Stop Listening to Your Users

by Mike Cohn • 0 Comments • originally published in Better Software on 2005-04-01

It’s time we stop asking users what they want. Asking a user what she wants in a new software system assumes that not only does she know what she wants but also that she’s qualified to answer the question. **Just because you have a problem does not mean you are the best person to solve it.** For example, I was with a team that had just released a great new system for its call center agents. The entire project was undertaken to reduce call duration, and it was a stunning success. A month or so after release, however, the users came to the development team with a request to add a very large free-text field to the system. They wanted to type notes into this field after the call ended. This would effectively bring call duration right back up to what it was. What were we to do with this direct request from the system’s users?

I sent a few of the developers into the call center to observe calls and ask the users what they would type into this large field after each call. We discovered that the things they wanted to type were things the system could capture and summarize in a view-only field at the end of the call. This would give the users peace of mind and keep call duration from increasing. The users thought they knew the solution to their problem but they were wrong.

So, if we don’t ask users what they want, how will we know? The answer came from Norway in the early 1970s. Kristen Nygaard, one of the co-inventors of SIMULA and object-oriented programming, became involved in a failed software development project for the Norwegian Iron and Metalworkers Union. The project was failing because users were involved in the project only as peripheral participants and objects of study by the requirements analysts. Nygaard corrected this situation by inverting the roles. The union member users would now lead the requirements effort with assistance and support from the software developers. The shift was subtle but important—from developers asking, “What should we build them?” to union member users asking themselves, “What do we need next?”

This model for software development was repeated throughout Scandinavia and became known as participatory design. In participatory design, users become a part of the team designing the behavior of their software. They do not become a part of the team through management edict.
(“Thou shalt form a cross-functional team and include the users”). Rather, the users become part of the team because they are given the opportunity; they view the development of the right software as important; and the developers use requirements gathering, design, and testing techniques that are accessible to the users.

In participatory design, for example, users assist in the prototyping of the user interface from the beginning. They are involved even before an initial prototype is available for viewing.

So, stop listening to your users. On your next project, instead of asking users what they want, try going further and truly involving them as participants in the process. The results may astound you.
About the Author

Mike Cohn specializes in helping companies adopt and improve their use of agile processes and techniques to build extremely high-performance teams. He is the author of User Stories Applied for Agile Software Development, Agile Estimating and Planning, and Succeeding with Agile as well as the Better User Stories video course. Mike is a founding member of the Agile Alliance and Scrum Alliance and can be reached at hello@mountaingoatsoftware.com. If you want to succeed with agile, you can also have Mike email you a short tip each week.