Researcher: Hello. In this research, we are trying to understand Turnover Turnover of software engineers, that is, for what reasons professionals leave their jobs.

Researcher: So, we are interviewing several professionals to better understand this aspect. Thank you for agreeing to participate in this study. Our goal is to have a good conversation with you.

Researcher: In your previous experiences, did you receive any training when you joined the company? If so, what types of training were received and how were they applied? Do you believe this aspect contributes to the turnover of a software developer? Why?

P02: Well, all the companies I worked for, I never had training. Are you talking about training with tool sheets, the environment, or training more about the company's culture?

Researcher: Tools.

P02: Tools, no. Normally the companies I come to work for already have software already in production, there are demands to be met within a certain deadline, and what happens is that the business is presented, and then we, as analysts, have to see the particularities and levels of difficulties and develop from there, right. I hardly remember working in a place where I had training. Sometimes it is, the company, it makes it available in the environment, such as alura, udemy, for you, of your own free will, to get there, and take the course, but nothing like that, the company: I will prepare you here for a certain tool, or else one of the things I feel, right, when migrating from my last company to my current one. I wanted to work with AWS, right? I studied AWS, I did some things, but I never worked with AWS. But at the same time the companies wanted the guy to know... AWS is a world, you know, companies did interviews and didn't know exactly what they wanted, I said I knew this, that, but at the same time they didn't know that , kind of ruled me out from the start because I didn't have long-term experience with AWS, and that kind of, maybe if the company arrived, even prepared the professional for that need, it would be interesting, but it's not the set. I don't really know the company that does this.

Researcher: Do you believe that obsolescence is a factor that contributes to software developer turnover? Why?

P02: Obsolescence is obsolete software, legacy systems, which are you referring to?

Researcher: It has a software point of view, and also a professional one. It's you becoming outdated.

P02: Yes, it was one of the reasons I left Salvador at a certain point, but there were other issues too. But it is one of the things that is crucial today. Because I realize that the more obsolete the system, the more difficult it is to find the professional, the higher the salary, but at the same time you remain stagnant in the market and it doesn't become as challenging, and it really is one of the main factors that makes me leave of a company, it's the question of, they give you legacy software, things that you won't be kind of, yeah, linked to what's happening in the market, even though it pays well sometimes, depending on, it varies a lot, but, for the long term It's not worth it, and I rarely stay in companies where I feel obsolete, that's for sure.

Researcher: In your opinion, do you believe that incentives offered by the organization

contribute to the professional's retention? Why?

P02: Look, it depends on the incentive. Here in São Paulo, I work with PJ, right, for me it is more advantageous than working as a CLT. CLT, most companies provide N benefits, right, daycare, several other things that I won't use, you know? But one of the things that I notice that actually encourages the developer to stay at the corporation is that he feels part of the corporation as the owner of the product, of what is being worked on, if he feels like just a tool, like something: beauty, I will allocate you to resolve this BO here. It just stays there, it wears out, right? And another thing I miss is the issue of companies, I haven't worked in any type of company like that yet, the closest I was in Salvador, where I worked at SSP, you know, it was practically me, and another colleague, so we had immense freedom to create solutions. It takes time for you to have an idea, for you to see that the company can, yes, create a product, something new. You have an insider there, and that thing, like: for the company itself, it's worth nothing, you know? In the current company I am in, I see several scenarios that could be applied, increasing the value of the product, from what it is today. But, from the team upwards, from the management side, people don't have this vision, so it's very complicated. This gradually starts to demotivate you, because you end up in that circle of, ha, you start the week with a

story. Then I always question, what is this story I'm doing here going to add? Will it have any positive impact, beyond what I'm doing here? Or is it going to be something that's just an adjustment that, you know? Nobody knows who I am outside of that box, you know? And that to me is very relevant. It's what motivates, in fact, it's what encourages in my view. Develop activities, things that make you feel part of the product, of the company, right. Yes, you are also heard within the company, this is very important. There's no point in saying: look, I have my notebook here, it only has eight gigs, I can't work, and the company, okay, we'll find a way. And the company doesn't pull any punches to provide you with a quality notebook, or something like that. These are things, you know? It's not even a matter of salary itself, the company arrives and I'm going to give you a ten percent raise, but the problems will continue, you know? And then, sometimes, it's not such a relevant part. The last company I left, one of the things that wore me down was: I had already been there for three years, you know, and I was in a kind of system, supporting the system that I already considered obsolete, right, and I went to a project with newer technology, yes, cutting-edge things anyway, you know? But, at the same time, I saw that the problems with the company's mentality in the old system were maintained in the current system, instead of the development of the work process being something stress-free, it ended up being something much more stressful, than when I worked with a more stressful system. , older architecture, understand? So, there are N factors, right? I don't know if I said too much, but.

Researcher: Did you have a good relationship and communication with other employees in your company? Do you believe that a good relationship with other employees contributes to the professional's retention? Why?

P02: Yes, definitely. One of my biggest fears when changing companies is the relationship issue. Yeah, when I came to São Paulo, right? Leaving SSP and coming straight to São Paulo, I worked in a company that I felt like every day. I only spent three months there, I don't even put it in my work history, because it was one of the worst experiences, right? So, because, firstly, it was a company that positioned itself as, no, we are a pluralist company, we really like diversity. It was a startup, right? This that. But the staff was boring, in the sense of hypocrisy, in a behavioral way. Firstly, it was the first company I left Salvador for and I already had a cultural shift, right? And people's behavior was completely opposite to what they said they were selling, right? Yes, there was the director there, from the startup, right? He spent the whole day making jokes about the Northeast, jokes about Bahia, this and that, and taking situations personally. For me, it was one of the worst places I've worked, in terms of relationships with people. Obviously I never treated anyone badly, but I left there, kind of, yeah, doubting my potential, you know? Because one of the things that we arrive at in the company as developers, at least me, is wanting to show

service, wanting to show that I'm capable, you know? No, I'm not the type of person who arrives and likes to, a, I'm going to look for that place where I can least show service. No, I like a challenge, I like having a problem that the company trusts, right? Look at yourself and see, give yourself a chance to solve it, not the company that passes it on to you. This company especially, I arrived, is the system that I was going to handle there, no one passed anything, they didn't provide source code, git. It seems like I wasn't welcome, that was the impression I had, so every day I spent there was a day of, kind of, suffering, like, I had to go there. But then I started to mature in this sense and realized that I didn't need to stay there, here in São Paulo as the market is very hot, I could have left a week later, but for the sake of having a Salvador culture, of staying a long time in place, being afraid of changing companies, I kind of accepted it for a while, but then I went to the second company, and one of the things that helped me a lot in adapting here was because Brasilprev, right? In the company I worked for, there were a lot of people from Bahia. We even call the place Bahia Prev, because the consulting companies hired a lot of people from Salvador, Bahia and there, so I kind of felt at home, you know? The jokes, the things, and it was even funny that the Bahians changed the behavior culture of the São Paulo people, right? The guys were more closed, they started to become more open, they joked, you know? There wasn't that boring conversation, which there was in the other company. It was a good place from the point of view of relationships between people, and it kept me there for three years, and the reason I left there was because I started to realize that this relationship between people was starting to break down, right? ? I saw that the problems that the company was passing on to the development team, which was kind of what we felt on a daily basis, was affecting the psychological, generating stress and I chose to look for another place, and my current place I felt quite embraced too, it was the first place that kind of, I'm the only Bahian, right? And people work from home, one of the things that is most difficult for me because I'm more closed, I'm more calm, I'm not much, I arrive at a place, and I arrived, and so on, I'm more calm, I go little by little and so on. And I felt a lot of support, because from the first day, the company gave me a lot of support in presenting the business, yes, showing how it works, how they work, right? They were already direct in the interview, saying that they didn't like people who were there to be stars, right? Because in our area there is a lot of this, people who join the team and talk, talk too much, and do too little, you know? So, I think the issue of relationships between employees is, one of the things for me, let's say, is sixty percent of maintenance, of continuity in a company.

Researcher: In your opinion, can a lack of organizational commitment lead a professional to leave the organization? Why?

Researcher: This commitment means that the employee is fully aligned with the organization's objectives and beliefs.

P02: Yes, yes. Yeah, I'm going to make the comparison, right? The first company I worked for in the IT area was at SSP, practically, yes, it didn't pay well there, but I had the opportunity to do things, and I didn't have the knowledge I have today, right? He was much more enthusiastic, curious. I arrived, studied and did it, showed something and people kind of saw it, they knew who I was in that corporation.

P02: Can you ask the question in the chat, just so I don't get lost here?

Researcher: Yes.

P02: Yes. SSP in general was such an access to responsibility that I had, which was one of the reasons, firstly: the salary wasn't good, there was no prospect of growth, right? And at the same time there was an excess of activity, an excess of responsibility for a single person. Then when I came to São Paulo, one of the only things I felt was the following, at least at Brasilprev it was the issue of how they had legal problems, this type of thing, kind of outsourced, was seen as a part of the organization , he knows? Not just one thing, so what happened? Kind of like the company's indexes, the things that the company had during a year of achievement, and so on, we kind of stayed apart, we just listened, we didn't participate in meetings that employees participated in, right? And when we go to, in the current company, there isn't that, that border, I'm kind of treated as an equal, you know? Yes, the company shows, I am aware of the company's reality, you know? I know what's happening, I'm not hearing through third parties, through others, what's going to happen in a week, no. Every week there are alignment meetings, so I'm there, I'm listening, I'm listening. So, it makes me feel part of the group, you know? Not just getting there and just turning on the computer to work, and having no idea of the context, the impact of what I'm doing, you know? This is important, knowing that the activity I'm working on will bring an important return, right? Not only for the company, but externally and, that's it, I think, there is no organizational commitment, the company just wants to know there. He's on your case, wanting to know, so what, is it over, is it over? Thinking that the question of having a good developer, of having a good collaborator in the corporation is just a matter of numbers, you know? Not only that, I think that the lack of organizational commitment leads professionals to leave the organization.

Researcher: In your opinion, exhaustion (excessive workload) is a factor that has big influence on software developer turnover? If so, could you talk a little about what causes exhaustion?

P02: Yes, definitely, and that's one of the main things, right? As I already said. For me, what causes the exhaustion problem is the following: lack of processes, it's not a matter of a lack of defined processes within the team, because, for example, the agile development process, right? We manage to achieve a certain point, we make an agreement with the team, with PO, with the scrum master, we manage to make the planning poker all nice, wow. But if they, the corporation, the company, do not have the real culture of agile development, of understanding that, what it is, why it is important, trying to apply this at other levels, this generates exhaustion. It's one of the problems I was experiencing in the last process I was in. It was, okay, we're trying to apply as much agile development here as possible, follow the whole beautiful cycle, but the director there, he wanted a deadline, like, it has to be ready. The idea, in one sphere we apply the process, but on top of that the manager has no idea what the process is like, so this lack of knowledge, of how it works, that in agile development you don't necessarily get there, and say: you're going to deliver a huge backlog in X amount of time, oh but we want a deadline, we can't have a deadline, what we can do is this, this and this. So, I think one of the biggest problems is this, the company has no idea how the agile process works. The current company I'm in, they're just starting to use the agile methodology, right? They didn't use it in the other project I was on, but I realized that even though there are difficulties, the guy at the top, the manager, is aware of the importance, he talks as an equal with the higher levels. technicians and this is very important. When the guy is up there, and has no idea what agile development is and doesn't have a connection between the more technical part and the business part, then the issue of stress, demands, absurd things, the PIO begins. They don't understand each other with the employee, poorly written stories. And this wears on, it was one of the things that wore me down a lot, it was a question of you are in a cool project, but the problems of the company's decisions to put, let's say, the carriage before the horse, right? It's what caused a lot of stress for me, and it's what I'm trying to avoid as much as possible. When I see a company and it starts to bring more stress to me, to my quality of life, than it benefits, right? I prefer to follow the path, it's not just salary. I earned very well there, very well indeed, I had immense morale, I asked for a raise two, three times a year, and I got a raise, you know? But it wasn't just money, I saw the issue of quality, I started to think a lot about it, and follow another path.

Researcher: In your point of view, the high demand in the industry for software developers

software, contribute to turnover? Why?

P02: The high demand you talk about is the amount of activities I have to do during a...? What would it be, what would be the understanding of high demand, or is it demand?

Researcher: That's right.

P02: Yes. Why? I don't like changing companies from time to time. I usually set a deadline. I want to stay there for a year, one year, two years. Because I think there comes a point when you kind of reach the top of the company, and then you're not going anywhere anymore, you know? Yes, a company that doesn't have a legal career process to motivate you to stay there. Yeah, then you start to think, wow, there's an interesting company here that has my friends who work there, someone I know who works there, who speaks very highly of it. What am I doing here, stressing myself out, if it is well there, and has a well-defined process, has a very good culture, and why am I going to stay here? So today, the issue of high demand is not just a question of salary, in my choices. It's also a question of how the processes are going, how things are going, how companies are culturally accustomed to the development process. Yes, the high demand increases the salary, inflates the salary, there is nowhere to run. We sort of manage to bargain, not bargain, negotiate a lot in relation to salary when we change companies. To give you an idea, there are days when I receive two or three job offers a day, and then I say: I haven't started yet, thank you and so on. But like, sometimes I'm like, oh, cool, I'll work with this, that, but no. I hold myself together and so on, but there are people, I've known, who jump from one place to another for two, three months, and they don't care. The important thing for him is increasing his salary range, quality of the company be damned, it's just a salary issue. But, for me, it does help. In general, from what I understand, when the developer, even more experienced, sees that the environment is orange, the guy doesn't stay, he immediately looks for another place. Today we have this... to a certain extent I think it's even positive for the developer, because in Salvador, turnover is practically very complicated for you to get a job, leave a company and so on, to go to another it takes a maximum of a month to you get it, pass it and it's there. Here, if I want to leave a company I'm in today to start next Monday, I can do it, you know? Because there's no reason for me to stay at my current company, if it's not doing me any good, if it's stressing me out more than it's helping me and there are companies that have a much more mature process, with much better teams, much more challenging projects, he knows? Inviting you there, knocking on your door, come on, come work here. So, this is notable.

Researcher: Did you have autonomy to develop your work and make decisions? If so, what autonomy did you have? Do you believe that autonomy has an impact on developer turnover? Why?

P02: Look, when I worked at SSP, my autonomy was one hundred percent, my autonomy was to configure a server, you know? Getting there, actually configuring a Linux server, configuring a web server, redesigning architecture, all of that, database, all of that, even the screen. On the one hand, I liked that, why? Because I was studying, so applying I wasn't stopped, I didn't get demotivated, I thought of other things to do, so I, but this generated another effect for me, which was excess responsibility, why? I put my hand in, I was the child's father, so all the systems, there weren't a few systems, there weren't a few solutions. I was wanted, wanted regardless of the time, weekends, at night. And I didn't get paid for it, that was the worst of all, you know? Then, when I came to São Paulo, I had a shock here, I had no autonomy. It was that little box there, an AI video, an already well-defined environment, there were things that I saw that the solution was very obvious to do, I needed other people, calls, I don't know what, to have permission to achieve install. It kind of wore me down, but I learned to deal with it, why? Yes, I could understand why that process was kind of frozen, so on. At a certain point I was able to kind of focus my autonomy on the things I had, which were more within my reach in that company, such as: the best solution for a certain story, a certain activity to be done over a period of time, I participated in planning, sprint. So it helped a lot, understanding the context as a whole of what I was immersed in there. The autonomy I had was much more from a business point of view than from the coding itself, you know? That was pretty cool. And in the current company I'm in, I feel like I have a certain level of autonomy that's very interesting, because I've kind of gotten used to it. People don't want to, they give you the freedom to propose the best solution, so I, right? Then you evaluate it, if I think I can use a language, or something else, a package that can solve a certain problem, that can make my life and the team's life easier, wow, people accept it without any problem, you know? The culture is very different, so I think so. That autonomy does have an impact on turnover, and you have to know it, kind of manage it. When you put your hand in, mess with the code there, and there's a problem, you become the child's father, right? There was even a motto that colleagues said, that proactivity… I forgot now, I'll send it later, I forgot now.

P02: I'm just going to be punished for my proactivity, like: cool, I have an IF here, an ELSE, which has nothing to do with it, for me it doesn't make sense, I'm going to have to kind of refactor it, then you refactor it , then give a big BO, then when they look for you, even if it wasn't you, as your name is there, you, because of your proactivity, are in the middle of the BO, you know? More or less that.

Researcher: In your opinion, is salary the main factor in keeping a software developer in the organization? Why?

P02: Look, for me it depends on the moment, you know? Today, until recently, salary wasn't so decisive, you know? Today I try to maintain a balance between the two things, for me the pillars are the issue of the relationship with the employee, stress, salary and what that can bring me in the long term, I always think about that, right? If I start to realize that, even though I'm earning a little less than what I was earning before and it doesn't bring anything in the long term, there's no reason for me to stay there. Obviously the salary, not to be an unfair salary, right? But it's not supposed to be a salary either, it's unrealistic. It has to be a salary, kind of tangible, from the point of view for you to maintain a good quality of life, and for you to be working well, you know? He's going to work. It's a somewhat difficult question for me, I could be in another company today with a salary even above the market in some ways, but I wouldn't continue, due to the problems that were most impactful for me, apart from the issue of stress, the relationship with employees, it kind of started to get more toxic, let's say. But I also wouldn't go to that company if they offered me a very low salary, right? A salary that was far from the reality that the company was offering me. Because here, we talk a lot with colleagues about the job market, how much each person earns, hours and so on. They have an idea more or less. So the company gets in touch with us and offers a salary below what the market is paying, even in comparison, knowing that your colleague is earning X, you won't want to earn less, you know? You will want to earn at least the equivalent, or equal to that. Yeah, but I don't think it's the one hundred percent decisive factor, you know? I think, when you're in a... If the salary is fair for that moment and you have other things to add, let's suppose, well: if today I went to a company, to earn a little less and that company had a project which would bring long-term results for me from a career point of view, right? I would agree without any problem, you know? But at the same time, today it would be much more difficult for me to go to a company to earn twice as much as I earn today, and it has the legacy system. I wouldn't, I wouldn't motivate myself, you know? So, it's something that's not so straightforward to think about, and it also depends on the times, there are times when you need money. So for me, the issue of focusing only on salary was very important. But salaries that are below the market, which do not match reality, will distance developers.

Researcher: Considering everything you said so far, could you talk a little about the factors mentioned, and not mentioned above, that can lead the developer

of software leaving the organization. You are free to comment anything.

P02: Let me think about something here. I think it's much more a matter of personal experience, right? One of the things that causes developers to leave the organization is when the organization itself sometimes underestimates them... but I think that's already been said, right? To give you an idea, when I left the other company, and went to Brasilprev, I did an interview with a team and the guy wanted to put me in the spotlight, and look, I was at a time, when I was in a really bad place: I arrived at a place here so empty of anything, I'm stupid. And the guys who were working on the team knew a lot less than me, but they knew how to sell themselves, you know? Today I'm kind of improving, in interviews, etc. I try to answer what they want to hear, but I don't know how to sell myself the way people sell themselves there, and in practice it's nothing like that. And so, then the guy made me full, right? And that made me feel so bad, man, I had been on the market for seven years and the guy put me on full-time. I haven't worked with a certain thing, most of the time you just have to study it, you're unlikely to go into much depth there in that environment, it's more like deploying in the cloud, these types of things can be learned, it's not something that is another world, you know? Even more so, people who… left UFBA, and UFBA is really a mess. When I looked at the things I had to do there, and looked at the job market, sometimes I laughed like that. Problems that people exaggerate too much in the workplace, you know? And what happens? The issue of underestimating, I remember that after I did the interview another team arrived, right? I was in need of a developer, my profile was the closest they wanted, I went for a senior position, so I could follow the game. But that thing where every day you're trying to prove that you're good, that the guy keeps doubting your work, that's horrible for me, you know? You seem like there's someone there on your back, or someone kind of... It was something that happened even at SSP, kind of looking at what you're doing, you know? Hold on, man. The issue of development, of coding, has a lot to do with the intellectual issue. Sometimes there are days when I'm here to solve a problem and I solve it in two hours, or less than two hours. There are situations where I'm not cool, I spend all day doing it, you know? So, sometimes charges, unnecessary things that the person above, for me, alienates the developer a lot, and another thing, it is not one of the motivating factors, but I would really like to work in companies where I could use things that, during academic life, things I learned at the academy, I could use in everyday life. I know that in most companies in general it's not easy to find, the processes aren't easy either, right? But that thing about you, you're actually solving a problem that's not a crud, not a business problem, but something that you say like: wow, I managed to reduce, I don't know, the processing of an algorithm, improve a performance, in short… you know? I think the challenge, when you have a challenging project, is what really motivates you to stay at work. That's the difference, like: I'm doing an airline reimbursement project here, but at the same time there's another guy, right? Who is doing another project to send a rocket to Mars. Which is the most challenging, which is the most thought-provoking? I know the road is long, and so on... But I just made this parallel to say it like this: when companies don't have many challenges, they are very lukewarm about it, they prefer to stay kind

of in their comfort zone. It also alienates the developer. So, I think that's it. I don't know if I said too much, but that's all that came to mind now.

Researcher: Great show. Very good.

P02: Like I even spoke to Professor Eduardo, right? I'm in a dilemma, like this: I don't want to reach forty years of development, as a developer. I have nothing against it, there are people who like it, like to get there and code that thing. But for me, I realize that every year that passes, well, a new technology comes, a new framework, I have to update myself, study for that, but the problems are the same, you know? What changes is just how the problems appear, but it's the same thing. Until recently, no one talked about microservices, right? Then this wave of microservices started, but in general, the problems end up being the same, you know? The process is the same, things are the same, and so, I really like the elementary part of things, fundamental, right, that more thing, yeah, why, and the job market doesn't give me that, I've already noticed. And in the why part, that's where you find a lot of things that encourage you to keep working, trying to wake up, thinking, writing here, and in the job market it doesn't give you that. He doesn't have a reason, he wants you to deliver something, and if there comes a time when the company is disposable, it will fire you, you know? And so, you will learn to use N tools, N frameworks, update yourself in Java, update yourself in Python and so on. But the most basic things, the things I miss, the job market doesn't bring me. So, today I'm kind of trying, I did a lot of research, right? And trying to move into the area of data science, because there are many whys to be answered, right? There are a lot of elementary things, a lot of mathematical things that need people, which I believe is my profile for acting, right? And I still realize that companies have not realized the power of data science. Every company I go to and ask, everyone boils down data science to just BI, BI, BI, you know? It's not just that, BI, I don't know, is one of ten of the potential, from my knowledge that it can bring gains to a company. So today, I'm kind of trying, little by little, to go into the academic field, I don't know yet which college, which place I'm going to work in, right? I have my problems with the academic environment, so, the way things were during my graduation, I'm kind of a little afraid, right? And so, at the same time it's something that I miss, you know? You wake up every day to solve something that few people can solve, I think that's really cool. And that.

Researcher: Perfect Renato. I want to thank you very much for your participation and availability.