Dawid Wozniak:

OK, so I've started the transcription right now and if you cannot see it on your screen and you can see it now on my screen. It's recording what we are saying, but it's not recording what is on the screen and it is not 100% correct so I will correct it afterwards and then I will send it to you so you can see what will be included in the report the project. Thank you one more time for participating and I will start with the short presentation to explain your role in the project at this stage of the project. then I will show you the project itself. I have some prepared questions that I will ask you so you can help me with the progress. So, let's start with the presentation. So, in short, I would say that I wanted to achieve is let's make a call with history great again and my research question is - which insights about commit messages are useful for stakeholders who are visualising a project using Git track? So, you can think about the Git track as a car that I have, and you need to help me to tune it a little. So, we have this commit messages. But we write and then we don't do much with them. You can, of course, just look what was changed with this comment. But it's just the raw data, we are not doing any automated processes to help us with our work. We don't really know how to do it. If you think about it in the context of the traffic, then you have the signs that you need to give to the user of the car in which direction they should go. But uh, there are many options how to do that. What should be transparent, what should be filled in? What should be the shape of the sign? Of course, we want to get there, so that's the ultimate goal. To use this information from the commit messages history to help us to have some automated process. In terms of our metaphor, you can help me to go there, and you can imagine yourself as a mechanic in this particular project and I will give you the tools. The tools are my skills and my time everything that I can do to fulfil your wish or just address the wish why it's not possible to do it at this particular point. But to be able to do that you need to provide me some feedback. I will give you the car. I will show you now what we have and you need to answer some question. You tell me what is good and what is bad. And of course, it will be nice if it's the honest opinion. Finally, we iterate over the solution. I have many testers that I talk to and then I need to summarise it. I need to take the most important things from each of the tester's feedback and then I will create the solution. I will mention it also at the end of the presentation. that's the end of this section and we start now with the tool. So it is a GIT track at the beginning you have the list of the repository where it was run and I will just show you the git track analysing the git track as we have a small issue with the scalability that I'm working right now, so the next presentation will be hopefully on some project that you work on. Then what you can see here is the title of the project. Here is the name of the branch when it was analysed, what the comment latest analysed was, how many files were analysed. Below, you have some information on how to provide feedback. Two types of charts are: bubble chart and map chart. It's something it is like this old file explorers and when you have the bubble chart then you have big bubble that contain other bubbles. It's the folder and you have some small bubbles that's a file. If the bubble is bigger than other bubbles that means that's bigger in terms of size of the file. So if you click here you can see that it is probably the biggest file in the repository. We have some metrics, so they are. Now, you have just the file extension, so we have some TypeScript files, some images and so on. And we have also number of commits metrics. When was the last change done, top contributor, truck facto. So, let's look at something else. So here we have the last change so you can see when this file was edited the last time in the whole history it stays with the file. And then finally we have this detail view. So, if you click on the particular bubble like here on this demo image file then you have the size, the number of commits, the last change that was done to this file where it's located and outer distribution. So, one of the 7th comments were done by Jonas and all of other commits were done by some automated bot and this is the commit history part that I'm responsible for. You just see three commits; you can also see they are order from latest to oldest they are always sorted like this. If you expand the view there is just a commit message here, we don't have that many comments,

but if you go to one of the components like details then there might be very long.

And if you open like second May view, then on this particular day there were more than one commit. The new commit message is just indicated that putting a new line. So, my first question is, what is your first impression about the product? And do you like it, or you don't really see how it's useful?

Jakub:

Well, to be honest, it kind of sounds cool, especially like the graphical view of the whole project seems very interesting and like the idea that you can basically visualise how big certain files are and like what's the structure of the project overall. Now that sounds really interesting and like something that could be really useful for like team managers or like business analysts I can imagine. And I kind of like that. Commit history part, there's this built in tool in right there where you can browse commit history in GitHub. But that is really hard to read and hard to understand. And then with a tool like this where you can basically like look at any file and like. I can see the potential there that you could basically see the comment history more easily and then be able to like see what's been done. I mean, I don't know if it's the time to talk about improvements, but I mean I you could also like add links to view the changes made in the particular file in the commit history if there is. If it is not there, I cannot really see that, but I don't think it is there.

Dawid Wozniak:
Now it's not there and there is definitely time to talk about the improvements. So, if you have any suggestion right now, you can just tell me about it.

Jakub:
Ohh yeah, yeah, so that's one thing. But yeah, I kind of feel that it could be very useful like to make the browsing the comment history simpler and gain more insight about like certain file and what's been done and like what, what were the changes over certain periods of time though.

I was thinking about something, but then I came up with the idea that it's actually not so not very used.

Dawid Wozniak:
Yeah, that's OK. Now, I have three questions that I would like to grade you from one to 10, when 10 means that I completely agree with the statement and one that you completely disagree with the statement. So, when you write or read the commit messages, it's important to me what is there. From one meaning that it's not completely important to you to ten that it's very, very important to you.

Jakub:
Like 5 because unless you have a specific policy in your company which tells you like what way to write the commit messages, then sometimes there is basically just like useless stuff that it's very hard to understand. So like it it's really depends on the project and the person who's writing the commit message.

Dawid Wozniak:

Okay. That's good feedback. Now I have a second question. So, if you look just in terms of the UI, so how you like the way how it's presented and the one is generally great for the whole product and one is just for this bit, so the commit history from one that you think that is completely not usable to ten that you think that there is nothing that can be improved.

Jakub:

Wait, so there are two grades, one for the UI.

Dawid Wozniak:

One for the product so how you like the visualisation, the layout in general and then separate grade just when you think about the commit messages part over here.

Jakub:

Yes, so uh for the product, I think that could be really useful. So there I would give it like an 8 because I mean it's probably take me some time to like get the full insight into like how this product could be used and see how much value would it bring. I've never run a company so that's like very hard to tell as well. But I imagine it's really good and then they commit history. This is the part that I especially like because. As I told you before, like the comment history, it's very hard to see stuff, especially when you have like those very long lines, meaning the commit history. And here you have like everything grouped together and you can just basically very easily view one file part. So, there I will give it like a 10 cause like it will be very useful especially if you could use it within the ID.

Dawid Wozniak:

Yeah, man, that's a very good feedback. And the last question in this part. So when you have the whole product like this, how beneficial would it be to use it for your own project or for your work project. In the scale, from one meaning it's completely not usable for your own project and for your work to 10 that you will use it basically every day.

Jakub:

Yes. So it's kind of hard to tell because like for my own project. Ohh I don't see how I would be using that because like I can see the the business value coming from using it where there's multiple people working on multiple projects and then if you are the only person I don't see how it would be useful. So, I would say like 6 something like that because then again it depends on like who uses it. When you have like a project or there's a couple of Members and I can see that it would be very interesting to use it to see what's been done and like basically kind of measure the flow of the whole company like and the business value that you are getting.

Dawid Wozniak:

Yep, that's very good feedback. And now I have some open question as the end of this meeting approaches. So, what would be your main goal to use Git track? You mentioned some things about the business value and to see the progress of the project. Do you wanted to add something to this?

Jakub:

Hmm. Yeah. So, I can imagine that it could be very nice to like be able to see the structure of the project and then kind of like gain more insight into what people are doing, which areas are the most developed and which are not. So then you could basically like see where maybe where you need to put more focus on, like how well designed your product is. I mean also you cannot tell how well the project is designed but still there are some metrics that you could use and like here it's usually visualizer. So, it's very comprehensible and I think that would given the right mindset and like the analyst mindset I think that could be very helpful when especially when running a company.

Dawid Wozniak:

OK. And you mentioned that it would be good to have some links to these commits when you click on the message of the commit itself. So, is there anything else that is missing related to the commit history?

Jakub:

To the common history. So yeah, the links. Ohh you have the dates. What else is there? Yeah, I think the links are the bits that are the most important for me. I mean I don't know how far you would you would like to go, but you could make like an inbuilt preview of like the changes or like a comparison. Comparison window where you could basically like take two of the comments and then compare the changes between those two. I don't know [tool] like that, but yeah, I guess the links are the most important. For me at least.

Dawid Wozniak:

Okay and what is the most useful insights from commit messages or commit in general, but especially commit messages? And here you can assume that you have like unlimited resources, unlimited time. What would be the most useful thing that you might get from those pieces of information about changes to this particular file, so basically commits.

Jakub:

You know, so I mean, if you we had like a very well defined way of writing it down, what's being done in given file, Then I guess that would be very nice because you could easily see where the changes were implemented and you could basically go back to history and like check the different files and like what's been changed, how is being done. Maybe also who approved it in all of that, so like, it's something that my insight is, maybe with a tool like that where it's like very simple to see it, people would be more motivated to write like proper messages, But then I can also imagine that you know it will depend a lot on the on the person's way to write the messages, so I guess if you like defined a way to write commit messages or like use of on one of the available ones like you know with the feat [feature indication in commits] and all of that and then somehow maybe made kind of like a way to detect those messages into like something that could be displayed in the UI, then that could essentially mean that you would get more insight and again something that would be amazingly visualised. I think something like that would be nice to use.

Dawid Wozniak:

Okay. And is there anything else that you would like to say about Git track? Something that I haven't asked about maybe.

Jakub:
You know, like for now, I don't see anything that would be more interesting to me. I mean you could maybe do like defining a certain size of the bubble. Maybe you could have like labels on top of the bubbles to be able to see what the file is named before clicking it. I would like a button that would allow.

Dawid Wozniak:
Yeah. So now when you hover on it. You have the name of the file.

Jakub:
Yeah, that's true. But then like maybe you could add the button that would enable like displaying the names, because I can imagine that now it would be hard to navigate because like let's say you have the component right and then you're looking for a certain component, you don't know how big that component is in terms of, you know the size, the amount of lines of code, then I guess it would be nice if you have it. There was like an option to look for a certain file or like highlight it somehow. So that you would find it very easily because now in the worst-case scenario we have to hope over every single bubble inside components.

Dawid Wozniak:
So here you search. So, if you know that you are looking, let's say for the file that is called site panel, you can find it here and you click there so then it zoom in to the place where it is and it's also displayed in the detail view and it is blinking. I agree you need to spend like a minute to find it, especially if it's small.

Jakub:
Okay. Yeah, yeah. I didn't notice that. Yeah. So that that's very nice like. I guess that's something to mention because it's very useful again to get more insight to find a certain file. I mean I think overall like the the product looks good and I would need some time to think about like the whole overall structure. But I think it's for now what it's. The first impression is very nice.

Dawid Wozniak:
Great. So hopefully you will accept my invitation to the next round of feedback. But before I talk about it, it was my last question about the product itself, but now I have some questions about you. So I can then make some statistic depending on some factors of my testers. So how many years have you work in the IT industry, including part time jobs? It doesn't need to be very precise to the mouth. You can say in years.

Jakub:
Almost 5, I'll say yeah, something like that.

Dawid Wozniak:
OK. And what was your primary role at that time?

Jakub:
Backend developer, but now it switched to full stacking developer.

Dawid Wozniak:
Yeah, that's great. You probably have been in many teams. So how big in average was your time? It is including developers, QA, product managers, product owners.

Jakub:
But you mean like project team or the like the whole company?

Dawid Wozniak:
The project team. So, if you have an project inside your company, how big is a team when you calculate developers involved in the project, QA, product managers, product owners in average?

Jakub:
So on average I would say like 6. So like usually within the companies I work with like 5 to 6 developers. And then you would have like CEO of that or CTO or somebody who's like taking care of the business. But he was not working with the team daily.

Dawid Wozniak:
Yeah. So that's a good measure that you took. And now when you have your average project, how many comments are there? And here you can take per day, per month, per year, what is more comfortable for you.

Jakub:
I ought to be honest, I would have to look at my GitHub page and.

Dawid Wozniak:
But you can just guess if you probably didn't make this statistic to look at it, so if you need to guess what it would be.

Jakub:
Ohh I would say like 5 comments a day. Depending on how how much coding is involved, like whether it's a new feature or something that I'm fixing, then obviously it's going to be less right.

Dawid Wozniak:
Yep, that's right. And the last question, so when you have the average project, how many files are there? In this project there's 105. And when you have your average project, what is the number of files?

Jakub:
That's a really hard question, like the personal ones I would say have. Like I don't maybe 50. Something like that. Like our company with companies, I would say like 200 and more cause like you know, they're usually way bigger, like between 50 and like 200 and then above for like the very big projects.

Dawid Wozniak:
OK, so that was the last question in this part of the survey, there would be a next part when I implement suggestion and it would be that the next way of feedback. So, then I of course don't ask

you the statistical question one more time we will talk about the changes, and I will address the suggestion that you gave me why it was implemented like this or why we need to skip it for a time being where my focus is so I will send the invitation and hopefully we will see each other again. Thank you one more time for participating and hopefully see you again when the next round starts. Thank you one more time and I will stop the transcription now.