

Customer Requirements and User Requirements: Why the Discrepancies?

Mary Deraitus, Usability Engineer
Ann Miller, Project Manager
Boeing Commercial Airplanes
P.O. Box 3707, MS 7J-90, Seattle, WA 98124 USA
{Mary.B.Deraitus, Ann.E.Miller}@Boeing.com

Development teams often speak about the need to satisfy the customer and to meet their requirements. But exactly who is the customer, how do they determine their requirements, and how do these differ from user requirements? If the customer requirements don't realistically (or objectively) present user requirements, can the developers still build a usable product? And if not, who's to blame when the product meets the requirements of the customer, but not the user?

Introduction

Your IS group is contacted to develop a moderately complex application. You and your colleagues spend the next few weeks meeting with the customer and their Subject Matter Experts (SMEs) to design the product. You build the prototype and hand it off to the usability engineer, who tests it with a number of typical users, only to find out that the users can't perform their work with the system. What went wrong?

Customer and user requirements

The IEEE Standard 830-1998 defines customers as those who pay for the product. In most cases, development teams subscribe to this definition, with the added assumption that the customer represents product users. The customer determines the product requirements, often in meetings with the developers. The developers then build the product based on those requirements.

Customers, in many cases, sincerely believe they are knowledgeable about the needs of product users. They may even make use of some requirements engineering (RE) techniques, such as those that utilize SMEs, to determine the needs of users. But how objective can the customer really be when gathering user requirements? If user needs fall outside of the initial project scope,

expanding both project budget and timeline, is there the temptation to dismiss those needs? Equally important is the possibility that those who conduct the research are not trained in the techniques, allowing for possible negative user and researcher experiences, as well as inaccurate findings.

Assumptions by both customers and development teams about implementing a formal RE process may explain why unusable products continue to be developed. RE seems to be viewed as an add-on, an extra and unnecessary step, even for large systems projects at major companies. Why?

Research on this topic is still in its infancy. Kauppinen, et al, suggest that product developers be may be open to learning RE techniques, if they are simple to learn and easy to implement. However, research by Lucas asserts that "systems professionals often become accustomed to conversing with each other in terms that are unfamiliar to users... due to the existence of a jargon among designers that is based upon their involvement with computer technology... at times this jargon is utilized intentionally to confuse users or to make excuses for project delays." This indicates that developers may not make good candidates for conducting requirement elicitation.

In this presentation, we will go through case study examples of approaches used on IS development projects. We will examine communication disconnects, the importance of being aware of verbal and physical cues, designing questions for interviewing users, customer and developer acceptance of user feedback that causes projects to fall out of scope, and contrasts between formal organizational processes and "the way it really works."