Researcher: In this research, in collaboration between <CONFIDENTIAL-NAME> and <CONFIDENTIAL-NAME>, we are trying to understand the differences that exist in whiteboard design/architecture meetings that are entirely done by novices, by teams of mixed experience, and entirely by experienced architects.

Researcher: We are interviewing a range of designers and architects, in order to uncover what those differences might be.

Researcher: Thank you for being a willing participant in this regard. By and large, all we want to really do is have a nice conversation with you about this topic. We do not have a set of questions that we must ask you; we are more than happy to let you lead the way, highlight your thoughts and feelings, and so on. Of course we will ask for detail, clarification, particular areas of interest, etc. But let's do this nice and informally.

Researcher: To set the stage a bit, can you tell us your own level of familiarity with whiteboard design (e.g., how often do you participate in such meetings, lead them, witness them happening, …)?

Architect: What do you mean by whiteboard design meetings?

Researcher: For example, you go with other architects or the development team for the whiteboard to discuss what would be the architecture before implementing it, did you understand? Or before discussing it in the architectural document or implementation.

Architect: Got it. In this case, I have been working with architecture since the time of the Rescuer project and that was the first opportunity and after the training of <CONFIDENTIAL-NAME> also in their method of architecture development. The meetings in this first context, I met with an architect from there. Basically in this project, we divided the activities of elicitation, architecture specification and architectural decisions. I joined the project in the middle and the initial decisions had already been taken and I participated more in the evolution of the proposal. As the meetings were online, so it was more a distributed specification. In this case, the concept of whiteboarding was more of a sharing. We made the sketches of drawings where the format of the sketches were completely free, in this case, the architect left me free to express the kind of decision I wanted, and there was no notation or something. Then, I made the drawings of how the flow of information would be among some components of the architecture and I sent to him to see if he understood what I drew and we scheduled a meeting to discuss based on that drawing and to him map the drawings into architectural models according to his method. He did the same there too. He made the sketches first and brought them to the meeting, and these sketches served as a basis. We discussed based on the drawing, the critical points that would be decided. We ended up sharing the work. I was dealing with some aspects and he was with others. We always did the extra work meeting and came back to the meeting to consolidate the information.

In the other two projects, as I was the technical leader, I took the role of architect too. I was responsible for the decisions with the developers. We did not have an architecture team. I was the responsible architect. In this case, discussions in one of the projects did not concentrate much around whiteboards, we used an architectural

description approach based on non-functional requirements and quality attributes. We knew the quality attributes to be met, we specified how they should be met, and the discussions involved scenarios. But in this case, we wanted a more direct activity, so the discussions were scenario-based. I was responsible for writing the scenarios and from those scenarios we would verify the best alternatives in terms of decisions to be made. I was also responsible for identifying alternatives to each quality attribute scenario. I presented these alternatives and put my point of view to discuss with the team and they brought their point of view. The decision process was quite argumentative. Although I was the lead of the project, there was no determination on my part. The idea is that the team should bring arguments to accept or refute the proposals given. Thus, we have been able to design the architecture for the second project. The third is still in the beginning and the complexity of the project does not require a very heavy architectural description, it only involves some decisions, but the process itself is very similar. We defined quality attributes that the customer considered most important and based on these quality attributes, we identified the best technology to be used in the project. Because we were hired for a very specific context and did not have much of architecture to be done. In this project too, we used the concept of whiteboard and sketches and it was more related to elaborate the quality attributes and we did the evaluation through a pilot project to identify which technology would fit best within those quality attributes. So the process of experimenting or checking the best architectural decisions was based on the pilot experience. So it was a longer project than the other two that were more related to design and use scenarios. In this project, we knew that the way to define the decisions would be through a pilot. So these were basically the three different scenarios, we do not have a specific method for this, since each project has a different need. We adapt the approach according to the need of the project. So, we tried to adopt the method that we were trained by Fraunhofer IESE, the ACES, which they have already published and it has been a very interesting approach to use since it is focused on quality attributes. Despite the different executions, we always have this core of concern to develop the architecture.

Researcher: Regarding time, you said that there were three projects. When these projects wered developed Can you tell me quickly?

Architect: I do not remember dates. But the Rescuer I stayed two years in the project and it lasted 3 years and a half, 4 years I would say. And I started there by 2014, more or less. The other, Living Lab project started in late 2016 and the development part ended in late 2017. So the project lasted 1 year. And this one from Intelbras started now, it should be about 2-3 months.

Researcher: Can you give me an example or two of such recent meetings in which you engaged? What was the setting, how many people were there, what problem were you trying to address, how long was the meeting, …

Architect: In the first scenario were meetings around an hour basically. In terms of problems, the approach used in the project divided the specification into different perspectives, such as data, components, etc., and sometimes, the meeting was related to some specific problem in some of these perspectives. Generally, we identified the existing problems in the architectural specification and worked on that problem. So we did the sketches to see the best solution. In terms of problems exactly, I cannot be precise because it has been a long time, but they were problems for managing the state of the application to maintain consistency of application states with the various components of the system. In general, it had two people: I and the

architect of <CONFIDENTIAL-NAME>. In the others, it is difficult to say because we did not have a meeting to say that it was related to architecture validation. As we had a more agile approach in these other two projects with a more limited scope, the context of the first project was distributed, a very large team, etc., we needed to have a greater control of the activities. These other two, as the team was smaller, 8 people and uses a more agile approach to development, the meetings were basically face to face and sometimes we had a meeting with everybody to discuss the general aspects of the architecture. However, these meetings happened more when I had to present the architecture proposal to be validated and if there was nothing else to be discussed. So we had some meetings with more people. In general, I ended up discussing with 1-2 experienced developers and making the decisions more informally. So the meeting time, I cannot estimate. These were 20-30 minute meetings depending on the content of the conversation, usually with only one person, face to face, and we solved some things. And the other way was the meeting with the whole team and it was more a review meeting of a sprint and the people ended up giving their feedback in relation to what was specified. The problems were related to the design and were focused on the quality attribute itself, whether it was a performance problem or interoperability. It varies a lot from architecture to architecture, and what is a priority in the project.

Researcher: What was the size of the first project which you said that it had a large team?

Architect: The project involved more than five institutions and had 20-25 people considering the whole project. But it changed a lot the people and I cannot be precise.

Researcher: Given this, can you tell us your perceptions of these kinds of meetings and what you think differs when they involve participants of different levels of experience (e.g., more or less all novices, some mix, or mostly or all experienced)?

Architect: It is a really tricky aspect because sometimes when you have a discussion with developers, even more junior or senior developers, sometimes the contributions to architecture itself are minor, and sometimes the discussions are unproductive, most of the time. The task of the architect is not to discard any ideas from the developers, any kind of feedback. However, dealing with less experienced developers or even with experienced developers with a lower maturity level is complicated and you have to exercise your argument much more as architect to try to convince that guy that your alternative is a reasonable alternative to the others. And sometimes because of the lack of maturity, the person does not want to give up the alternative proposal that he has been advocating and ends up prolonging the meeting and making the discussions move to things that are not productive, so it is shocking. I think this has a big impact when you have a more unbalanced team. On the other hand, when you have discussions with more experienced developers, even the discussions are longer because more experienced developers like detail and seek to better understand the alternative and how it can solve the problem, no matter how long that you spend on this, the content of the discussion for a decision-making is much more productive. So you end up having a better result in terms of the final result for the architecture.

Researcher: In these meetings, What stands out to you? Why? Can you give examples?

Architect: For me most is the inflexibility of less experienced software engineers. It is amazing how you cannot have a more productive discussion simply because the owner of the idea is not able to let it go for the architecture of the project itself. This ends up generating a wear of time and on the team. Because at a certain moment, you end up having to impose something, because otherwise you would reach the impasse. So this inflexibility is complicated. Amazingly, I have not found this in experienced developers. As much as they defend their ideas, they are willing to discuss and understand each other's roles on the team. Although we are in a team with agile practices, we do not follow at all a specific agile method. So there are specific assignments for each one on the team and the experts understand that. They know that in my position as an architect / technical leader, they go against you to some extent. However, in some point they leave the discussion or else myself in my role of architect I take, when I see that my argument is not so strong for a decision, I take this up for the project manager to help in the decision and to mediate this situation a little so that this is solved in the best possible way. So the issue of inflexibility for decisions is complicated, because each one has in his head the best solution and leaving it is a bit complicated.

Researcher: In your thinking, is it always better to have only experienced architects there? Why or why not? What about the quality of the end product of the meeting in this regard?

Architect: It is a tricky question. I think that it does not matter if you have experienced architects on the meetings but these experienced architects are not up to date with the development trends, new technologies, etc., to improve the project. What I mean is that sometimes you have a guy who is a good architect, with a good line of argument, who can discuss very well, but for a particular project he does not have much technological background of trends for contributing to the discussion. And perhaps, what he would spend with research in that field, etc., does not replace the experience of a guy who is less experienced than this architect that I am talking about, but this less experienced had opportunity to work with technologies related to the project. And this guy can contribute a lot more even he being inflexible, etc. So if you can manage this scenario, he can contribute a lot more than the experienced architect. So the architectural meetings within a project are much more tied to the technical capacity of the team to be able to identify the best decisions for that specific project, than the fact that an architect is more experienced or not. Of course, in the case of CONFIDENTIAL COMPANY NAME, as we are dealing with innovative projects, it is very difficult for you to have a team that already has the profile that that project needs. Generally, we must along the way empower ourselves for that project because each one is different from the other. So in this scenario where you have developers and architects who do not have the technical knowledge of the project context, then this lack of knowledge makes the developer experience more important. But in terms of priority would be the technical knowledge that the project needs and then the level of experience and maturity of the developer. I think that maybe the word maturity is even more important because sometimes you have a developer/experienced architect but they do not have enough maturity to behave well in a team.

Researcher: If you could provide advice for other designers/architects when they are designing with you, what would it be? If you could provide advice more generally, what would it be?

Architect: Regarding to advice, I said before. It is more the question of you always trying to look for alternatives that can support a certain architectural decision that you are making, the search for alternatives is always very important. On the other hand, it is the role of any member of any software engineer on the team, that you defend your proposal, but do not be hostage to it. You can discuss and understand the arguments, defend your proposal and also have the discernment, although difficult, that if another alternative presented is better than yours, you can also understand that it is the one that should be chosen. We know that in architectural decisions when you present alternatives, there are always pros/cons of any of them. So knowing which is the best trade-off to be chosen is very important. So it is the task of the developer/architect to recognize also that when investigating a particular solution, not only seek the best aspects of that approach, but also those against, the deficiencies, since this is what often leads the project technically to failure. So the advice would be precisely to seek the solution in these two points of view and be always open to what the team has to say and is open to consensus as well.

Researcher: So, these are your more general recommendations?

Architect: Yes. And in addition, as in our context at CONFIDENTIAL COMPANY NAME, we always work with new projects and different from previous ones, the training to search for new technologies is also very important. It is very difficult for you to work in a project center and expect the next project be similar to the previous one. So you will be always looking for new information, new knowledge to make the best decisions. And sometimes this kind of empowerment is complicated. So, generally, we try to do some training, take some course and also if possible, if the project has schedule for this, try to develop a pilot project to experiment with the new technologies in practice. It gives a very important input to the guy who has never had contact with the technology that will be used. This is also another important aspect in the context of companies working with projects in different types of technical solutions.

Researcher: Having contemplated the above, how would you describe experience now (“what is experience”)?

Architect: Experience for an architect is the guy who has done the architecture of some system before. In my own case, despite knowing the concepts of architecture, etc, I had the first experience before <CONFIDENTIAL-NAME> training in the first project. So I had an experienced architect to give me assistance and tell me what should be done in terms of architectural activities in practice. So, with that, I acquired this experience in the project so that I can, in the next, conduct the architectural activity in the best possible way. So experience first involves experience in previous projects with architecture itself. The second point is that although you are not an expert in the various technologies which system development requires today, it is very important that you be aware of these trends, know that they exist and what works, and the best contexts they can apply. So having knowledge of the technical aspects to different paradigms, whether it is a web system or real time, etc., it does not matter, you are always knowing what it is that can be used. So that is a second point. The third point that involves experience is the ability to communicate with the team. Sometimes, communication problems are the worst that can happen in a project. So you have to know how to criticize well, you have to know how to make critic observations. There are technical people who do not criticize the team in a way that they can evolve. So knowing how to communicate well with the team is interesting. The question also of interaction between the technical side represented

by the architect and the management of the project. So the architect needs to always communicate with the project manager to report as early as possible the possible risks, which can cause the project to be hampered in the future or not, and define mitigation actions. So the communication has to be always direct and transparent with the project manager. Thus, the architect can be aided in risk management since sometimes the architect's decisions need to be supported by the manager because the manager who will need to solve something in order to make that possible. So this interaction with the manager would be another point. Basically that. These four dimensions, I think, bring a good experience for the architect. The second part I do not remember anymore.

Researcher: What are you looking for in an experienced architect when you bring them into a meeting?

Architect: The idea, if I put an experienced architect in the meeting, the first thing beyond these dimensions that I said, is the critical look. Because sometimes when we design an architecture and put it in discussion with other architects, we hope that other architects be able to see faults and architectural problems that we cannot see with a limited vision and for this you require a technically well-prepared architect to be able to see the proposed solution and try to identify in that solution something that will generate a problem in the future. So it requires a very prepared guy with enough experience from other projects because he can bring this to the reflection of the current project and a guy who knows how to communicate in a cordial way. It is useless to be experienced but have a very problematic personal and emotional personality because it can end up bringing problems for the project. Basically this.

Researcher: Do they live up to that expectation? What happens when they do? What happens when they do not?

Architect: Meets the expectations considering the first project. The responsible architect always brought interesting solutions to the problems found. Things that I did not have maturity in that time to identify or model. So he usually met the expectations very well. When these situations occurred it was basically some architectural problem of some quality attribute problem being solved and I could not find a reasonable solution to it and the architect brought a solution that fit into the context and you simply judged that it was good and approved. So it usually does not have a different path to it. However, when it does not meet the expectations of the architect, you have to know how to handle the possible decision-making if you are responsible. You have to evaluate his proposal, his participation, present arguments that not only him, but the whole team can see that it is not the best solution and in case there is some friction or it does not occur, you have to find some way to discard that decision in order to adopt another one. So when it does not meet expectation, that's it. Mediating this decision-making is an important aspect since it is a point that can generate friction later.

Researcher: To help us calibrate, how would you rate your own level of experience? Can you provide an estimate of how many whiteboard design/architecture meetings have you been part of over the past 5 years?

Architect: Estimating it is very difficult. Considering that I will put a meeting every 15 days, it would be 24 every year. Around 120-130 meetings both distance and face to face.

Researcher: How would you rate your own level of experience?

Architect: My experience level is intermediate. Generally between junior/senior/ full, I would be a senior beginner. I am in the third year with architecture activity experience itself, so I would put myself as senior.

Researcher: Any final thoughts? Fell free to comment.

Architect: Sometimes informal meetings are the best, rather than the meetings where you put everyone in a room. The responsible people in a room take great care of what they are going to say, sometimes they do not speak, especially the inexperienced, they are afraid to speak something with fear of criticism. There is usually some fear. And sometimes the most productive meetings are the ones you sit on the guy's side and start discussing. So an architect who is responsible for a project can have much more productivity in the meetings if he can speak individually, most of the time, and periodically consolidate this information with the team whenever necessary. It is good that he has previously been able to collect the ideas of all to be able to elaborate a proposal that is good for the project and that has considered the point of view of the majority of people even because thus the architecture produced is not a surprise for anyone, you already anticipate decisions. So the value of small meetings is always very important.

Researcher: One more time, thank you for your time.