Project management

Jana Kostičová, 09.12.2025

What is project management?

Project

“A temporary venture that exists to produce a defined outcome that leads to the achievement of intended benefits (value)” *(Axelos / PRINCE2®)*

Project types

- SW / IT projects - include SW development
- ! But also construction projects, engineering projects, transport projects, research projects, ….
- ! But also personal projects (e.g., writing a master thesis)

Project management

“The discipline of applying specific processes and principles to initiate, plan, execute and manage the way that new initiatives or changes are implemented within an organization.” *(Axelos / PRINCE2®)*

SW / IT projects

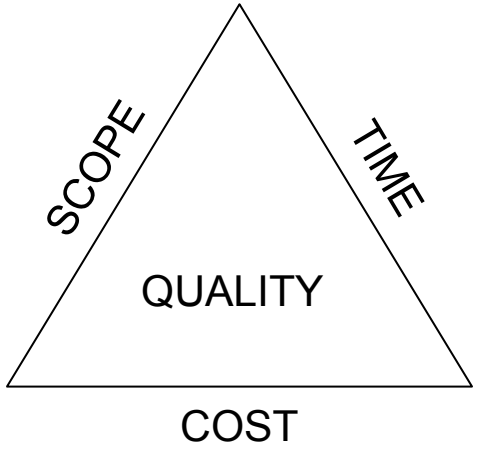
“New” project type (~100 years of IT vs thousands years of e.g., construction)

+ IT projects often take place in a dynamic environment - they must quickly adapt to innovations and rapid technological changes

Common problems

- Unclear project visions and business goals
- Too ambitious scope with respect to the resources and deadlines
- Insufficient requirements traceability
- Inadequate planning and poor risk management
- Poor communication (both internal and external)
- …

Project management triangle



- Three constraints of the quality of work
- The project manager can trade between constraints
- Changes in one constraint necessitate changes in others to compensate or quality will suffer

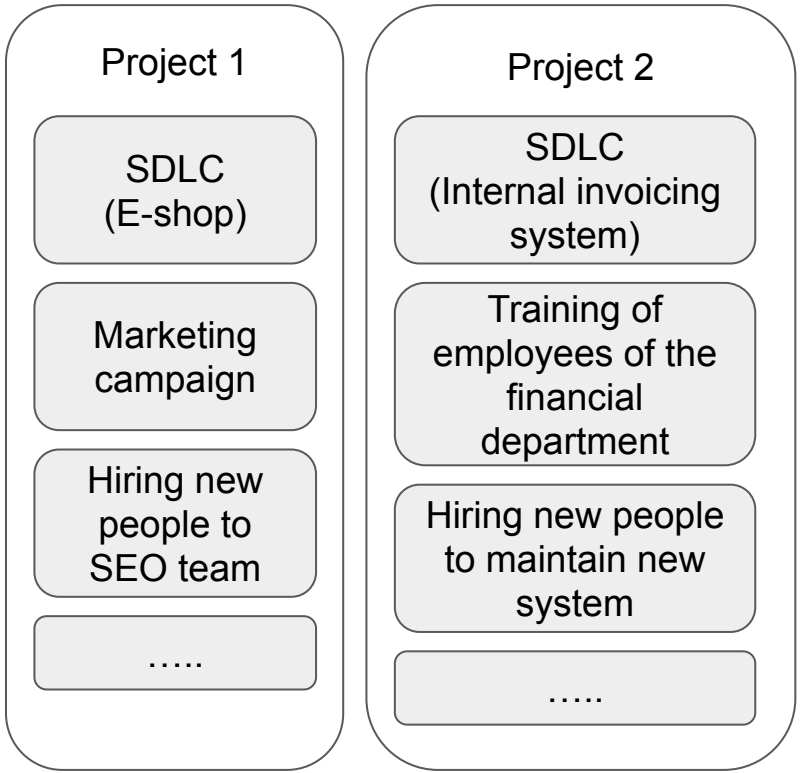
Project vs process

Software development process (also known as SDLC)

- A structured set of activities that aims to design, create, test, and maintain software and systems
- The process is applied repeatedly during different projects (although various methodologies can be used)

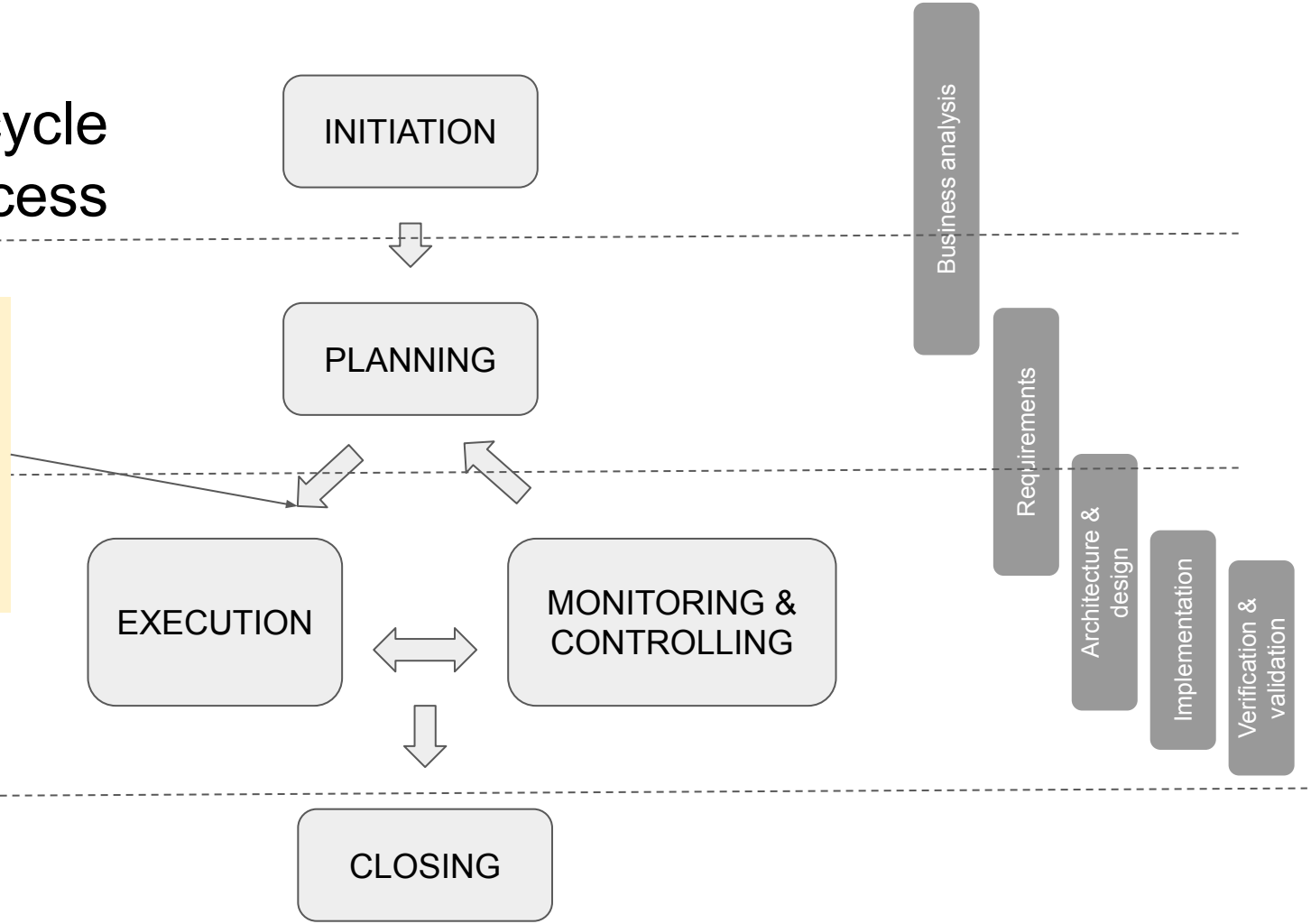
SW / IT project

- One-time, temporary organisational activity
- Typically broader initiative, may have various goals, requirements, and deliverables beyond just creating software
- Includes also (a part of) SDLC in some form - depending on the chosen methodologies

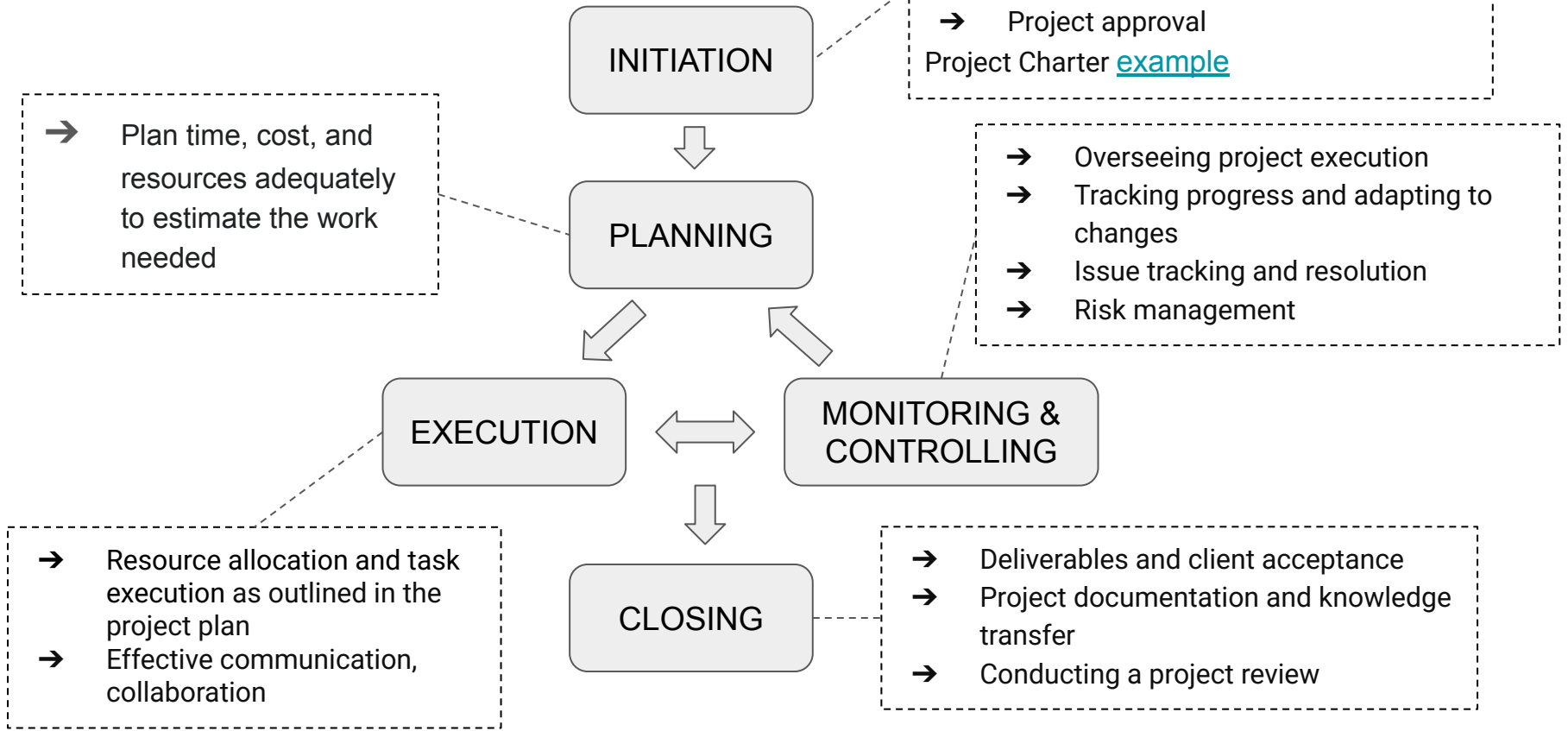


Project lifecycle vs. SW process

Between the planning and the execution phase, a project kick-off meeting typically takes place



Project lifecycle



Project planning

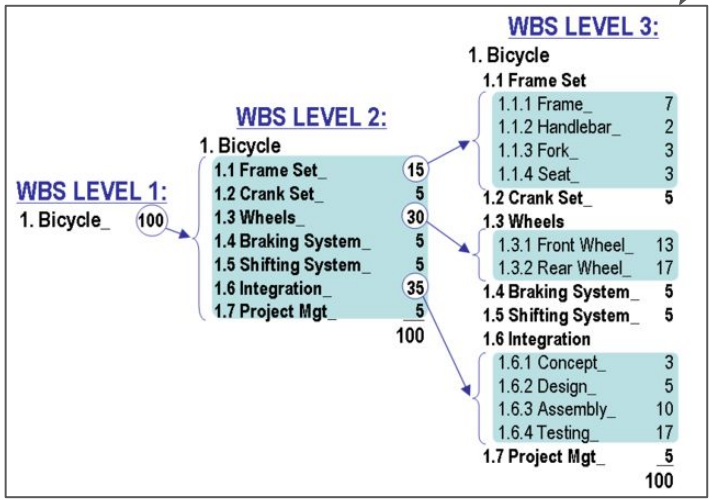
Project plan

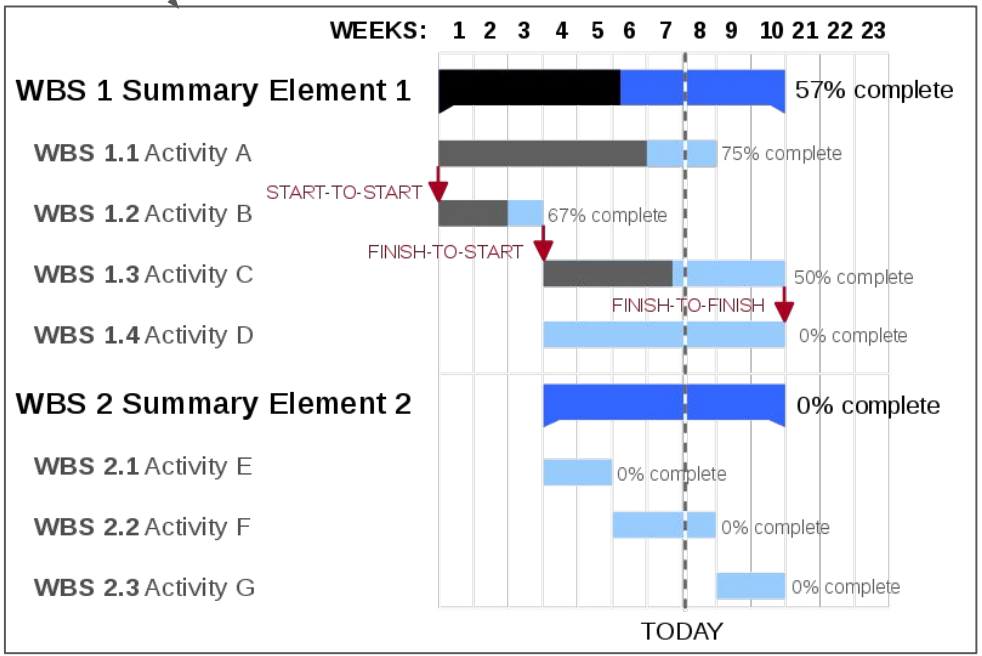
"...a statement of how and when a project's objectives are to be achieved, by showing the major products, milestones, activities and resources required on the project." (PRINCE2)

Methods:

Work Breakdown Structure (WBS): Task identification

Gantt charts: Project schedule, dependencies, current status





Software tools for project management

- MS Project
- Jira
- Trello
- Asana
- Smartsheet
- …

Some of them provide issue tracking directly, other support integration with issue tracking SW

Version control systems help a lot

- Git + web based platforms such as GitHub, Bitbucket

Issue tracking

- Issue = work item to be done / solved
  - Typical attributes: Title, Description, Assignee, Issue type, State, Priority, Relationships to other issues, Start date, Due date, …
  - Workflows (define transitions between issue states)
- Example: Jira
  - It directly supports various agile methodologies (Scrum, Kanban,..) but is flexible enough to cover other methodologies or their combinations
  - Integration with communication platforms (Slack, MS Teams), version control (GitHub) and other SW
  - Apache Cassandra Jira

Methodologies

Traditional / sequential methodologies

- Waterfall
- Iterative-incremental

PMBOK (by Project Management Institute)

- A set of standards

Agile approaches

- Pure Agile
- Scrum
- Kanban
- Extreme Programming (XP)
- Feature-driven development
- SAFe (Scaled Agile Framework)
- …

Process oriented

- Lean management
- Six sigma
- …

Other

- Prince2 (Projects IN Controlled Environments)
  - Traditional (sequential phases)
  - Agile “version”

Methodologies - project vs. process

Project management (PM) methodologies vs. SDLC models & methodologies

- Closely interrelated - the chosen project methodology affects the SDLC model / methodology and vice versa

Generally it holds

- Sequential PM methodologies align well with sequential SDLC models
- Iterative-incremental PM methodologies align well with iterative-incremental SDLC models

Waterfall & iterative-incremental approaches

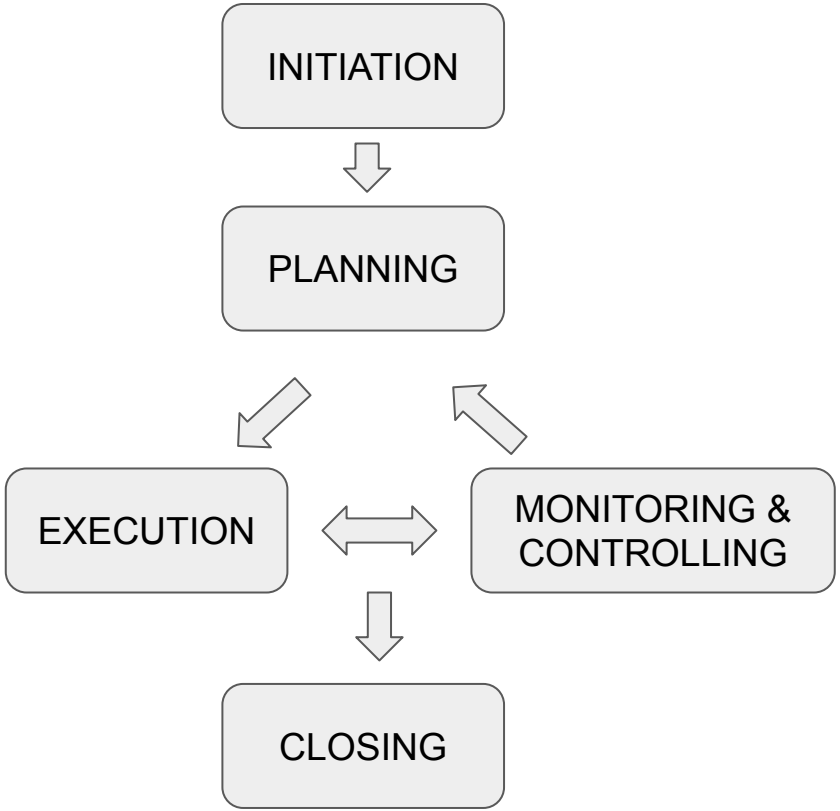
Similar principles to SDLC

Waterfall

- Project phases follow one another sequentially (note that “execution” and “monitoring / controlling” are inherently parallel)
- Planning is usually done only once, although the project plan may be updated during subsequent phases

Iterative-incremental approaches

- Breaking down a large project into smaller, manageable iterations with incremental releases
- Basic approach - “small” interconnected waterfalls
- Other approaches - e.g. agile methodologies



Critical path method - CPM

= is a project management technique used to plan and manage complex projects (also Critical Path Analysis - CPA)

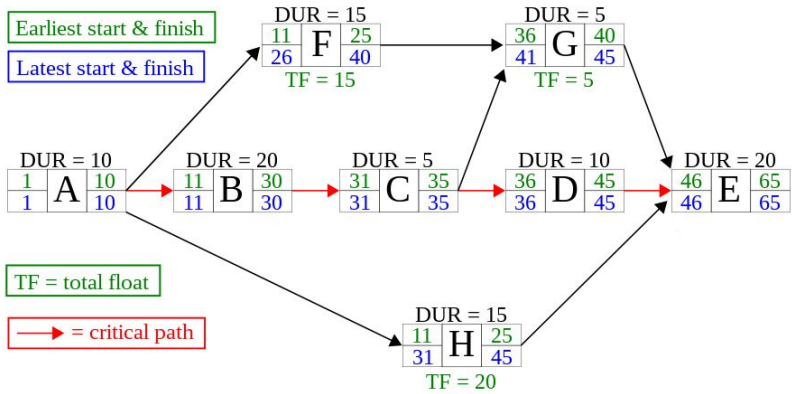
Critical path (CP): The longest path through the project, any delay on this path would directly impact the project's completion date. CP determines the shortest time possible to complete the project.

Critical activities: Activities on the critical path

Total float (TF): The amount of time an activity can be delayed without delaying the project.

- Critical activities have zero TF
- Non-critical path activities have positive TF, indicating flexibility in their scheduling

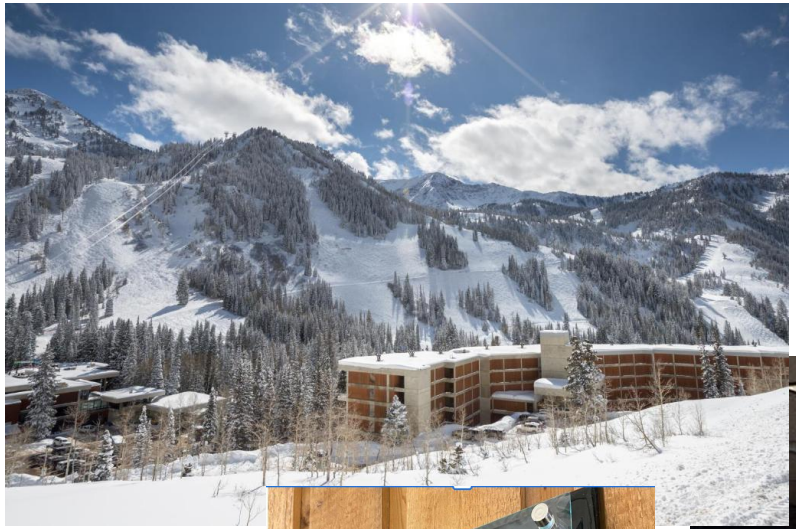
CPM is frequently used in the Waterfall model, however it is generally not restricted to this approach



Inputs:

1. Activities - typically by means of WBS (A - H)
2. Duration of each activity (DUR)
3. Dependencies between activities (arrows)

Computed values: Critical path, Earliest / latest start & finish, TF



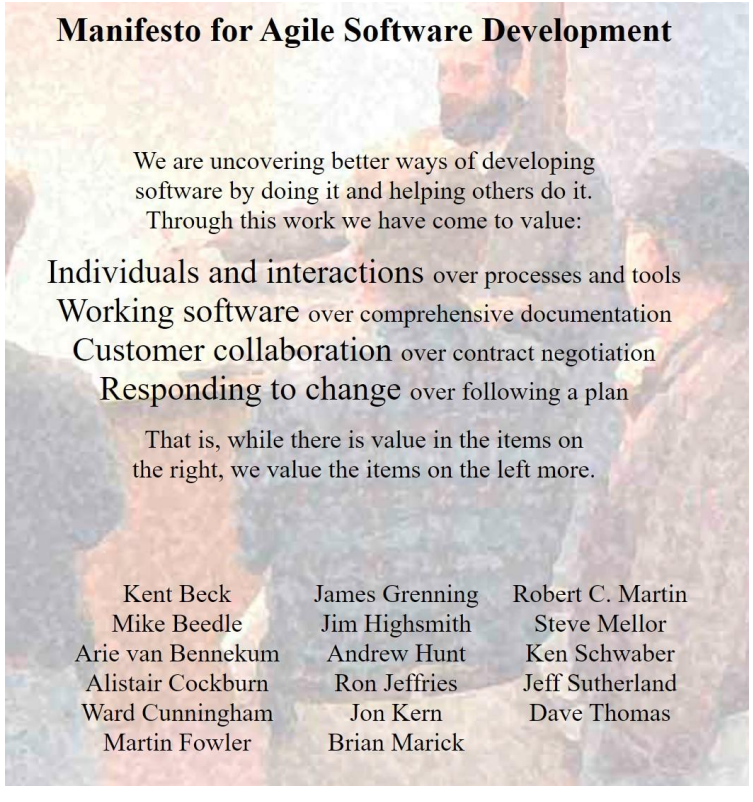
Probable timing of discussions, unless something major counters this suggestion:

7:30 - 12:15 ..... Morning discussions
12:15 - 1:00 ..... Lunch
1:30 - 4:30 ..... Skiing or afternoon hot tub
5:30 - 7:30 ..... Evening discussion
8:00 - ... .Group dinner

AGILE MANIFESTO

February 2001
The Lodge at Snowbird, Utah

Agile



- 4 key values and 12 principles (link)
- Created in 2001 as a response to the challenges and limitations of traditional software development methodologies
- It reflected a distillation of ideas and practices that had been evolving within the software development community
- 17 representatives from various agile methodologies came together to articulate a set of guiding principles
  - Extreme Programming, SCRUM, DSDM, Adaptive Software Development, Crystal, Feature-Driven Development, Pragmatic Programming, ..
- The manifesto mentions software development, but the ideas can be applied also to other industries

Pure agile:

- An approach to agile development that strictly adheres to the core principles and values outlined in the Agile Manifesto

SCRUM

= an agile framework for managing and organizing work, used software development and other industries

- Introduced in the early 1990s

SCRUM team

- A self-organizing, cross-functional team

Scrum Master

- Facilitates the Scrum process

Product Owner

- Represents the (business) stakeholders
- Responsible for defining and prioritizing the product backlog (the list of features and tasks)

Development Team (3-7 people, including UX, QA)

- Responsible for delivering the product incrementally

What about other traditional roles? - in some cases external roles can be involved, for example

- An analyst when additional expertise on requirements analysis is needed
- An enterprise architect responsible for defining and guiding the overall enterprise architecture in larger organizations or when dealing with complex and interconnected systems
- ….

SCRUM process

Sprint: A short iteration, typically 2 weeks (no longer than 4 weeks)

Official Scrum ceremonies

- **Sprint Planning:** A meeting at the beginning of the sprint where the team plans the work to be done during the sprint
- **Daily Scrum:** A short daily meeting where the team report progress, set plans for the day, and identifies and addresses impediments
- **Sprint Review**: A meeting at the end of the sprint to review and demonstrate the completed increment to stakeholders
- **Sprint Retrospective**: A meeting at the end of the sprint after the sprint review where the team reflects on the sprint and identifies opportunities for improvement

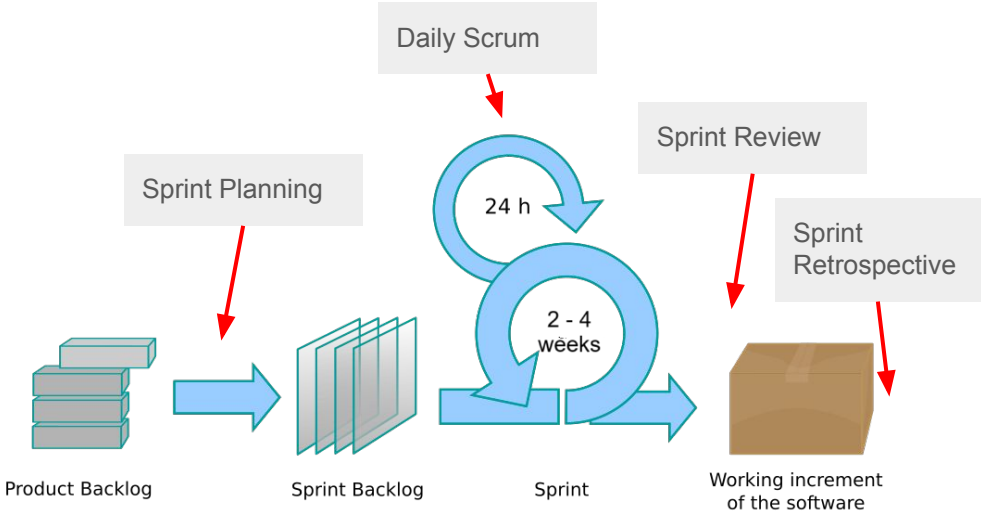
Other meetings

- **Backlog refinement (grooming):** Collaborative sessions to review and prepare product backlog items for upcoming sprints, occur regularly throughout the sprint.
- Release planning. Sprint review preparation, Design / architecture reviews, Technical debt sessions, Dependency review meetings, Daily sync of Scrum of Scrums, …

SCRUM artifacts

Product Backlog

- A prioritized list of all features, enhancements, and bug fixes that need to be addressed in the product
- Backlog items are often in the form of User stories



Sprint Backlog

- Subset of items from the product backlog to be completed during a specific sprint

Increment

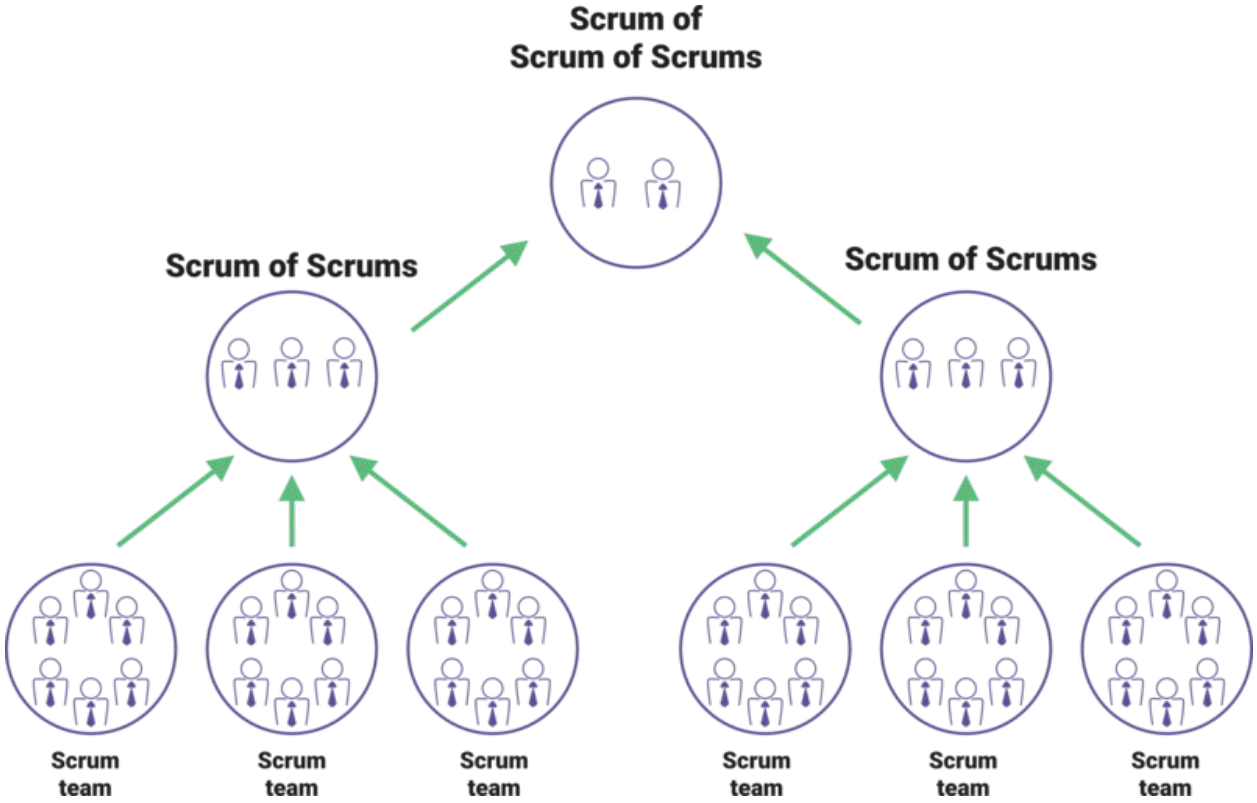
- The sum of all completed items from the sprint backlog, representing the new version of the product

Source: Wikimedia

SCRUM limitations

- Scrum does not cover certain phases of SDLC:
  - **Requirement analysis:** Scrum assumes that the product backlog contains the requirements for the product. However, it does not provide detailed guidance on specific requirements analysis techniques.
  - **Architecture planning:** Scrum does not prescribe detailed architectural planning processes.
  - **Deployment and operations:** Scrum does not provide detailed guidance on deployment and operations.
- Scrum cannot be directly scaled to large projects
  - For projects that span multiple products/systems, additional methods need to be used to synchronize different teams, such as
    - Scrum of scrums
    - Large-scale scrum (LeSS)
    - SAFe

Example - Scrum of Scrums



Source: [Wikimedia]

Extreme programming (XP)

= a complete methodology for software development based on agile principles

- Introduced by Kent Beck in the late 1990s (Chrysler Corporation's C3 project)
- **Unlike SCRUM**, it encompasses both project management and technical practices, aiming for high-quality software.
  - E.g., Test-Driven Development (TDD), pair programming, refactoring, continuous integration
- **Similarly to SCRUM**,
  - It uses short development iterations
  - It delivers small valuable product increments
  - It allows for continuous customer feedback to adapt to changing requirements

Why “Extreme”?

- The best practices from traditional software engineering are taken to "extreme" levels

Scrum and XP can be combined

Kanban

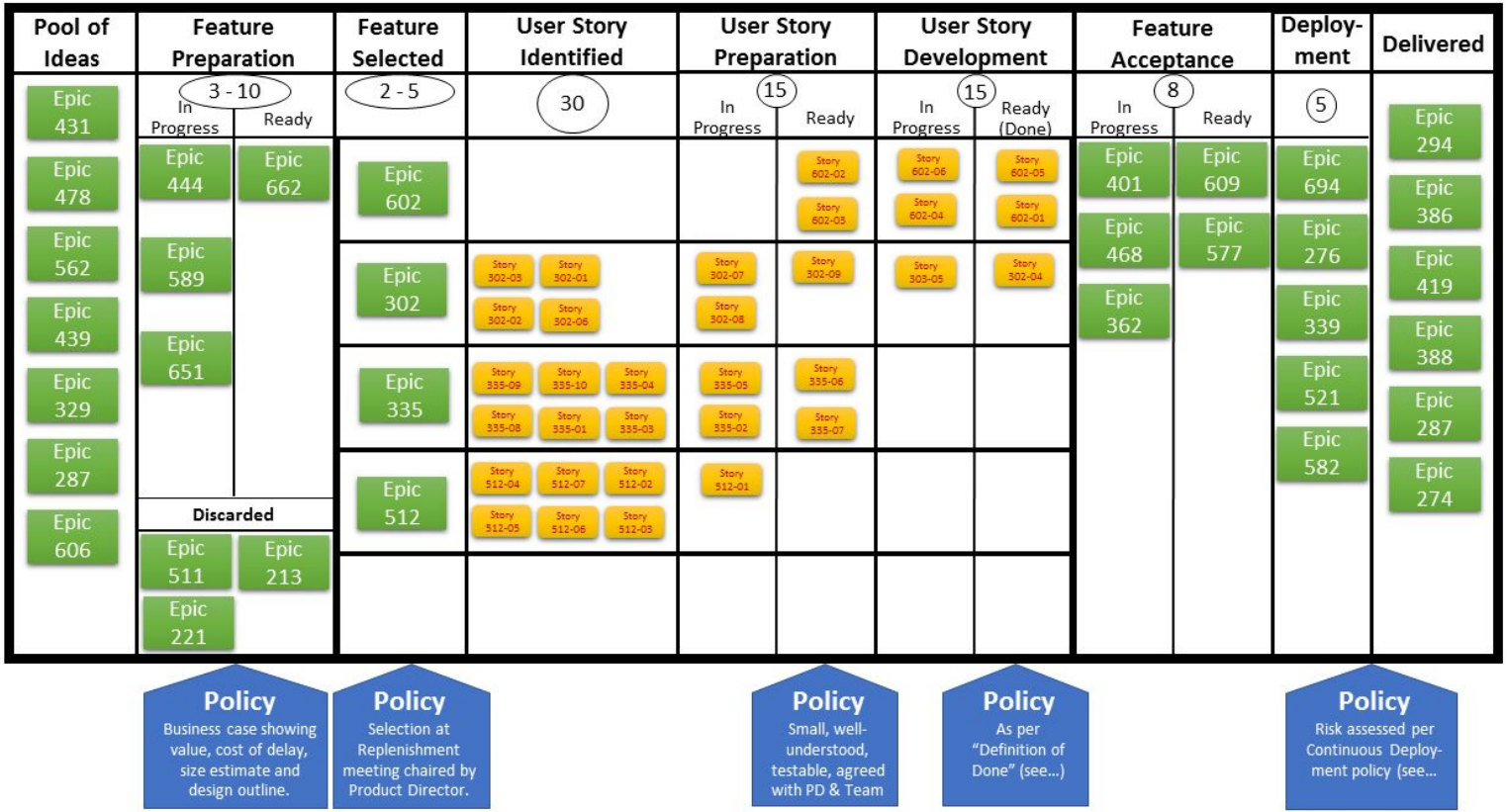
= a visual management method for optimizing work processes and improving efficiency

(Kanban = “signboard” in Japanese)

- Developed in manufacturing (Toyota, 1940s}
- Visual boards
  - Represent workflow of tasks or work items

- Work-in-progress (WIP) limits
- Pull system
  - New work is pulled into the system only when there is capacity to handle it
- The goal is to achieve a continuous flow of work through the system
- Often combined with other agile methodologies and lean principles

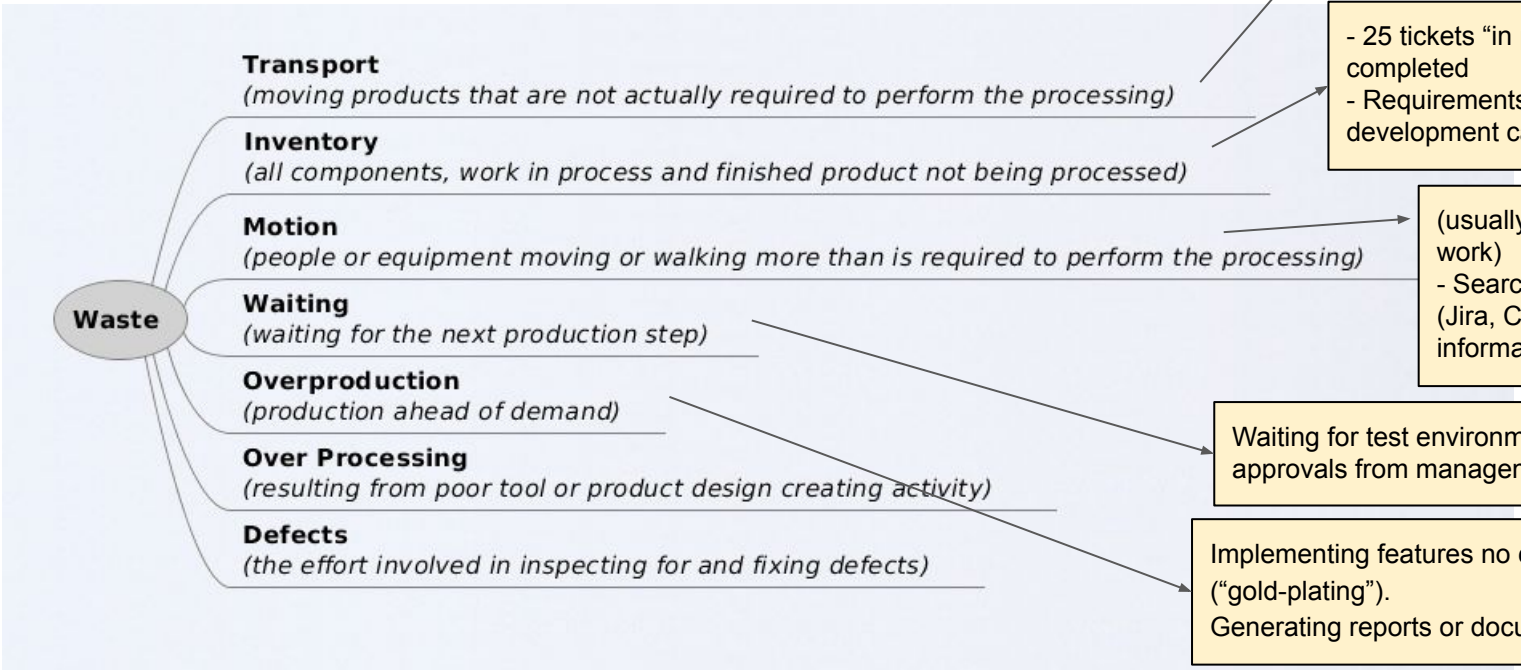
Example



Zdroj: [wikimedia]

Lean management (1)

= a management philosophy and set of principles that aims to maximize customer value while minimizing waste in all aspects of business processes



- Repeatedly passing issues across teams with incomplete information.

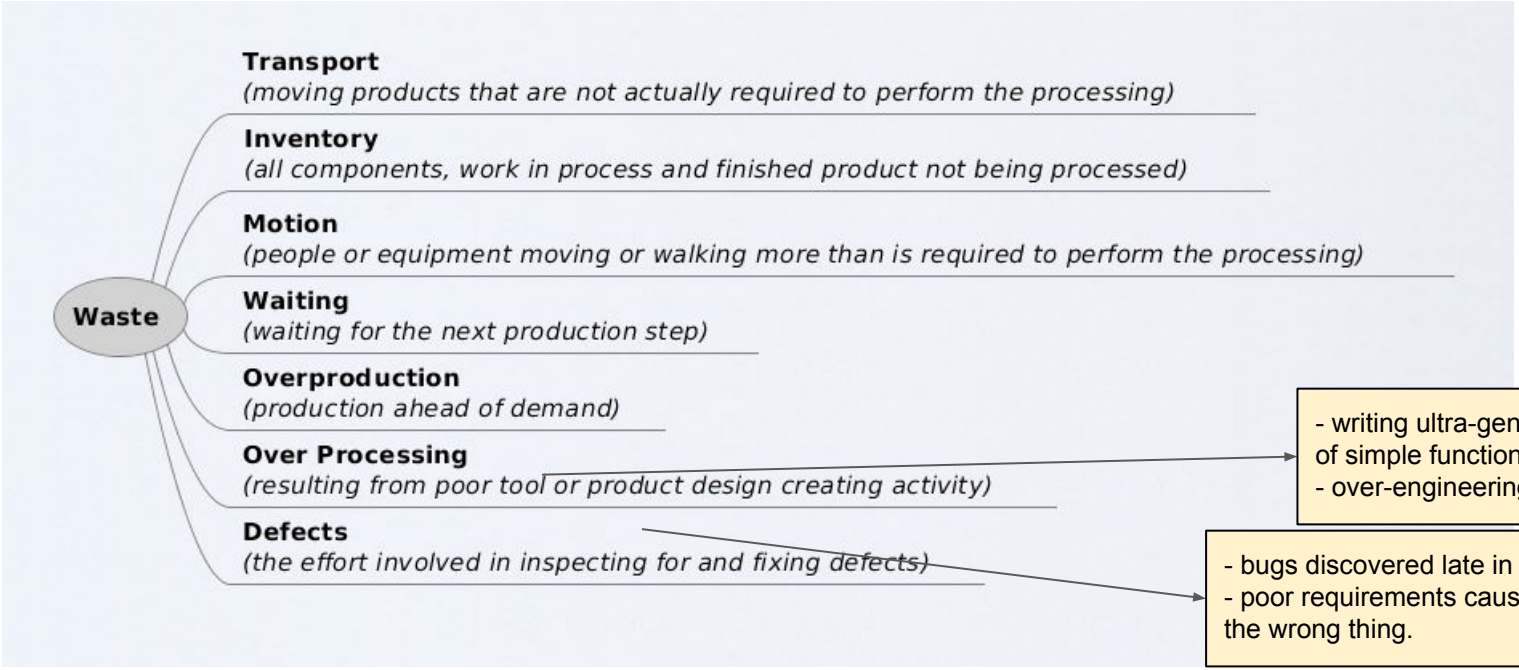
- 25 tickets “in progress”, but none being completed
- Requirements documents piling up while development cannot start

(usually digital motion in software work)
- Searching across multiple tools (Jira, Confluence, Slack, Drive) for information

Waiting for test environments to be available, for approvals from management or product owners, …

Implementing features no customer asked for yet (“gold-plating”).
Generating reports or documents that no one reads.

Lean management (2)



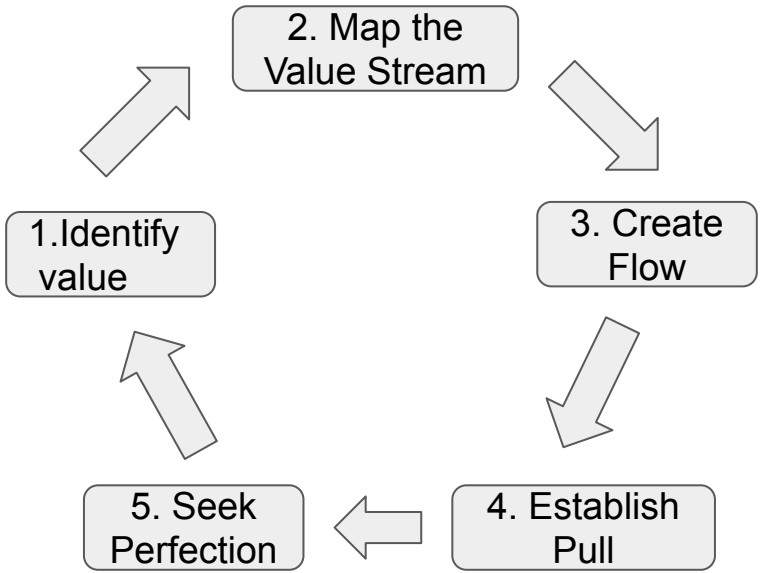
- writing ultra-generic frameworks instead of simple functions
- over-engineering

- bugs discovered late in QA that force rework
- poor requirements causing developers to build the wrong thing.

Zdroj: Wikimedia

Lean management (2)

- Originated from manufacturing (Toyota, 1950s)
- Applies the five principles of lean thinking to project management
  - **Value:** Identify what customers value
  - **Value stream:** Identify all steps that bring product to the customer. Eliminate steps that do not create value.
  - **Flow:** Turn value stream into a smooth and continuous movement of work (e.g., in the form of Kanban Board)
  - **Pull:** Implement a pull system where work is initiated based on customer demand.
  - **Perfection:** Repeat the process again.



Example: Value stream

User story: Add email notification …

Step	PT (active work)	WT (waiting)	Notes
Requirements clarification	2h	1 day	Waiting for PO answers
Coding	8h	—	Actual development
Code review	1h	2 days	Slow reviewer response
CI pipeline	20 min	40 min	Queue time + run
QA testing	1h	1 day	QA backlog
Deployment	10 min	4h	Release window

SUMMARY

Total Process Time: ~12 hours
Total Lead Time: ~4.5 days
Value-Adding Ratio:
12 hrs / 108 hrs ≈ 11%

89% of the time is waiting

Example: Value stream - improved

Step	PT	WT	Notes (Improvements Applied)
Requirements clarification	1h	2h	Clear PO availability, daily refinement
Coding	6h	—	Smaller stories, pairing optional
Code review	30m	2h	WIP limits, SLA same-day review
CI pipeline	10m	5m	Automated, parallel
QA testing	30m	2h	Shift-left, automated tests
Deployment	5m	0	Continuous deployment

SUMMARY

Total Process Time: ~8 h
Total Lead Time: ~14 h
Value-Adding Ratio:
8 hrs / 14 hrs ≈ 57%

Resources & further reading

- Axelos (Certification PRINCE2): What is Project Management?
- Wikipedia - Project Management
- Wikipedia - Project Management Triangle
- Wikipedia - Work Breakdown Structure
- Wikipedia - Gantt chart
- Wikipedia - Critical path method
- Agile manifesto
- Wikipedia - SCRUM
- Ken Schwaber and Jeff Sutherland: The Scrum Guide (PDF)
- SAFe
- SAFe Project Handbook
- Wikipedia - Extreme programming
- Wikipedia - Kanban
- Toyota production system
- Wikipedia - Lean management
- Aziz Moujib: Lean Project Management