Chapter 4:
PRODUCT MANAGEMENT VS. DESIGN

Understanding User Experience Design

Many product people complain to me that their company doesn't staff or even understand user experience design, and they know their product suffers for it. Most say that the UI engineers just do whatever they can and that, by default becomes the design. Sometimes it's the product managers who wade into the design waters and do whatever they can. And in other cases, companies try to outsource some visual design at the end of the product development process to add in pretty veneer just before the product goes into QA.

Others tell me that their company values good user experience design, but they don't really understand the roles or how a good design comes about.

This is a very serious problem that not enough companies are aware of.

It seems to me that the design community hasn't been doing enough to address this lack of recognition for their importance to a product. While they do a good job communicating among themselves (and there are some outstanding talents in the design community, including Mark Hurst, Hugh Dubberly, and Alan Cooper, to name a few), in general I think these guys spend a lot of time preaching to

the choir. The message about the value they deliver is most needed by the teams without designers. One way to do this is to work on educating the wider product team about the need and benefits of designers to a product, especially the product managers.

The reason I care so much about this problem is simple. A good product requires a good user experience. And a good user experience requires the close collaboration of product management and user experience design.

This is a big topic, so let's start by getting on the same page in terms of what design includes. In this chapter I spell out what I consider the design-related roles essential to creating a good user experience. Note that I'm emphasizing roles rather than people, as it's possible to find people that can competently handle more than one role. But one way or the other you need these roles if you want a good user experience:

Interaction Design. These people are responsible for developing a deep understanding of the target users and coming up with the tasks, navigation, and flow that are both valuable and usable. Generally, the interaction designer maps product requirements to a design represented by wireframes, and passes them to the visual designer.

Visual Design. These people put the flesh on the wireframe and create the actual pages and user interface look and feel, which includes everything from the precise layout, colors, and fonts, but more importantly, the visual design communicates and evokes emotion in the product (which is far more important than you may think).

Rapid Prototyping. The prototypes work to capture the ideas of the product manager and designers into a prototype that can be tested on real users, and iterated upon.

Usability Testing. This person specializes in research and analysis of the users, evaluating whether products or prototypes allow a given

user to easily achieve objectives. It includes recruiting appropriate test subjects, administering the tests, evaluating the results, and recommending alternatives.

The four design roles above work closely with the product manager to discover the blend of requirements and design that meet the needs of the user. The idea is to get to the point where the software is both *usable* (users can figure out how to use it) and *valuable* (users actually want to use it).

You will also need to ensure the software you're designing is *feasible*, so you need to have a software architect reviewing the progress and prototypes. More on this later.

For large products—especially at consumer Internet service companies—you really do need all four roles represented on your team. If you're an enterprise company and you'd like to differentiate your product from your competition, one of the easiest ways to do this is to create a good user experience. As a general rule, most enterprise products are very weak in this respect.

For smaller products, you may be able to double-up some of the roles. For example, I recently worked with a consumer Internet service startup in the Web 2.0 space which had assembled a terrific team of three: a product manager, an interaction designer (who also covered usability testing), and a visual designer (who also covered prototyping). This team of three worked extremely well together to quickly come up with numerous prototypes that they then tested with target users.

One other important note: Many companies realize they need to do something in this area, but they think they can outsource this user experience work to a design firm. To some degree you can, but beware that certain functions are more appropriate than others. For example, I don't recommend outsourcing the interaction designer role because of these three reasons:

1. It takes time, over the course of several projects, to truly develop the necessary understanding of users and customers. Most design contracts don't have the time to do that, and even if they do, that knowledge is lost when the next release comes up;

2. The interaction designer needs to be on hand and deeply involved all the way through the project, from the beginning to launch. Hundreds of detailed questions will come up during development and test—having an interaction designer there to make the right decisions immediately is critical;

3. The user experience of the product is simply too core to the company to not have in-house. Given the option, it's a better choice to outsource QA.

You can get away with outsourcing visual design, as there are a number of studios that can do what you need, especially if you have a strong interaction designer on staff. You can also outsource user research and/or usability testing, although it's often expensive and I'm a big fan of informal testing (see the chapter *Prototype Testing*). The product manager and interaction designer can often team up to cover this.

For the rapid prototyper, you can borrow a developer from your engineering team, as long as you make very clear to that person that this is completely different from production-level coding, and that he or she should not try to build a prototype where any of it can be reused later in the real product. In fact, they should consider all the code in a prototype as throw-away.

There's a great deal more to say on this critical topic than can be covered in this brief chapter, but hopefully this discussion lays the foundation. Which of these roles are currently covered within your product team and which ones are missing?