

The Art of Influence

by Elisabeth Hendrickson

“THEY JUST DON’T SEEM TO WANT TO hear what I have to say!” Abby looked like she was about to hit something; I got ready to duck. “How can I get them to listen?” she demanded.

Abby had been attending a series of design meetings with the developers working on a new release. As a skilled tester, Abby had hoped that she could help the team uncover bugs early, while they were still on paper rather than entrenched in the code. But things hadn’t worked out as she’d hoped. It seemed to her that whenever she made a suggestion, the developers dismissed her comments as irrelevant.

Invisible. Ignored. Frustrated. Abby was close to giving up.

Abby’s experience is not unusual. Most groups are not very good at listening to input from other groups. Technical writers may bristle at grammatical corrections from developers; testers may reject test advice from non-testers.

So how could Abby exercise influence over a group where she has no control, no authority, and perhaps a credibility gap?

See Them as People

The first thing Abby needed to do (after calming down) was to realize that what was happening wasn’t about her.

The developers—Bruno, Carla, and Dean—were accustomed to working together. They knew each other. They were a team. They held no ill will toward Abby, but they saw her as an outsider foisted on them by management. They didn’t understand why she was there, and they were just hoping she would stay out of the way.

In other words, they weren’t reacting to Abby as a person. They were reacting to her in a role: outsider-foisted-on-us-by-management. Abby’s first task was to

get the developers to see her as a person, not as that-tester-who-just-filed-ten-bugs-against-my-code. And she needed to start seeing them as individuals instead of those-developers-who-don’t-listen-to-me.

As the newest member of the team, Abby had to invest some time beyond just showing up at the design meetings. She tagged along for lunches, made a point of talking to the others in the halls, and also touched base with each of them, asking questions outside the meetings.

Ask Insightful Questions

Initially, Abby tried asking the developers, “How does this work?” Her questions were met with blank stares. Abby thought they were unwilling to answer. In reality, they were frantically thinking, *How does this work? Does she have two months for me to explain it to her in detail? Even if she does, I don’t. Where do I start? Does she want to know about the transaction model? The class structure? What does she mean, “How does this work?”* The question was too broad.

When Abby pressed again for a broad explanation in one of the design meetings, Carla finally snapped, “Abby, I don’t know what you’re asking. Ask me a question I can answer, and I’ll answer it.”

“Oh.” Abby was caught between being furious at being put off yet again and feeling sympathy for Carla, who was clearly equally frustrated. A day later, she tried asking more specific questions:

- “It looks like the client waits for the server to confirm the transaction before dismissing this window. Is that correct?”
- “How can the user customize reports?”
- “What happens if I try to view the Web-based front end through an unsupported browser?”

Abby was delighted with the depth and thoughtfulness with which Bruno, Carla, and Dean answered her specific questions.

Decide What You Want

Even after improving her ability to ask questions, Abby still had difficulty getting the developers to listen to her. They weren’t trying to ignore her. They were wrapped up in design issues. Testing and defects were concerns for some nebulous point in the future.

However, Abby did notice that sometimes the issues she raised resulted in Bruno, Carla, and Dean considering alternate solutions. She realized that although the developers weren’t acknowledging the testing issues she raised, they were reacting to what she said.

Abby needed to decide what she really wanted. Was she seeking acknowledgement of her contributions? Did she need the developers to implement specific features to improve testability? Or did she want to influence good development practices that would lead to defect prevention?

On reflection, Abby decided that while she would like the group to acknowledge her input, she really wanted to effect changes in the code to improve testability and prevent bugs. Abby had learned that if she wanted a specific response, she had to ask a specific question. Her next task was to determine what, exactly, she wanted Bruno, Carla, and Dean to do differently. She made a list of ten specific requests she believed would vastly improve the testability and prevent defects.

Lobby

With evangelical zeal, Abby visited the developers outside the design meetings and made an informal pitch for the changes she wanted to see made. All three said, “Great ideas. Bring them up at the next design meeting.” There was a time when Abby might have interpreted this response as just another way to get her to go away. Now she saw it as an opportunity.

At the next design meeting, Abby listed the specific changes she wanted. This time, the developers listened. By now, they knew Abby well enough to know that she usually had good ideas. In the

QUICK LOOK

- 5 tips for getting people to listen
- Communicating with developers

end, Abby didn't get everything she wanted. But at least she knew they'd heard her.

Show Value

The critical moment came later, when Bruno, Carla, and Dean were presenting an overview of a new interface. Ordinarily, they would not have spent much time discussing the interface. It was relatively straightforward and required little engineering effort to create. However, now that they saw Abby as a team member, the developers decided that it was worth a scheduled meeting to keep her informed.

Abby listened, asked a few questions, then said, "So what part of the system is responsible for user authentication?"

Silence.

The trio looked at each other, then back at the diagram, then at Abby. "Authentication..." Dean mumbled. "Good point!"

Abby smiled inwardly. One major bug avoided!

Being an Effective Influencer

By establishing her relationship and her

credibility with the developers, Abby had made it possible for them to hear her when they most needed to: when her knowledge enabled her to see the system in a way they didn't.

In a short amount of time, the developers had accepted Abby and were treating her as one of them. In turn, Abby realized that Bruno, Cara, and Dean weren't those-darned-developers-in-an-ivory-tower but were instead real people working under the same kind of pressure she was.

Abby became effective at influencing the development process by connecting to her developers on a human level, making specific requests, and contributing substantially to the team effort.

We ran into each other a few weeks after our first conversation.

"Still feel like hitting something?" I asked.

"Well, there's this product manager in marketing..." she sighed.

"You know, he's probably a nice guy once you get past his MBA-in-a-suit appearance." I offered.

"Yeah. I know. I'm savoring my last

few minutes of cursing him as that-clueless-suit-who-writes-useless-memos. I'm meeting him for lunch so we can talk about requirements."

I grinned. Abby had indeed learned to be an effective influencer. **STQE**

Elisabeth Hendrickson is an independent consultant with twelve years of experience working with leading companies in software quality and management. An award-winning author, Elisabeth is a frequent speaker at major conferences on these topics. Read more about her ideas on quality and testing at www.qualitytree.com, or contact her at esh@qualitytree.com.

STQE magazine is produced by STQE Publishing, a division of Software Quality Engineering.