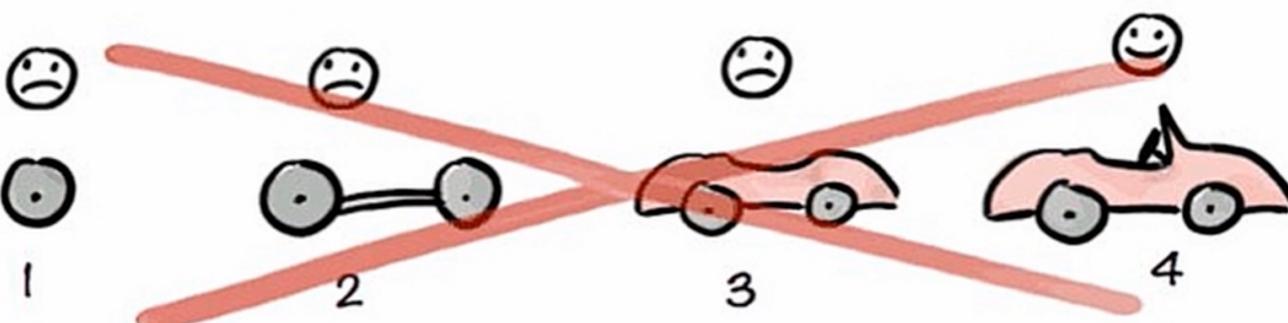


# Agile Course

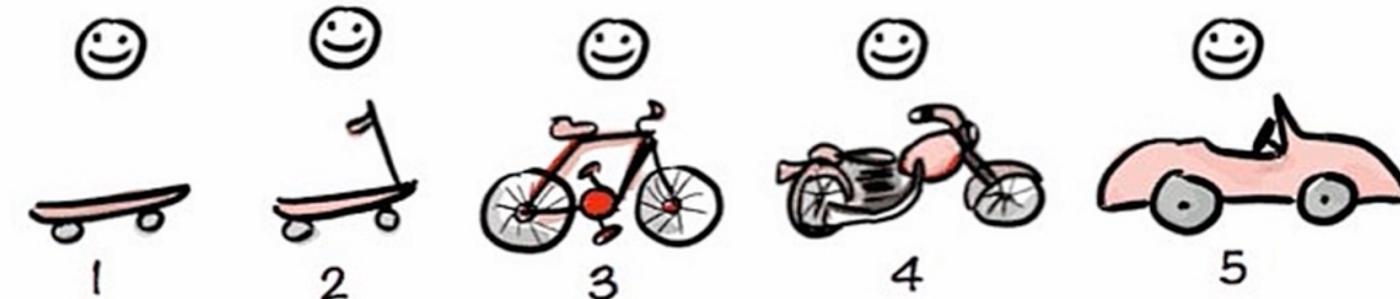
# What is Agile?

Not like this....



Traditional Project Delivery

Like this!



Agile Project Delivery

Traditional

Don't build this...



Agile

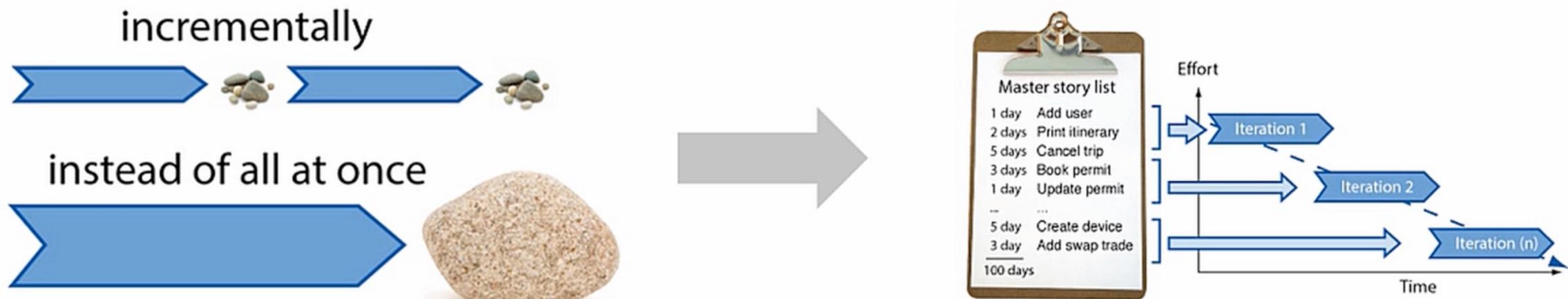
if all you need is this.



# What is Agile?

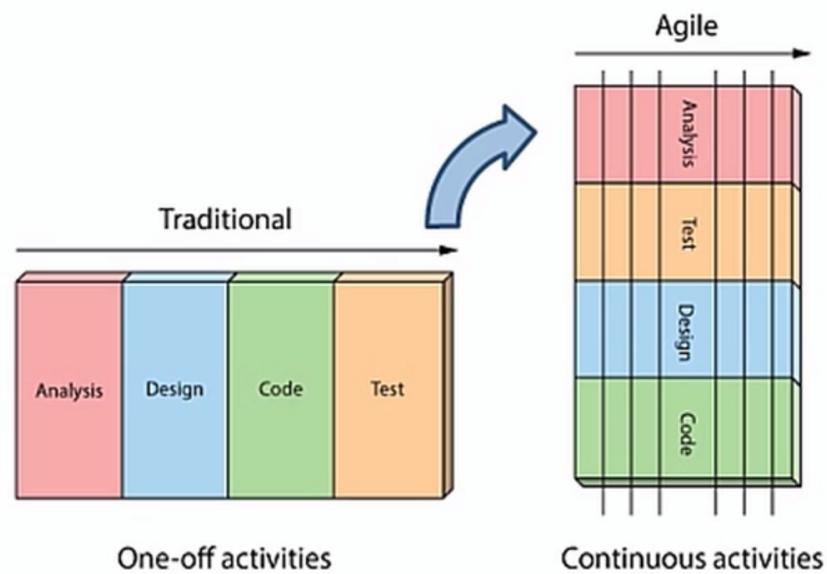
Agile is a methodology that delivers software or value incrementally from the start of the project instead of all at once at the end. It's iterative and time boxed.

It works by breaking projects down into little bits of user functionality called user stories, prioritizing them, and then continuously delivering them in short 2-week cycles called sprints (iterations).



# How is it different?

Analysis, design, coding, and testing are continuous activities.



Scope can vary.

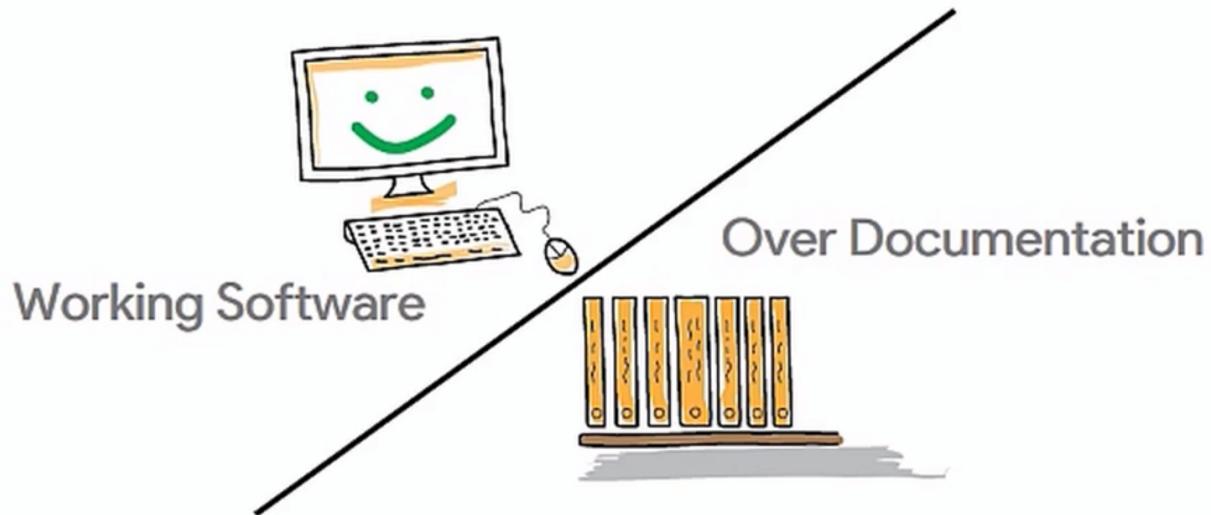


Development is iterative.



# How is it different?

Working software is the primary measure of success.



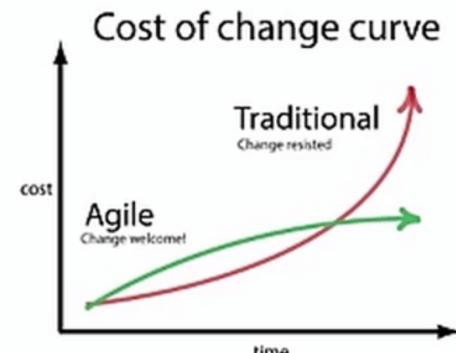
Roles blur.



Planning is adaptive.



Requirements can change.



# Agile Principles

1. Customer satisfaction by early and continuous delivery of useful software.
2. Welcome changing requirements, even late in development.
3. Working software is delivered frequently (weeks rather than months).
4. Close, daily cooperation between business people and developers.
5. Projects are built around motivated individuals, who should be trusted.
6. Face-to-face conversation is the best form of communication (co-location).
7. Working software is the principal measure of progress.
8. Sustainable development, able to maintain a constant pace.
9. Continuous attention to technical excellence and good design.
10. Simplicity— only build what is really essential.
11. Self-organizing teams.
12. Regular adaptation to changing circumstance.

# A little bit of History

Incremental software development methods trace back to 1957. In the mid 90's light weight methods were developed (e.g. scrum, extreme programming and others), now referred to as "Agile methods," but the real game changer was the publication of the Agile Manifesto in 2001.

The Agile Manifesto was published by a small group of developers who met in the U.S. and defined a set of concepts and principles mainly driven by:

**"Individuals and interactions over Processes and tools  
Working software over Comprehensive documentation  
Customer collaboration over Contract negotiation  
Responding to change over Following a plan."**

# Agile FAQs

- Is Agile only for Developers?
- Can I apply Agile on my Project?
- Is Agile only for IT?
- Is Agile hard or complex?
- Will Agile solve all my problems?

