

Agile Design: Intentional Yet Emergent

by Mike Cohn • 14 Comments

The [agile process](#) favors an incremental, just-in-time approach to design. As such, Scrum projects do not have an upfront analysis or design phase; all work occurs within the repeated cycle of sprints. This does not mean, however, that design on a Scrum project is not intentional. An intentional design process is one in which the design is guided through deliberate, conscious decision making.

The difference on a Scrum project is not that intentional design is thrown out, but that it is done (like everything else on a Scrum project) incrementally. Scrum teams acknowledge that as nice as it might be to make all design decisions up front, doing so is impossible. This means that on a Scrum project, design is both intentional and emergent. The design emerges because there is no up-front design phase (even though there are design activities during all sprints). Design is intentional because product backlog items are deliberately chosen with an eye toward pushing the design in different directions at different times.

As an example of how the product backlog items can be sequenced to influence the architecture of the system, consider a workflow system I worked on. The system supported a fund-raising company that produced specialized T-shirts and similar products. School-age children would go door to door selling these items. The sales revenue would be split between the company and the organization the kids represented, such as a school, sports team, or other group.

For each sale, the kid would complete a form and send it to the company, where it was scanned, sent through an optical character recognition (OCR) process, and converted into an order. To keep shipping costs down, orders from the same organization were batched together and sent back to the organization, after which the kids would hand-deliver the items.

Our software handled the entire process—from when the paper was received by the company until the shipment went out the door. Kids have notoriously bad writing and are bad spellers, so our system had to do more than just scan forms and prepare packing lists. There were various levels of validation depending upon how accurately we thought each order form had been read.

Some forms were routed to human data-entry clerks who were presented the scanned form on one side of the screen, the system's interpretation on the right, and an additional space to make corrections. Because thousands of shirts were processed on the busiest days, this process needed to be as automated as possible. I worked with the product owner, Steve, to write the product backlog.

After that, I met with the development team to discuss which areas of the system were the highest risk or we were the most uncertain about how to develop. We decided that our first sprint would focus on getting a high-quality document to run through the system from end to end. It would be scanned, go through OCR, and generate a packing list. We would bypass optional steps such as deskewing crooked pages, despeckling pages, and so on but would prove that the workflow could be completed from start to finish.

This wasn't highly valuable but it was something that needed to be done, and it let the developers test out the general architecture. After we accomplished this, we had a basic database in place and could move documents from state to state, triggering the correct workflow steps. Next the developers asked the product owner if they could work on the part of the system that would display a scanned document to a human who would be able to override the scanned and interpreted values. This was chosen as the second architectural goal of the project for three reasons:

- It was a manual step, making it different from the workflow steps handled already.
- Getting the user interface right was critical. With the volume of documents flowing through this system, saving seconds was important. We wanted to get early feedback from users to allow time to iterate on usability.
- After this feature was added, users could start processing shirt orders.

The project continued in this way for a few months and was ultimately tremendously successful, meeting all of the prerelease targets for reliability and throughput. A key to the success was that

the product owner and technical personnel worked together to sequence the work. The closest the team got to a design phase was the first afternoon in the conference room when we identified risky areas and dark corners and decided which one we wanted to tackle first.

From there the design emerged sprint by sprint, yet was intentionally guided by which product backlog items were selected to illuminate the dark corners and risks of the project. [Succeeding with Agile](#) goes into much more detail on agile design, including how the roles of architect and user experience designer change with Scrum, the concept of emergent design, and how teams work together and with the product owner to deliver increments of functionality that guide the design of the final product.

Posted: December 4, 2009

Tagged: product backlog, product owner, sprinting, teams, product management
