Unit 3 Module 3: Agile Development Methodologies

Dylan Lane McDonald

CNM STEMulus Center
Web Development with PHP

August 28, 2014

Outline

Agile Development Methodologies

Quote of the Day

"All the world's a stage,
And all the men and women merely players:
They have their exits and their entrances;
And one man in his time plays many parts,
His acts being seven ages."
$\approx$ *William Shakespeare*

Agile Development Methodologies

Quote of the Day

"All the world's a stage,
And all the men and women merely players:
They have their exits and their entrances;
And one man in his time plays many parts,
His acts being seven ages."
$\approx$ *William Shakespeare*

Agile development methodologies is an "umbrella term" for many different methodologies, all of which are "agile" and adapt to changing requirements very fast.

Common Features of All Agile Methods

All agile methodologies have the following features in common:

- **Iterative**: The development process is done small slices and releases are scheduled regularly
- **Communication**: The product owner is very involved and constantly providing feedback to the development team
- **Quality**: High quality standards are maintained through methods such as continuous integration, unit testing, test-driven development, and refactoring
- **Adaptive**: Software development evolves as new features, bug reports, and client needs are given in the communication cycle

Agile Actors

All agile methods have various roles in the development process. Please add all these roles to your lexicon.

- **Product Owner**: The individual who represents the client's interest in the software development process
- **Project Manager**[1]: The individual who supervises the development team, approves sprints, and assigns tasks
- **Developer**: Team member who, in addition to writing software, writes documentation, tests and verifies all units, and provides feedback to the project manager

Software teams in an agile environment are self-organizing. Developers organize based on current projects and how their talents are best suited to particular tasks.

[1] Also known as a "master"

Agile Phases

The agile process consists of various phases. Please add these to your lexicon, too:

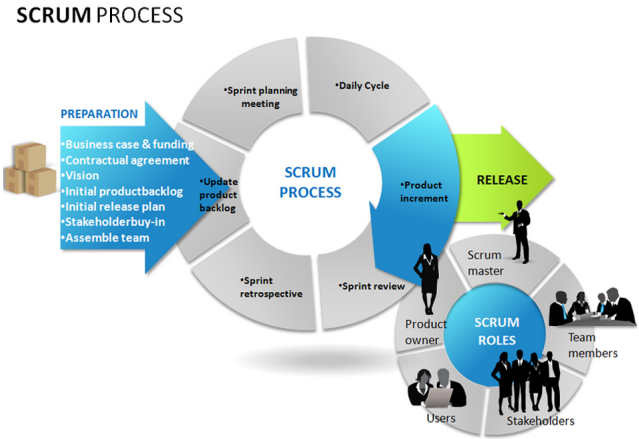
1. Tickets are taken from the **backlog**, which is a group of tickets waiting for development
2. This group of tickets is known as a **sprint**; the sprint is then given a time limit
3. Daily **stand up meetings** designed to keep everyone up to date; these meetings are designed to be only 5 - 10 minutes
4. As developers complete tickets, they write tests for all their code, known as **unit tests**
5. The sprint is then worked on by the developers and the tickets will gradually become complete; the rate at which this happens is known as the **burn down**

Different Agile Methodologies

There is no one agile methodology. Organizations will often create "hybrid" methods of two or more methodologies. As always, there are advantages and disadvantages to each.

- **Extreme Programming (XP)**: Method where developers use **pair programming**: one keyboard, two developers
- **Kanban**: Kanban stresses just-in-time delivery and limiting the tickets that are in progress
- **Scrum**: A flexible methodology that emphasizes the ability of the team to adapt to a product that may or may not be well defined
- **Test Driven Development**: A methodology that enforces all unit tests be written before the actual unit

Scrum


SCRUM PROCESS
PREPARATION
•Business case & funding
•Contractual agreement
•Vision
•Initial productbacklog
•Initial release plan
•Stakeholderbuy-in
•Assemble team
•Update product backlog
•Sprint planning meeting
•Daily Cycle
SCRUM PROCESS
•Product increment
RELEASE
•Sprint review
•Sprint retrospective
SCRUM ROLES
Scrum master
Team members
Product owner
Users
Stakeholders

In Class Agile

Like many job sites, this class will be using a "hybrid" approach to agile development. Specifically, the following features will be used from the processes discussed:

- Roles, stand-up meetings, sprints, and burn downs will be used from Scrum
- In progress tickets will be limited as per Kanban
- **All** JavaScript and PHP code must be unit tested before it is written, as is seen in test driven development

Essentially, the class will be using Scrum with a few added features from Kanban and test driven development.

Unit Testing

Unit testing has already been alluded to in this set of slides. But a more complete definition would be:

Definition

A **unit test** *is a function that tests another function (a unit) for a specific set of inputs and conditions and expects a specific outcome and outputs.*

Unit Testing

Unit testing has already been alluded to in this set of slides. But a more complete definition would be:

Definition

A **unit test** *is a function that tests another function (a unit) for a specific set of inputs and conditions and expects a specific outcome and outputs.*

Notice the use of the word **unit** in the term unit test. In agile, programming units are the **smallest functionality possible** that accomplish the given task or objective. One keystone of agile design and methodologies is the fact that a problem is composed of many small units. This is what facilitates Agile's flexibility.

Unit Testing Terminology

Add all these terms to your lexicon.

- **Assertion**: A testable condition the unit test is testing for. For example:
  - *Assert x is in array A*
  - *Assert* $x < y$
  - *Assert x is not* **NULL**
- **Throws**: An assertion that succeeds if and only if an exception is thrown
- **Setup**: A function that sets up testing conditions to be run before the unit test is executed
- **Tear Down**: A function that destroys the test data created by setup or the unit test after the unit test is executed

QUnit Unit Test

```
test("hello test",
    function() {
        ok(1 == "1", "Passed!");
    }
);
```

Listing 1: QUnit Unit Test

QUnit unit tests take the form as seen in Listing 1. After setting up the HTML and JavaScript, units can be tested using the `test()` method. Full examples and document are at `http://qunitjs.com/`.