Chapter 21:
PRODUCT VALIDATION

Evidence of Valuable, Usable And Feasible

The past few chapters have had references to what I call product validation. This refers to verifying that the product spec is describing a product that you have *evidence* will be successful, but doing so without actually building out and deploying the product.

This used to be a very expensive and difficult thing to do, and was generally only done for products that were very expensive to tool and manufacture, such as automobiles. However, for just about every type of product today, the costs to produce effective prototypes or simulations has come down so far that I am amazed I continue to encounter product teams that don't do this.

One of the biggest and most common mistakes product teams make is to have far more confidence in their product specifications than they should. They move forward, thinking they'll adjust the product—if necessary—once they get beta feedback. But of course beta is far past the time for major changes, and it is little wonder so many initial product releases are so far off the mark.

As product manager, it is your responsibility to ensure that this doesn't happen to your product. The key is to prove to yourself and to the rest of the product team that the spec you give them describes a winning product. You can do this, and it costs far less than you probably think.

There are three important types of validation that you need to perform before you hand over a final product specification to the engineering team:

Feasibility Testing

One immediate question is whether or not the product is going to be buildable, with the technology time and funds currently available. Your engineers and architects should be very involved in investigating technologies and exploring possible approaches. Some paths will be dead ends, but hopefully others will prove viable.

What is most important is that, if there are obstacles the engineering team considers insurmountable in this product's timeframe, it is important to know this now rather than much later in the process—after the time and money has been lost.

Some products have more technical risks than others, but if yours has significant risks regarding feasibility, make sure you address them early.

Usability Testing

Your interaction designers will be working very closely with you to come up with ways of presenting the functionality in the product so that users can figure out how to use it.

Usability testing will often uncover missing product requirements and, also—if done well—identify product requirements that are not as necessary as originally thought. You should plan on multiple iterations before you come up with a successful user experience.

The purpose of the prototype is to have something to test on real people, and usability testing is the art and science of getting useful feedback on your product from your target customers. Certainly, the product manager and designers will use the prototype and learn a great deal from it, but there is no substitute for putting the prototype in front of real people from the target customer base.

Note that for usability testing purposes, it is perfectly fine if complicated back-end processing is simulated—the key is to evaluate the user experience.

Value Testing

Finally, it is not enough to know that your product is feasible to build and will be usable. What really matters is whether or not your product is something users will find *valuable* and want to buy—that is, how much do users and customers like and value what you're doing?

This testing can typically be combined with the usability testing process, and the prototypes used are the same. But in usability testing, you're seeing if users can figure out how to do the necessary tasks, while in value testing you're seeing if they actually care about those tasks and how well you've solved them.

For a few small product efforts, simply working your ideas out on paper may be sufficient. But for most products—with complex user interactions or new uses of technology—prototypes are absolutely critical in order to assess whether or not the product will meet its objectives.

Most often the prototype is simply quickly assembled and clickable pages representing the eventual web site or software service. But for other types of products the prototype may be a physical device or a combination of device and software. The key is that it needs to be realistic enough that you can test the prototype on actual target customers and they can give you useful feedback.

Until recently, there was debate over the relative merits of "high-fidelity" prototypes (what I'm describing), versus "low-fidelity" prototypes (essentially paper drawings). Today, I consider this debate meaningless because the cost of high-fidelity prototypes has dropped so low, and the quality of the feedback is so much higher.

In the past, there were two major obstacles to these prototyping approaches. The lack of good prototyping tools meant that it could take a long time to construct the prototype. Another problem was in unenlightened management not understanding the difference between a prototype and the real product. Here, teams would be pressured to use the prototype as the basis for the final product, with predictable results in the quality of the implementation.

Today, there are outstanding prototyping tools that can let engineers or designers rapidly create very functional prototypes (often in hours or days) that can effectively emulate the future product—to the degree necessary—and form the basis of realistic user testing. Moreover, most managers today understand that building a simulation and building the actual product are very different animals—akin to building a scale model of a house versus building the actual home.

These are not the only ways to validate your product—especially for Internet services, there are other techniques that are also easy and effective. But I can't emphasize enough how important and valuable it is to validate your ideas before you actually go and build the product. There are always surprises, and it is far better to discover them early rather than when the product is in beta or released. Further, once the real engineering begins, a special type of inertia sets in—it becomes very difficult to make significant changes and the costs of these changes rise dramatically.

In the chapter *Prototype Testing*, I explain in detail the techniques for the usability and value testing.

Examples

You can see example high-fidelity prototypes, and a list of tools for creating them, at www.svpg.com/examples.