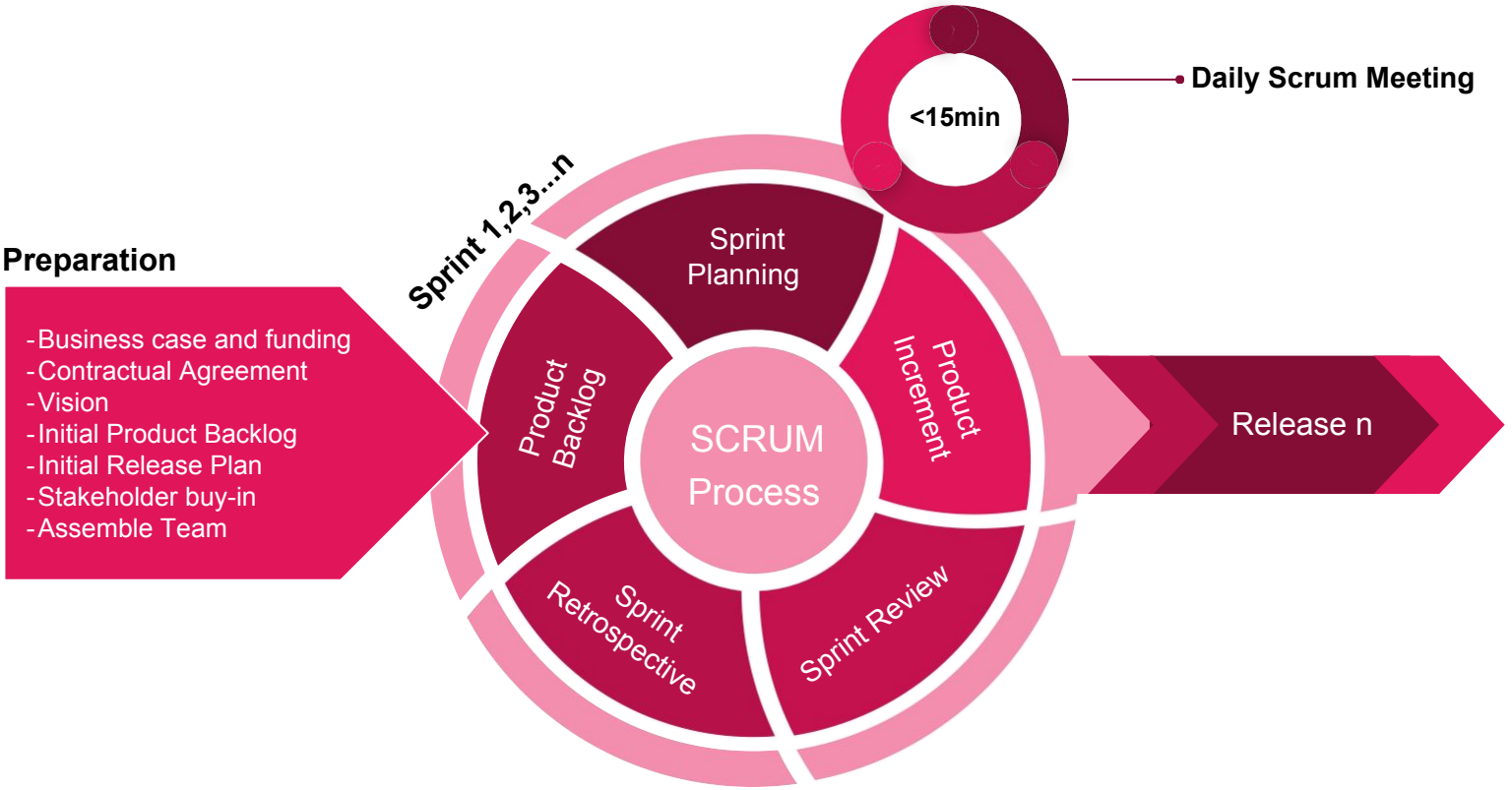
Product Increment

- Product Increment is the sum of all completed product backlog items in a given sprint, plus the value of previous increments.
- Output: A working, tested shippable product

Scrum - Events

Ceremony Time	Box	Input	Output	Value
Backlog Refinement	1hr/week	Draft User Stories, Epics from Product Owner	Finalized User Stories Technical Stories Ranking for top PBIs Sprint Backlog	Product Backlog and Team is ready for Sprint Planning
Sprint Planning	2hr/week	Ranked Product Backlog with Acceptance Criteria	-Selected stories + estimates -Tasks + estimates	Team has a plan to implement Sprint Backlog
Daily Stand-up	15min	In-progress Tasks	-Tasks updated -Impediments Raised	Team on same page: Sprint Progress and Impediments
Sprint Review	1hr/week	Demo prepared for completed stories	New Stories, based on review by PO. Ranking may be revised	Ensure appropriateness of deliverables
Retrospective	1hr/week	Things which we should: -Start Doing -Stop Doing -Continue Doing	Shortlist of improvements for next sprint with owners	Learn from experience, enable continuous improvement

Scrum Process



Scrum Process - Preparation

Product Owner Prepares the first sprint:

- Talks to stakeholders to form a vision of the product
- Arranges stakeholder buy in
- Gathers high level features
- Creates initial product backlog
- Creates rough release plan

Planning??

- How do you plan a Vacation/Trip?
- What all things you will consider while planning for a vacation/trip?
- Sample Project Plan

Scrum Process - Product Backlog & User Stories

Product Owner own and prioritize this:

- Prioritize list of features and work
- Estimated in how much work it will take relative to each other

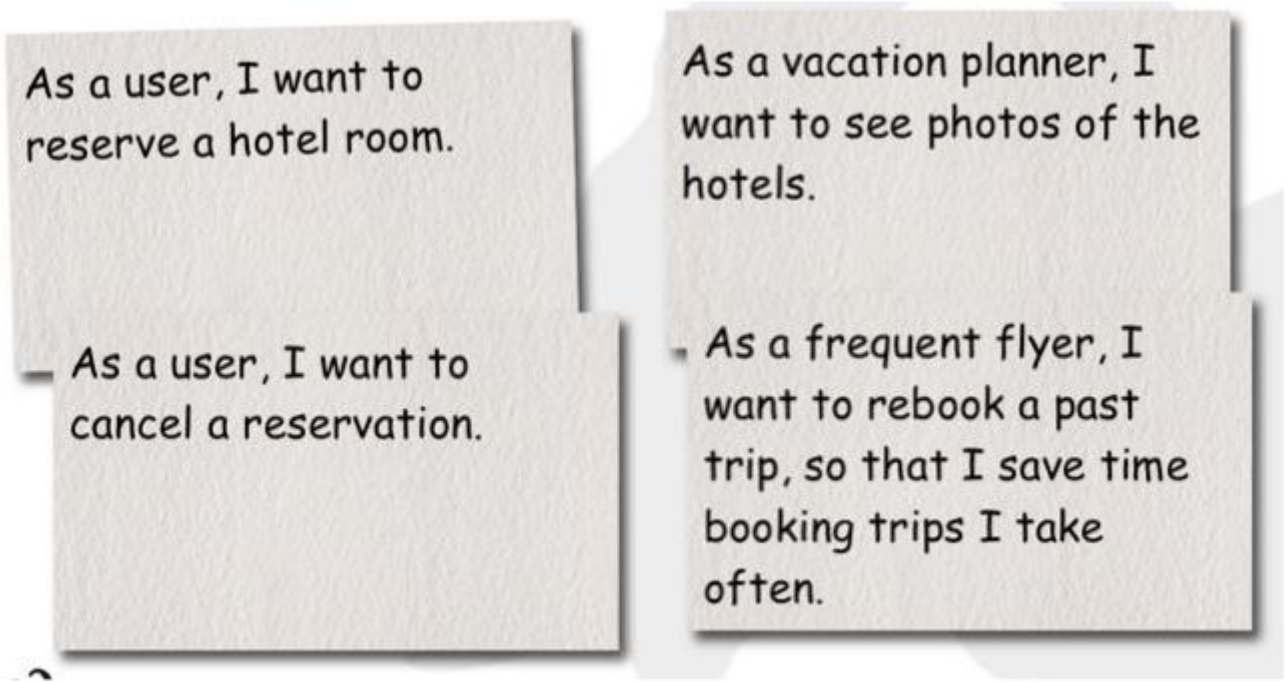
User Stories

- A user story describes functionality that will be valuable to either a user or purchaser of a system or software.
- Business writes user stories like:

As a [stakeholder] I want to [feature] so that [business value]

Scrum Process - User Stories

Examples of a User Story:



Scrum Process - User Stories

Good Stories should be:

Independent **N**egotiable

Valuable

Estimable

Sized Appropriately

Testable

Scrum Process - Sprint Planning

Planning Meeting

- Set up Sprint goals
- Select User Stories for the sprint
- Product owner sets sprint goal and clarifies requirements
- Team commits to set of user stories based on its ***velocity***

Velocity

Velocity is a measure of the amount of work a Team can tackle during a single Sprint and is the key metric in Scrum.

Scrum Process - Estimations

Planning Poker:

- Involves all team members
- Each has deck of cards: e.g. 0, ½, 1, 2, 3, 5, 8, 13, 20, ?
- Product owners explains user story and answers any questions
- Each team member picks his card
- Discuss different estimates
- Re-estimate -> Converge (meet at a point)

Scrum Process - Product Increment

- Product Increment is the sum of all completed product backlog items in a given sprint, plus the value of previous increments.
- At the end of a Sprint, the new Increment must be "Done," which means it must be in useable condition and meet the Scrum Team’s definition of "Done"
- The increment is a step toward a vision or goal.
- The increment must be in useable condition regardless of whether the Product Owner decides to release it.

Scrum Process - Daily Scrum Meeting

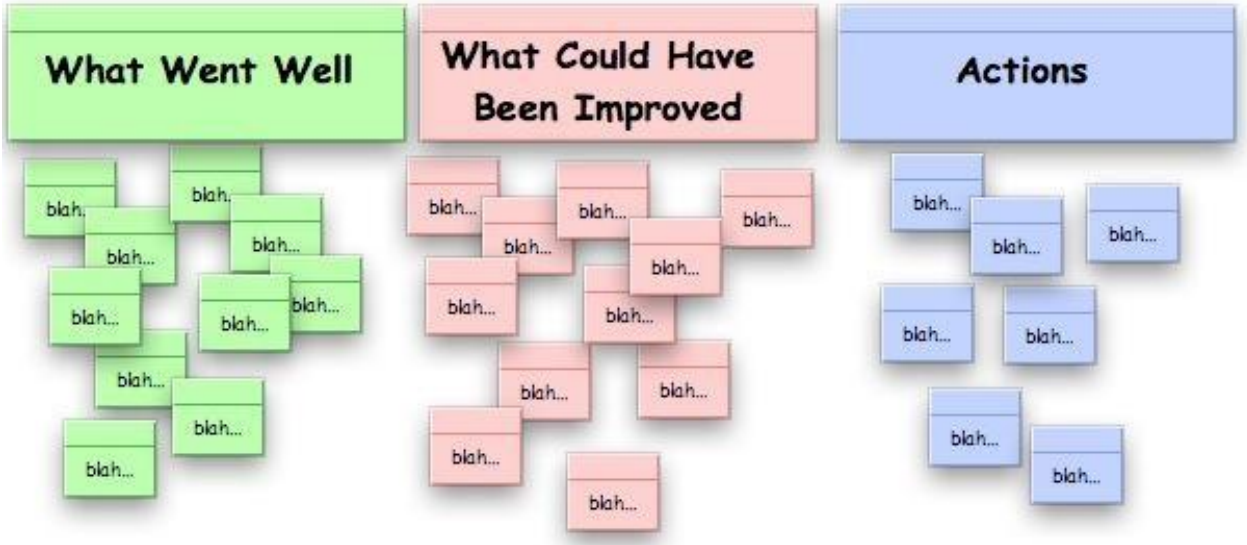
- Should be of maximum 15 mins at same time and same place
- Broadcast individual updates to everyone
- It’s not a status check-up meeting: Team member making commitments in front of peers
- No Problem Solving should happen in this meeting
- Team members describe:
  - What did I get DONE yesterday?
  - What will I get DONE today?
  - Any impediments blocking me?
- Sample Video: https://youtu.be/q_R9wQY4G5I

Scrum Process - Sprint Review

- Also known as ‘the demo.
- Team presents what is accomplished during the sprint
- NO SLIDES
- Get feedback from the customer/stakeholders
- Feels great if they like it
- Get immediate response/feedback

Scrum Process - Sprint Retrospective

- How can we improve our way of working?
- Things which we should:
  - Start Doing
  - Stop Doing
  - Continue Doing
- Action Item from
  - Previous Retro
  - Current Retro



Scrum Process - Release

A working, tested shippable product

Q & A

Thank You!