

New York City Spring 2010

Agile
Firestarter 

Agile
Firestarter 

Agile Estimation

Dan Berlin

daniel.l.berlin@gmail.com

Wikipedia: Estimation is the calculated *approximation* of a result which is **usable** even if input data may be incomplete or uncertain.

Common Problems with Estimation

Manager: “How long will this take?”

Engineer: “Do you mean if I worked full time on
JUST THIS, or with my other work?”

Calendar time is not a measure of effort.

Common Problems with Estimation

Manager: “Joe said it would take three weeks.”

Engineer: “What!?! He’s wrong, the MountainCrusher library already does that, and I have a copy right here.”

One’s mountain is another’s molehill.

Common Problems with Estimation

Manager: “Why is this taking so long!? You said this would only be two days!”

Engineer: “I said that 4 months ago! The current system architecture makes this feature much harder to build.”

Estimates have a short shelf-life.

Common Problems with Estimation

Engineer: “I don’t know how long it will take until I know every detail.”

Manager: “I gave you everything we have, now just take a guess.”

You will never know every detail up front.

Common Problems with Estimation

Manager: “Now how long will it take to do Z?”

Engineer: “Well, I haven’t even done B yet, so I don’t have any way to know!”

Manager: “How long for B?”

Engineer: “About a week.”

It is easier to estimate in the near future.

Agile estimation avoids these common problems (and more) by taking a more *practical approach* to estimating work.

How to Estimate

1. Define your work (User Story Modeling)
2. Collaborate and compare to increase accuracy (Planning Poker, Relative Sizing)
3. Use a unit of work, not time (Story Points)
4. Measure real progress (Velocity)
5. Adapt to change (Re-estimation)

User Story Modeling

(Who) wants (what) so that (why)

- Keep your user stories small
- A story is a conversation starter, and gets more detailed over time

User Story Modeling

Good stories satisfy

INVEST:

- Independent
- Negotiable
- Valuable
- Estimable
- Small
- Testable

GOOD:

Billing wants to see a summary page of all unpaid accounts, so that they can collect payments.

BAD:

Our company wants a new website to increase sales.

Users want rounded corners on the search button.

User Story Modeling:

Acceptance Criteria

Story: Users want to import music from a folder so that they can include their own music in the library.

Given a user is on the "Library Folder" screen

- *When* the user clicks Add *Then* the Add Folder screen shows a radio option for "Search for Music"
- *When* the user saves a folder with "Search for Music" checked *Then* the folder is added to the list of folders.

Planning Poker

- After all the stories are written, *prioritize!*
- Choose which stories to estimate first
- Estimate by consensus
 - Have story owner give a brief overview
 - Only open discussion if the estimates don't match
 - Record estimating assumptions

Story Points

Story Points are a unit of size/complexity, not duration

Relative sizing: estimate based on other stories

An alternative unit: Ideal Days

Don't worry: You *will* calculate duration later...

Story Backlog

- This is just a bucket for all your stories
- Keep it organized!
- Pull stories off this list to create an iteration

Iteration 1

- Usually 2-4 weeks
- Do high risk / high priority work first
- Make sure everyone can be kept busy
- How much can you do in one iteration?
 - Take the estimates off the stories at first
 - Try not to kill yourselves...

Team Velocity

- Velocity is the number of points you can complete in one iteration
- If you can demo it, you get credit. No partial credit!
- Over time this number will stabilize (usually after 3 iterations)

Points of work / Velocity * Iteration length =
Calendar Time!

Re-estimation

- Re-estimate when you have new information that affects your previous estimates
- Review all your story estimates before each iteration
- Re-estimate if you change a story

Lab - Planning Poker

- Break up into groups
- Define roles you will represent:
 - Business users (2) and team members (design, dev, QA, etc)
- We work at Expedia.com and have 20 million customers. You are doing an add-on for the Customer Service team. Do a planning poker for this user story:

“Customer Service wants to search for customers by their first and last name, so that they can quickly retrieve customer information when on a call.”