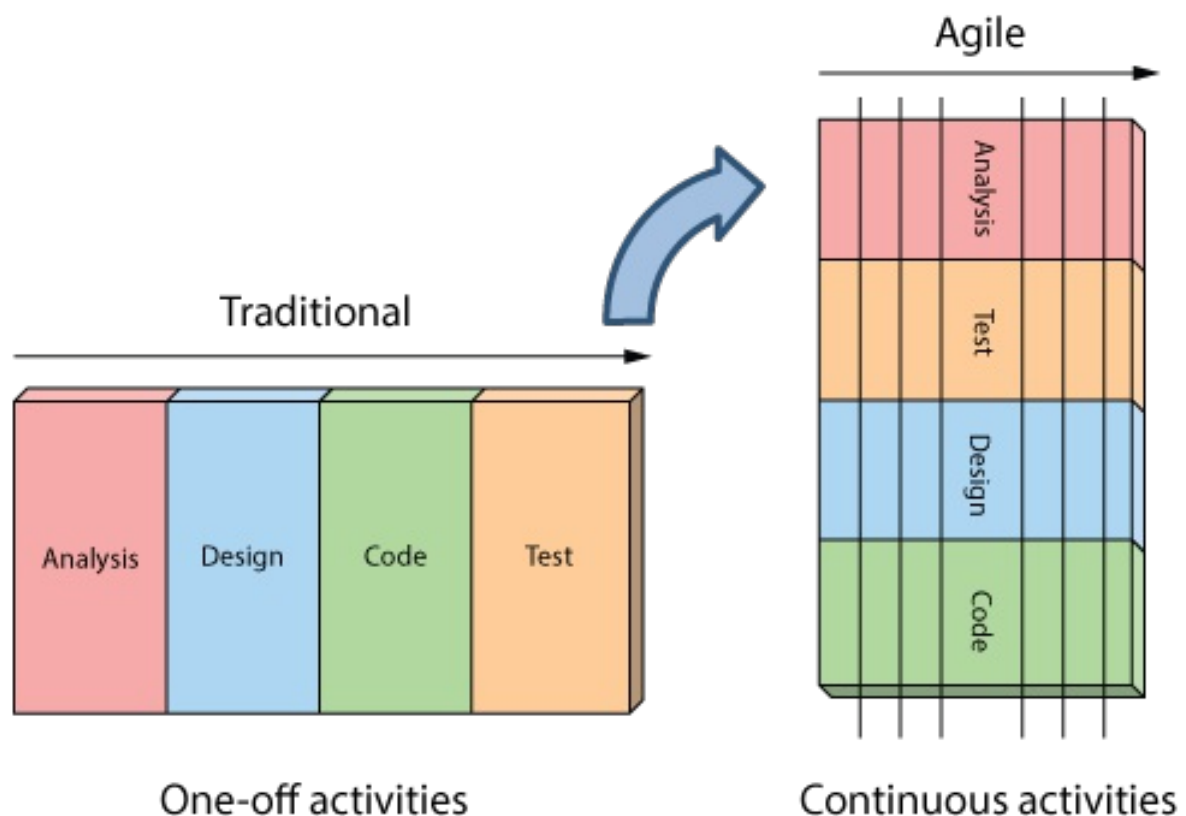# Agile

Core principles:

- Flexibility and incremental changes (short feedback loop)
- Efficient, face-to-face communication
- Working software is the best measure of progress (and also the best sales tool)
- Continuous communication with client/customer

Incremental changes

While in traditional project management different phases of the project are typically split up into different blocks of time, in agile software development you are constantly and continuously analyzing, testing, and coding.

One-off activities   Continuous activities

Essentially, the goal of agile is to take constant, baby-steps forward, so you can check and test every step of the way and easily change your plans if needed.

Sprint Planning

In agile, you break your project into 'sprints'. A sprint is typically a two week period (but can really be as long or as short as your team decides). At the beginning of each sprint, your team will get together and decide which features to try to complete during the sprint.

- Create backlog (aka list of all potential new features, bug fixes, and refactoring)
- Estimate how long each feature will take
- Prioritize features
- Assign features to different team members
- Start executing
- Update and track progress as you go using project management software (like waffle.io)

MVP

Daily standup

- Cover progress since last standup

- Talk about priorities for the day
- Any blocking features or impediments

Resources

- [Agile Software Development Wiki](#)