Software Engineering Fundamentals:
Agile Methodology

What is it?

In short, agile methodology is the ability to create and respond to change. Because software projects are largely considered an engineering process, it was thought that many of the same principles could be applied to software design. These ideas included project management methods, such as waterfall. As software engineering processes developed, many engineers began to realize that the typical engineering process was not as beneficial to the software environment. This was due in large part to how software projects needed to adapt to changing requirements.

With a typical engineering process, the requirements can be very static. Take a building for example. The floors are typically the same height, doors operate the same way, etc. There is not a lot of changing once the blueprints are decided upon, imagine changing the structure of a building once construction starts. In contrast, look at a software project. Many times the requirements can change as the development process is moving forward. This is the nature of product development in software. Traditional waterfall methodology leads to many issues such as stale development, wasted development time, and wasted documentation process. Agile aims to serve a rapidly changing environment and mitigate some of the drawbacks of waterfall.

How is it Applied?

Agile is more of a guide to how a software project should be managed. The Agile Methodology is applied through frameworks. Some of the more common frameworks are Kanban and Scrum.

Scrum Framework

Scrum applies the agile methodology through concepts such as sprints, roles, and ceremonies. This framework is a highly adaptable to change and allows teams to iterate effectively.

Sprints

A sprint is an allotted amount of time to accomplish tasks, usually two weeks. The team is responsible for accomplishing the tasks in the sprint and then reporting on them at the end. The tasks are usually assigned in the sprint planning ceremony and the end result is demonstrated in the sprint review ceremony.

Ceremonies

Sprint Planning

This ceremony is where the team along with the product owner plans that tasks that will be accomplished during the next sprint. Usually tasks are assigned story points and sprints will have a maximum. This allows a sprint to be manageable. These typically last hour.

Daily Stand-Ups

This ceremony happens daily, usually in the morning, it involves the team members, scrum master and the product owner. This meeting is a short what you have accomplished and what you plan on doing, typically only lasting about fifteen minutes.

Sprint Review

This ceremony happens at the end of a sprint, it involves the team members, scrum master, product owner, and optionally the stakeholders. This is where the team talks about what was accomplished and demonstrates a functional product.

Sprint Retrospective

This ceremony happens at the end of a sprint and involves the developers, scrum master, and product owner. During this meeting the participants typically discuss how the sprint went, focusing on what went well, bad, and how things could be improved.

Product Backlog

The product backlog is the place that all the user stories, issues, and tasks are stored. Generally, this is where everything required for the product implementation is listed. In the sprint planning session requirements are selected from the product backlog to go into the sprint backlog.

Sprint Backlog

The sprint backlog is like the product backlog but is all the tasks that will be completed during the sprint. The tasks in this backlog have been assigned points and are ready for development.

Points

Points are a way to measure the amount of work associated with a task. Point values are typically arbitrary, usually they will be decided in the first sprint planning sessions. Typically they will be based on the Fibonacci sequence (e.g. 1, 1, 2, 3, 5, 8 ...), with three representing easy task, five is for intermediate tasks and eight is for complex tasks. If a task is above an eight they should be decomposed into smaller tasks.

Roles

Product Owner

The product owner manages the product. They usually have an understanding of the business needs as well as the technical needs of the system.

Scrum Master

The scrum master is responsible for managing the tasks in the sprint and making sure things are flowing within the sprint.

Development Team

Developers and architects that complete the engineering and documentation tasks for the product.

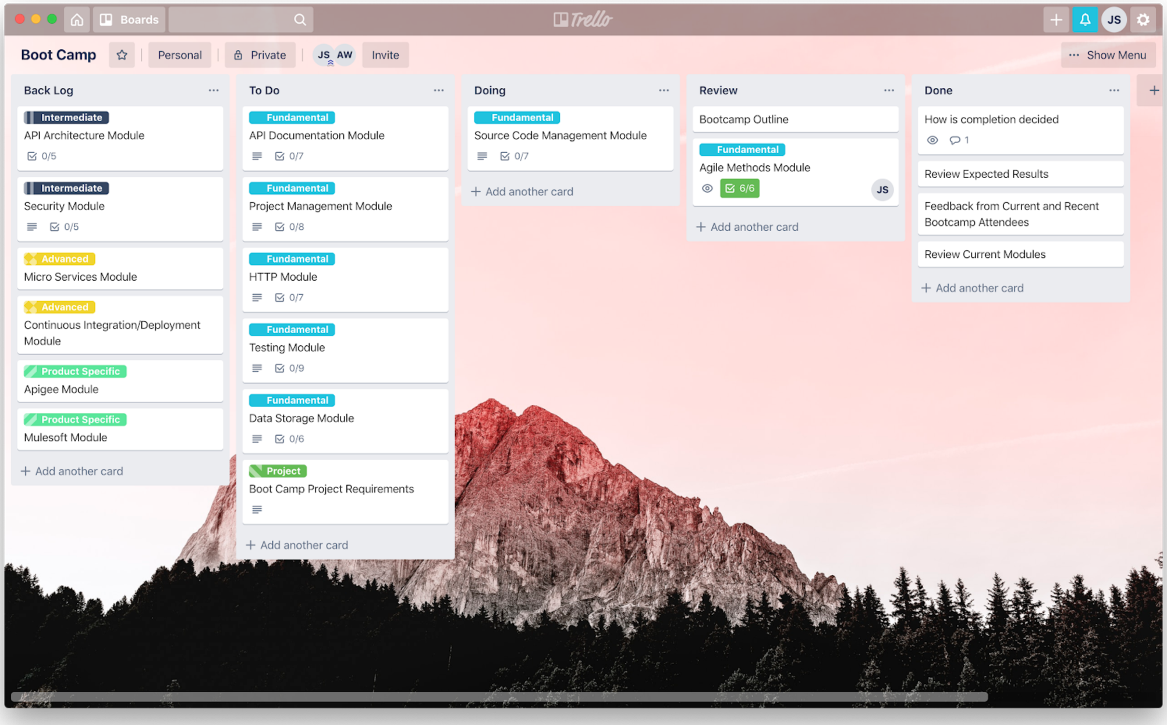
Stakeholders

The stakeholders typically represent the business needs for the product.

Kanban Framework

Kanban applies the Agile framework through concepts such as boards, cards, work in progress limits, and transparency of work. In many ways this framework is easier and less process intensive than the Scrum framework. Instead of complicated story points, strict time sessions and ceremonies, this framework builds on logical segments of work and limits to work in progress.

Boards



Boot Camp

Back Log
- Intermediate — API Architecture Module — 0/5
- Intermediate — Security Module — 0/5
- Advanced — Micro Services Module
- Advanced — Continuous Integration/Deployment Module
- Product Specific — Apigee Module
- Product Specific — Mulesoft Module

To Do
- Fundamental — API Documentation Module — 0/7
- Fundamental — Project Management Module — 0/8
- Fundamental — HTTP Module — 0/7
- Fundamental — Testing Module — 0/9
- Fundamental — Data Storage Module — 0/6
- Project — Boot Camp Project Requirements

Doing
- Fundamental — Source Code Management Module — 0/7

Review
- Bootcamp Outline
- Fundamental — Agile Methods Module — 6/6 — JS

Done
- How is completion decided — 1
- Review Expected Results
- Feedback from Current and Recent Bootcamp Attendees
- Review Current Modules

The Kanban framework uses a board that represents all the work that needs to be done on a project. These boards are organized into columns that represent status for the tasks of the project workflow, for example to do, doing and done.

Columns

Columns in the Kanban framework represent status in the project. A project can have as many status as needed but the amount should be limited. Workflows in the Kanban framework should be simplified.

Some of the more common columns are:

To Do

This column usually represents work that needs to be started. Typically these tasks have not been assigned to a developer and there are no work in progress limits.

In Progress

This column is for cards that have been assigned to a developer. This column typically has a work in progress limit that represents how much work load a team can handle at any given time.

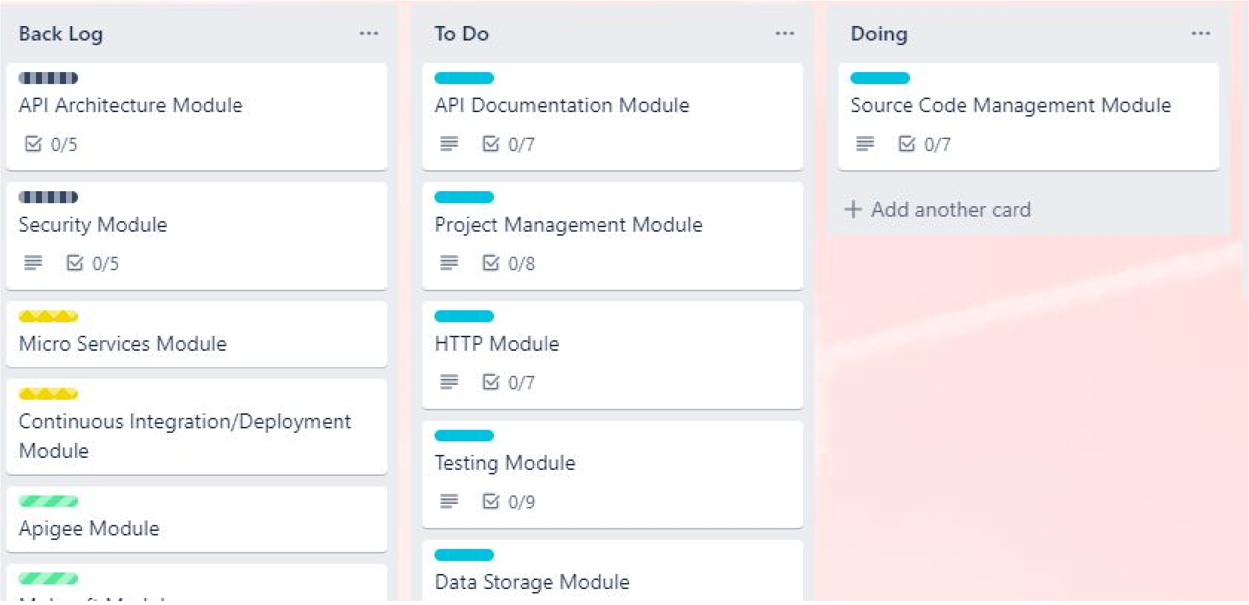
Blocked

This column is for cards that have something blocking progress. It tells the team that there are issues development on a task. Usually the work in progress limits are low on this column so that issues can be handled quickly.

Done

This column is for completed work and serves as a place that completed tasks can be reviewed.

Cards



Back Log

API Architecture Module
☑ 0/5

Security Module
☑ 0/5

Micro Services Module

Continuous Integration/Deployment Module

Apigee Module

To Do

API Documentation Module
☑ 0/7

Project Management Module
☑ 0/8

HTTP Module
☑ 0/7

Testing Module
☑ 0/9

Data Storage Module

Doing

Source Code Management Module
☑ 0/7

+ Add another card

Cards are considered the unit of work in the Kanban Framework. They represent a task that needs to be accomplished. As the task progresses, the card is moved between columns on the board. This allows as much time that is needed to work on a task and therefore tasks should not be very granular.

Cards are considered the unit of work in the Kanban Framework. They represent a task that needs to be accomplished. As the task progresses, the card is moved between columns on the board. This allows as much time that is needed to work on a task and therefore tasks should not be very granular.

Work in Progress Limits

Work in Progress Limits are numerical limits to the number of cards that can be in a column on a board. In the Kanban framework this is what manages work expectations and the number of outstanding issues for any given column. Work in progress limits are usually set on in progress and blocked columns. In progress WIP limits are usually based on the team velocity when completing tasks. For blocked tasks the WIP limit is usually low so that issues blocking progress are dealt with quickly. Similar to point limitations in sprints, they manage the number of tasks that can be in any one state.

Recommended tools

There are a few tools that will help when using frameworks such as Scrum and Kanban.

- https://trello.com
- https://www.atlassian.com/software/jira

Resources

- https://www.agilealliance.org/agile101/
- https://www.atlassian.com/agile
- https://www.atlassian.com/agile/scrum
- https://www.atlassian.com/agile/kanban