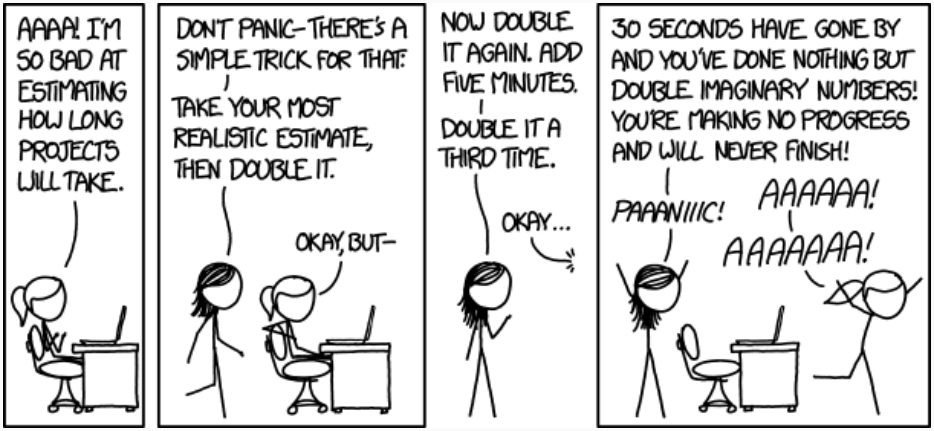
Quantified Tasks

January 25, 2024

Engineers for Exploration, UC San Diego

Obligatory XKCD



[1]

[1] https://xkcd.com/1658/

Fundamentals: What do we need to plan?

- What do we need to accomplish?
- How long is that going to take?

ENGINEERS FOR EXPLORATION

The Church of Agile

The 10 Commandments of Agile

- Thou shalt stop all work once a day for a period of time to stand in the sacred scrum circle.
- Thou shalt not blab about bullshit that no one cares about during the scrum.
- Thou shalt honor thy scrummaster.
- Thou shalt not touch, mark, or stick pins in the storyboard, for the storyboard is a sacred place.
- Thou shalt only use Post-It brand sticky notes on the storyboard.
- Thou shalt not use Post-It notes which have sticky crap all over the back.
- Thou shalt not have a scrum bigger than 7 members, for 7 is a sacred number..
- Thou shalt never speak ill about Agile, for Agile is perfect.
- Thou shalt waste a day after a sprint for retrospective and planning.
- Thou shalt spread the word about Agile to everyone thou knowest.

The Deadly Sins Of Agile

- Making cards with large tasks instead of breaking them down into smaller tasks.
- Peeling Post-It notes from the bottom instead of from the side causing them to curl up when they are stuck to the storyboard.
- Putting developers on tasks they don't enjoy or aren't their specialty. Just because there is an open task and a free developer doesn't mean they should be put on that job. Use the developer that makes the most sense by skills and by what the developer wants to do. Happy employees are more productive, write better code, and tend to be more loyal to the company.
- Putting developers in teams where they don't agree on coding style and tools and such. (See Horseman 2 Of The Scrumpocolypse)
- Editing or adding a new Post-It note on the storyboard after the sprint has started. Post-It notes placed on the storyboard should be considered READ-ONLY.

Coding Style Manifesto

There is also a coding style that all Agilists should use: The Coding Style Manifesto

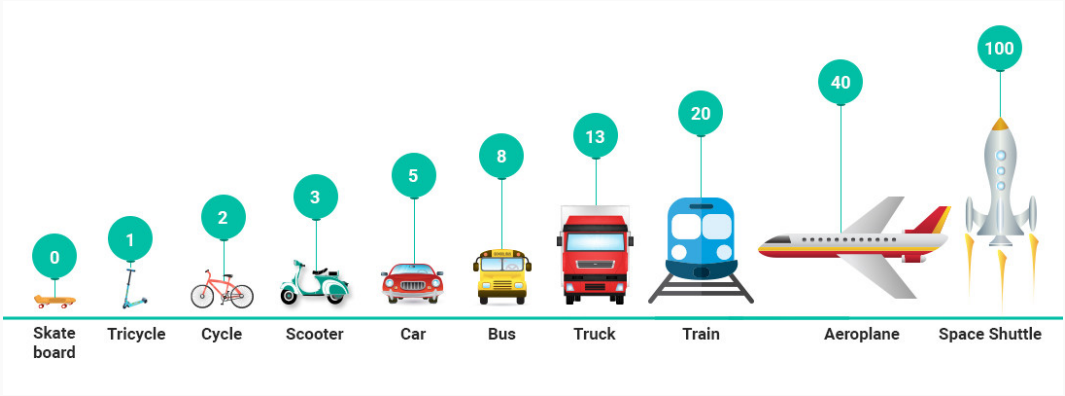
Have a question or comment about Agile? Send an email to Luthor The High Priest (luthor.priest@aol.com) and your question shall be answered. You can also follow us on Twitter https://twitter.com/churchofagile

2

[2] https://www.thechurchofagile.org/commandments.php

ENGINEERS FOR EXPLORATION

Agile: Story Points



3

[3] https://www.linkedin.com/pulse/understanding-agile-story-points-suren-gaur

ENGINEERS FOR EXPLORATION

Exercise #1

Let's define making eggs and bacon as 3 story points.

How many story points is making a hamburger?

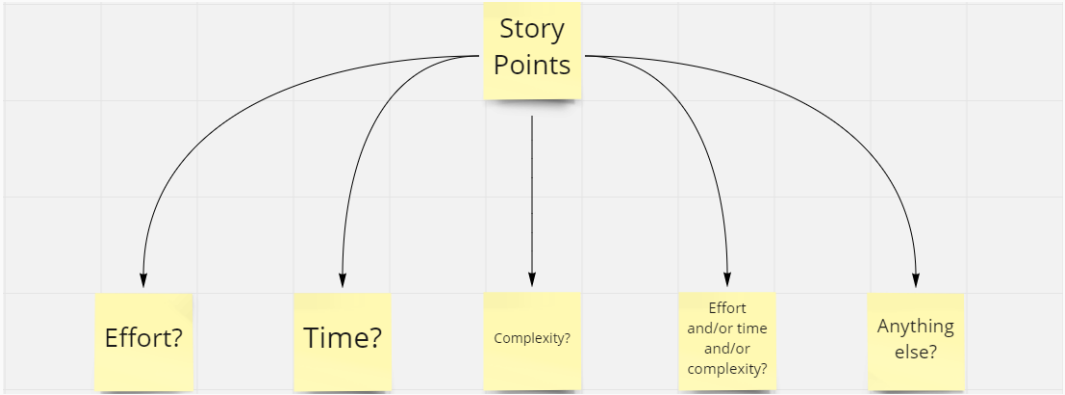
ENGINEERS FOR EXPLORATION

Exercise #1

How about making Beef Wellington?

ENGINEERS FOR EXPLORATION

Agile: Story Points



4

[4] https://maciejjarosz.medium.com/the-conundrum-about-story-points-pointless-or-not-b5d715180c96

ENGINEERS FOR EXPLORATION

What do we actually want to know?

- Planning
  - Impact
  - Gravity
  - Priority
- Estimation
  - Distance
  - Friction
  - Relativity
- Stability
  - Origin
  - Caught

See `https://www.quantifiedtasks.org/p/overview`

ENGINEERS FOR EXPLORATION

Planning Measures

Impact

- Importance of Task to goals of the Solution
- 1 - Wishlist
- 5 - Essential
- Defined by product owner/stakeholders

ENGINEERS FOR EXPLORATION

Gravity

- Importance of Task to the next release
- 1 - Not slated for inclusion
- 5 - Must-have
- Defined by product owner/stakeholders/developers

ENGINEERS FOR EXPLORATION

Priority

- Urgency of Task
- 1 - Not on the schedule
- 4 - Now
- 5 - Emergency!
- Defined by product owner/developers

ENGINEERS FOR EXPLORATION

Estimation Measures

Distance

- Estimated time frame for completion
- **Time frame if you understood everything about the task**
- 1 - Single work session
- 5 - Exceed a sprint

ENGINEERS FOR EXPLORATION

Friction

- Available documentation and precedent for Task
- 1 - Complete tutorial available (i.e. follow the tutorial)
- 5 - Pure invention

ENGINEERS FOR EXPLORATION

Relativity

- Likelihood that the task is a “black hole” with an indeterminable completion date
- 1 - No known uncertainty (i.e. no schedule risk)
- 5 - Virtually no certainty (i.e. extreme schedule risk)

ENGINEERS FOR EXPLORATION

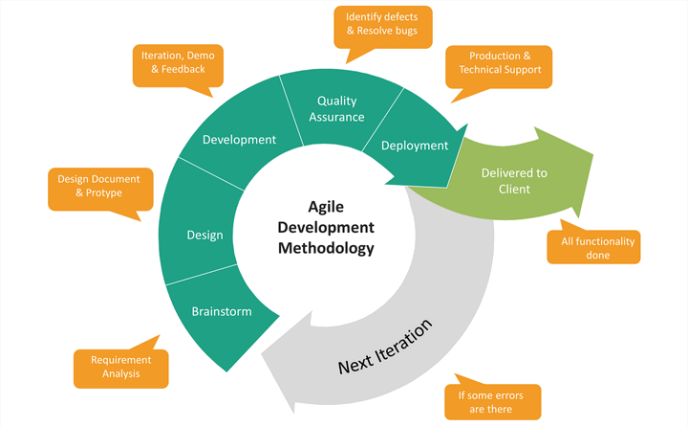
Energy Points

$$(Distance + Friction) \times Relativity$$

ENGINEERS FOR EXPLORATION

Stability Measures

Software Development Lifecycle



5

[5] https://serenagray2451.medium.com/agile-software-development-life-cycle-b3ed0f0f7212

ENGINEERS FOR EXPLORATION

Origin

- Stage at which an Issue **originated**
- 1 - Planning
- 2 - Design
- 3 - Implementation
- 4 - Validation
- 5 - Production

ENGINEERS FOR EXPLORATION

Caught

- Stage at which an Issue is **detected**
- 1 - Planning
- 2 - Design
- 3 - Implementation
- 4 - Validation
- 5 - Production

ENGINEERS FOR EXPLORATION

Volatility

$$(Caught - Origin) \times Impact$$

ENGINEERS FOR EXPLORATION