



World Leaders in Research-Based User Experience

Tracking Research Questions, Assumptions, and Facts in Agile

Summary: User-related questions and assumptions are not tracked throughout a product's lifecycle, causing misalignment and overconfidence. Documenting these questions and assumptions in a knowledge board differentiates them from real facts.

By [Rachel Krause](#) on December 15, 2019

Topics: [Agile](#), [Research Methods](#)

User research can be difficult to conduct in Agile environments and, due to time or budget constraints, teams tend to make assumptions about their users in order to move forward in the development process. In an ideal world we want to make decisions based on actual data, but sometimes we have to make assumptions first and test them later.

An **assumption** is something we take on faith or as a best guess despite a lack of proof. We can't know everything about users or their environment before starting a project, so some assumptions are usually necessary. The danger with assumptions is that they are often treated the same as facts later in the project, when team members forget that the assumptions have a shaky base. But decisions based on incorrect assumptions can have serious consequences, so teams need to be aware of their assumptions by properly documenting them and creating action plans to turn them into real data.

With all of the moving parts in an Agile process, formally tracking questions and assumptions is commonly neglected. In this article, we show how Agile teams can track and test assumptions.

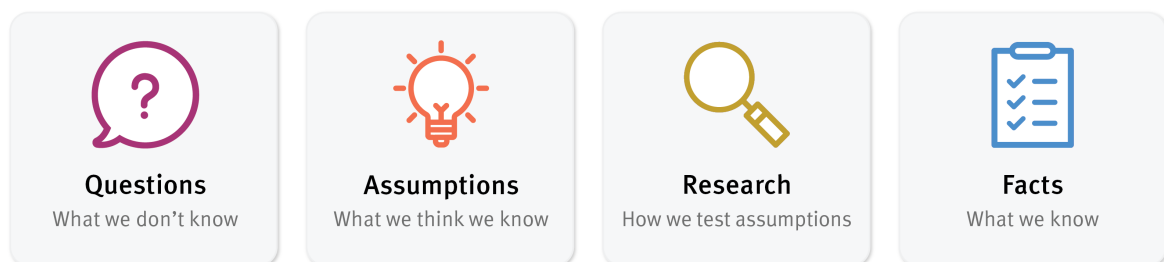
Creating a Knowledge Board

There are four types of user-related statements and activities that typically inform Agile development:

1. Questions about user behaviors, attitudes, or motivations
2. Assumptions about these behaviors
3. Research to test assumptions
4. Facts based on collected user data

The same statement can start as a question, become an assumption, and then become a documented fact based on user research. (During the progression through these phases, the information rarely stays constant: usually, as the evidence solidifies, the conclusion needs to be modified.) Let's discuss each of these individually.

KNOWLEDGE BOARDS IN AGILE



NNGROUP.COM **NN/g**

1. Questions: What We Don't Know

Agile teams have a product backlog that they pull from in order to determine what will be committed in the upcoming sprint. This product backlog is comprised of new features and technical debt in the form of [user stories and epics](#). The backlog is where we want to formulate research questions within our teams to set

ourselves up for user research.

Brainstorm these research questions together with your team so that everyone has a shared understanding of your **knowledge gaps**. You won't know the answers to these questions yet, but they will help you formulate a plan for uncovering the answers. Do this consistently as part of your backlog-refinement meetings so that you uncover gaps throughout the entire product lifecycle.

Let's look at a couple of examples:

	Example 1	Example 2
Feature from Backlog	Provide offline functionality	Build a user profile
Research Question	<i>Will our users want to use our application offline?</i>	<i>How much personal information will users provide?</i>
Next Steps	Do additional research before building functionality that users may not need	Determine how much information to collect and gauge how much users trust our product

Once you can guess the answer to a question or have an idea of what users might do, it's ready for the second phase.

2. Assumptions: What We Think We Know

Phase two of this process is making a guess — or an assumption — as to the answer of one of your research questions. You may have some data to back up your guess — whether it's a persona, an insight from early user research, or something your client or stakeholder passes along to you. This data does not *directly* answer the research question, but it allows you to make a semi-informed guess and move forward. Wild guesses are the weakest assumptions, whereas estimates derived from empirical information or insights are better, though still not the same as proven facts.

Document the assumptions you're making and add them to the corresponding

research question, whether you're doing so in a text editor or on a Kanban board.

Aim to include the following information for each research question:

- The assumption(s) you're making
- Where each assumption came from (client, persona, interview, etc.)
- Next steps for research

The next steps for research will list whatever research you need to test your assumptions. They could include [usability testing](#), [field studies](#), website [analytics](#), or additional [user interviews](#).

Using our examples from phase one, here's how we might document what we think we know:

	Example 1	Example 2
Research Question	<i>Will our users want to use our application offline?</i>	<i>How much personal information will users provide?</i>
Assumptions	<p>Users may be in a basement without access to internet</p> <ul style="list-style-type: none"> • Data collected from stakeholder on October 31st 	<p>Our target market will likely provide name, email address, phone number, and current location</p> <ul style="list-style-type: none"> • Based on case study from our closest competitor
	<p>Users don't want to use their personal cellular data at work</p> <ul style="list-style-type: none"> • Insight from user interviews during discovery phase 	

Next Steps for Research	Conduct field studies to observe users who may use the application in a basement and determine what features will be useful for them	Run A/B tests of two different account-creation forms to determine the most applicable account fields that users will complete when signing up

3. Research: Testing Assumptions

Once you've documented assumptions and you know the next steps for gathering data, it's time to conduct research and test your assumptions. After the designated research has been conducted, you will move forward in one of two directions:

1. If your assumptions were wrong and you need additional information, refine them and continue documenting them until you feel confident in your data.
2. If your assumptions are validated, you're ready to move on to phase four.

4. Facts: What We Know

Now that you can confidently answer the research questions documented in phase one, you can turn them into factual statements. It's important to keep these documented and continue to revisit them throughout the product lifecycle, so that your team is aligned.

Looking at our same research questions as above, we can now provide answers based on the data we collected from our studies:

	Example 1	Example 2
Research Question	<i>Will our users want to use our application offline?</i>	<i>How much personal information will users provide?</i>
Findings from Research	In our field studies, participants matching our	Participants who encountered the layout

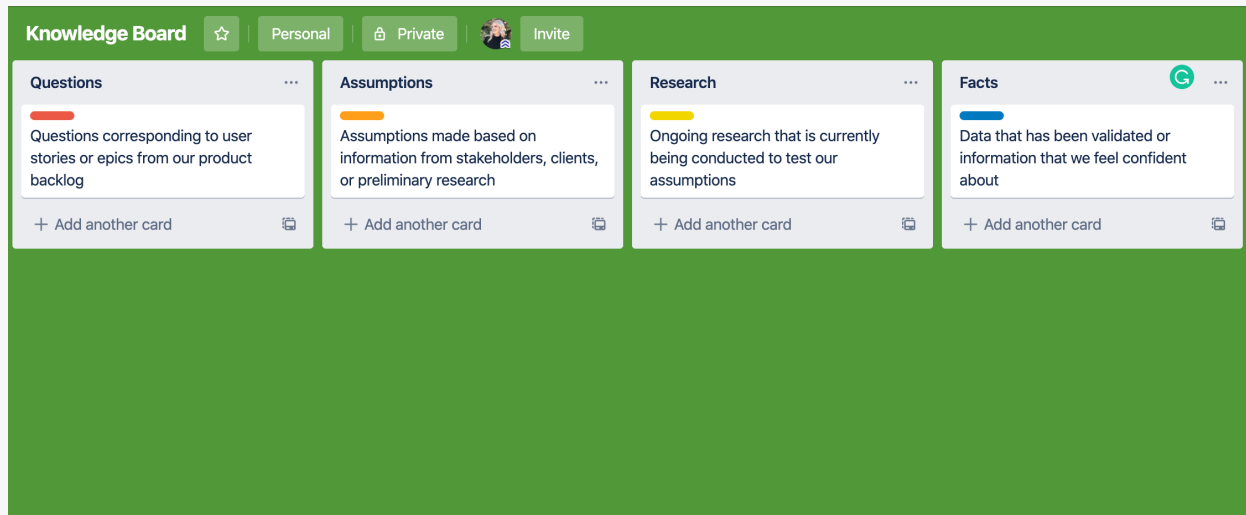
	<p>primary persona spent half of their workday in a basement without stable access to a network connection.</p>	<p>with the phone-number field were more likely to abandon the task altogether, whereas participants who encountered the layout without the phone number were more likely to complete the form.</p>
Factual Statement	<p>Since our application allows these users to optimize a large portion of their workflow, we have concluded that users will want to use our application offline.</p>	<p>We've concluded that users will provide a name, email address, and current location, but not a phone number when creating a profile.</p>

Documentation Methods

Just like there is no definitive rulebook when it comes to [Agile backlogs](#), there is no one right way to document the difference between assumptions and facts throughout your process. Choose a method that will mesh well with your team's current process and is easily accessible to the team.

Kanban Board

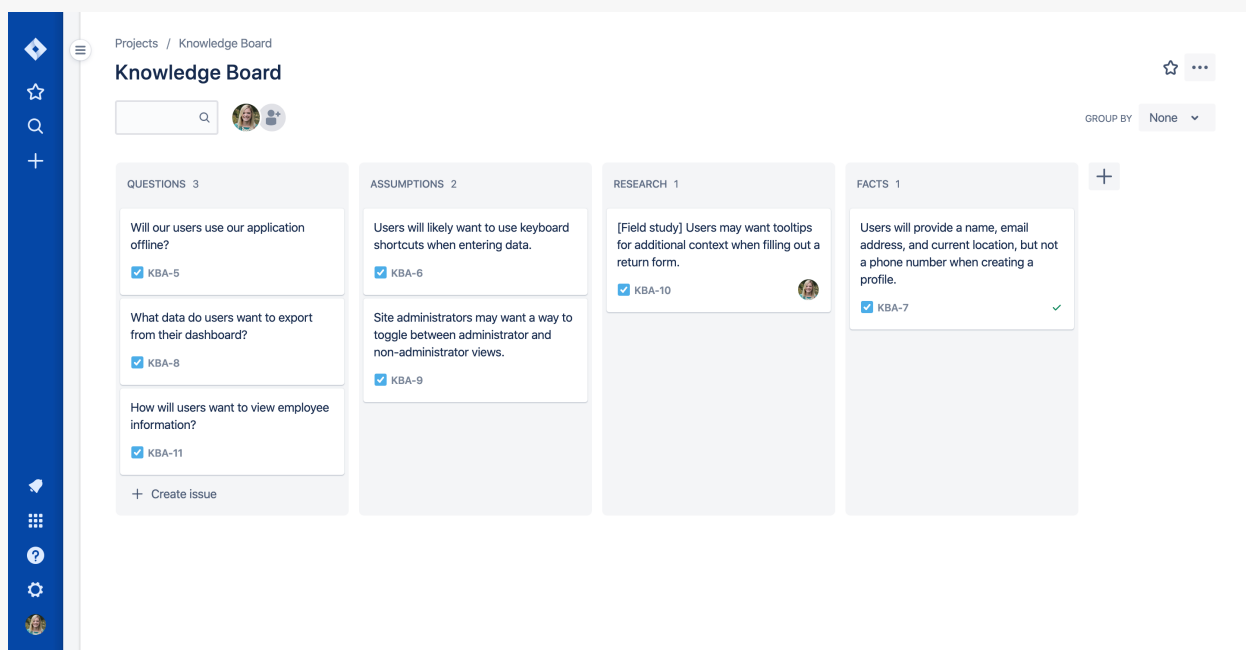
This is my recommended method, as it's easy to see exactly where knowledge is in its journey. Create a column for each phase of the process. A digital Kanban board is ideal for easy access by all team members, but you may also consider a physical Kanban board for colocated teams for constant visibility.



Trello.com: A Kanban board is ideal for tracking at which stage each research question sits. Cards are movable from one column to the next; you can also specify due dates, owners, and tags for each question.

Project-Management Software

Some teams find it easiest to document knowledge in the same way that they document their product backlog. They create a separate backlog of research questions and document their progress in the same way they would in the product backlog. The details of each question will live within each item and can be transferred over to the product backlog easily if applicable.



Jira Software: The knowledge board can live within your existing project

Jira Software: The knowledge board can live within your existing project-management software.

Text Document

If your team is more comfortable using Google Drive, Dropbox, or Sharepoint as a repository of information, you may consider keeping a running document tracking the assumption information. If you choose this route, however, make sure to use the outline feature of your text editor so it's [scannable](#) and easily contributable.

Conclusion

There are many instances in Agile where we have to make assumptions, whether it's because we're short on time or we don't have instant access to users. The goal is to turn these assumptions into validated facts through user research.

A second goal is risk management, based on recognizing your "known unknowns." Even if you can't turn all unknowns into facts, you're not going to be completely blindsided if one of them turns out to be different than what you had guessed (because you knew that your guess could be wrong). Much worse to be hit by the infamous "unknown unknowns," which are the things you never even thought of. Such surprises can truly doom a project, because you can't manage a risk that you aren't aware of. Reducing this risk is a key reason to spend time documenting what you don't know.

Reference

Kirkland, Kieron. 2017. *How a knowledge Kanban board can help your user research*. Retrieved from <https://userresearch.blog.gov.uk/2017/02/16/how-a-knowledge-kanban-board-can-help-your-user-research/>

Share this article: [Twitter](#) | [LinkedIn](#) | [Email](#)

Copyright © 1998-2022 Nielsen Norman Group, All Rights Reserved.