

Intro To Process Milestones, Estimation, Planning

17-313 Fall 2026

Foundations of Software Engineering

<https://cmu-17313q.github.io>

Eduardo Feo Flushing

Administrivia

- Project 1B due today
- Recitation tomorrow
 - Meet your team!
- Project 2A will be released tomorrow
- The first problem set will be released this week
 - Quiz #1: Next Tuesday

Today's Learning Goals

- Recognize the importance of process
- Identify why software development has project characteristics
- Understand the elements of Scrum
- Create and evaluate user stories
- Use milestones for planning and progress measurement
- Understand the difficulty of measuring progress

What does this mean?
What else can we do apart
from coding?
☐ *Processes* are **key**
concerns.

**Software Engineering principles,
practices (technical and
non-technical) for confidently
building high-quality software.**

Outline

- Software Processes and why we need them
- Software Process Models
- Planning: Task and progress estimation

Outline

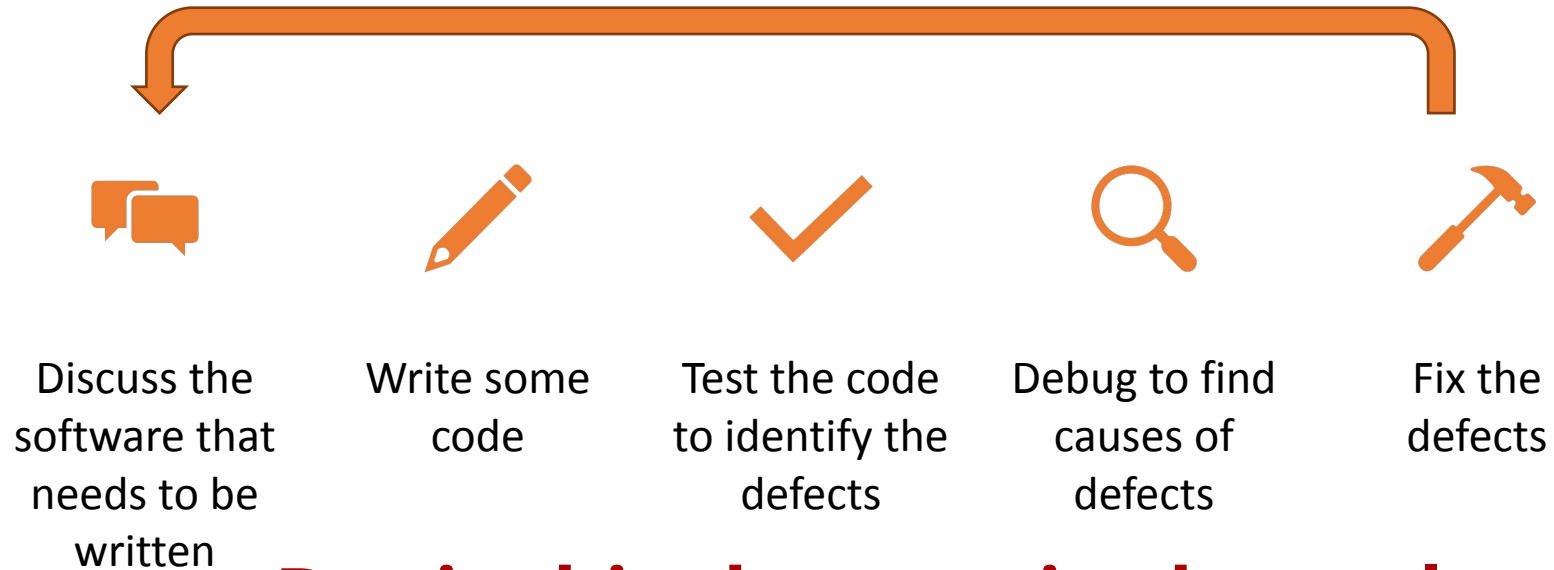
- **Software Processes and why we need them**
 - **Burden or benefit?**
 - **Plan-oriented vs iterative**
- Software Process Models
 - Agile and Scrum
- Challenges of distributed and remote teams

Software Process

“The set of activities and associated results that produce a software product”

Sommerville, SE, ed. 8

How to develop software???



But is this the case in the real world?

Software Development

Coding

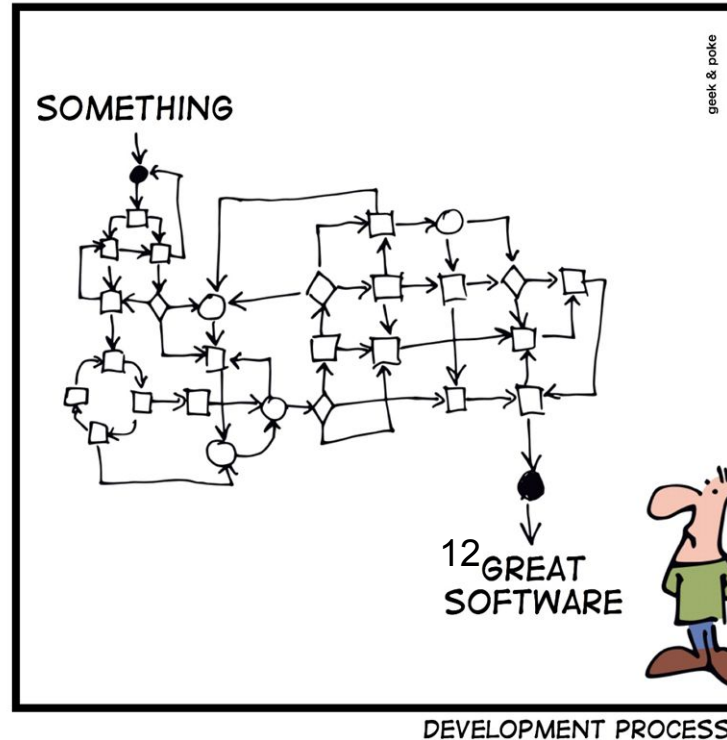
Today Was a Good Day: The Daily Life of Software Developers

André N. Meyer^{ID}, Earl T. Barr^{ID}, Christian Bird, *Senior Member, IEEE*,
and Thomas Zimmermann^{ID}, *Senior Member, IEEE*

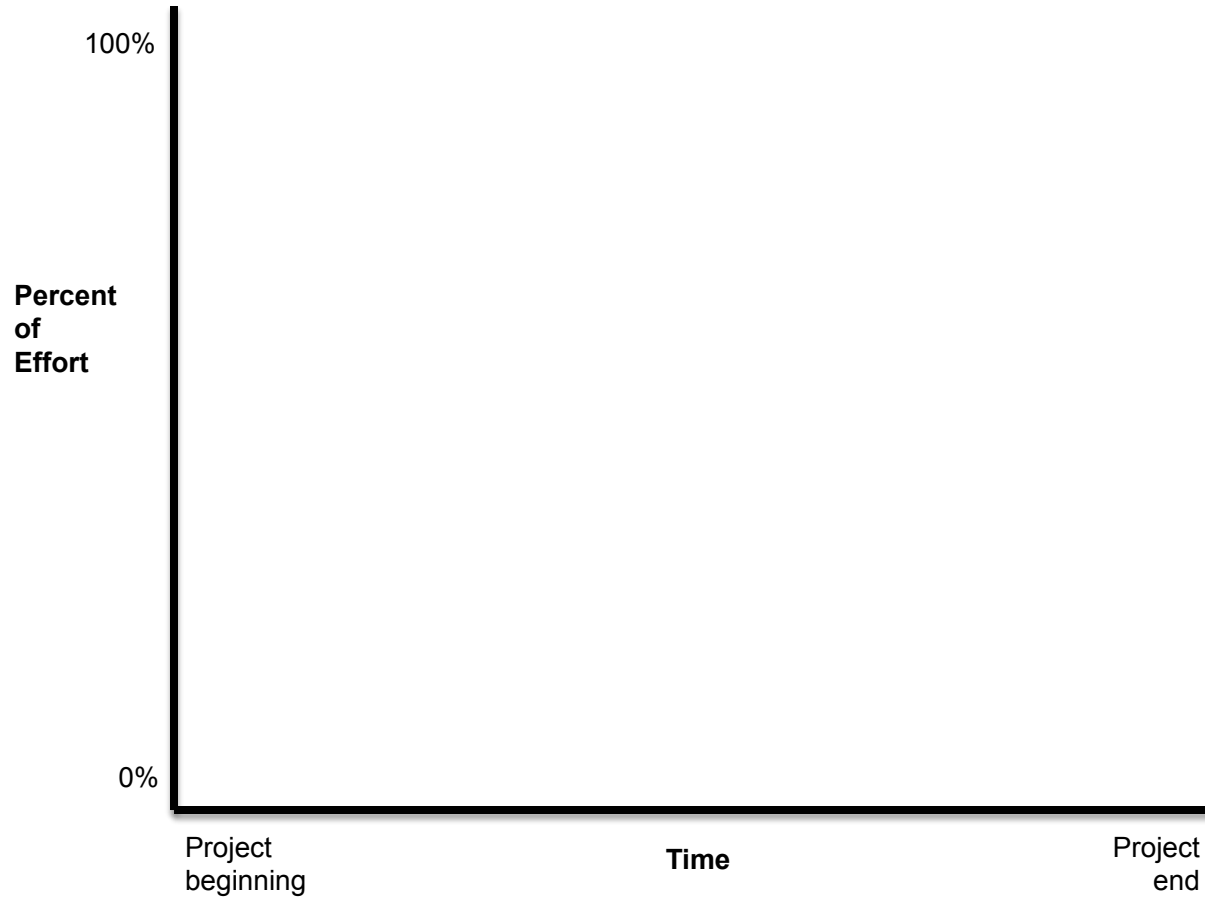
“developers spend surprisingly little time with coding, 9 to 61 percent depending on the study”

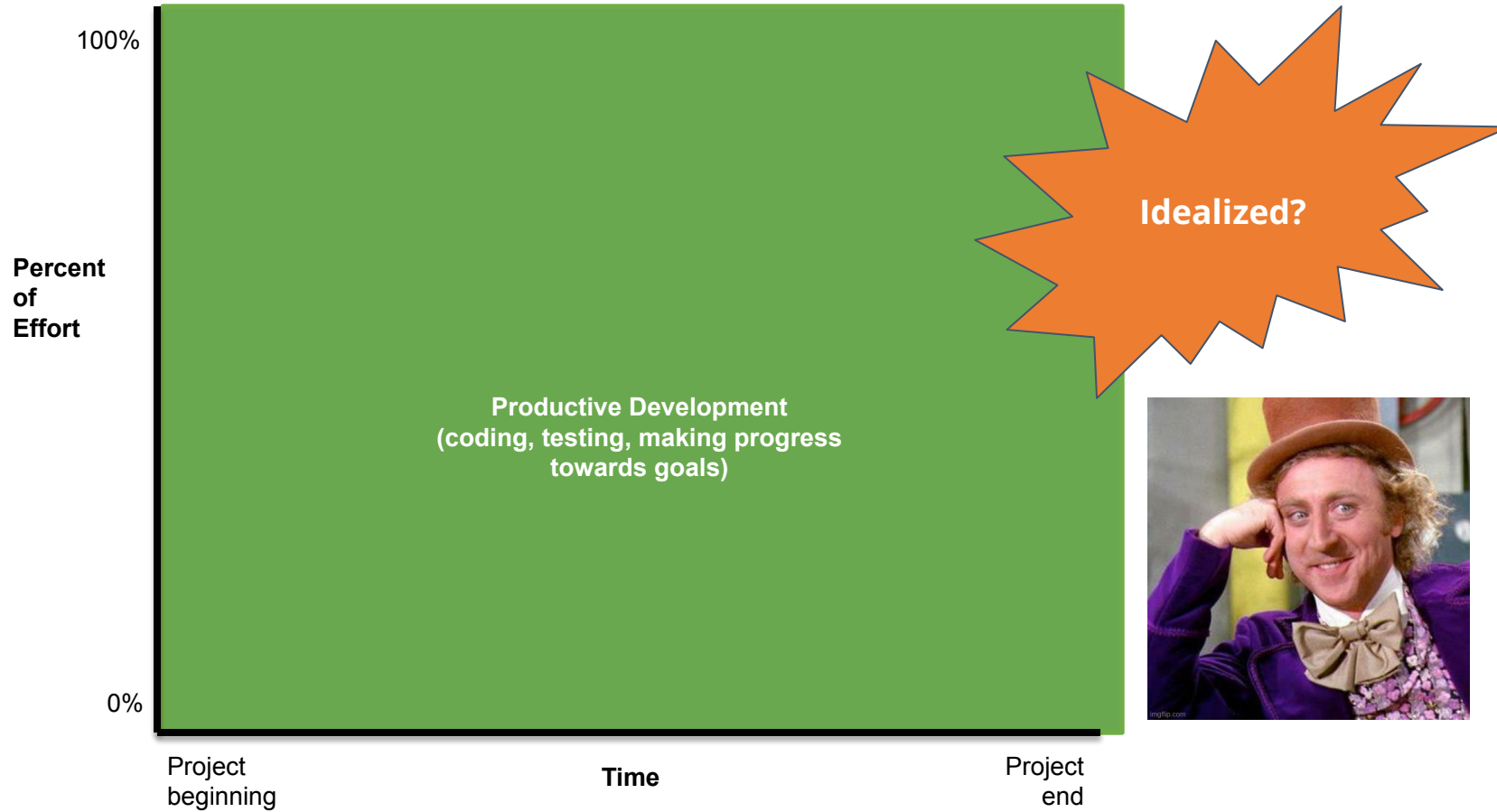
Software development methods

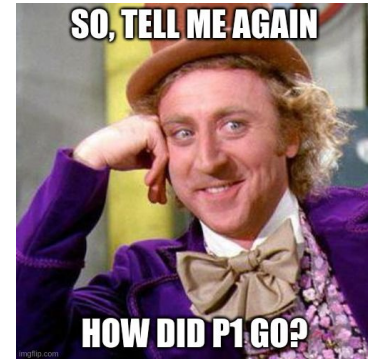
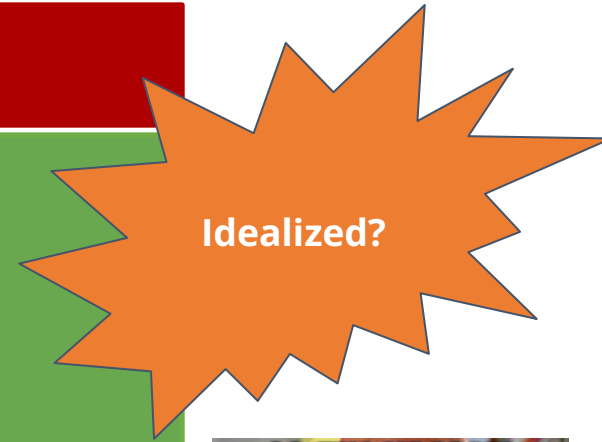
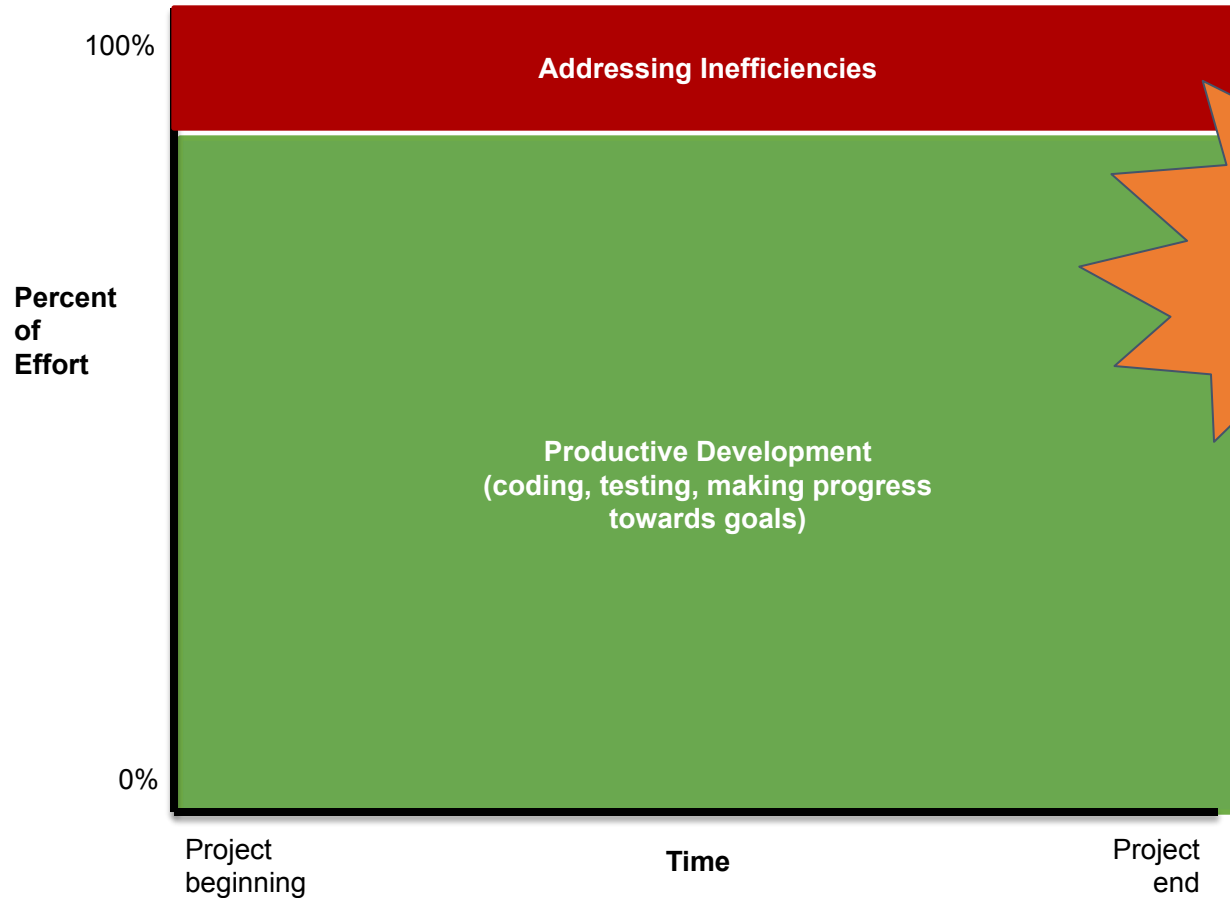
Software development methods are codified practices that support the work of a group and are based on an idea of how is best to work to achieve a particular business goal



Process: Burden or Benefit?

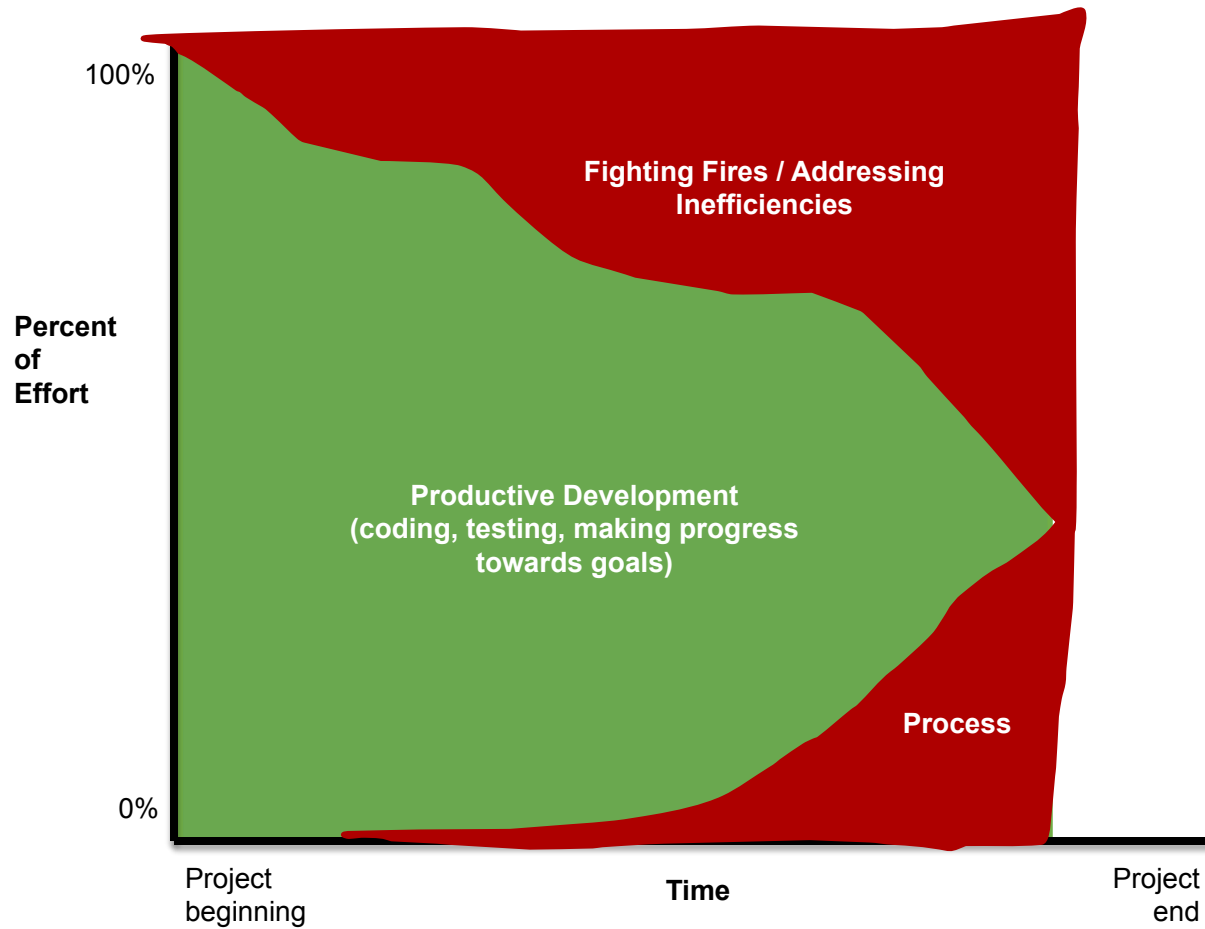






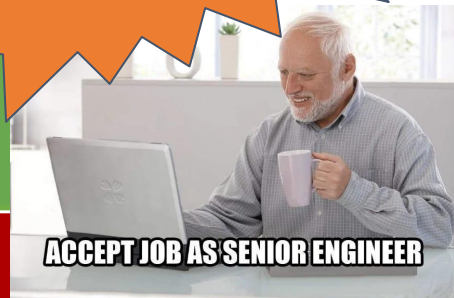
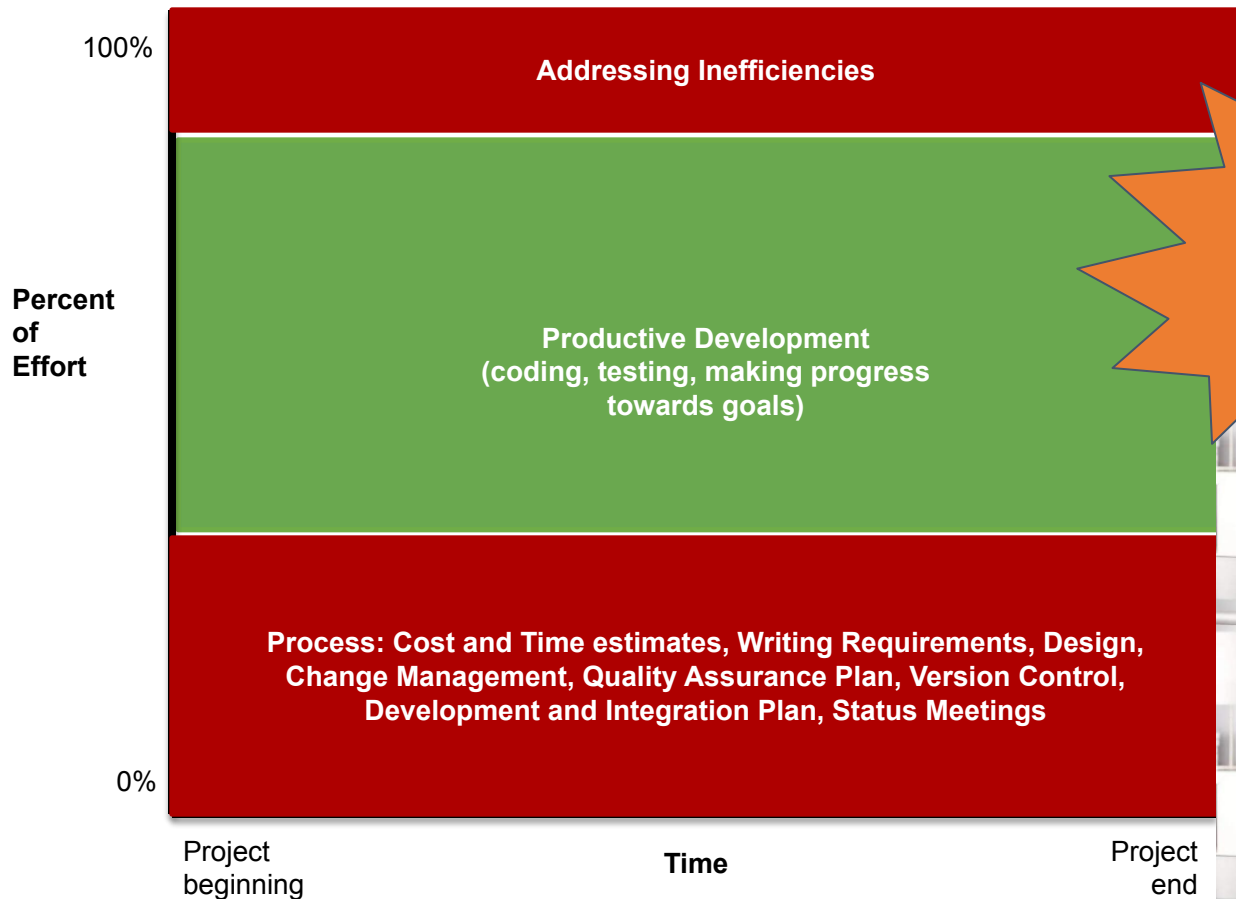
What happens when ...

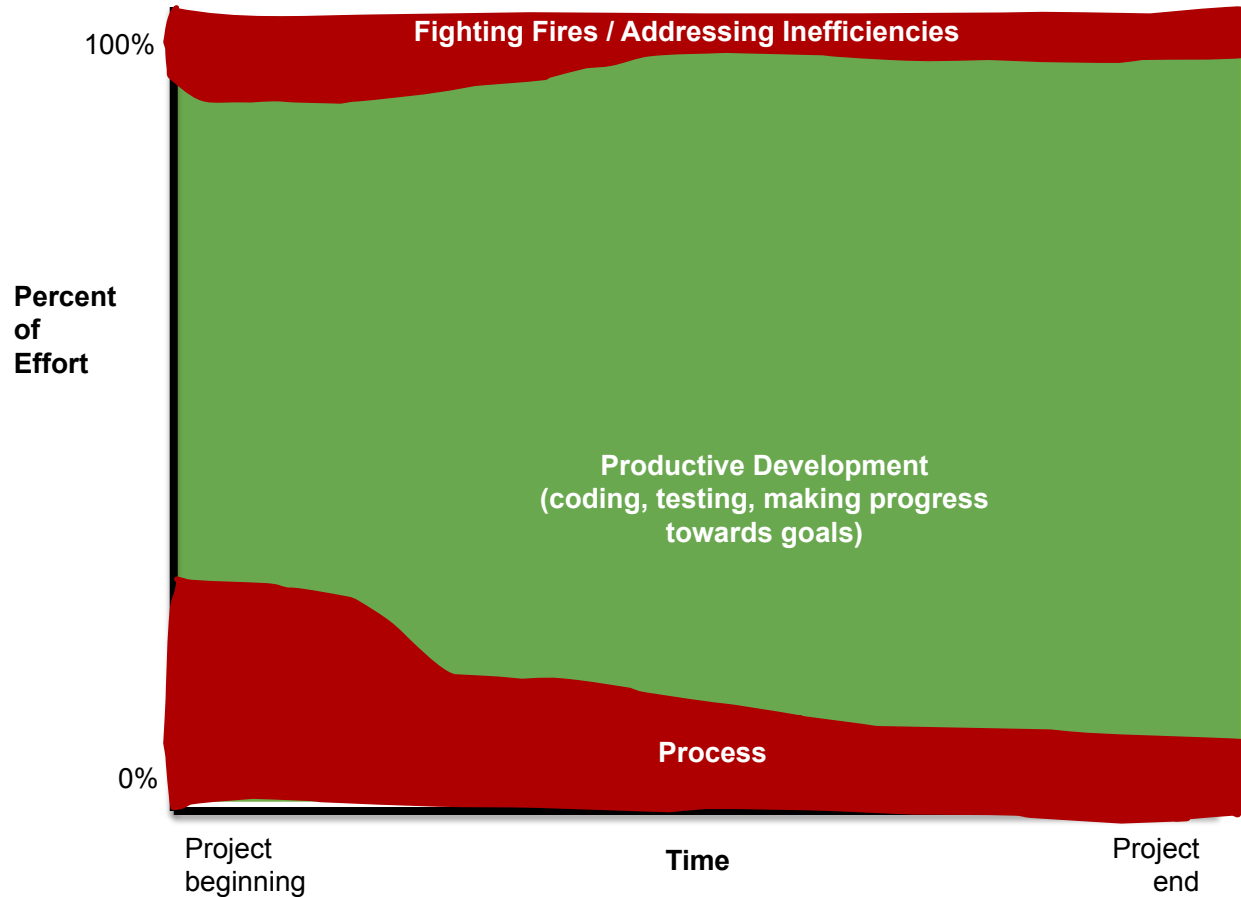
- Mid-project informal agreement to changes suggested by customer or manager. Project scope expands 25-50%
- Late detection of requirements and design issues.
- Bug reports collected informally, forgotten
- Integration of independently developed components at the very end of the project. Interfaces out of sync.
- Accidentally overwritten changes, lost work.
- When project is behind, developers are asked weekly for new estimates.



Let's improve the reliability of this process

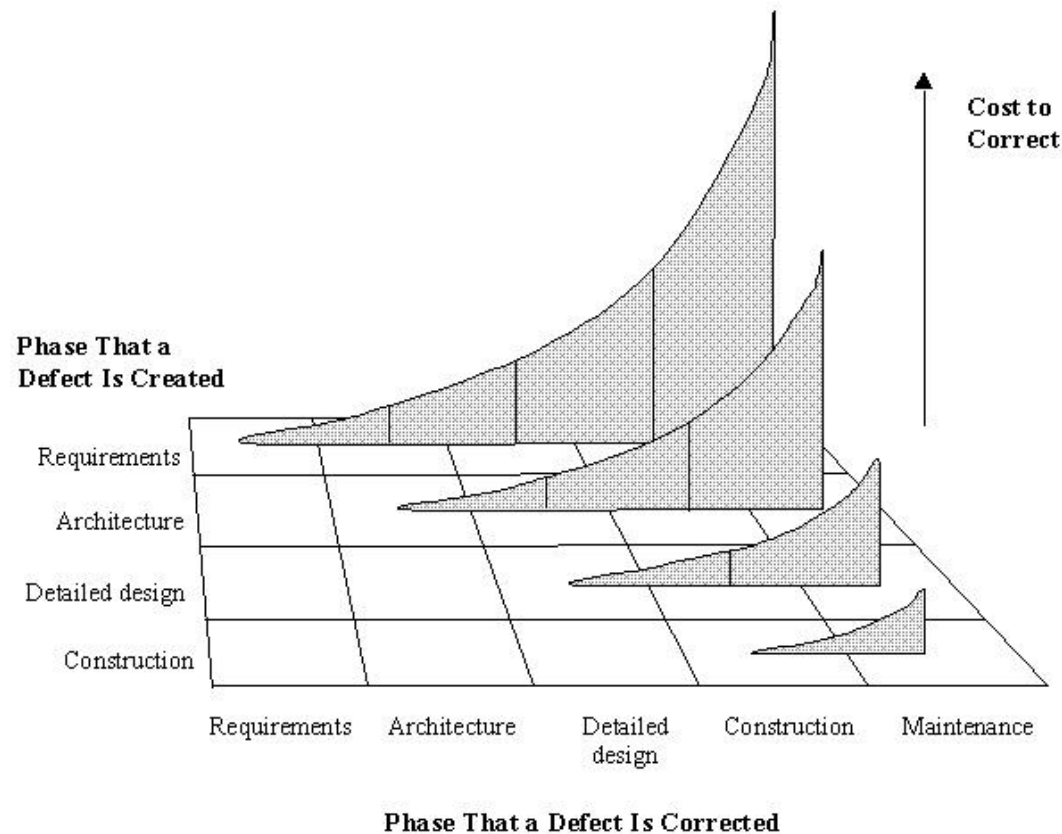
- Writing down all requirements
 - Review requirements
 - Require approval for all changes to requirements
- Use version control for all changes
 - Code Reviews
- Track all work items
 - Break down development into smaller tasks
 - Write down and monitor all reported bugs
 - Hold regular, frequent status meetings
- Plan and conduct quality assurance
- Employ a DevOps framework to push code between developers and operations





Hypothesis: Process increases flexibility and efficiency

Ideal Curve: Upfront investment for later greater returns

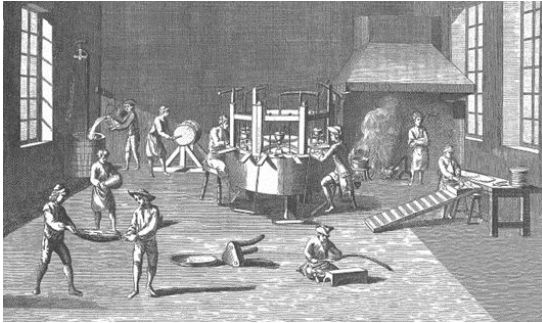


Copyright 1998 Steven C. McConnell. Reprinted with permission
from *Software Project Survival Guide* (Microsoft Press, 1998).

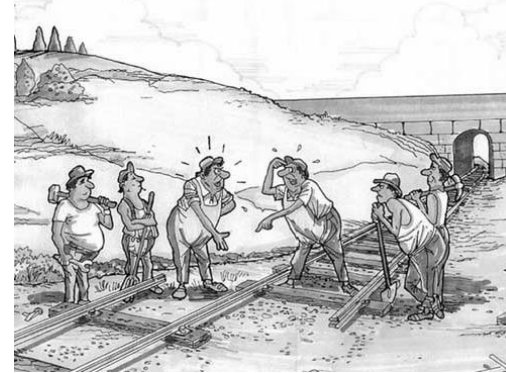
Outline

- Software Processes and why we need them
 - Burden or benefit?
 - **Plan-oriented vs iterative**
- Software Process Models
 - Agile and Scrum
- Challenges of distributed and remote teams

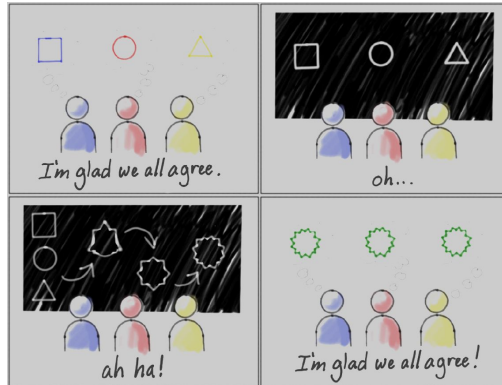
Successful projects require that their members:



... divide the work ...



... coordinate their activities and resources ...

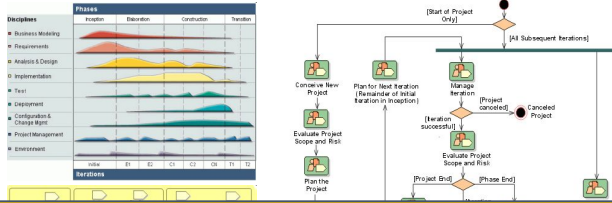


... achieve a shared understanding of the work to be done ...



... and align their efforts

Software process models are often categorized as plan-driven versus iterative



In practice, most processes combine elements of both



	Plan oriented	Iterative
What it values	The software will be delivered as promised, on time and on budget	We'll get you something working faster
Scope	Work management and technical practices. End-to-End	Mostly work management. Currently focused on construction
Knowledge management	Knowledge is made explicit through extensive documentation	Knowledge is mostly tacit
Human resource management	Specialized workforce Team members join as required	Cross-functional workforce Team members are assigned to the project at its outset, they are collocated and dedicated full time
Coordination	Reference to higher authority, preparation	Mutual adjustment
Work control	Project manager (bureaucratic control)	self-managed (clan control)

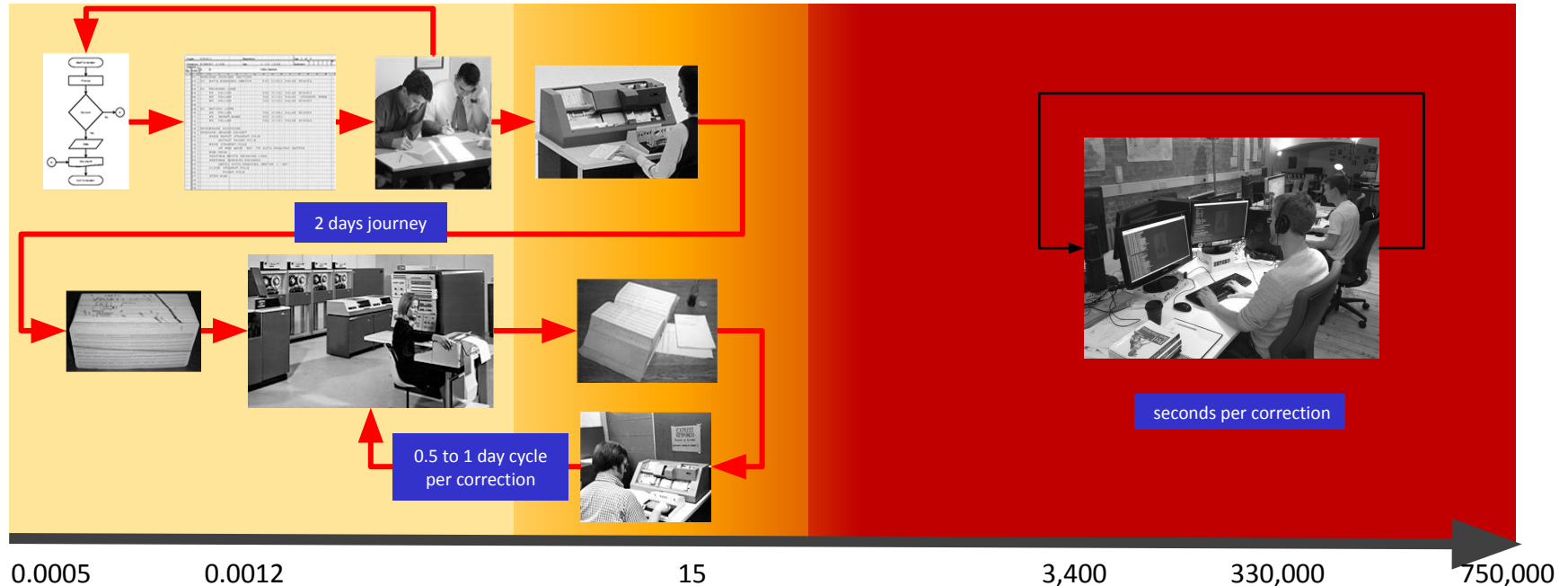
Quick Discussion: Process Trade-Off Scenarios

- Work in groups
- Scenario
 1. A medical record system for a hospital.
 2. Social media app for students.
 3. A one-week hackathon project.
- Would you use a more plan-driven or iterative approach for this project? Why?

The evolution of software development methods: A hardware perspective

Focus on resource utilization

Focus on time to market



MIPS per 1,000\$

1968

1970

1984

1986

1996

2001

2010+

NATO conference
coins the term
Software Engineering

Royce describes (&
criticizes) the "waterfall"
model of software
development

SEI is instituted by DoD
Maturity Models

Takeuchi & Nonaka. The
New Product
Development Game

Xp is born out of the C3
project at Chrysler

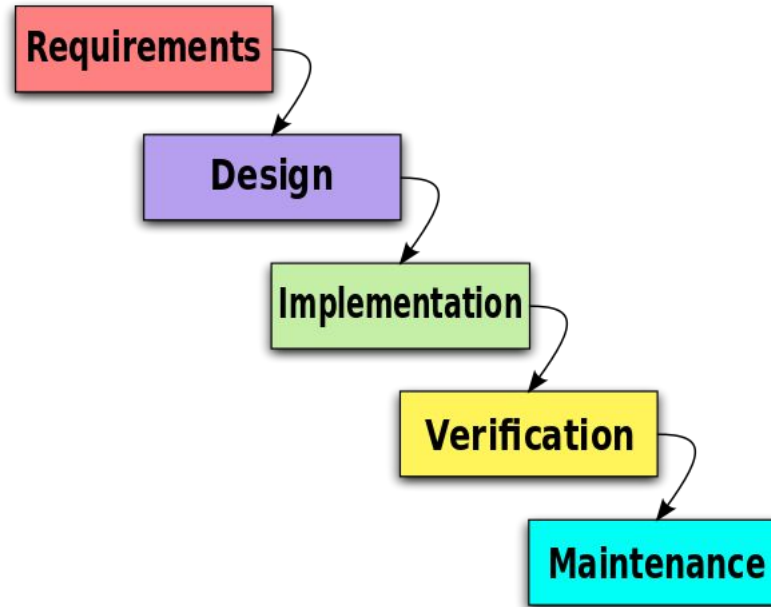
Agile Manifesto

DAD, Safe, DevOps

Outline

- Software Processes and why we need them
- **Software Process Models**
 - **Agile and Scrum**
- Planning: Task and progress estimation

Waterfall model was the original software process



Waterfall diagram CC-BY 3.0 [Paulsmith99](#) at [en.wikipedia](#)

... akin to processes pioneered in mass manufacturing (e.g., by Ford)



Lean production adapts to variable demand

Toyota Production System (TPS) Late 1940s

Build only what is needed, only when it is needed.

Use the “pull” system to avoid overproduction. (Kanban)

Stop to fix problems, to get quality right from the start (Jidoka)

Workers are multi-skilled and understand the whole process; take ownership

Enabling teams to have autonomy and control to change/improve quickly/continuous improvement (kaizen)

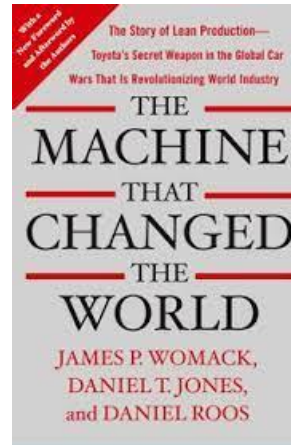


Taiichi Ohno

From TPS to Agile



1986

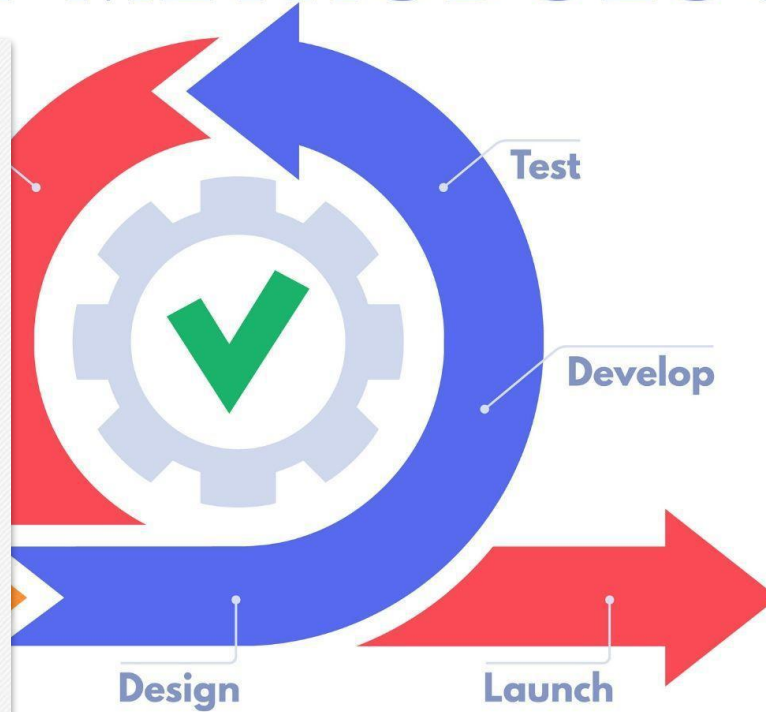


1990



2001

AGILE METHODOLOGY



Scrum

(Only a brief intro)

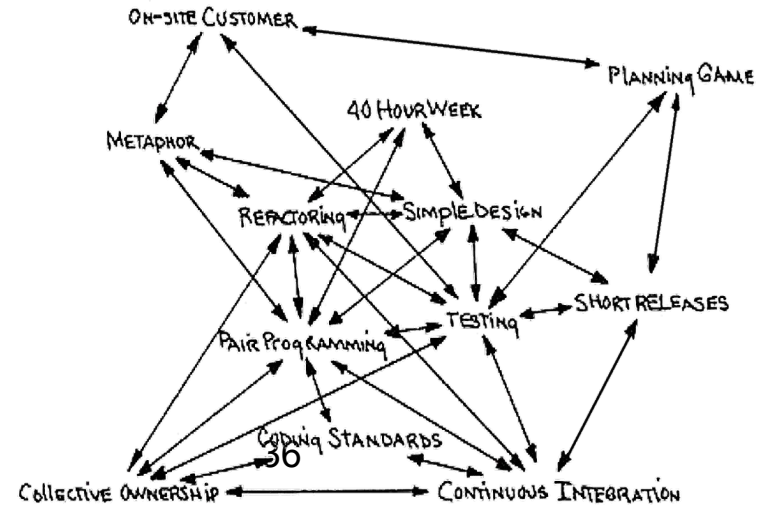


If we are going to change any of a method's practices, we need to understand the role they play and the interactions between them

"Any one practice doesn't stand well on its own. They require the other practices to keep them in balance."

Figure 4 is a diagram that summarizes the practices. A line between two practices means that the two practices reinforce each other. I didn't want to present this picture first, because it makes XP look complicated. The individual pieces are simple. **The richness comes from the interactions of the parts**"

XP: Extreme Programming Explained, K. Beck, 1999

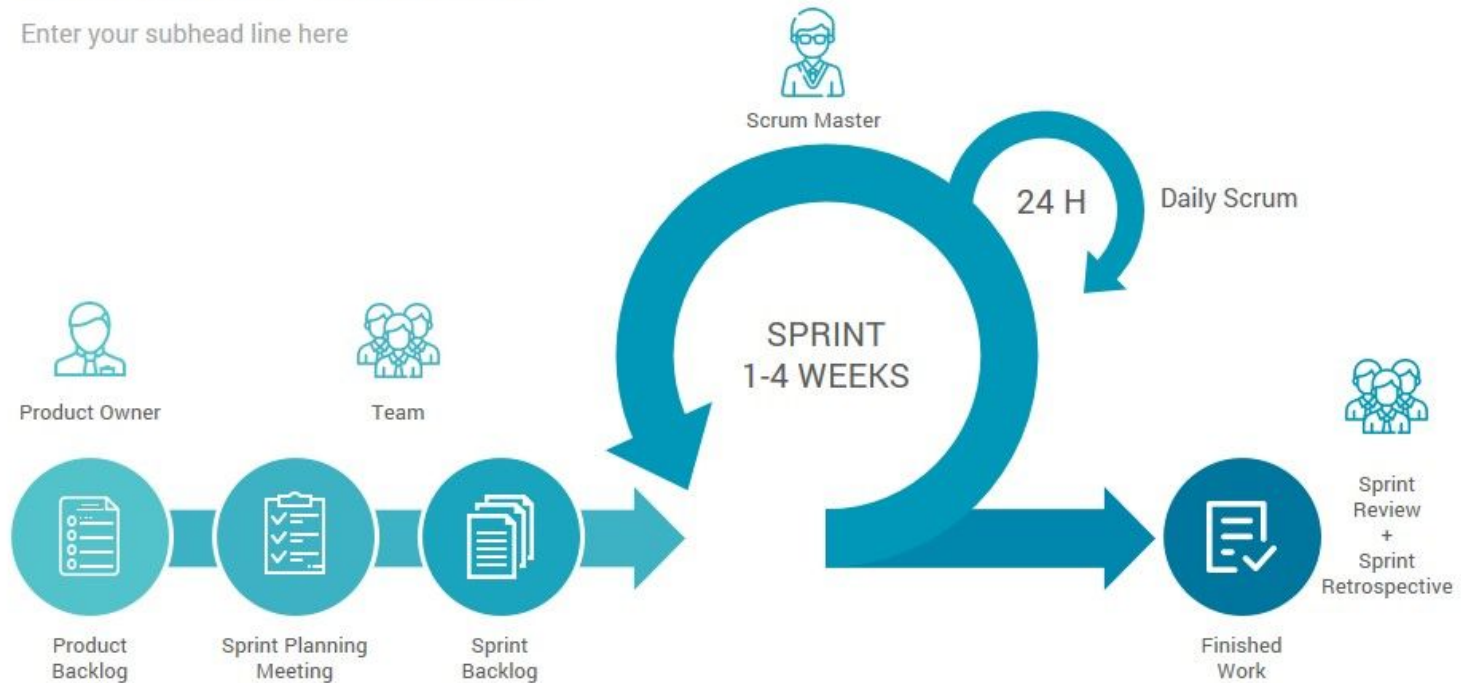


Method's practices support each other

Elements of Scrum

Scrum Process

Enter your subhead line here



Backlogs

The **product backlog** is all the features for the product

The **sprint backlog** is all the features that will be worked on for that sprint. These should be broken down into discrete tasks:

- Fine-grained

- Estimated

- Assigned to individual team members

- Acceptance criteria should be defined

User Stories are often used

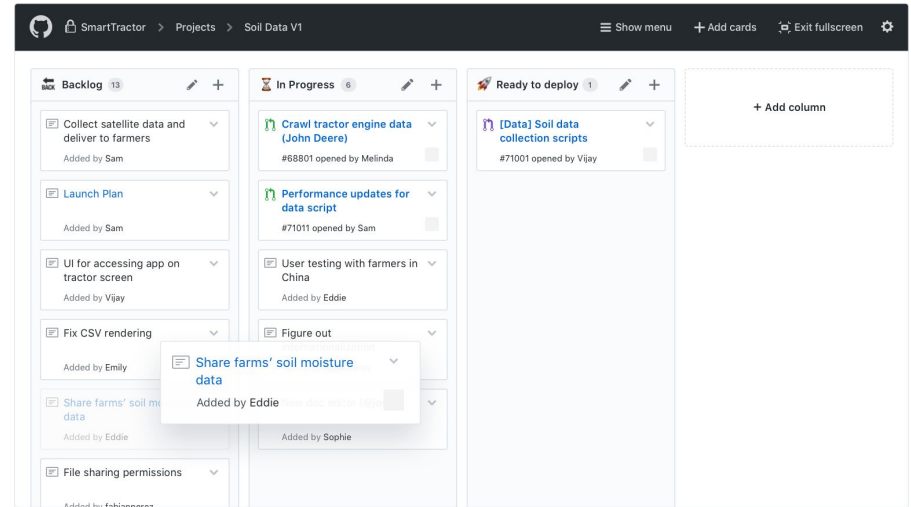
Backlog: ordered list of everything that might be needed in the product.

ToDo List

ID	Story	Estimation	Priority
7	As an unauthorized User I want to create a new account	3	1
1	As an unauthorized User I want to login	1	2
10	As an authorized User I want to logout	1	3
9	Create script to purge database	1	4
2	As an authorized User I want to see the list of items so that I can select one	2	5
4	As an authorized User I want to add a new item so that it appears in the list	5	6
3	As an authorized User I want to delete the selected item	2	7
5	As an authorized User I want to edit the selected item	5	8
6	As an authorized User I want to set a reminder for a selected item so that I am reminded when item is due	8	9
8	As an administrator I want to see the list of accounts on login	2	10
Total		30	

- Features / new functionality
- Bug fixes
- Technical work (e.g., refactoring, infrastructure tasks)
- Knowledge acquisition (e.g., research tasks)

Kanban boards



Scrum Meetings

Sprint Planning Meeting

Entire Team decides together what to tackle for that sprint

Daily Scrum Meeting

Quick Meeting to touch base on :

What have I done? What am I doing next? What am I stuck on/need help?

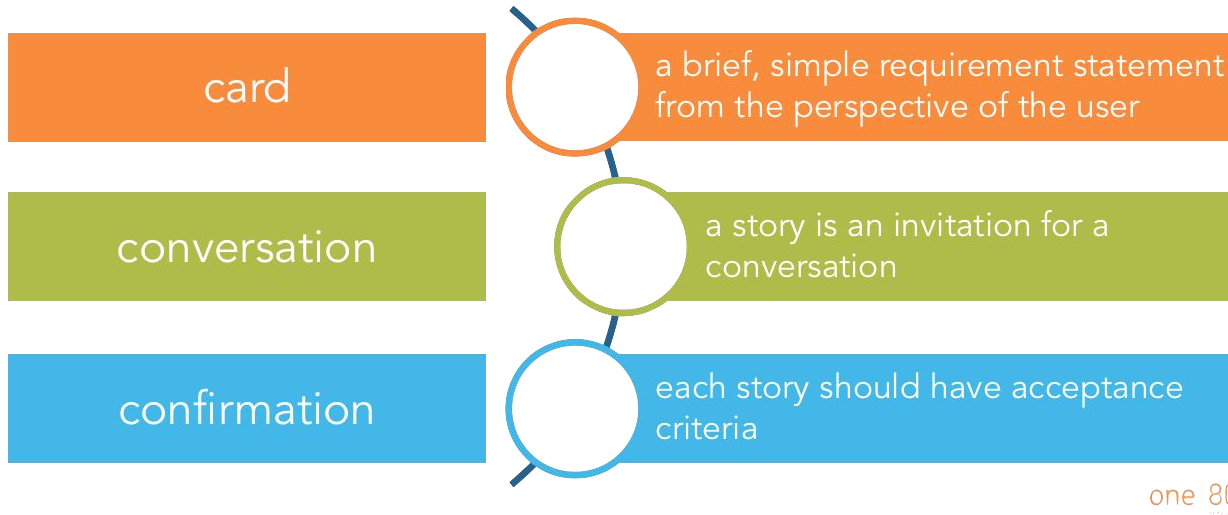
Sprint Retrospective

Review sprint process

Sprint Review Meeting

Review Product

User Stories: How to write backlog entries for features



User story cards (3"x5")

"As a [role], I want [function], so that [value]"

Conversation

- What must a developer do to implement this user story?

Confirmation

- How can we tell that the user story has been achieved?
- It's easy to tell when the developer finished the code.
- But, how do you tell that the customer is happy?

How to evaluate user story?

Follow the INVEST
guidelines for good
user stories!



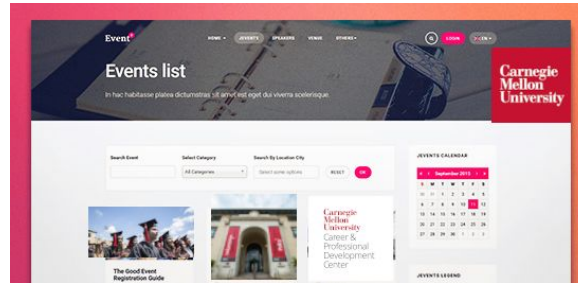
Source:

<http://one80services.com/user-stories/writing-good-user-stories-hint-its-not-about-writing/>

one | 80
services

Example

The university is looking to enhance student and staff engagement by creating an online platform where all university-related events are easily accessible. The goal is to provide a user-friendly website that serves as a central hub for information on various activities, ranging from academic seminars to sports events and club meetings.



Independent



- Schedule in any order.
- Not always possible

Counterexample

As a student, **I want to** receive notifications for events that are about to start, for those I have shown interest in, **so I** don't miss them.

Acceptance Criteria:

- An option is provided to 'Set a Reminder' for each event.
- Notifications are sent to users who have opted for reminders, shortly before the event starts.

Assumes that the homepage with an event calendar is already in place!

I	independent	☒
N	negotiable	☐
V	valuable	☐
E	estimable	☐
S	small	☐
T	testable	☐

Negotiable



- Details to be negotiated during development
- Good Story captures the essence, not the details

Counterexample

As a student, I want to view the upcoming events at the university, **so I** can decide which ones to attend.

Acceptance Criteria:

- Add an interactive grid layout of upcoming events at the top of the homepage.
- Each event card in the grid is visible for a 2 seconds before automatically rotating to display the next set of events.
- Each card in the grid includes the event's name, type (e.g., seminar, sports game), duration, a brief description, and scheduled times.
- This grid of events is displayed under a prominent H1 heading that reads "Discover What's Happening on Campus!"

Too many
implementation
details and constraints

I	independent	☒
N	negotiable	☒
V	valuable	☐
E	estimable	☐
S	small	☐
T	testable	☐

Valuable



- This story needs to have value to someone (hopefully the customer)

Counterexample

As the Events Coordinator, **I want** a database to store details of students and staff interested in university events.

Acceptance Criteria:

- A database is constructed to manage user information.
- The database stores details such as name, email, phone number, favorite event types, date of birth, and history of event attendance or registrations.

Why is this feature important?!

I	independent	☒
N	negotiable	☒
V	valuable	☒
E	estimable	☐
S	small	☐
T	testable	☐

Estimable



- Helps keep the size small
- It should provide enough details to estimate the amount of effort needed
- More on estimates later...

Counterexample

As an undergraduate student, **I want to** be able to filter university events, **so I** can choose the ones that align with my interests.

Acceptance Criteria:

- Filters are added to the event listings on the website.

Too vague. Very difficult to estimate

I	independent	☒
N	negotiable	☒
V	valuable	☒
E	estimable	☒
S	small	☐
T	testable	☐

Small



- Fit on 3x5 card
- At most two person-weeks of work (one sprint)
- Too big == unable to estimate

Counterexample

As a student, I want to easily find information about upcoming events, **so** I can participate in activities that interest me.

Acceptance criteria:

- A homepage is created displaying the university's name, motto, location, email, and contact information.
- The homepage features a calendar of upcoming university events.
- The event calendar includes details such as the event title, type (e.g., seminar, sports game, club meeting), a brief description, location, date, and time.
- Users can filter the event list by event type, date, and hosting department or club.
- The admin can update the event calendar as new events are planned or existing events are modified.

Too large. It can be decomposed into multiple tasks.

I	independent	☒
N	negotiable	☒
V	valuable	☒
E	estimable	☒
S	small	☒
T	testable	☐

Testable



- Ensures understanding of task
- We know when we can mark task “Done”
- Unable to test == do not understand

Counterexample

As a student, **I want to** easily view promotional videos or trailers of university events, **so I** can decide which events to attend.

Acceptance Criteria:

- Promotional videos can be embedded on each event detail page.
- Videos are of high quality.
- The embedded video is well-integrated into the page design.
- The video size is large enough to ensure clarity.
- The video controls are user-friendly.

Acceptance criteria is subjective

I	independent	☒
N	negotiable	☒
V	valuable	☒
E	estimable	☒
S	small	☒
T	testable	☒

Activity: Evaluate using INVEST

Follow the INVEST
guidelines for good
user stories!



one|80
SERVICES



User Story #1

As the Events Coordinator, **I want** the website to seamlessly integrate with various academic calendars and departmental schedules, **so that** event information is always synchronized and accurate.

Acceptance Criteria:

- The website integrates with different academic and departmental calendars.
- Event information on the website reflects real-time updates from these calendars.

How can you fix it?

Select the most serious flaw



I	independent	☐
N	negotiable	☐
V	valuable	☐
E	estimable	☐
S	small	☐
T	testable	☐

User Story #2

As a student, **I want** the website to have an intuitive navigation system **so that** I can find events effortlessly.

Acceptance Criteria:

- The website's navigation is intuitive to users.
- Users can find events with minimal effort.
- The navigation system feels natural and easy to understand.

How can you fix it?

Select the most serious flaw



I	independent	☐
N	negotiable	☐
V	valuable	☐
E	estimable	☐
S	small	☐
T	testable	☐

Discuss in groups. Compare your answers with each other

User Story #1

As the Events Coordinator, **I want** the website to seamlessly integrate with various academic calendars and departmental schedules, **so that** event information is always synchronized and accurate.

Acceptance Criteria:

- The website integrates with different academic and departmental calendars.
- Event information on the website reflects real-time updates from these calendars.

User Story #2

As a student, **I want** the website to have an intuitive navigation system **so that** I can find events effortlessly.

Acceptance Criteria:

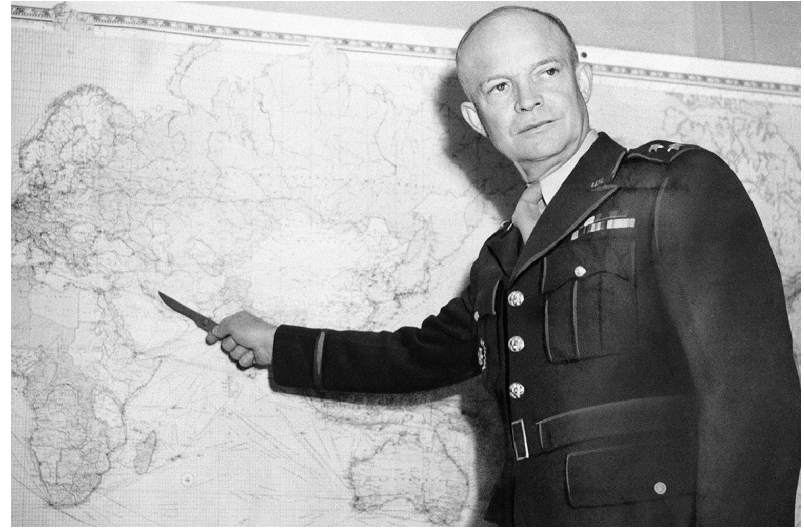
- The website's navigation is intuitive to users.
- Users can find events with minimal effort.
- The navigation system feels natural and easy to understand.

Outline

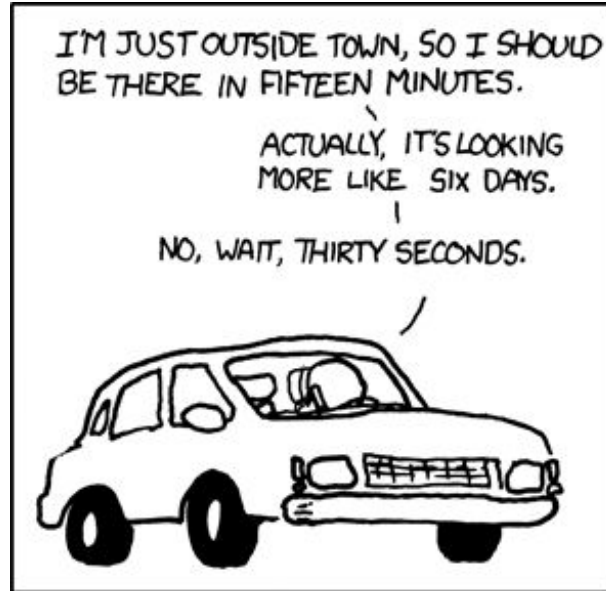
- Software Processes and why we need them
- Software Process Models
 - Agile and Scrum
- **Planning: Task and progress estimation**

*“Plans are nothing,
planning is everything”*

-Dwight D. Eisenhower

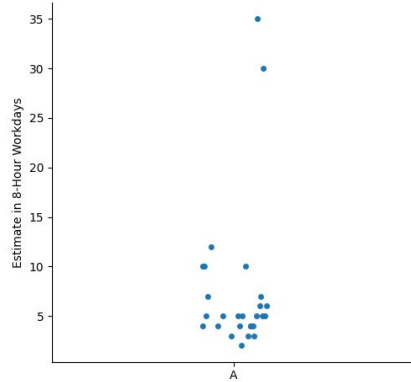
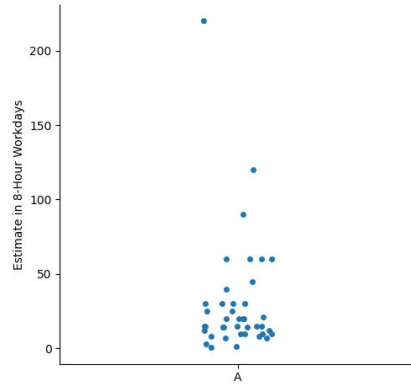


Time estimation

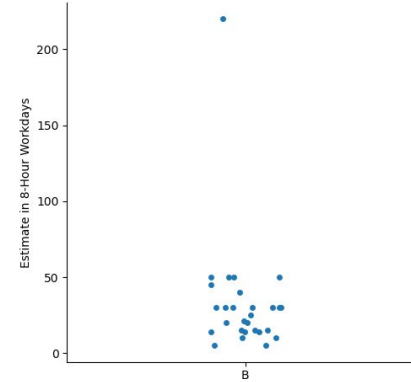
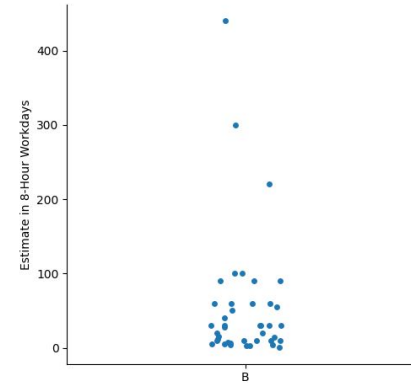


THE AUTHOR OF THE WINDOWS FILE
COPY DIALOG VISITS SOME FRIENDS.

Task 1



Task 2



Improving Time Estimates

- Prevent conformity bias
- Do you have a comparable experience to base an estimate on?
- How much design do you need for each task?
- Break down the task into smaller tasks and estimate them.



XS



S



M



L

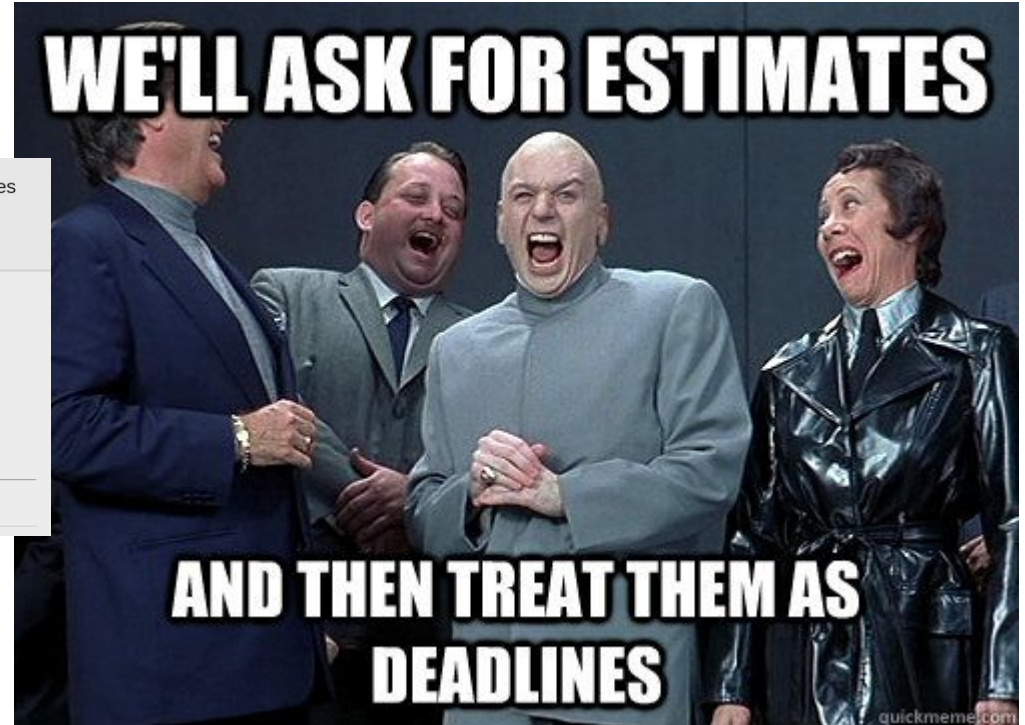


XL

made by **:codica**

codica.com

Is Estimation Evil?



Ron Jeffries

About Search Site Categories

Estimation is Evil

© Feb 1, 2013 • [\[Agile-Related, estimation\]](#)

The following article is recovered from the February 2013 issue of the Pragmatic Programmers magazine.

Overcoming the Estimation Obsession

Ron Jeffries's essay [Estimation is Evil](#)

Measuring Progress?

- Developer judgment: x% done
- Lines of code?
- Functionality?
- Quality?



Measuring Progress?

- “I’m almost done with the app. The frontend is almost fully implemented. The backend is fully finished except for the one stupid bug that keeps crashing the server. I only need to find the one stupid bug, but that can probably be done in an afternoon. We should be ready to release next week.”

Milestones and deliverables make progress *observable*

Milestone: clear end point of a (sub)tasks

- For project manager
- Reports, prototypes, completed subprojects
- "80% done" is not a suitable milestone

Deliverable: Result for customer

- Similar to milestones, but for customers
- Reports, prototypes, completed subsystems

What you need to know

- Recognize the importance of having a software process
- Main ideas of Agile/Scrum
- Understand backlogs and user stories
- Understand the difficulty of estimating tasks and progress
- We use milestones for planning and progress measurement