

Agile Practices

Lecture 8-Journey of Agile

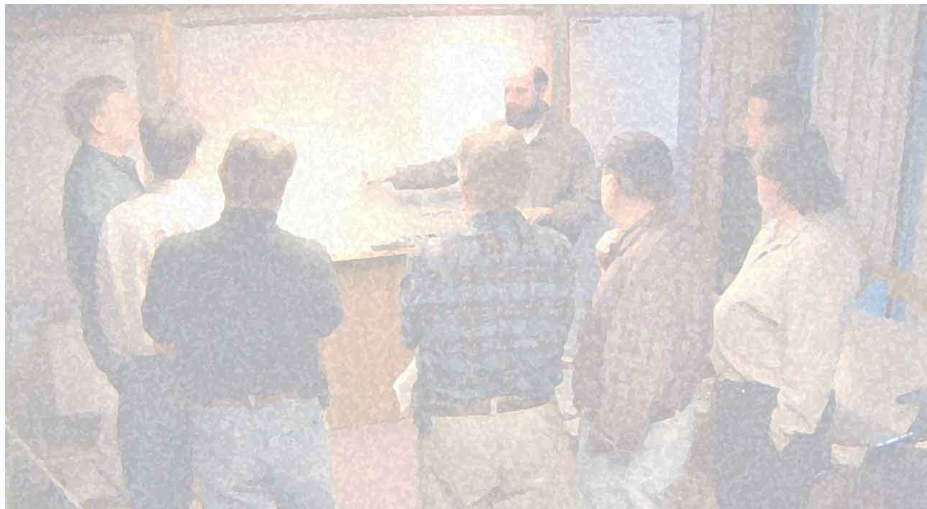
Omkarendra Tiwari

September 1, 2022

Agile History

Picture taken by Ward Cunningham at Snowbird, Feb 2001^a

^a<https://agilemanifesto.org/>



Manifesto for Agile Software Development

Preamble and Values

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

While there is value in the items on right, we value items on left more.^a

^a<https://agilemanifesto.org/>

Principles behind the 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. Agile processes harness change for the customer's competitive advantage.
- 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.
- 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.

In words of Uncle Bob

- **At the Beginning** Dates are frozen and requirements are constantly changing
- **Analysis** No concrete definition of when done
- **Design** No definition of done or limit on improvement
- **Implementation**
 - There is certainly a criteria for when its done
 - Needs to be aligned with design
 - Needs to be functional
 - Needs to provide expected performance
 - Needs to be done within time line
 - If done within time line; great!
 - Not; Inform Stakeholder; face the angst; manage to borrow more time.

Decision Making

- **At the Beginning** Dates are frozen and requirements are constantly changing
- **First thing we do** Divide the time into small chunks (#Weeks)
 - One week or two week
 - Not too short and not too long
 - Enough to get some work done and not much that debug takes longer
- **And, Iteration begins** Start working on small chunks of software;

Whats the Deal of Iterations

Things we do in iterations

- User stories; features
- Analysis; estimation
- Design; planning
- And, keep doing it all

In Agile, Analyzing and Planning never stops

What Agile is about!

Special Ingredient

- Produce data about all the important aspects
- Analyze data; how good or bad we r doing
- Continuous update on estimation
- We are failing is not an issue; knowing early is important
- Continuous monitoring

Saving the day

- Agile produces lots of data
- Each iteration will have some (un)finished stories
- Finished/Unfinished stories gives us data for estimation
- It helps Managers to keep an eye on the state of the project
- It allows them to reshape the future plan

Trade-offs

- Software: good, fast, cheap, done; can have three but not all
- Managers need to keep revising
 - **Schedule**: Dates are frozen; they may be coupled with Business events;
 - **Staff**: Brook's law (adding staff to late project makes it later)
 - **Quality**: Quick and dirty is myth; The only way to go fast, is to go well.
 - **Scope**: Ask/Convince to prioritize features; deliver in parts

Further readings

Recommended

- <https://www.theatlantic.com/technology/archive/2017/12/agile-manifesto-a-history/547715/>