

Good User Stories are INVEST

Agile teams usually capture requirements in the format "As a <role> I want <solution> so that <value>". The whole team - business and development people together - improve stories by making them:

I ndependent

Independent stories can be freely re-ordered in the product backlog. Sometimes you can't get rid of an order dependency but it should be an exception.

Negotiable / Negotiated

A user story is the reminder to have a conversation. In that conversation the team negotiates the concrete solution, the "I want" part. The story may be enhanced or rewritten.

V aluable / Vertical

Each story adds something useful for the end user / customer - the "so that" part. This leads to vertical increments: E.g. a working slice of front end, scripts & DB, instead of a finished DB without front end.

E stimable

You need a rough effort estimate to guestimate ROI and order the backlog. If you can't estimate, you need to a) break the story into pieces or b) better understand what value it's meant to add or c) explore unknown tech in a time-boxed research spike.

S mall

Small stories are easier to estimate and test and hide fewer misunderstandings. "Small" can be 1 day in a web shop or 3 person-weeks for a medical product. At the very least, the team must be able to finish a story ("done done") in 1 iteration.

T estable

It must be possible to write a test (at least in theory) for each story. Otherwise, how will you confirm that the story is done? Sometimes test cases are given as acceptance criteria. If you can't think of a test, the story is probably to fuzzy.

