Open Educational Resources   John Jay College of Criminal Justice

2019

Lecture 6: Mobile Application and Product Development

NYC Tech-in-Residence Corps
rdomanski@sbs.nyc.gov

Mobile Application and Product Development

Agile Project Management

Agenda

- Agile vs Waterfall
- About Agile Project Management
- Agile Frameworks
- SCRUM
- SCRUM Roles
- User Stories
- Planning Poker
- The Sprint
- Daily Standup
- Sprint Retrospective
- Agile Case Study: Netflix Culture

Agile Manifesto

We are uncovering better ways of developing
software by doing it and helping others do it.
Through this work we have come to value:

Individuals and interactions over processes and tools

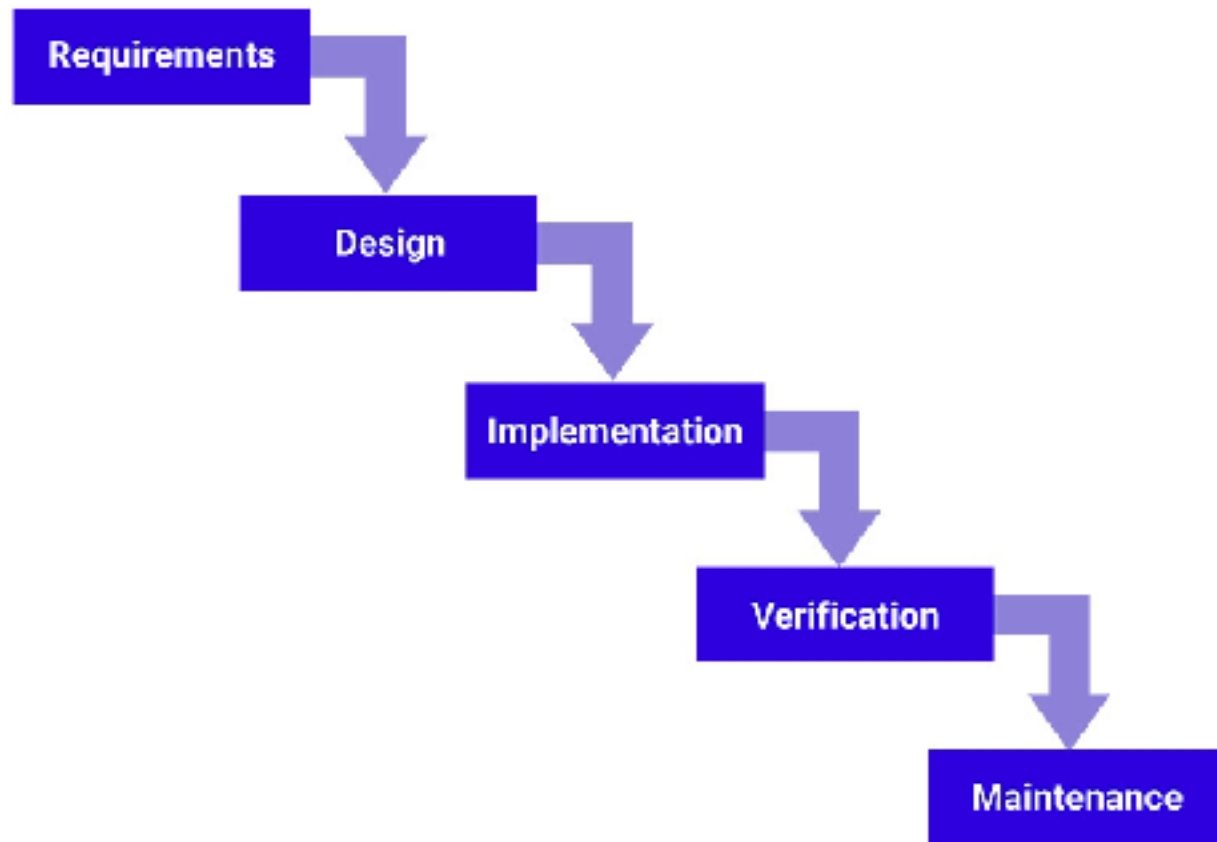**Working software** over comprehensive documentation
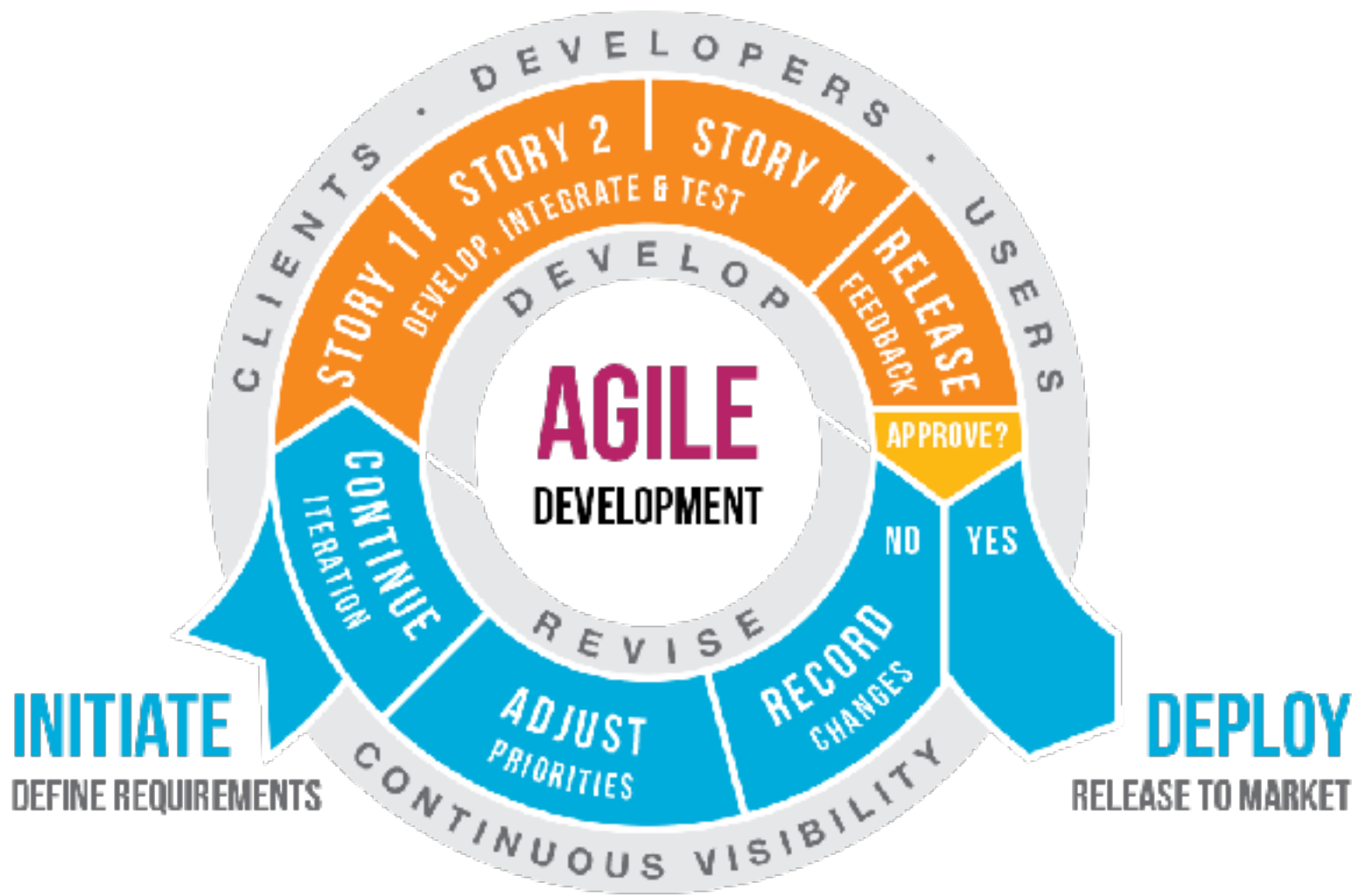
Customer collaboration over contract negotiation

Responding to change over following a plan

That is, while there is value in the items on
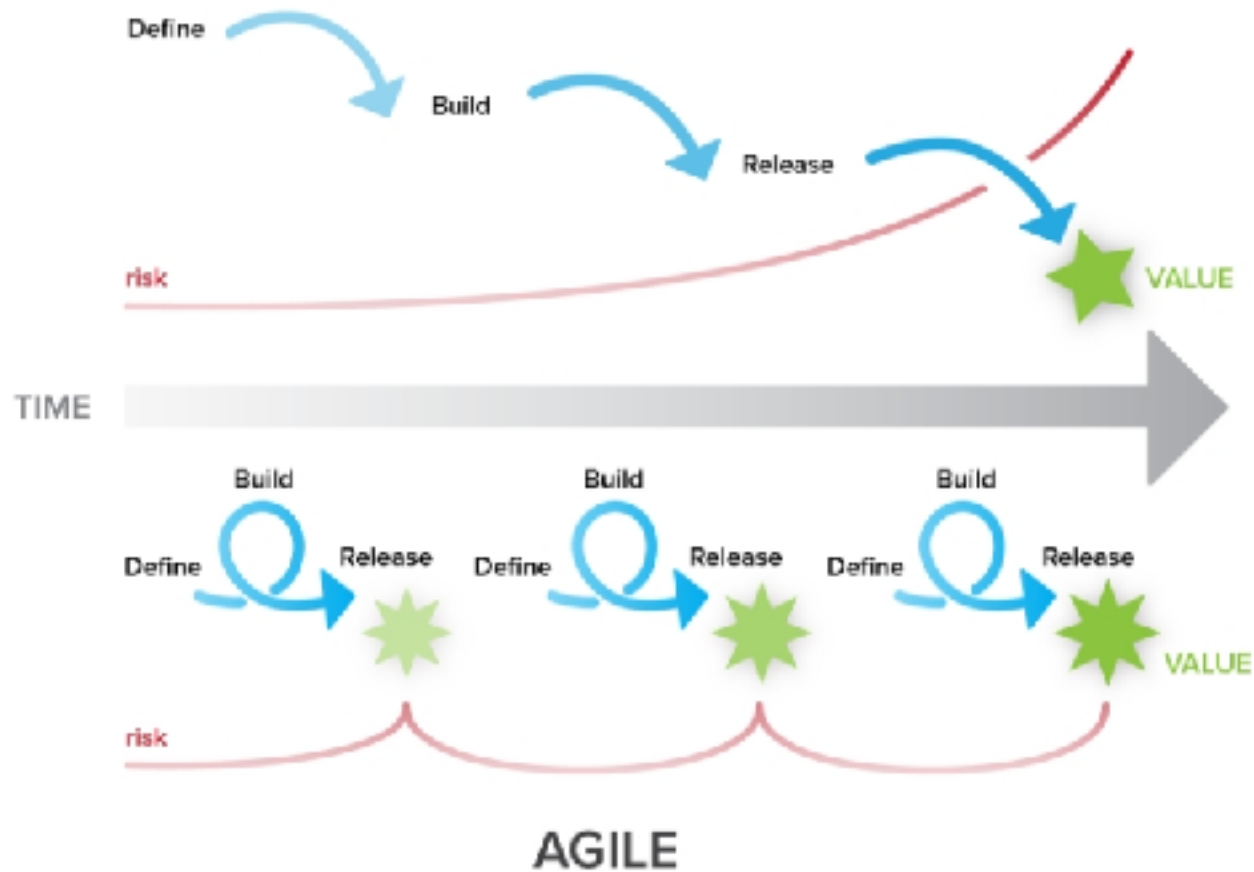the right, we value the items on the left more.

https://agilemanifesto.org/

Waterfall

AGILE
DEVELOPMENT

CLIENTS · DEVELOPERS · USERS

DEVELOP

STORY 1 | STORY 2 | STORY N
DEVELOP, INTEGRATE & TEST

RELEASE
FEEDBACK

APPROVE?

NO   YES

CONTINUE
ITERATION

REVISE

ADJUST
PRIORITIES

RECORD
CHANGES

CONTINUOUS VISIBILITY

INITIATE
DEFINE REQUIREMENTS

DEPLOY
RELEASE TO MARKET

Agile Project Management

Shared collaboration and responsibility

Agile Umbrella

Agile

Scrum
Crystal
Kanban

XP
FDD
DSDM
RUP

and few more...
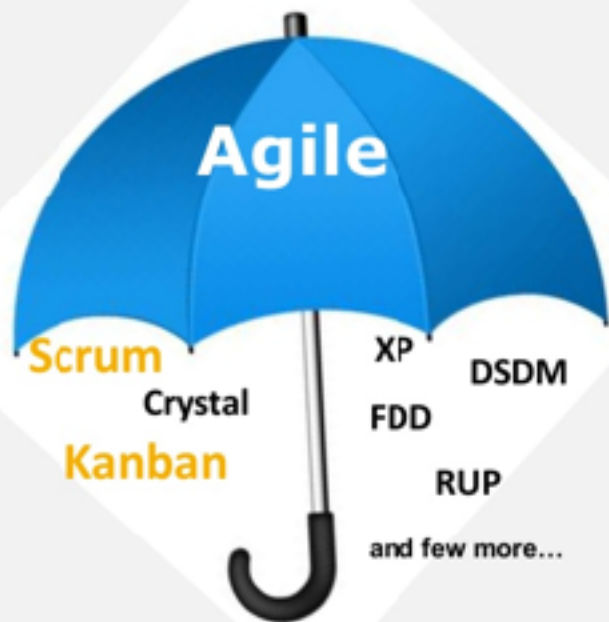
More Prescriptive
more rules to follow

● **RUP (120+)**
RUP has over 30 roles, over 20
activities, and over 70 artifacts

● **XP (13)**

● **Scrum (9)**

● **Kanban (3)**

● **Do Whatever!! (0)**

More Adaptive
fewer rules to follow

SCRUM

- Agile framework for managing knowledge work, with an emphasis on software development.
- Designed for teams of 3 - 9 members, who break their work into actions that can be completed within timeboxed iterations, called sprints
- Less than one month, usually 2 weeks
- Track progress and re-plan in 15-minute time-boxed stand-up meetings, called daily scrums

Other Agile Frameworks

- **RUP** - Rational Unified Process, an iterative software development process framework created by the Rational Software Corporation, a division of IBM
- **XP** - Extreme programming, a software development methodology which is intended to improve software quality and responsiveness to changing customer requirements. Advocates frequent "releases" in short development cycles, which is intended to improve productivity and introduce checkpoints at which new customer requirements can be adopted
- **Kanban** - (signboard or billboard in Japanese) is a scheduling system for lean manufacturing and just-in-time manufacturing

SCRUM Roles

- Product Owner
  - Project's key stakeholder. Has a vision of what he or she wishes to build, and conveys that vision to the SCRUM team.
- SCRUM Master
  - Facilitator for an agile development team. Keeps everyone on schedule and removes blockers.
- SCRUM Development Team
  - A self-organizing, cross-functional team of people who collectively are responsible for all of the work necessary to produce working and validated product
  - Usually Engineers, QA, DevOps, Designers.

Scrum Roles: Product Owner (PO)

- Scope management
  - Definition of the product and what the project is trying to create
- Schedule Management
  - Changes schedule based on features
- Cost Management
  - Oversight of cost

Scrum Roles: Agile Team

- Risk Management
  - Team is responsible for the risk
- Quality Management
  - Team is accountable for building in quality
- Task Assignment
  - What needs to be done and who can tackle it

Scrum Roles: Scrum Master

- Protects the agile framework
  - Ensures that everyone is following the process
- Removes blockers for the team
  - Watch for and eliminate anything that gets in the way of team performance
- Promotes healthy team behaviors
  - Open and frequent communication and collaboration
- Lead continuous planning
  - Owner of the planning process
- Monitor and measure progress
  - Ensures that product owner and the team are aware of their performance

User Stories

User stories are part of an agile approach that helps shift the focus from writing about requirements to talking about them. All agile user stories include a written sentence or two and, more importantly, a series of conversations about the desired functionality.

Example:

As a < type of user >, I want < some goal > so that < some reason >.

As an administration, I want to put the site in maintenance mode so that I can safely deploy new code

Product Backlog

- List of features in the backlog
- New stories are pulled into a sprint during Sprint Planning session
- Stories are ranked in order of importance

Backlog Grooming

- prioritizes each work item, with the most important work listed at the top
- includes fully-formed user stories the development team can begin to execute on
- contains an up-to-date estimate for each work item

Scrum: Teams in Space

Plan   Work   Report   Board ▾   »

🔍   QUICK FILTERS: **Tech Debt**   Product   UI   Server   Only My Issues   Recently Updated

EPICS

All Issues

SeeSpaceEZ Plus ▪

Space Travel Partners ▪

Summer Saturn Sale ▪

Afterburner Plus ▪

Large Team Support ▪

Local Mars Office ▪

Hyper-speed shuttles ▪

New launch platforms ▪

Delicious Space Nutrition ▪

Spacetainment ▪

Microgravity Delight ▪

⌄ **Sprint 6** 4 of 24 issues visible Clear all filters   🔵 🟡 🟢

24/Nov/14 11:05 PM • 08/Dec/14 11:05 PM   Linked pages

👩 👨 👨 •••

▪ ↑ **TIS-67** Developer Toolbox does not display by : `Large Team Support` 👩

▪ ↑ **TIS-68** Homepage footer uses an inline style - : `Large Team Support` 👨

▪ ↑ ~~TIS-56~~ Add pointer to main css file to ins `2.0` `Large Team Support` 👩

▪ ↑ ~~TIS-8~~ Requesting available flights is now t `2.0` `SeeSpaceEZ Plus` 👨

Backlog 7 of 44 issues visible Clear all filters   Create Sprint

▪ 🚫 **TIS-9** After 100,000 requests the SeeSpac `2.0` `SeeSpaceEZ Plus` 👤

▪ ↑ **TIS-7** 500 Error when requesting a reservation `2.0` `SeeSpaceEZ Plus` 👤

▪ ↑ **TIS-10** Bad JSON data coming back from h `2.0` `SeeSpaceEZ Plus` 👨

▪ ↑ **TIS-55** Links should not be in italics `SeeSpaceEZ Plus` 👤

▪ ↑ **TIS-57** Buttons need to be red `SeeSpaceEZ Plus` 👩

▪ ↑ **TIS-59** Look into API errors `SeeSpaceEZ Plus`

▪ ↑ **TIS-61** Small CSS error `2.0` `SeeSpaceEZ Plus` 👨

🌀 Teams in Space / **TIS-67**   •••

Developer Toolbox does not display by default

Details

Status: **CODE REVIEW**

Component/s: None

Labels: None

Affects Version/s: None

Fix Version/s: None

Epic: `Large Team Support ✕`

People

Reporter: Jennifer Evans

Assignee: Jennifer Evans

Dates

Created: 18/Oct/14 9:43 AM

Updated: 20/Dec/14 1:46 PM

Issue Links

Planning Poker

- Consensus-based, gamified technique for estimating, mostly used to estimate effort or relative size of development goals in software development
- Estimate based on complexity (points) vs hours
- Stories are typically estimated by size using Fibonacci Sequence
  - 0, 1, 2, 3, 5, 8, 13, 21, 34, 55
  - Stories > 21 points should be simplified

The Sprint

- Usually lasts 1 - 4 weeks
- Work is pulled from the Backlog by the SCRUM team during the Sprint Planning session
- Includes 15-minute daily standups
- Release at the end
- Followed by a Sprint Retrospective

Sprint Planning: Roles

- **Scrum Master** - facilitates the meeting
- **Product Owner** - clarifies the details of the product backlog items and their respective acceptance criteria
- **Entire Agile Team** - define the work and effort necessary to meet their sprint commitment.

Sprint Planning: Process

1. Remind the team of the big picture or goal
2. Discuss any new information that may impact the plan
3. Present the velocity to be used for this release
4. Confirm team capacity
5. Confirm any currently known issues and concerns and record as appropriate
6. Review the definition of DONE and make any appropriate updates based on technology, skill, or team member changes since the last sprint
7. Present proposed product backlog items to consider for the sprint backlog

Sprint Planning: Process (continued)

8. Determine the needs, sign up for work, and estimate the work owned
9. Product Owner answers clarifying questions and elaborates acceptance criteria
10. Confirm any new issues and concerns raised during meeting and record
11. Confirm any assumptions or dependencies discovered during planning and record
12. ScrumMaster calls for a group consensus on the plan
13. Team and Product Owner signal if this is the best plan they can make given what they know right now
14. Get back to work

Backlog 7 of 44 issues visible   Clear all filters

Create Sprint

🚫 **TIS-9** After 100,000 requests the SeeSpace 2.0 **SeeSpaceEZ Plus**	
↑ **TIS-7** 500 Error when requesting a reser	**Send to**
↑ **TIS-10** Bad JSON data coming from	Sprint 6
↑ **TIS-55** Links should not be in italics	Top of Backlog
↑ **TIS-57** Buttons need to be red	Bottom of Backlog
↑ **TIS-59** Look into API errors	Add flag
↑ **TIS-61** Small CSS error	View in Issue Navigator
+ Create issue	View in Excel
	Bulk Change

Kanban Board

Kanban Board in JIRA

# The Daily Standup

SCRUM - The Daily Standup

- Daily 15-minutes meeting
- In person, or via video
- Execution team attends
  - Product Owner
  - Project Manager
  - Engineering Team
  - QA Team
  - DevOps (when needed)
- 3 Questions
  - What did I accomplish yesterday?
  - What am I working on today?
  - Any blockers?

The Daily Standup

Sprint Retrospective

- Meeting with all the stakeholders after Sprint is complete
- 2 questions
  - What did we do well?
  - What can be improved?
- SCRUM Master summarizes the action plan
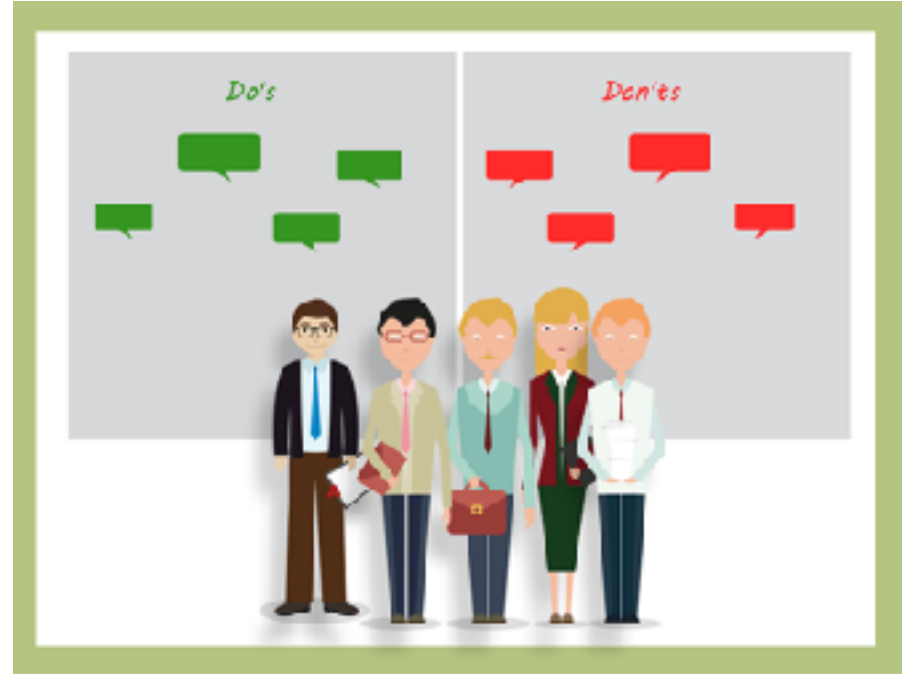
Pop Quiz: What are the roles in SCRUM?

1. Product Manager, Project Manager, Development Team, QA Team
2. Product Owner, Scrum Master, Agile Team
3. Project Manager, Scrum Master, Development Team

Pop Quiz: What 15-minute meeting happens daily?

1. The Retrospective
2. The Stand Up
3. Sprint Planning
4. Backlog Grooming

Pop Quiz: How long can a Sprint last?

1. 1 week
2. 2 weeks
3. 3 weeks
4. 4 weeks
5. All of the above
6. None of the above

Agile Case Study

Netflix Corporate Culture

https://jobs.netflix.com/culture

Agile Case Study: Netflix Employee Culture

people over process

1. encourage independent decision-making by employees
2. share information openly, broadly, and deliberately
3. are extraordinarily candid with each other
4. keep only our highly effective people
5. avoid rules

Agile Manifesto

We are uncovering better ways of developing
software by doing it and helping others do it.
Through this work we have come to value:

Individuals and interactions over processes and tools

Working software over comprehensive documentation

Customer collaboration over contract negotiation

Responding to change over following a plan

That is, while there is value in the items on
the right, we value the items on the left more.
https://agilemanifesto.org/