CONSIDERATIONS FOR A SUCCESSFUL IMPLEMENTATION OF AN AGILE SDLC FOR MEDICAL DEVICES

Adam Darmstadt

Nov 07, 2012

Summary

Agile and Lean development methodologies can not only expedite time to market, but also increase quality. With the right customization on your Agile development process you can also meet regulatory requirements.

This presentation will focus on:

- How Agile principles can contribute to quality
- Meet IEC 62304 requirements
- Get and use tools for collocated and distributed teams
- Continuous improvement

Biography

2008 – …	Bio-Rad Laboratories Software Development Manager (6 product lines)
2000 – 2008	Carl Zeiss Meditec Software Group Manager, Architect
1995 – 2000	HEAD acoustics GmbH Software Engineer
1992 – 1995	PREMA GmbH Software Engineer

12+ Years in Medical Device Development
20+ Years in Software Engineering

http://www.linkedin.com/in/adamdarmstadt

I would like to share …

… what I believe:

- Safety and Effectiveness are not optional
- Quality is not optional
- Documentation is not optional, but don't overdo it
- Processes and Tools, if done right, help you meet your business objectives
- … that all of the above is true even for non Medical Devices

I would like to learn about your problems and concerns

What is Agile?

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

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

http://agilemanifesto.org/

What is Agile?

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

That is, while we value the items on the left more, we still do the items on the right.

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.

Principles behind the Agile Manifesto

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.

Principles behind the Agile Manifesto

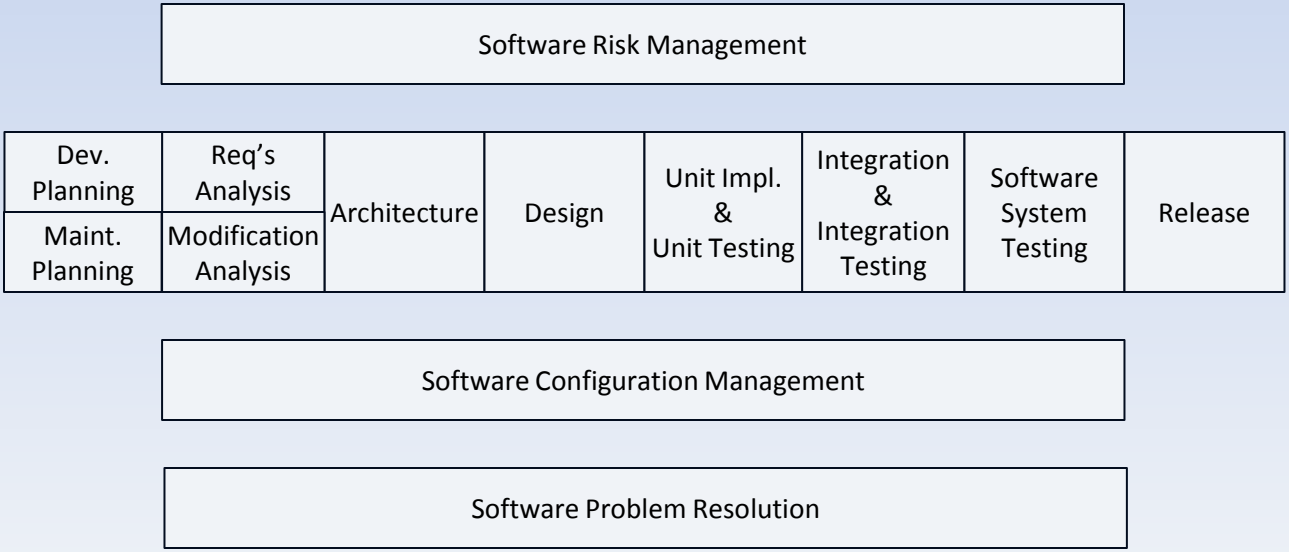
9. Continuous attention to **technical excellence and good design** enhances agility.
10. **Simplicity** – the art of maximizing the amount of work not done – is essential.
11. The best architectures, requirements, and designs emerge from **self-organizing teams**.
12. At regular intervals, the team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

http://agilemanifesto.org/principles.html

Benefits of Agile Software Development

- Decreased Time to Market
- Reduced Risk that the Project fails
- Earlier Customer Feedback
- Higher Adaptability to Change
- Increased Quality

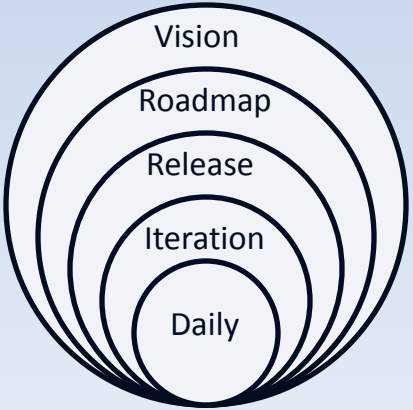
IEC 62304 Processes and Activities



Planning

5 Levels of Planning

- Product Vision
- Product Roadmap
- Release Planning
- Iteration / Sprint Planning
- Daily Standup / Commitment



Product Vision

- Product Vision
- Product Roadmap
- Release Planning
- Iteration / Sprint Planning
- Daily Standup / Commitment

- Documenting the vision
  - helps as a guidance for decision making
  - helps managing scope and prioritizing features

Product Roadmap

- Product Vision
- Product Roadmap
- Release Planning
- Iteration / Sprint Planning
- Daily Standup / Commitment

- The roadmap helps to
  - manage scope and prioritize features
  - create a future ready architecture
  - avoid the “last release” feature creep
  - align software releases with hardware changes

Release Planning

- Product Vision
- Product Roadmap
- Release Planning
- Iteration / Sprint Planning
- Daily Standup / Commitment

- Initial Development Plan
- Risk Management Plan
- Resource Plan (Team Size)
- Estimate team's velocity
- Define your "release acceptance criteria"
- Define your System Integration Plan
- Define your Software Validation Plan

Iteration / Sprint Planning

- Product Vision
- Product Roadmap
- Release Planning
- Iteration / Sprint Planning
- Daily Standup / Commitment

- Clarify user stories and define their acceptance criteria
- Schedule relevant tasks
- Negotiate scope with Product Owner

After each iteration / sprint:

- Deliver working software and demonstrate to product stake holders
- Conduct retrospective reviews and improve continuously

Daily Standup

- Product Vision
- Product Roadmap
- Release Planning
- Iteration / Sprint Planning
- Daily Standup

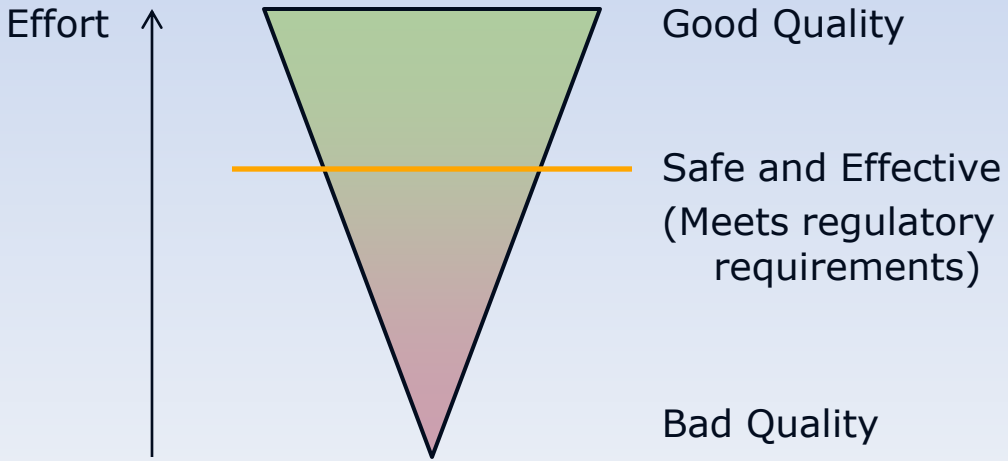
- Share progress and plan for next day
- Remove obstacles as a team

What is Working Software?

- Your SDLC should define what "Working Software" means to your organization or your project.
  - It is not only the code!
- As needed, include
  - Documentation updates
  - Unit tests
  - Updated integration and system level tests
  - Design reviews (incl. code reviews)
  - …

→ **Define your acceptance criteria, your "definition of done".**

Quality Focus



Your Agile Team

- Is your team motivated? Do you trust it? (principle #5)
- Keep it motivated! Keep trusting it!
- Plan for constant training, that your team can return this with technically excellent design. (principle #9)

The Team vs. The Individual

- What goals do you define in your annual objective?
- How does that fit into your performance review process?
- Can your stronger individuals still get the recognition they deserve?
- Is your culture conducive for team members to raise issues during stand-ups?

Agile Configuration Management

- Managing all artifacts in one tool simplifies traceability.
- Build your change package around a user story and include all related items, e.g.
  - User Stories
  - Requirements
  - Risks
  - Tests
  - Source Code, incl. unit tests

Tools

- Use tools where all team members have access to. (Check your IT policy, if you want to use tools in the cloud.)
- Manage all artifacts of your SDLC as much as possible in integrated tools:
  - Document Management
  - Product Backlog
  - Requirements
  - Risk
  - Test Cases
  - Source Code
  - …

Distributed Teams

- Face-to-face meetings is still the best form of communication. (principle #6)
  - Plan regular meetings, preferably at iteration planning times.
- Use communication tools that are easily available to team members.
  - Instant Messaging
  - Video Call / Chat
  - Online Meetings
  - Online Whiteboards
- Have a (proxy) Product Owner at all sites.
  - Easy access for team member.
  - Minimizes issues, if sites are in different time zones.

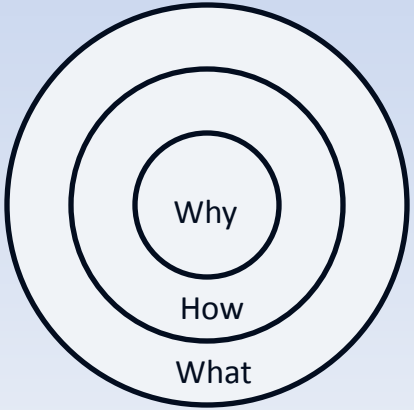
Almost Last Comments

- How does an Agile SDLC fit into your Product Development Process?
  - Is your PDP a waterfall like process with phase gates?
- Scalability to large projects need special considerations.
  - E.g. "Scrum of Scrums"
- Discipline to follow the process is required from the team and all the stakeholders. Agile is not an ad hoc development method.
- If you are implementing an Agile SDLC or changing what you have currently implemented, involve all your stakeholders early.
- Manage your learning curve. Have "test iterations" to establish processes and team velocity.

A few more Comments

- Plan to train your organization.
- Know why you are transitioning or using an Agile method. It is not a goal in itself.
- Integration with Hardware: Use of hardware simulators for automated software testing makes integration with real hardware easier.
  - Maintain your hardware simulator even when real hardware is available.

Simon Sinek: The Golden Circle



Why – We want to create the best medical devices in the world, and improve patient care,

How – by speeding up time to market, increasing quality, improving productivity.

What – Let's implement an Agile SDLC, and create new SOPs, templates, forms, and checklists, …

http://www.ted.com/talks/simon_sinek_how_great_leaders_inspire_action.html

Questions

adam@adamdarmstadt.com