Dawid Wozniak:
OK. So let's start. Thank you for being here. So, I will start with the presentation of the tool and then we can talk about your feedback. So, this tool is called Git track and when you open it on your local machine, you're going screen like this. So here you have some repositories and some of them were preanalyzed. So, then you can just view what was there and they don't need to be analysed. Otherwise, it'll take some time. Because of the problem with the scalability, we just take 5000 latest commits and based on that we generate some visualisation. So here we have this data. Now we just create how it should be visualised. It is a huge repository. So it takes some time. Using this time, I can tell you that if you would like to just test it without my assistant, you can do it. If you have npm installed, then you use this command: npx -y git-truck-beta@latest. Then you can actually run this code and when you load this on the repository, you will get something like this. So, on the left side you see the name of the repository, the current branch, when it was analysed. The short version of the commit hash, how many files there are, some feedback box. So, we are going to use it now. Here it is the bubble chart. So, each bubble corresponds in the size of the real file size so here in this repository this bubble „platform" it's the biggest folder and you can deduct that based on this size.

Daniel:
That basically takes commits per file with grouping via folder?

Dawid Wozniak:
Yeah, so it's, basically reading all the commits and then try to visualise something for you, so the first visualisation type is a bubble chart. You can also use the tree map. It will take some time to regenerate this. Then it looks a bit like the old File Explorer when you have the same idea, but it is just the squares instead of the bubbles. When it is generating, I'll describe the rest and then we can also play a little more with that. It's gonna be the problem. So it happens from time to time but it loads at some point. So, we also have a metric, the default one is the file extension and on the right side you can see that it's colour coded. They are so small that you cannot actually see the colours. But if we click on some of the bubbles then you will actually see that there are some colours. They corresponds to the file extension. Other metrics that we have is the number of commits. So how many commits are actually detected on the particular file? So It is not that important for my master thesis. This is important just to understand how it works. And yeah, so then another metric after number of commits, then we have the last change. So, when the last commit was detected, you have track factors so how many people actually touched this file. Then we have single author so if you have a file that only one person committed to, then you will get the information there. Next option, it was a chart depth there when you have a chart you can see that there's a lot of bubbles. If you have a product that is more complicated than you can actually specify that you would like to see just one level, two levels and full levels. Sorry for that actually, maybe just my computer today slow. I think that it works. It is just not working well with the bigger repositories. We planned action on our side, but it is just not topic of this master thesis and it is difficult to solve. So, I need to somehow limit. I can try to disable animation. It should help. It's very costly to actually generate all these bubbles or all other generation of the files. So now I will talk about this as it will probably take one minute to load and here you have some object, so one of all. When I click on some of these then you will get a detailed view that will be on the right side and then we have some detail information including the commits and this is actually the topic of my master thesis. So, this what I said and now it is just the introduction. It's slow because my computer it's slow.

Daniel:
So did that all that on your own or no?

Dawid Wozniak:
It was the existing tool developed as a part of the other master thesis. And then on the top of that, I just developed the functionality I'm going to show you. Now I will just show you how the chart more or less work. So, then you can increase it to two and if you increase more then it's more and more costly. I would just leave it on two then it's faster for sure. Let's see what happen if you zoom in. Let's say go just to the „platform," then it will follow you. So, you can see two levels from your current destination there. And if you go somewhere where we have actually some files. So, let's say, yeah here. So, you can see that there are some colours and they actually correspond to those colours here. There is a lot of extension in this repository. So, if you would like to see the whole list, it will be like this and then you can see OK, maybe it's similar. So, if you hover over it, you can see here what it was you can see the name of the file and the extension. So, you know that, so you can say, ok, let's close that. So, you can also see how many commits were there. Last change so which files were recently changed and then you have single author and top contributor. So based on the comics, who should be the owner of this file. It is a nice transition to actually what I wanted to show at the beginning. So here we have those details, the one is "General tab" that is used to be there. So, it is the folder. So, we have the information how many files are there, how many folders are there, what is the distribution of the authors and finally, we have a part that I'm responsible for it. So, this number of commits, I will collapse, that's how we have more place and that's something. I'm responsible for this tab. So here by default you have commits sorted by date when it was performed. So, you have more than one commit on one day then it is displayed like this and you can have it from latest to oldest or another way around. You can also sort by author. So, then you can say OK, all these commits belong, I mean, were committed by Damien here. It was committed by some other person and so on. We have also filters, so maybe you are just interested in… I'll stick to the date because it's more convenient for this test. Maybe just interested in „fix" and then you see that all commits that are there, they need to include this phrase. You can also include it in that commit description. Then you might be even more specific and say, OK, I'm just interested what Tom did. Actually nothing here with this filters, so I can just exclude Thomas. And now I will see all the commits that were done by other people. Finally, you have the time frame. So, if you just want to see what happened, let's say from the 1st December, then you would just select here and it's reflected. This functionality does not work here because it is designed for GitHub. So, if you have GitHub and you use merge commits then you can also show them or not show them. In the DevOps, we use different strategy, and it is not designed for DevOps. So it was quite a long introduction. But my first question is what you think about the UI. So if I give you like the sentence that the information about the commit messages is presented in the clear and easy to navigate UI. To what extent do you agree with it from zero, that is completely don't agree with it to ten that actually you agree with it and it is perfect. Nothing to correct actually.

Daniel:
So the commit view as I am concerned. So first you need to find that before it comes with you which is somewhat of like by default the right pane is not really visible, so you click some on it and then it's not selected by default. So, you need to see that it's, you know, you need to navigate to commits and then there is the sorting and the filters. So that's a little bit of a cognitive element.

Dawid Wozniak:
And then when you found your commit, because I forgot to mention it, you can also click it here and then you have more information and you can click „show edited files" then it will fetch which files were changed and you will get the list here and they will be highlighted in the visualisation.

Daniel:
Yeah. So, like scale was… what's the zero to ten? Which do you say is the highest?

Dawid Wozniak:
Yeah, 10 is the highest. Like, nothing perfect UI, noting to change, to zero that it is completely not usable.

Daniel:
Yeah, I mean. It's falling somewhere in between, so it's like super simplistic and minimalistic and it's also not following like, not like a common line towards something that is like super utilitarian. I would say it's probably I don't know... Seven.

Dawid Wozniak:
Yeah, that's a good grade. So that I have another question, it is the last question still with this scala thing. So, if you look at this whole product and I know that it's a quick introduction, but if you think about your work. How beneficial would it be for you to use the tool as it is right now, so you can actually do some chart types metrics. You can also analyse the commits. From zero that means if you have access to it, you will never use it except this presentation to ten that you will use it for all your project multiple times a day.

Daniel:
Yeah so I see roughly like 3 scenarios. So. if you're a person task with some sort of team organisation, so when you are like team lead manager or whatever level. You might be interested in to see it. If you have a project spanning multiple teams, there is a nice way of breaking the responsibility to multiple smaller teams, so that would be kinda useful just to see more or less how you might split the domains so they follow more or less you known better project's structure from the perspective of individual contributors. It might be useful, especially at the beginning of the project when you're just first starting, just to get a grasp of the complexity, because the line count itself is a reasonable measure of complexity. But the activity of the file in terms of the number of commits, that's probably a better predictor of complexity, especially in terms of what do you need to focus on when you're onboarding on the project. So I would say that it's useful for that purpose and I was also thinking about something else. It's the first then other than that, you also show me. So there was another… That was the... So here you have talked about metrics. So, you have file extension, number of commits…

Dawid Wozniak:
Yes. When it was last changed, if there is a single author. So basically, we show like single author and not single author files. So here just this person committed to this file. Maybe it was just renaming or moving this file. Then we also have this information, let's say. But here we have more and we just have multiple authors and then you can also see who should be top contributor. So, if I need to change this file, who should I contact? And we measure that based on the comments. Yeah. So then we can calculate this and we have tracked factor. So how many people actually committed to this file and there is also useful. Yeah. It's like 1 outer. Then you have track factor one. But if you have more then you get some other number.

Daniel:
Yeah. So, in terms of like, once you've already been on boarded the project and you know more or less where the complexity lies and everything. They try to factor who might be useful, for somebody like a manager to distribute the load and put maybe a little bit of an emphasis on sorting certain

types of bugs or feature requests and delegating them to particular people just to make sure that everyone's exposed to the certain pieces.

Dawid Wozniak:
Yeah. So, on this scale from zero to 10, how useful that this.

Daniel:
Usefulness from the presentation. I would say it's probably something like six… seven. The UI doesn't really guide you that much towards what is expected and results. It's more like you, you know ahead of time what can be done. And then you're like, OK, so what if I need to do use it like this, then I can do it like that. It's not like design for it. It doesn't really. It's maybe a little more flexible, but it doesn't guide you all that well.

Dawid Wozniak:
OK. Yeah, that's a nice suggestion. So, we can work on that in the future of this project and you mentioned some scenarios that you would use this product for. So, if you think about this stuff that I developed, what do you think would be your primary usage to go here and to do some sorting, some filtering, checking some stuff? Do you have some scenarios that can be fulfilled using this functionality here?

Daniel:
So based on a given subset of system from the perspective or like so for a given nested level of the structure, if I can see.

Dawid Wozniak:
Yeah, the commit message, author or date time and then use that to filter. Yeah, you can also do it on the global level, so you can set the global bubble position and then we'll have all commits but you can also zoom into some of these.

Daniel:
Yes, sure. I mean, the list of commits itself is not really that useful, but combinations of this app system with the commit list might be useful, so usually in in my day-to-day I will be looking at the others commits or blame of a specific file to the get an idea for who might have touched this last, because what you care about is either the very line that you suspect of being broken. Like you know, if somebody has an idea about that or on the scope of a method so it usually just resolves problem. Your resolution is either like line, method or maybe a file and it isn't too big. That's all you really need, but what this is useful is if you do need to zoom out and you for instance like want to evaluate, maybe the scope of contribution of somebody like you know if I don't know if John on your team or somebody is more skilled in working with this module. That module that might be useful for this and that or you know within the scope of like let's say last year when we've onboarded this project in my team or something like this, who is working on what, right? Where we've made the biggest contribution. So, it's less, I would say useful for individual contributors and more like, you know, evaluating the impact of people.

Dawid Wozniak:
Okay. Yeah. So now I have another question that is - what is still missing regarding this commit history, something that is not there but you would like to have it here. Do you have any ideas how it can be improved somehow?

Daniel:
So of the top of my head, not really. I would start maybe with a use like getting a list of use cases

where, then what? What I might be interested in and then go from that point versus like what could be added to those you know…

Dawid Wozniak:
This question is meant to give the answer if you have like immediately idea - I would like to have that and it's not there. So I expect that there is no something like this and it was my last question about the product but ,aybe there was something that you would like to say and I haven't asked about that. So, if you have any comments, anything that you think should be included and it's important you can say it now. Otherwise, I have a few questions about you.

Daniel:
Yeah, no worries. So yeah, we can move what I would just say like you know, depending on what, what, what your audience might be, it's often a good idea to start with like presenting common use cases in an accessible way. The tools that have, like lots of different toggles, are good for like explore exploration over domain. So, and it's fine. I mean it's not a problem, it's just a question of assumptions for me, as I primarily do things on an individual contributor level and for me, the breakdown that I see on the on the resolution of the file is usually enough.

Dawid Wozniak:
Yeah, so now I have a few questions about you. So how many years have you work professionally with IT projects and you can include part time jobs and it is in years so you don't need to be very precise. It is more range question.

Daniel:
I think it's gonna be like 11 years.

Dawid Wozniak:
OK. And what is your primary role? How would you define that right now?

Daniel:
I'm on the level of like my daughter is a senior software developer. So yeah, I primarily work primaly on the contribution level here.

Dawid Wozniak:
So if you have your average team, how big it is, including roles like developers, QAs, PMs, product owners but not the high management or manager of your manager should also not be included

Daniel:
Usually between 5 to maybe 12 people.

Dawid Wozniak:
OK. And if you have your average project, how many commits are there and you can say per day, per week, per month, per year? If you have a squash merge or merge with the comments, you can also take that into the account for your team is average project. So, if your average project is more like your team or more like private project, you decide.

Daniel:
In my current role, we have something like maybe 10ish projects. We primarily work in maybe like 3 or 4 at a time and then each of those you would have something like three comments per day, something like that. And they're the ones with that were currently like working on.

Dawid Wozniak:
Yeah. So do you know how many files are there? So here we have like 53,504 files.

Daniel:
We're currently in the scope of… what I'm working on right now, that they will be probably almost, I don't want to say microservice level but they are like service oriented and therefore smaller than this. What we do is maybe like 1000 something like that.

Dawid Wozniak:
Yeah, that's OK. And it's also just informative. Yeah. So, it was the last question. Now it's time for some gratification. So, you can choose which chocolate would you like to take with you. And the final question is that when I finish my paper for my master thesis, would you like to be informed about the results? So basically, get the paper to read.

Daniel:
Yeah, sure.

Dawid Wozniak:
So thank you one more time and for helping me I will stop the transcription right now.