



Creating user-tested designs, properly

An Agilesphere white paper

Version: 1.0

August 16, 2018

Author: Hiren Shah, hiren.shah@agilesphere.co.uk

Abstract

One thing government IT projects do well (compared to the private sector) is focus on user research. User research is an essential part of these projects and is led by user needs – letting the product develop into its rightful state. Effective user research requires efficient designs for user testing.

Efficient designs incorporate user needs, keep service goals in mind and have business change at the heart to reach the suitable end state every time.

The designs can have a low or high fidelity, depending on which stage of the project you're in. Once the designs are ready, they need to be tested by the end users and updated as per the feedback gathered. Finally, the tested user designs are attached to the user stories and implemented into the product.

The process here is simple – user needs are adapted into designs, tested and implemented. It might sound straightforward enough, but there are a few stumbling blocks along the way. To prevent these issues from happening, it requires a little more thought into making the process as lean as possible and working like a well-oiled machine.

Stumbling blocks you could run into:

- if the user testing happens after implementation, the change is quite expensive
- if the product owner isn't involved in the entire process, the user needs might be met by the design, but the service goal and transformational element could be lost
- if the business change aspect of the service is not tested and articulated properly, the user testing could lead to misleading feedback

This paper looks closely at how to develop a lean process in order to create user-tested designs from user research to be used by feature teams.

How do teams create user-tested designs?

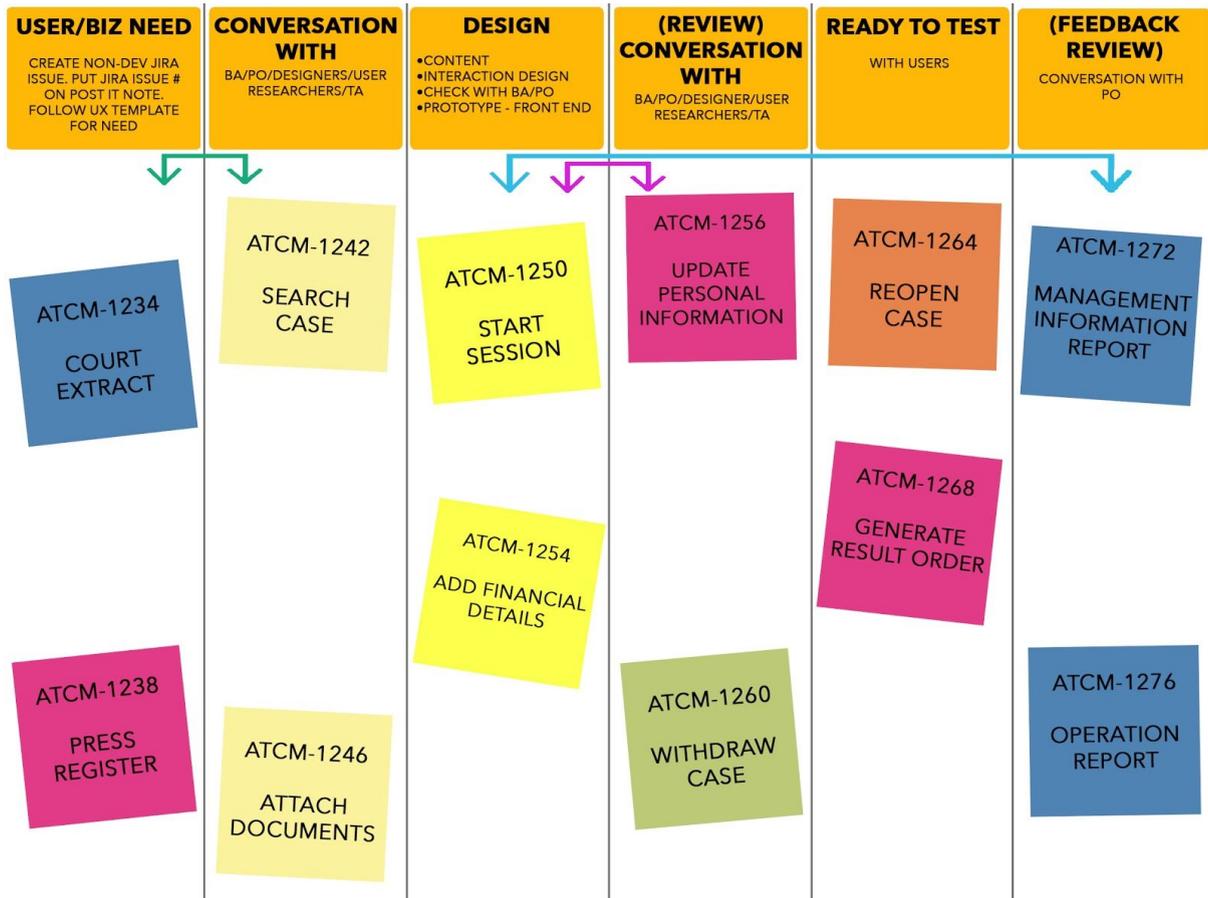
Different teams use different approaches to create the designs, based on user needs and outcomes:

- some teams perform design with low fidelity prototypes as soon as user needs are obtained. After, these are tested and improved based on the user testing. When the testing phase is complete they're finally developed
- some teams prefer to perform user testing after the development has been done
- some teams use high fidelity prototypes to perform user testing and invest more time in the design stage to get these prototypes in front of users before development
- some teams don't involve the product owner during the design and user testing phase – often they then find out (the hard way) that some designs don't meet the service end goal, even though they may pass the user testing

Our process

The first step was to visualise the workflow following the lean principle.

It took us a few sprints, but we were able to refine our workflow using a Kanban board. This helped us visualise it:



Our workflow

1. **User researchers come up with the user needs**
2. **User needs are discussed with designers and the product owner**

If there's a separate content designer, it's wise to involve them too. This makes sure the designers and product owner understand the user needs, with the designers being informed from the beginning.

The product owner has a crucial role to play in this conversation as they can prioritise certain user needs over others, based on the service goal. And more importantly, they can remove any ambiguity in terms of understanding of the user needs from the various parties.

On some features, we'd also recommend involving technical architects in the conversation to know as early as possible if the technical solution is at risk of being

blown out of proportion or if there are any opportunities for reuse, if we choose that route.

3. Designer and content designer come up with a design (ideally a low fidelity prototype)

4. Review the designs

This is followed by another conversation between the designer, the product owner, technical architect and the user researcher to review the designs developed. This internal checkpoint make sure user needs are captured in the design and the product owner has an opportunity to review the design – making sure not to deviate from the overall service goal.

A decision is also made during this conversation as to whether the design is ready for user testing or needs to be iterated.

5. User researcher takes prototypes to user testing

The users and the findings of the user testing are discussed with the product owner and designer. It can go back to the design stage from here to iterate and incorporate the findings of the user testing.

6. User tested designs ready for development

These user tested designs are finally attached to the user stories going into refinement with the development team by the business analysts.

Superficially, it seems there are a lot of conversations in process, with many people involved. But really, it's not complicated.

Each week we have an hour-long meeting in which the designers, product owner, user researcher, technical architect and business analysts are involved. The stickers from the Kanban board, which are ready in the conversation column, are picked up and presented, either by the designer or the user researcher. The product owner decides whether it's ready for testing. If it is, the ticket's moved to the next column.

Bi-weekly stand ups

We have bi-weekly stand ups with the user researcher, designer, content designer, product owner and business analysts around the Kanban board.

The most frequent questions asked are:

- For the user researcher
 - Are there any user testing sessions booked that require designs?

- For the designer
 - Are you working on the right designs needed for user testing and consumption from the user stories upcoming for refinement?
- For the business analyst
 - Do you need any designs for stories that the team is going to pick up for the story refinement sessions?
- For the product owner
 - Are the user researcher and designers working on the designs and user needs for the features and stories that are part of the top 20% of the prioritised backlog?

Often, prioritisation decisions are made along with the product owner to make sure the right designs are being worked on. We also check the business analysts aren't starved of designs for the user stories, which are picked up in the refinement sessions.

Last (but definitely not least), we identify blockers and any required decisions and create actions to resolve them.

What are the advantages of this process?

User testing happens *before* development begins

The big advantage of our process is that user testing happens before the development begins – it's more expensive to incorporate user testing feedback on developed software than at the prototype level.

We'll never be able to avoid **all** rework, but following this process makes sure the rework is minimum and more cosmetic in nature when the developed software is showcased to the users. The major user testing feedback is addressed at the design stage, which is significantly cheaper.

Bi-weekly stand ups keep us updated

Our stand ups make sure the tickets are moving smoothly between the columns and highlight any blockers – helping us to come up with the most appropriate action. The board also provides a visual radiator to any stakeholder in terms of what the team is working on and which designs are in progress.

Working on designs relevant to top stories in our product backlog

We usually print the top 20% of our prioritised product backlog on the Kanban board. This means the designers and user researchers know exactly what's coming up and prioritise

the tickets that are relevant to the top 20% of our product backlog. The aim is that the designs are ready for at least two of the next upcoming sprints.

Initially, the whole project was in the Kanban board – we were tracking and following this with rigour. As time passed, there were a few designers who started working from home, but by the very nature of the work of the user researchers, the designers needed to be in the field testing and researching the user needs.

To overcome this challenge (the Kanban board not being available for those not in the office), we created a user story in Jira and all conversations and design decisions were captured in the ticket. This provided us with the audit trail of the activities and decisions made on a particular design as it moved through the columns.

Synergising design elements across programme

Another benefit of the process is that when there are multiple projects in a program, it's easier to synergise design components and elements. The designers from other teams can be invited during the conversation phase and design elements can be reused to increase rapid prototyping.

It also gives us a sense of how much work is remaining at the design stage, to make sure the design team are ahead of the development by at least two sprints.

And finally

The team should fine tune and come up with their own process of capturing user needs, repeating designs, user testing and playing them in development sprints.

The process detailed throughout this paper can be used as a reference to make a start, based on the makeup and skills within the team.

What works for us may not necessarily be something that works for all scenarios, hence the importance of the team repeating this process – capturing the lessons learnt and acting on these lessons is essential.

[Hiren Shah](#), Delivery Manager