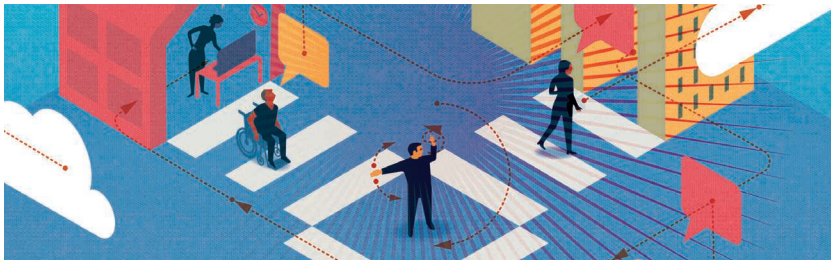
Unscheduled Meetings in Hybrid Work

Tor Sporsem, SINTEF

Audun Fauchald Strand, NAV

Geir Kjetil Hanssen, SINTEF

// By studying four development teams in the Norwegian companies NAV and Entur, we found that unscheduled meetings are maintained by using tools like Discord and Zoom to create virtual rooms. From our findings, we provide three recommendations for hybrid work. //



INFORMAL COMMUNICATION, SUCH as unplanned and ad hoc conversations, is crucial in software development to improve problem-solving.[1] These kinds of communication are called *unscheduled meetings*; they happen spontaneously and are more effective than scheduled meetings for discussing complex problems and making decisions.[2,3] Unscheduled meetings occur more often when developers are colocated and little time has elapsed since the last face-to-face meeting.[2]

As the COVID-19 pandemic is tapering off, people are allowed back to the office. However, many developers are returning to a whole new way of working, heavily influenced by new practices that emerged during the lockdown. Developers are keen to retain the flexibility of working from anywhere, a privilege they gained during the lockdown.

Digital Object Identifier 10.1109/MS.2022.3229554
Date of current version: 13 February 2023

Despite a considerable number of studies on distributed development, very little is known about unscheduled meetings in distributed or hybrid work. And yet, some research indicates that unscheduled meetings are as crucial to developers when they work distributed as when they work colocated (see "Related Research on Unscheduled Meetings in Distributed Work Settings"). With this in mind, we pose the question: How do developers maintain unscheduled meetings in post-lockdown work life? To find answers, we studied four development teams in two Norwegian public companies: NAV, the Norwegian labor and welfare administration, and Entur, a national supplier of digital infrastructure to public transportation. Both companies are modern development organizations with autonomous teams responsible for delivering products to very large customer bases.

This article presents three stories on how the four developer teams experimented to find new ways of ensuring unscheduled meetings when working hybrid. These stories offer valuable insights into how they experimented step by step and what solutions they found. The teams were able to use the technology of audio and video channels in whole new ways. We have known this technology for years, such as Skype, which became mainstream about 15 years ago.

Drawing on the three stories respectively, we have elicited three actionable recommendations for development teams and software companies to consider for moving forward:

1. Experiment with tools like Zoom or Discord to create virtual rooms. They stimulate unscheduled meetings by disclosing what others are doing and revealing whether it is acceptable to interrupt them.

2. Create smaller Slack channels with fewer members to make developers feel safe to ask questions even though they think their questions might seem stupid.
3. Let teams experiment with different ways of solving tasks that demand unscheduled meetings. Teams tend to find sufficient ways when given the freedom to experiment.

Unscheduled Meetings Kept Alive: Three Stories

Studying Entur and NAV allows us to tell three stories from the trenches on how developers learned new ways of maintaining unscheduled meetings during the lockdown and brought them into post-lockdown work life. See "NAV Case Study" and "Entur Case Study."

Story 1: Virtual Rooms Show Developers Whether an Unscheduled Meeting Is Appropriate

Developers started using breakout rooms in Zoom and voice channels

RELATED RESEARCH ON UNSCHEDULED MEETINGS IN DISTRIBUTED WORK SETTINGS

The pandemic has stimulated a growth in studies that provide insight into how distributed work affects IT professionals, including how unscheduled meetings are managed in distributed work.

A study by Stray and Moe (based on data before the pandemic) showed that distributed teams spend more time in unscheduled meetings than scheduled ones. Furthermore, they found that the absence of organizational support for unscheduled meetings is a barrier to effective coordination across sites.[2] Along the same lines, other studies before the pandemic showed the *value* of unscheduled meetings, where discussions in unscheduled meetings lead to more effective decision-making compared to scheduled meetings.[S1] A recent study during the pandemic provides more nuance; Smite et al. found that spontaneous meetings are essential both to socializing and problem-solving, but that developers reported that there were no ''over-the-shoulder'' conversations anymore, for good and evil. On the one hand, people could not shout out for quick answers; on the other hand, they felt less distracted, experiencing uninterrupted flow.[1]

In our previous work, we have identified that using virtual rooms can lower the threshold for quick chats and intra-team discussions, which is needed to maintain effectiveness in distributed teams.[S2] Related to this, Gratton explains how real-time virtual interactions make it possible to do synchronized tasks,[S3] but that this can also introduce constant communications and interruptions that disrupt focus.[S4]

In a global survey to understand the impact of working from home during the pandemic, Nguyen-Duc et al. observed, among other aspects, that more than a third of the respondents found it more challenging to keep an overview of who does what in projects when working remotely[S5], this is valuable information when someone needs to initiate an unscheduled meeting. Related to this, Blanchard found that the need to be seen, e.g., via video solutions, is particularly important within smaller groups.[S6]

Such early but not yet fully systematized findings indicate that people need to compensate for the benefits that physical presence and colocation at the office give.

References

S1. B. Eisenbart, M. Garbuio, D. Mascia, and F. Morandi, "Does scheduling matter? When unscheduled decision making results in more effective meetings," *J. Strategy Manage.*, vol. 9, no. 1, pp. 15–38, Feb. 2016, doi: 10.1108/JSMA-03-2014-0017.

S2. T. Sporsem and N. B. Moe, "Coordination strategies when working from anywhere: A case study of two agile teams," 2022, *arXiv:2204.03978*.

S3. L. Gratton, "Four principles to ensure hybrid work is productive work," *MIT Sloan Manage. Rev.*, vol. 62, no. 2, pp. 11A–16A, 2021.

S4. L. Gratton, "How to do hybrid right," *Harvard Bus. Rev.*, vol. 99, no. 3, pp. 66–74, May/Jun. 2021.

S5. A. Nguyen-Duc et al., "Work-from-home and its implication for project management, resilience and innovation—A global survey on software companies," 2022, *arXiv:2202.04950*.

S6. A. L. Blanchard, "The effects of COVID-19 on virtual working within online groups," *Group Processes Intergroup Relations*, vol. 24, no. 2, pp. 290–296, Mar. 2021, doi: 10.1177/1368430220983446.

in Discord to create virtual rooms that somehow mimicked their offices. Observing each other in different virtual rooms made developers aware of what others were doing and whether it was appropriate for an unscheduled meeting. This differed from our assumption that status indicators in Slack (for example, green for available and red for busy) sufficiently supported developers in interpreting whether it is appropriate to approach each other.

To illustrate, we describe how one team at Entur used Discord (Figure 1). They observed that when they were at the office, they could see which rooms their colleagues were in, making it easy to judge whether or not it was

NAV CASE STUDY

The national welfare administration (NAV) is Norway's largest governmental agency, responsible for distributing one third of the federal budget. NAV has 2000 employees, almost 400 product developers, and 150 product teams organized in 10 product areas. During the last five years NAV has insourced and employed half of the 800 developers, the rest are consultants. The insourcing was an essential strategic measure moving toward a DevOps mindset and going from a few releases to production a year to more than 1500 a week.

On 11 March 2020, every employee was sent to work from home because of COVID-19. NAV didn't have the time to create any central guidelines, meaning that teams had to discover new ways of working remotely themselves while working on delivering our products. Prior to the pandemic, NAV used tools such as Slack, and during the first hectic days of working from home, they started using tools such as Zoom and Mural to cooperate. This allowed NAV to continue working in teams despite going fully remote in two days.

More than 12 months later, the offices reopened, but it was up to the teams to figure out a suitable way to handle this mix of remote and colocated work without any managers setting rules. We studied how this played out in two teams. One was a product team consisting of six developers, one designer, one product owner and one security champion. They were responsible for delivering "my pages" for citizens using NAV's services. The other was a platform team with 12 developers that offers an application platform to all product teams at NAV.

ENTUR CASE STUDY

Entur is a public company that provides a digital infrastructure to the Norwegian public transport system. For example, they provide an app where travelers can find buses, trams, trains, subways, ferries, scooters, and city bikes and plan travel across transportation modes and providers. They also provide components like payment solutions to transport companies.

Entur has more than 100 developers organized into 20 development teams, and each team is responsible for its part of the digital infrastructure. The two teams in this study are described as autonomous, meaning they choose freely how they solve their tasks and what methods they use. The teams included front-end and back-end developers, designers, and product owners. One team was responsible for the app used by travelers, while the other gathered data from travel companies and structured them into products that other teams use to build features.

Prior to COVID-19, the teams used tools such as Slack, Jira, and Confluence. During the lockdown, teams were free to explore new ways of working that fitted themselves. Tips and tricks soon spread among teams as they tried different tools and practices. When offices reopened after approximately one year, managers trusted their teams to, again, find suitable ways of working. Some teams are even hiring developers from rural areas in Norway, unlocking much-needed competencies.

acceptable to interrupt them for an unscheduled meeting. To gain a similar effect when working from home, they recreated the physical rooms from the office in virtual rooms, organized by voice channels. They created a "Team-room" that mimicked their shared space at the office. A "One on one" room resembled meeting rooms where developers could retreat for private discussions. "Do not disturb" was like a quiet room. The idea came up when teams were forced to work from home at the start of the pandemic, and one team member who was also an experienced gamer proposed using voice channels to create virtual rooms in Discord to allow quick clarifications and short exchanges of information the same way online gamers do.

These rooms enabled them to observe each other's presence in different rooms; this helped them to "see" their coworkers' state of mind. "I can see, for example, that Maria and Peter are sitting in another room and having a meeting. […] you know where they are [mentally]," said one developer. Awareness of what others were doing helped developers to interpret whether it was appropriate to interrupt their team members. "Discord matches how we work when we sit near each other in the office. We can get quick clarifications like 'can you have a quick look at this? Looks OK?'" one developer said. Having a feeling that a person can be approached lowered the threshold for contacting them and helped developers progress in their tasks. All informants in NAV told almost the exact same story about how they used Zoom and breakout rooms.

So, why were status indicators in Slack not enough to reveal whether it was appropriate to start an unscheduled meeting? The reason was a mistrust of the status indicators in Slack. Unclear statuses make it hard to know for sure when you can approach coworkers. "You don't know if you are interrupting people when you contact them on Slack. […] you have no idea what they are doing. […] I don't update it [my status indicator] much myself. Based on how I use it myself, I do not fully trust it," said one developer. "Yellow or orange or red… I don't dare trust them," said another. As we have seen, such challenges can be mitigated by using virtual rooms.

Implementing virtual rooms required experimentation. "In the beginning, everyone had their microphone unmuted to make it feel like you were in the office, but at home, you also have other sounds that come from the kitchen or children or cats and stuff, so it did not work well," said the team's designer. Experimentation led to a practice where all team members kept their loudspeakers/headsets on and microphones muted when not speaking. That way, they could unmute and ask questions or address someone while everyone heard it. When asked if this was annoying for others in the same virtual room, all informants said that the practice enabled transparency and offered opportunities to include themselves. "If you do not like it, you can always turn off your sound; it will be like putting on headphones in the office," the designer continued. "I thought it might be a little tiring, but it's not. People are very respectful and do not bother each other," another developer said.

An interesting point was that teams found virtual rooms redundant when they came to the office because they physically observed each other's mental presence. Those few who worked virtually on such days stopped relying on the virtual rooms to determine teammates' mental presence. Teams solved this issue in two ways: by introducing common office days, so all members came to the office on the same days, or by making it mandatory

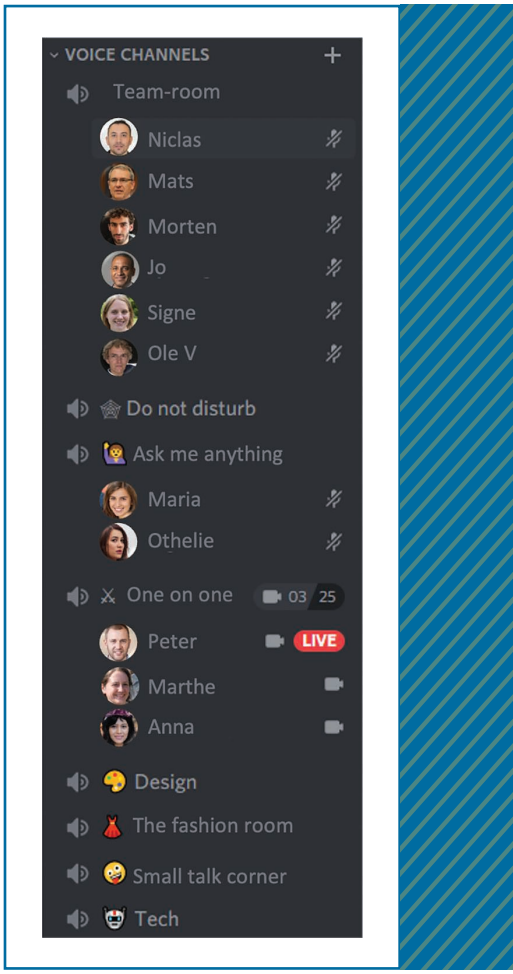


FIGURE 1. The virtual rooms and their participants (pictures are generated by an AI for anonymity) are shown. Six members are present in the "Team-room," all muted but with their speakers on, simulating their shared team space at the office. No one is present in "Do not disturb." While two are present in "Ask me anything," they are also muted. Three members have a live discussion in "One on one" with their cameras on. The other rooms, "Design," "The fashion room," "Small talk corner," and "Tech," are empty.

to use Discord/Zoom for those in the office as if they were at home. Both strategies were used by teams experimenting with different setups and reflecting upon them in retrospectives.

Virtual rooms have proven valuable within teams (intrateam), helping developers to cross the threshold of contacting team members for an unscheduled meeting. However, this effect is lacking across teams (interteam) because teams do not participate in each other's shared virtual spaces. One team in NAV has started experimenting with opening up to outsiders by sharing their Zoom-link openly. That way, outsiders can visit their Zoom rooms, observe, and even enter the breakout rooms—mimicking office life where one could roam freely.

> As a countermeasure, one team at NAV, who always worked in pairs, started pairing up those who stayed remote with those who frequently came to the office.

We find it interesting that developers can still find new ways of adapting existing technologies, such as Discord and Zoom, to new ways of working when they are allowed to experiment freely. In this story, they found new ways of maintaining unscheduled meetings.

Story 2: Developers Feel Safer Asking Questions in Smaller Slack Channels

What was earlier, before the pandemic, a short unscheduled meeting to ask a question now became a post in a Slack channel. A significant difference is that most Slack channels are open and often have numerous members—features that prove challenging for some.

Some developers perceived large Slack channels as a risky medium for asking questions. "It feels like imposter syndrome—if I ask questions in large channels, it feels like I will reveal my incompetence," an experienced but young developer told us. Several developers said this makes them avoid asking specific questions. They usually try to find other ways of asking them, i.e., in direct messages (DMs) to someone they know or in smaller channels. "It is easier to ask a question in a smaller channel because you don't expose your weaknesses to *that* many," says the same developer.

Moreover, several developers felt there is a social cost to asking questions in Slack as the questions are saved in logs and may create doubts about developers' future performance. "There is a threshold to formulating a message, [...] the stupid question stays there forever, but in a physical conversation, the only one who heard how stupid you were is the one you were talking to," one young developer told us. Newcomers were concerned about how they express themselves; if their formulation is perceived as naive, newcomers fear that this may have a social cost as it reveals their weakness, ignorance, and a general lack of ability.

So, why do all these large Slack channels exist? One of the teams we investigated called themselves a *platform team*, building and maintaining an application platform used by other developer teams. To handle communications from NAV's approximately 150 teams, they created a Slack channel with more than 700 members where developers could post questions and requests and discuss issues. This way, everyone could see what questions were asked and what answers were given, relieving the platform team from answering the same question several times. However, this created a steady flow of individual DM's to the team's members, creating an unmanageable flow of information.

Smaller slack channels led to more unscheduled meetings. Smaller Slack channels were found to be the solution to this problem. The platform team started creating small channels dedicated to teams often reaching out in direct messages. "In such channels, we are able to conduct more in-depth discussions," a platform team member said. Developers outside the platform team praised these small Slack channels, describing them as a way of feeling safe to ask whatever they need without worrying about being judged by someone they do not know.

Further, the platform team experienced discussions that started in a small Slack channel often moved on to a Zoom call—creating an unscheduled meeting. This usually happened if the discussion required interpretation or screen sharing. "If it's more convenient, we move the discussion out of Slack and into a Zoom call," said one platform team member. They consider this a favorable way of

retaining the unscheduled meetings with developers outside their team.

Other researchers have found that less experienced developers and newcomers communicate less frequently than experienced people.[4] Our findings indicate that small Slack channels might create a sense of security that encourages juniors to communicate more.

Story 3: Two Divergent Strategies Emerge

When we first started studying the four teams, we hypothesized that complex tasks would fit better for colocated office time and simple tasks would fit the home office. However, we found early on that developers were able to do quite complex tasks when working from home. On the other hand, one team revealed that tasks requiring frequent decision-making and discussions in unscheduled meetings had piled up in their backlog during the lockdown's work-from-home period. These tasks usually had an interpretive element that demanded frequent clarifications and discussions. We discarded our early hypothesis and made a new one: Tasks requiring unscheduled meetings are preferably done when colocated in the office. As we report next, this hypothesis also needed some adjusting as time progressed and teams continued experimenting.

Common days in the office versus "virtual first". Two of the four teams decided on a policy where everyone would come to the office together two days per week. Developers blocked their calendars to ensure that the colocated time was used for unscheduled meetings, not accepting meetings such as stand-ups that merely shared information or reported status. Such meetings were just as well suited for remote work and planned on days when the team worked from different locations. "If I come to the office and sit in Zoom meetings all day, then something is wrong," said one designer.

However, this strategy was not without challenges. "On those days we were at the office, the other teams weren't," one developer said, showing us that maintaining unscheduled meetings across teams was still an issue. Our informants speculated on various reasons: It is more comfortable in the office when there are fewer colleagues to share the space with; the best meeting rooms are available; time is precious for the teams to meet internally and build cohesion.

In contrast, the two other teams decided on a "virtual first" policy, allowing team members to work from anywhere and leaving it up to each individual to determine whether they wanted to go to their office. Those who went to the office still had to use Zoom and breakout rooms as if they were remote. Of course, this was easier said than done. Those colocated naturally used the physical rooms for unscheduled meetings, thus distancing themselves from the virtual rooms. As a countermeasure, one team at NAV, who always worked in pairs, started pairing up those who stayed remote with those who frequently came to the office. This strengthened the link between colocated and remote enough to maintain unscheduled meetings within their team.

The "virtual first" teams found it challenging to obtain unscheduled meetings with other teams. As described in story one, their approach was to open up their virtual rooms for outsiders by publicly posting an invite link and effectively mimicking how someone could visit them in their shared prepandemic office space. "Sometimes someone [outside the team] just appears in our Zoom room to see who's available to chat when they need something," said one developer on the platform team. This practice is still emerging as the idea of visiting other teams' virtual rooms is still a novel one.

This story shows that different strategies work for different teams in maintaining unscheduled meetings. It also shows the importance of unscheduled meetings when conducting tasks requiring frequent clarifications, decisions, and discussions.

Facilitating Experimentation

From the stories, we derived the three actionable recommendations listed in the introduction: 1) test virtual rooms, 2) create small Slack channels, and 3) let teams experiment to find new ways of maintaining unscheduled meetings. These solutions may only work for some; you need to test these ideas in your own context and build your own experience. The question then becomes: What enabled the four teams to succeed with this experimentation?

Tweaking Tools: The teams we followed were experienced software developers and highly competent users of collaborative technologies and had been so for years. Hence, they had the skills needed to quickly exploit advanced features of tools such as Zoom and Slack. Senior developers were able to tweak Zoom, literally in seconds, to enable advanced features and improve remote work.

Experimentation Culture: They were also highly experienced with agile processes and team autonomy—experimentation seemed to be in their DNA. They dealt with the challenge of being forced to work from home the same way they tackled any other technology-related problem; ideas were created, tested, and evaluated quickly.

Sharing Knowledge: Another contributing factor to experimentation

STUDY METHOD

As we were addressing an abrupt phenomenon, we used an open and qualitative approach to understand the change toward fully remote work and, over time, how a mix of remote and co-located work became the new normal.

Data collection: Our primary data source was interviews with agile team members, such as developers. In addition, we observed a full day at one of Entur's teams. Eleven interviews at NAV and eight interviews at Entur were recorded and transcribed. The interviews were all based on the same semistructured interview guide and started when workers were allowed back to the office in November 2021 and lasted until April 2022. In addition, the teams indicated that our findings would soon be outdated as the developers kept experimenting rapidly. In response, we conducted three interviews in NAV in September 2022 to verify that our findings were still valid.

Data analysis: The nineteen interviews were recorded, transcribed, and coded (constant comparison method[6]) in NVivo™ to identify and structure common codes into larger constructs, describing the balance of colocated and remote work. Findings were presented back to the four teams in May of 2022 to eliminate any mistakes.

This study's preliminary findings were published in a research article in June 2022[6] which contains a more detailed description of the research method.

ABOUT THE AUTHORS

TOR SPORSEM is a research scientist at SINTEF, 7012 Trondheim, Norway. His research interests include internal software startups, coordination in large-scale agile development, continuous software engineering, and tacit knowledge. Sporsem received his M.Sc. in project management from the Norwegian University of Science and Technology. Contact him at tor.sporsem@sintef.no.

AUDUN FAUCHALD STRAND is a principal engineer at NAV, 0661 Oslo, Norway. His research interests include increasing development speed and making systems more resilient. Strand received his M.Sc. in information systems from the University of Oslo. Contact him at audun.fauchald.strand@nav.no.

GEIR KJETIL HANSSEN is a senior research scientist and research leader at SINTEF, 7012 Trondheim, Norway. His research interests include agile software development methods in general and agile development within critical domains in particular. Hanssen received his Ph.D. from the Norwegian University of Science and Technology. Contact him at geir.k.hanssen@sintef.no.

was the strong learning culture at NAV and Entur that is based on openness. These teams were already good at sharing knowledge, and we observed how good results and experience with new ways of doing unscheduled meetings in one team spread to others. Dedicated Slack channels were also created related to remote work practices, enabling teams to share experiences.

In conclusion, our study tells the overall story of four teams that tackled a dramatic change to fully remote work when society locked down, which they learned from to establish new norms of unscheduled meetings as a permanent improvement for working hybrid.

Finally, our findings seem to cast doubt on the established truth that physical distance between people's offices results in an exponential drop in the frequency of communication;[5] that those within a 25-m range are likely to communicate at least once per week, while those beyond 25 m are not likely to communicate at all. Developers in the four teams we studied have demonstrated that experimenting

with new ways of using existing technologies kept unscheduled meetings alive, regardless of distance. Please see "Study Method" where we explain how we collected and analyzed the data for the case studies.

Acknowledgment

We thank the four teams in NAV and Entur for their warm welcomes and openness during interviews and visits. Also, we are grateful to Knowit and The Norwegian Research Council for funding this research (Project 321477).

References

1. D. Šmite, N. B. Moe, E. Klotins, and J. Gonzalez-Huerta, "From forced Working-From-Home to voluntary working-from-anywhere: Two revolutions in telework," *J. Syst. Softw.*, vol. 195, Jan. 2023, Art. no. 111509, doi: 10.1016/j.jss.2022.111509.
2. V. Stray and N. B. Moe, "Understanding coordination in global software engineering: A mixed-methods study on the use of meetings and Slack," *J. Syst. Softw.*, vol. 170, Dec. 2020, Art. no. 110717, doi: 10.1016/j.jss.2020.110717.
3. F. Calefato, D. Damian, and F. Lanubile, "Computer-mediated communication to support distributed requirements elicitations and negotiations tasks," *Empirical Softw. Eng.*, vol. 17, no. 6, pp. 640–674, Dec. 2012, doi: 10.1007/s10664-011-9179-3.
4. V. Stray, N. B. Moe, and M. Noroozi, "Slack me if you can! Using enterprise social networking tools in virtual agile teams," in *Proc. ACM/IEEE 14th Int. Conf. Global Softw. Eng. (ICGSE)*, 2019, pp. 111–121, doi: 10.1109/ICGSE.2019.00031.
5. T. J. Allen, *Managing the Flow of Technology: Technology Transfer and the Dissemination of Technological Information within the R&D Organization*, vol. 1. Cambridge, MA, USA: MIT Press, 1984.
6. C. B. Seaman, "Qualitative methods in empirical studies of software engineering," *IEEE Trans. Softw. Eng.*, vol. 25, no. 4, pp. 557–572, Jul./Aug. 1999, doi: 10.1109/32.799955.

Digital Object Identifier 10.1109/MS.2023.3236829