

# REFINEMENT AS A QUALITY ENABLER

**Quality isn't measured solely by the absence of bugs. Derk-Jan de Grood explains how refinement helps to achieve quality by means of clearly defined and well-thought-out backlog items.**

**Derk-Jan de Grood**

**A**s a software tester, I tend to think about testing when talking about quality in an IT context. I've written many times about my profession and often treated it as finding bugs in the software and verifying whether the product meets expectations. Of course, quality is about more than just the absence of bugs: it's also about developing the right solution.

In Agile development the backlog items define what the teams will be working on. In general, these work packages start as a rough idea and will be split into epics, features and eventually user stories that are small enough to fit the sprint. Quality in the context of backlog items should focus on how clearly the items are described and how well thought out the solution is.

Fledgling Agile teams are sometimes tempted to start their sprints with unclearly defined backlog items. Many of you have probably experienced that the time the team thinks it will save by just diving in is lost during the sprint, to discussions and clarifying the details. Not to mention the massive risk of building something that may work but doesn't match the stakeholders' or users' needs. Consequently, development takes more time and its velocity becomes unpredictable. The team might have given its commitment at the start of the sprint, but the sprint goals end up unmet.

## Clear and relevant

I currently work as an Agile coach in a sales environment. This is a volatile

environment, with the backlog items representing improvements enabling the sales team to perform better and boost its efficiency and revenue. There are some major differences with IT development: it's not about software, and working on the backlog items isn't the team's primary activity. Sales is. The backlog items are also much smaller, so that sales employees can pick them up and complete them whenever they have some spare time between two customers.

I've noticed that many times the team fails to complete the items in the sprint backlog. In discussing this, a few members mentioned they were busy with their primary activity and forgot about it. Another team member stated that she started to work on an item, only to find out that she didn't understand what she needed to do. I believe that what's

happening here also holds true for IT development teams, but that it shows up more clearly in a sales environment.

I draw two conclusions. First: a clear definition of backlog items ensures that team members understand what is needed and enables them to deliver the right output. When this isn't the case, they'll deliver a wrong solution or not deliver at all. Second: when backlog items don't seem relevant enough, team members choose other activities instead.

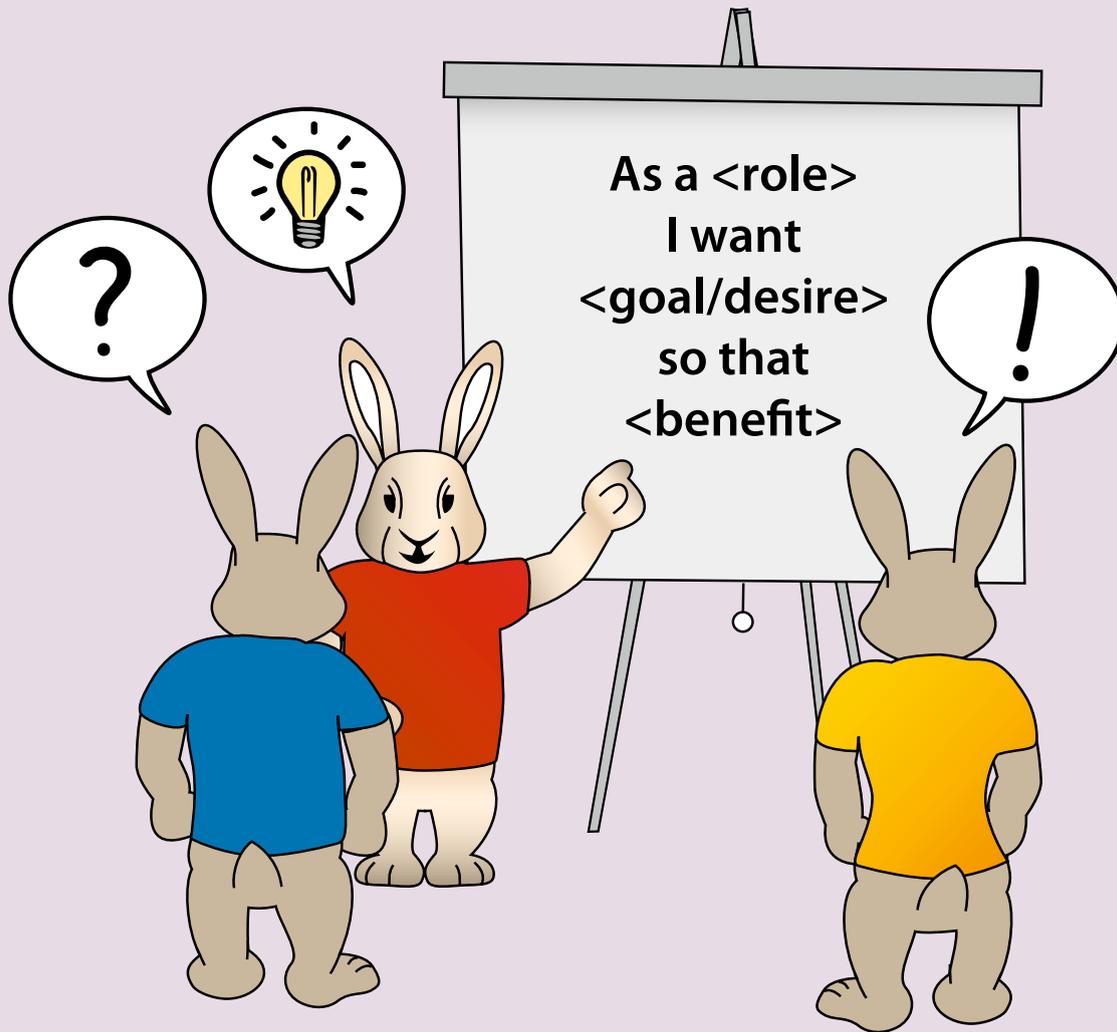
## User story

Refinement is the process in which the team spends time during the sprint preparing the backlog items that will be picked up during the next or a later sprint. A good refinement process ensures that backlog items have sufficient detail before they are

## Advantages of refinement

A well-designed refinement process:

- ensures that teams deliver the highest value first;
- increases flexibility because changes in priority don't affect team effectiveness;
- reduces surprises during development;
- brings 'first time right' to your fingertips;
- provides accurate information on progress and technical complexity, which can be used to plan and prioritize the product backlog;
- enables the effective distribution of work across the various teams;
- lets teams also take into account architectural and non-functional requirements;
- leads to thoughtful solutions (for example, generic instead of multiple similar functions);
- gives stakeholders the time to define and discuss their needs with their bosses, such as operational staff.



added to a sprint. This makes the team more efficient and predictable. Since refinement leads to well-defined solutions, size and dependencies are known. This enables the product owner to prioritize the product backlog more precisely and ensures that the highest-value items are delivered first (see the box for an overview of refinement's advantages).

In the sales environment the backlog items are smaller and less complex. The focus therefore shifts from the details to thinking carefully about what solutions actually produce business value. I stated above that quality is also about working on the right solutions, and my non-IT experience has taught me that thinking about what solution we need is just as important as defining what it should look like.

A good refinement discussion starts by clearly defining what the backlog items are all about. I've noticed that formulating this precise-

ly not only helps the team understand what to do, but also enables team members and stakeholders to challenge the initial idea and ask clarifying questions. The user story notation 'As a <role>, I want <goal/desire> so that <benefit>' can be very useful as a starting point. Although many teams leave out the benefit part, I think it's crucial if you want to challenge the proposed solution.

### Customer perspective

Let me illustrate this with a real discussion from the sales environment. A team member proposed we boost customer satisfaction by giving away a small seasonal present, such as a tulip. 'We shouldn't just give away presents,' another team member replied. 'We should add an incentive.' So we discussed where we wanted to boost sales and what audience would be interested. Talking about the benefits of the incentive action, it became

clear that we wanted customers to change their behaviour. 'A one-time reward might not be sufficient, then,' a clever team member remarked. 'We want them to repeat their action so they get used to it and experience the advantages.' We came up with a stamp card, plus a present after, say, five repetitions. From that point onwards the discussion really took off. We asked ourselves questions like: are there other customer groups we want to address, are there other incentives they would prefer, what customer needs do we address and could we fulfil these in another way as well?

Critical questions helped the team and the stakeholders to be more creative and think outside the box. The process we went through was about more than just filling in the details of the initial idea, but doing so helped us come up with other solutions. This made us discard the initial solution and replace it with far more effective ideas, resulting in better-quality solutions. Since these backlog items were clearly defined, the team not only understood their relevance, but could work on them effectively as well.

*Derk-Jan de Grood works for Valori as a senior test manager and Agile coach. In 2016 he published the book 'Agile in de echte wereld – Starten met Scrum' (Agile in the real world – Getting started with Scrum). At the Software-Centric Systems Conference, 4 October in Eindhoven, he will lead an interactive session on continuous integration and development.*

**Edited by Nieke Roos**