

Example of a good submission - Talk

The following is an example of a good talk submission accepted at SGZA19.

Beyond Tech: Confusion, Conflict, and Lost Productivity

Speaker(s): Riaan Nel

Session Format: Talk (45 mins)

Level: All

Description:

Developers do not exist in isolation - our success is bound to the success of the organizations that employ us. In order to be effective, we need to understand not only the technical aspects of the work we do, but also the business domains and the processes within which we operate. We need to engage with our users, ask the right questions, and be able to make meaningful contributions that go beyond simply writing code. With that said, it's not always easy to communicate with users - we don't always speak the same language and it takes effort and understanding to get everyone on the same page. This goes both ways - while we try to understand their world, we should also seek to help our business users understand our world.

In addition, we need to remain cognizant of the fact that things change - and few things are as demoralizing as having to throw out weeks' worth of work. However, when we develop systems, we learn and refactor as we go along. Isn't it really the same thing when business requirements change?

In this talk, I want to explore how we can understand our business domains, build teams that function well across business and tech, communicate effectively with our business and our users, and deal with the frustrations that arise in the process. I'll also share some of my past experiences and what I've learned from them.

Learning Outcomes:

- Understand the importance of a common understanding across business and technical teams.
- Understand what motivates other people that you work with.
- Understand how to communicate effectively.
- Understand the value of close collaboration.

Information for the Review Team:

The whole idea for this talk was triggered by a meeting with my team where the team was getting frustrated with frequent changes from our business users. Yet, as developers, we learn things as we go along and then we refactor our code. Why can't business users do the same?

This led to thinking about understanding business context and being able to effectively engage with business users, which I think is a critical skill for building a successful career in software development.

I've been an IT professional for over 10 years, and in the process, I've spent a lot of time interacting with users and figuring out business domains. I think I can offer some insight on this.

Introduction (5m)

- Who am I?
- My career in a nutshell.

Stories and past learnings (15m)

- Different languages spoken by techies and non-techies.
- Understanding user needs and communicating with users.
- Learning business domains and the challenges that go with it.
- Agile and collaboration - it's about principles, not practices.
- Lean principles - going both ways.

Tips and resources (15m)

- How to communicate effectively.
- Continuous improvement.
- Mutual respect.
- How to figure out what to learn, and where.
- Books and videos.

Q&A (10m)

- Audience questions - I'd also like to open the floor to discussion and have attendees engage with each other.

Notes to organizers:

None