This post was published to Code Blog at 1:25:00 PM 2/16/2017

Scrum Process Notes for a Developers Mind

# ​Intro

This is a dumping ground for general notes on the scrum process and how an engineer can decode what is expected from a scrummaster or managers.

## High Level Process Notes

* 2 weeks sprint
  + 60 hrs. per developer (6 hrs./day) for 2 weeks.
  + The goal is to burndown tasks from 60 hours to 0 hours in a perfect world for one individual developer.

* Vertical slicing - The goal is to try to build one small piece of the app for the user to consume as opposed to horizontal slicing which means to build the whole monolithic database layer, then the middleware, then the UI. The real goal is to delivery customer value fast and get them using the app ASAP, even if only one piece is completed.

## Scrum Work Hierarchy

1. Epic
   1. Top of the scrum hierarchy
   2. Huge things to be done for the entire project
2. Features
   1. Second in the scrum hierarchy
   2. Typically a feature takes 1-2 weeks to complete and are done within a sprint
   3. Maybe have 2 features per sprint depending on team size.
3. Users Stories
   1. Third in the scrum hierarchy
   2. To be completed within a sprint.
   3. Typically there are 2-3 user stories per feature for one developer
   4. Typically a user story takes about 2-3 days
   5. Typically there are 5-6 user stories per sprint
4. Task
   1. Last in the scrum hierarchy
   2. Typically completed within one day
   3. Usually written by the developer and is contained in a user story
   4. Typically 3-6 tasks per user story

## Scrum Lifecycle

1. **Grooming** (occurs every 1 week, comes before sprint planning, groom user stories)
   * Typically the developer see's user story for first time
   * If user story is bigger than 2-3 days figure out how to break it up
   * Get rid of dependencies

1. **Sprint Planning** (occurs at the start of every 2 weeks)
   * Team commitment, teams figures out what they want to deliver…make sure there are no dependencies
   * Break down user stories into tasks (10-15 tasks normal per person per 2 week sprint)
   * Could see 50-75 tasks per sprint

1. **Sprint Review** (occurs at the end of every 2 weeks)
   * Show off every 2 weeks what we have done, typically with stakeholders and Aaron

1. **Retro** (occurs at the end of every two weeks)
   * Team reflects back on what they have done. Can get technical in nature. Basically how can you get better.

## Example Board for 1 developer (TODO make more detailed...maybe take a real projects data)

Epic - build a pricing app (1 month)

Sprint 1 (2 weeks)

Feature 1 - Get customer prices for products (1 week)

User Story 1 - Customer Prices Data access layer

Task 1 -  Plumbing for REST calls

Task 2 -

Task 3 -

Task 4 -

User Story 2 -

Task 1 -

Task 2 -

Task 3 -

Task 4 -

User Story 3 -

Task 1 -

Task 2 -

Task 3 -

Task 4 -

Feature 2- Get price per price for products (1 week)

User Story 1 - Price per Product Data access layer

Task 1 -  Plumbing for REST calls

Task 2 -

Task 3 -

Task 4 -

User Story 2 -

Task 1 -

Task 2 -

Task 3 -

Task 4 -

User Story 3 -

Task 1 -

Task 2 -

Task 3 -

Task 4 -