Requirements Elicitation is one of the most important stages of system development lifecycle, where relevant information regarding the functionality and constraints of a system is gathered. This is where users and developers both play a crucial part, all parts of the process with their strengths and weaknesses.

The main asset of users during the requirements elicitation activity is their domain knowledge. Users typically have years of experience dealing with the business processes and problems that the software is supposed to solve. Your expertise in your field is key as you want to ensure the system meets the real needs of the organization and its stakeholders. Also, elements with systems or workflows that are currently in place can give a lay of the land, informing critical feedback on existing structures, or inadequacies (Zowghi & Coulin, 2005). This firsthand insight allows for the creation of software solutions that are practical, user-centered, and aligned with organizational goals.

A fundamental strength of users, despite being one of their major weaknesses, is their inability to express their needs in technical terms. More often than not, users do not have technical knowledge to explain their needs, which results in misunderstandings or incomplete specifications. Moreover, users might not have a complete understanding of the technical limitations or capabilities of the system, leading to requests that are either unfeasible or excessively complex. This gap in understanding may result in ineffective communication with developers, which can lead to misaligned expectations (Kotonya & Sommerville, 1998) or costly rework.

Conversely, developers have some advantages and disadvantages related to the requirements elicitation process. Technical expertise is a major strength of developers because they can translate user needs into actionable system requirements. This can help users better understand

the limitations and possibilities of technology, directing them toward more realistic and attainable solutions. They will also help systems design by identifying risks, this may not be transparent to users.

In fact, requirements elicitation is not easy either for the developers. There are a few things that they do not do well when it comes to Enterprise-level purchases; for instance, they get too technical and do not understand the business context of the user. It can produce solutions that are technically perfect but do not respond to the user's needs or expectations. And at times, developers believe they know enough about the requirements without further engaging in conversation with the user which can cause them to assume details that can set the project up to fail (Zowghi & Coulin, 2005). Time pressure or a failure in the communication channel between developers and users can this problem much worse.

Summary: The users have an intuitive sense of what they need whereas the developers have limited knowledge of the application domain. Users provide deep subject matter expertise but may not articulate their requirements well, while developers bring strong technical skills but may miss the business narrative. Effective collaboration and communication between both parties are crucial to overcoming these challenges and ensuring the successful elicitation of requirements.

Wordcount: 478

References

Kotonya, G., & Sommerville, I. (1998). *Requirements engineering: A good practice guide*. Wiley. https://www.wiley.com/en-us/Requirements+Engineering%3A+A+Good+Practice+Guide-p-9780471974444

Zowghi, D., Coulin, C. (2005). Requirements Elicitation: A Survey of Techniques, Approaches, and Tools. In: Aurum, A., Wohlin, C. (eds) Engineering and Managing Software Requirements. Springer, Berlin, Heidelberg. https://doi.org/10.1007/3-540-28244-0_2