



DEVOPS

Ing. Oscar Garcia Colon
UFM

Objetivo

- Solid foundation in Agile frameworks
- Best practices work in an Agile environment

Agile Alliance

- Lightweight
- Agile Manifesto
- eXtreme Programming
- Scrum
- Higher quality software in shorter frames
- Collaboration, Self-organizing, cross-functional

Agile Manifesto

- Forms the basis for most methods currently in use today, including Scrum, eXtreme Programming, Lean, Crystal Methods and Others.
- Do you think this course is about Programming?

Agile Manifesto

- 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 Manifesto

- ❑ Heavier Processes
- ❑ Overuse of tools
- ❑ Comprehensive documentation
- ❑ One over the other?

Empirical Process Control

- Transparency
- Inspection
- Adaptation

Guiding principles

- ❑ Iterative
- ❑ Incremental
- ❑ Frequent Reviews
- ❑ Adaptation
- ❑ Uncertainty and risks during execution
- ❑ It isn't a “defined” process; it's a way of being and doing

Agile vs. Waterfall project management

- Top-Bottom
- Requirements -> Delivery
- W: Preplanned & Change Control

Few Remarks

- Respond quickly to changes
- Agile principle needs:
 - ▣ Understanding
 - ▣ Internalization
- Avoid current process flow

Introspection

- Guidelines
- What Processes
- Interpersonal skills
- Quality
- Risk
- Backlog
- Priorization



Cool!

Whatever works for you is cool, and that's the beauty of agility.



Scrum and Extreme Programming (XP)

What is Scrum?

- A framework within which people can address complex adaptive problems, while productively and creatively delivering products of the highest possible value.
- Scrum was designed to be a framework, not a process.
- This framework is lightweight, very simple to understand but inherently difficult to master.
- Scrum theory was founded or based upon the empirical process control theory, or empiricism.

Pillars



- Transparency
- Inspection
- Adaptation

Transparency

- Mutual understanding
 - ▣ Standards
 - ▣ Common Language
 - ▣ Definition of done
- issues. The entire focus is based on the understanding that everyone knows what everyone else is doing and how they are doing it.

Inspection

- Frequency
- How often???

Adaptation



□ Definition of Agile

Scrum Values

- Commitment
- Focus
- Openness
- Respect
- Courage

Grooming the backlog

- Value: Grooming
- Review
- Revised
- Add detailed estimates

The Scrum Teams

- Scrum teams;
 - ▣ Consists of a Product Owner, the Development Team and a Scrum Master
 - ▣ Are self organizing and cross-functional
 - ▣ Choose how best to accomplish their work, rather than being directed by others outside the team
- The team model in Scrum is designed to optimize flexibility, creativity, and productivity.
- The ideal size of a Scrum team is between 3 and 9 members

Roles – The Product Owner

- The PO is **solely responsible** for the Product Backlog.
Accountable for;
 - ▣ Clearly expressing Product Backlog items
 - ▣ Ordering the items in the Product Backlog to best achieve goals and missions
 - ▣ Ensuring the value of the work the Development Team performs
 - ▣ Ensuring that the Product Backlog is visible, transparent, and clear to all, and shows what the Scrum Team will work on next
 - ▣ Ensuring the Development Team understands items in the Product Backlog to the level needed

Roles – The Development Team

- The Development Team consists of professionals who do the work of delivering a potentially releasable Increment of “Done” product at the end of each Sprint.
- Development Teams have the following characteristics:
 - They are self-organizing and cross-functional
 - Scrum recognizes no titles for Development Team members other than Developer, regardless of the work being performed by the person
 - Individual Development Team members may have specialized skills and areas of focus, but accountability belongs to the Development Team
 - Development Teams do not contain sub-teams

Roles – The Scrum Master

- The Scrum Master is responsible for ensuring Scrum is understood and enacted. Scrum Masters do this by ensuring that the Scrum Team adheres to Scrum theory, practices, and rules.
- The Scrum Master is a servant-leader for the Scrum Team
- The Scrum Master can coach and guide the team but is NOT a Project Manager
- The Scrum Master removes impediments and facilitates Scrum events as requested or needed

Scrum Events

- Sprint Planning Meeting
- Daily Scrum
- Sprint Review
- Sprint Retrospective

Sprint Planning Meeting

- Plans the work to be performed in the next Sprint
- Is a collaboration between all members of the Scrum Team
- Time-boxed (2wk sprint = 4hrs, 4wk sprint = 8hrs)
- Meeting has two equal parts
 - What will be delivered in the Increment resulting from the Sprint?
 - How will the work needed to deliver the Increment be achieved?

Sprint Planning Meeting – Part 1 (50% timebox)

- What will be delivered in the Increment resulting from the Sprint?
 - ▣ The Product Owner presents the ordered Product Backlog
 - ▣ Inputs
 - The Product Backlog
 - The latest product Increment
 - The past performance of the Development team
 - ▣ Outputs
 - A selection of items from the Product Backlog to go into the Sprint
 - The Sprint Goal

Only the Development can decide what goes into the sprint

Sprint Planning Meeting – Part 2 (50% timebox)

- How will the work needed to deliver the Increment be achieved?
 - ▣ The Development Team decides how it will build this functionality into a “Done” increment
 - ▣ The Sprint Backlog is the selection of items off the Product Backlog plus the plan for delivering them
 - ▣ Often the Product Owner attends this part of the meeting
- By the end of the Sprint Planning Meeting, the Development team should be able to explain how it intends to work as a self-organizing team to accomplish the Sprint Goal and create the anticipated Increment

Daily Scrum

- A daily 15-minute time-boxed meeting for the Development Team
- Used to synchronize activities and create a plan for the next 24 hours
- Each team member explains
 - ▣ What has been accomplished since the last meeting?
 - ▣ What will be done before the next meeting?
 - ▣ What obstacles are in my way?

Sprint Review Meeting (2wk=2hr, 4wk=4hr)

- The Sprint Review Meeting is held at the end of the Sprint to inspect the Increment and adapt the Product Backlog if needed
- The meeting should include the following elements;
 - ▣ The Product Owner identified what has been “Done” (and not)
 - ▣ The Development Team demonstrates the work that has been “Done” and answers questions about the Increment
 - ▣ The Product Owner discussed the Product Backlog as it stands
- The result of the meeting is a revised Product Backlog that defines the probable PBIs for the next sprint

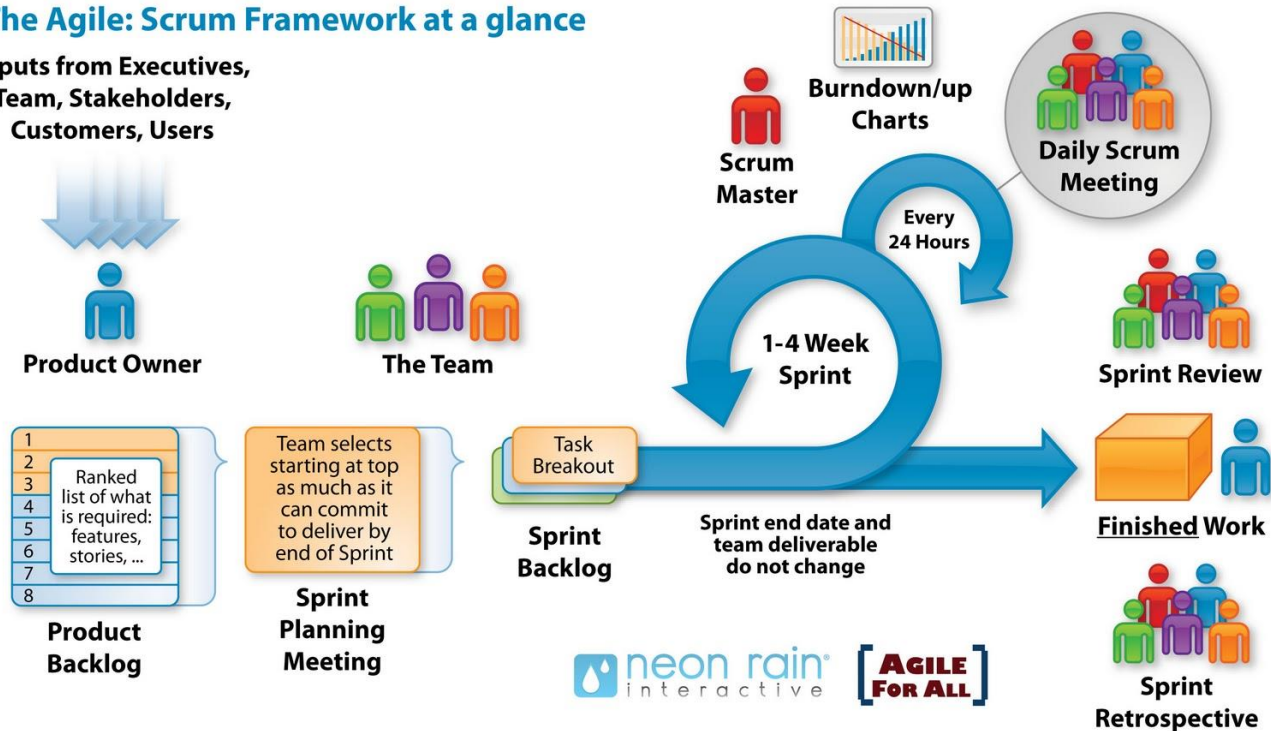
Sprint Retrospective (4wk=3hr, 2wk=90mins)

- The purpose of this meeting is;
 - ▣ Inspect how the last Sprint went with regards to people, relationships, process, and tools
 - ▣ Identify and order the major items that went well and potential improvements
 - ▣ Create a plan for implementing improvements to the way the Scrum team does its work
- The Scrum Team may adapt the Definition of “Done” as appropriate to increase quality

High Level Diagram

The Agile: Scrum Framework at a glance

Inputs from Executives,
Team, Stakeholders,
Customers, Users



Cancelling a sprint or iteration - Overview

- ❑ Only the **Product Owner** has the authority to cancel the Sprint, although he or she may do so under influence from the stakeholders, the Development Team, or the Scrum Master
- ❑ A Sprint would be cancelled if the Sprint Goal becomes obsolete or if it no longer makes sense given the circumstances
- ❑ Due to the short duration of Sprints, cancellation rarely makes sense
- ❑ Sprint cancellations are often traumatic to the Scrum Team, and are very uncommon

Cancelling a sprint or iteration - Process

- When a Sprint is cancelled, any completed and “Done” Product Backlog Items are reviewed.
- If part of the work is potentially releasable, the Product Owner typically accepts it.
- All incomplete Product Backlog Items are re-estimated and put back on the Product Backlog.
- The work done on them depreciates quickly and must be frequently re-estimated.

Definition of “Done”

- When the Product Backlog item or an Increment is described as “Done”, everyone must understand what “Done” means
- Although this varies significantly per Scrum Team, members must have a shared understanding of what it means for work to be complete, to ensure transparency.
- This is the “Definition of Done” for the Scrum Team and is used to assess when work is complete on the product Increment
- it is expected that their Definition of “Done” will expand to include more stringent criteria for higher quality

Determining Sprint/Iteration Length

- Sprints are limited to one calendar month. When a Sprint's horizon is too long the definition of what is being built may change, complexity may rise, and risk may increase
- Sprints enable predictability by ensuring inspection and adaptation of progress toward a goal at least every calendar month
- Sprints also limit risk to one calendar month of cost.

Less certainty = shorter sprint duration

Backlog items not completed in a Sprint

- ❑ The Development Team should always try to select appropriate items from the Product Backlog so they can be completed within a Sprint
- ❑ Incomplete work cannot be presented during the Sprint Review Meeting
- ❑ Unstarted work can be returned to the Product Backlog for consideration accepting into the next Sprint
- ❑ Frequently carrying incomplete work to the next Sprint can be detrimental to measuring velocity

Monitoring Sprint/Iteration progress

- At any point in time in a Sprint, the total work remaining in the Sprint Backlog items can be summed. The Development Team tracks this total work remaining at least for every Daily Scrum
- The Development Team tracks these sums daily and projects the likelihood of achieving the Sprint Goal. By tracking the remaining work throughout the Sprint, the Development Team can manage its progress
- Scrum **does not consider** the time spent working on Sprint Backlog Items. The work remaining and date are the only variables of interest

What is Estimation Poker?

- A relative estimation technique
 - ▣ Accuracy over precision
 - ▣ Based on Fibonacci number sequence
- Done by the development team members who will do the work
- Self corrects for specific team idiosyncrasies
 - ▣ Leverages Scrum's "Apples to Apples" dynamic
 - ▣ "Accelerated Reality"
- Enables more accurate long-term planning

How to Play Estimation Poker

1. Product Owner explains story
 2. Each Dev Team member selects a card (don't show it)
 3. Together, all Dev Team members show their card
 4. If different, the HIGH and LOW estimators briefly explain their choice and assumptions
 5. The Product Owner provides additional information, as necessary
 6. The Dev Team iterates the process, usually up to 3 times, until consensus is reached
-
- ☐ Sounds ridiculous?
 - ☐ Maybe...
 - ☐ But it works!

The PO is a...

- **Subject Matter Expert**

- Understand the domain well enough to envision a product
- Answer technical questions on the domain for those creating the product

- **End User Advocate**

- Describe the product with understanding of users and use, and a product that best serves both

- **Customer Advocate**

- Understand the needs of the business buying the product and select a mix of features valuable to the customer

- **Business Advocate**

- Understand the needs of the organization paying for the software's construction and select a mix of features that serve their goals

- **Communicator**

- Capable of communicating vision and intent – deferring detailed feature and design decisions to be made just in time

- **Decision Maker**

- Given a variety of conflicting goals and opinions, be the final decision maker for hard product decisions

The Product Owner role is generally filled by a single person supported by a collaborative team

Thing to watch for

1. Estimation
2. Sprint Review
3. Retrospective quality meeting
4. Quality on the Features and Product Backlog items

Common Pitfalls

- ❑ Lack of training
- ❑ Ineffective Product Owner
- ❑ Lack of automated testing
- ❑ Lack of professional transition support
- ❑ Inappropriate physical environment
- ❑ Poor team selection
- ❑ Discipline slips
- ❑ Lack of support for learning (blame game)
- ❑ Tweaking/Dilute till dead



Questions???