Stanford eCorner

Danger's Product Development Process

Andy Rubin, *Danger*; **Joe Britt,** *Danger*; **Matt Hershenson,** *Danger*

February 18, 2004

Video URL: http://ecorner.stanford.edu/videos/1138/Dangers-Product-Development-Process

The Danger product encompasses three major pieces of technology: the hardware, the operating system and graphical user interface, and the backend service. The founders drew upon their experiences at Apple and WebTV to know what to develop in-house and what to outsource. Looking back, they are not sure how it all came together and how they got it done, but it was a huge accomplishment when the product shipped.

Transcript

So there you have it. I mean you have an end-end solution where there is really three major pieces of technology, and three independent teams in the company that are all working on their areas of technology. There's a piece of hardware that you see on the table there. There is the operating system and the graphical user interface of the hardware, and then there's a backend service. Maybe you, guys, can talk a little bit about what it was like developing all that technology, because that's a big cookie to chew, right? Sometimes, we think about that. You can name large and successful companies which under took one of those three things independently. We chose to do it all at once. That's another solution thing. Sometimes, we think back and like, "Man, we were crazy." But it's a part of the shift that we're proud of. By taking on that much, we also were able to guarantee the quality of the product in a way that if you decide that you care.

Figure out "Here's a piece of technology that I'm going to build" and "Here's a piece of technology that I should actually just buy." And there were things that, for instance, in the hardware which is much more my responsibility, there's a wireless radio in there. It's a GSM-GPRS radio which is a particular wireless standard chosen by carriers in the US and in Europe. But we didn't want to take on the actual design of the radio itself, so we bought it as a module. We opened up one of those things and there's a tin can on the circuit board and that's a radio that somebody else built. We paid premium for that but there's a lot of engineering in there that we didn't have to do. It also allowed us to decide that we might have our eye on one or two different vendors because we weren't sure who is going to have the best solution for that piece of technology. So we were flexible with that choice. There were other things where we decided, "OK, that's something that we have to do because it's a lot more fundamental to the way things get done." That's true in other parts of the software as well, too. Yeah, we learned a lot about that at places like Apple, in WebTV, Catapult. We have a lot of engineers came from them.

And the philosophy, sort of, is that if you take a big piece of code from some place else -- a lot of people do this. They'll license something, license to the graphic engine, license an operating system. Depending on the team that you have and what their skill set is, it might take you the same amount of time to adapt whatever it is that you bought, as it would take to implement it from scratch yourself and have exactly what you want. We were able to get a bunch of great people, like Andy said, that came from places that had done pieces of the different things that we knew we needed for the device. And all had very strong beliefs in how to put it together and how quickly they could do it. And we were able to leverage that and build an operating system, a graphics toolkit, a communications sub-system, a Java interpreter -- it's one of the fastest Java interpreters on the

planet. A sound system. A sound system, and put all those pieces of software together. A Web browser. Yeah, a Web browser.

It was a Herculean effort, thinking back on it. I still can't believe that we got it to work. Yeah. It's kind of a combination of bravado and naivety that gets you to go off and do this kind of a thing.