

**Dawid Wozniak:**

Let's start the transcription. That's comfortable way. It's not always perfect, so I'll send this to review it. But yeah, it's working ok. So, I start with the short presentation and then we will get to the product. So, you can give me some feedback. It should not take a long time, so let's make a commit great again. It is the beginning of the presentation because my research question is which insights about commit messages are useful for stakeholders while visualizing a project using GitTruck. GitTruck is a tool that I will use to develop some extra logic to handle commit messages so you can imagine GitTruck is a car and your kind of mechanic that will help me to get something working, that is currently not working. So, in the current situation, I have this impression that we don't use commit messages to help us. I mean that we write them and then we just turn around and we are going back to what we need to do. So, if you think about the traffic, I will say that it is the sign that we go somewhere, then we turn around and go in the different directions. But the real problem is that we don't really know how to do this, how to help us with commit messages. So, what should we do with them? There should be useful for something more than just scrolling through them when you need some help. It is similar to the situation where you need to design some sign. So here we have the turn right sign and you can have multiple of them you need to think about the shape, about what you need to include there. Should it be there arrow that has some colour or should it be the background transparent or brings some colour? We, of course, want to get there. So, we want to have the right sign to go in the right direction and you're gonna help me to go there. So, it is your role as a mechanic that can help with my car. I will give you tools so my skills, my time and whatever I need to do to complete your wishes. What should we change in the product? You will need to provide me some feedback. So, take the car and say OK, this is what you need to change. This is what we need to do and then we will iterate over the solution with other people. Some of them have already looked at this product and there are new people, so fresh eyes, with you. So, let's go to the product. At the beginning I will just show you how this product is structured and how it's code. So, we analyse GitTrack using GitTrack as there is a small issue with the scalability. So, when it's fixed and we can do something more interesting. On the left side, you have some general information like the name of the repository branch, when it was analysed, which commit hash, how many files are in the repository. There are some options to give feedback, some settings of the chart types so it can be bubbled chart or tree chart. We can change it for a second to the tree map view. Then it looks something like it was in the old file explorers. You also have some metrics, so now each color here represents some extensions. So, you have TypeScript files, GitHub scripts and some markdown files. You can also have: last change, single author, top contributor, track factor. That's actually slightly redundant to single author and top contributor together. Then if you look here we have those big bubbles, they are just folders so like *source* or *components*. We have another bubbles in those bubbles that are just files. If it is bigger than others, it means that the size of the file is bigger. So let's say that you have this *demo* image. if I click on it, I can see some details. So, what is the size? How many times it was changed through the comments? When was the last change? What is the location? Who made the change? Here, it is like 8 commits so one commit was done by Jonas and seven commits were done by some automated bot. Below here you have the commit history, so it is the part that I'm taking care of and it is the very first version, very basic. So, by default you can see three latest commits then you can see also some other commits if you expand this data view. You just see the message and if we go to some more interesting file like *details* here in *components* folder. So it's more likely to get this file changed with your commit. So here we have many more commits, so let's expand them and on the particular day you can have multiple commits. So, let's say fourth May and if I open it, OK, it's not a good example. So, let's try another. So here it's second May and then you see that there's just a new line indicating that it is a new commit message. So, my first question to you is - what is your first impression about the whole product?

**Matti:**

I think that it looks great. I think it gives a nice overview over the code base, like immediately I think you can see where the complexity of this particular project is. If there's a folder that contains 90% of the code, then I know that's probably the most relevant part of the code base, and then I like that it's separated into the different file types that it looks relevant as well, so I know what kind of technologies I need to know to understand this project. I think that it's pretty intuitive that it's based on the folder structure and files. I think that makes sense.

**Dawid Wozniak:**

And what do you think about the commit history view for now?

**Matti:**

I think that it's not clear that every new line is a new commit. I think that it might as well have been interpreted as one big commit message on the 2nd of May, for example. So, I would probably add some kind of an icon to the left or something indicating that it's a new commit. The same way they use in in GitHub or something like that. Then, I guess that you could also show already in that overview which person has committed it.

**Dawid Wozniak:**

Yeah. We have access to all information from the commit actually. We can do whatever we want with them.

**Matti:**

Yeah, exactly. Yeah. So that might make sense.

**Dawid Wozniak:**

Yeah, OK, now I have a three statements that I would like to say to you and you just answer in the scale from one to ten when one means that you completely disagree with the statement and ten that you completely agree with the statement that the commit messages are important to you when you write them, when you read them, is it important that they have some meaningful text inside?

**Matti:**

Like five, maybe...

**Dawid Wozniak:**

OK. The information about those commits are presented in the clear and easy to navigate UI in this product. So you partially mentioned that they commit should be separated in the different ways and we should be adding the icons. So, from one to ten if you grade how, it looks now.

**Matti:**

Three.

**Dawid Wozniak:**

OK. I have some general questions. So, do you believe that it would be beneficial to use this tool for your work from one to ten, how is it likely that you will use it for some project that you are involved in?

**Matti:**

Seven.

**Dawid Wozniak:**

OK, great. What might be the main goal of using GitTruck?

**Matti:**

So if I focus on the commit, then I think that the only reason why I look in commits is to figure out why something changed. So, I'll do like the git blame and I'll find the commit hash and then I'll look up the pull request or whatever to find out what changed from the previous version to that version and why did people do it? I think that's the main thing. And in that context, then I think that message does actually means something and here would it be useful to show you, let's say, which other files were changed with this commit or and what kind of changes were done? So, the changes between those commits on the 2nd of May. For example, if there was some kind of button, that could show me what changed from one to another, that would be great.

**Dawid Wozniak:**

OK. So, do you think that this history view is missing anything? We haven't talked about.

**Matti:**

Nothing more.

**Dawid Wozniak:**

If you have like unlimited time unlimited resources, what would you do with the commit history? What would be the useful insights about the commit history, especially commit messages?

**Matti:**

I think that being able to link work items for example would be nice that I could see in the UI... I need to think about it more, I guess...

**Dawid Wozniak:**

Yeah, you can always catch up with me when you have some time, and you have some idea. Now the just the general question. If you would like to add something about the GitTruck, not only commit history, something that maybe should be improved or you think that it would be nice to implement.

**Matti:**

I think that maybe if you were able to add some kind of a tooltip for each of the bubbles, for each of the folders, like a small description in the brackets that says so like a quick description of what is inside, because it's not always that the folder name or the file name exactly describes what is there.

**Dawid Wozniak:**

You mean something like this...

**Matti:**

Yeah. Like the way you see the file name.

**Dawid Wozniak:**

So, you could hover over a bubble, if you go there and you just hover over some, then you will get your name.

**Matti:**

But it would also be nice that if people could manually add a description. Maybe, if the folder contained README file, then just use that README file. Select that in the drop down or something like that. I think the folder often will have that document in the big project. I might actually create the entire folder structure and then at least for each directory, I'll just quickly do a one line about what is inside of this directory. I'm thinking that it would also be nice to be able to tag if a folder is a legacy related or it's still active. For example, in our product at Microsoft right now we have some things that are just completely useless. It's legacy. It's not relevant for anyone like we have a folder containing the files to burn the product into CD. We have that in the platform repository... That is like a version without web related stuff. So being able to say this is just legacy, don't worry about. It would be nice, I think.

**Dawid Wozniak:**

OK. And that's good. So, I think that were all my questions about the product. For now, I'd like to also ask you some statistic questions. I know the answers, at least for some of them, but I need to have some proof that I asked about it. So how many years have you worked in the IT industry including part time jobs?

**Matti:**

Like nine years... yeah, nine years.

**Dawid Wozniak:**

And is your primary role developer?

**Matti:**

Software engineer.

**Dawid Wozniak:**

And when you have the average team, what is the number of people in your team? Like developers, software engineers, QAs, PMs, product owners and that kind of people. And now what is the average of all people in the team?

**Matti:**

So, it's like between five and twenty. So twelve.

**Dawid Wozniak:**

When you have a project, how many commits are in the project in average? You can say per day, per year, per month, whatever is easier.

**Matti:**

I have no idea. I know that over a few years then it's a big thing to reach 1000 commits like even over three years on a project where a few developers work on it every day 1000 commits is a lot. It's also a question if people squash the commits are not. Yeah. So yeah, I assume that it's that.

**Dawid Wozniak:**

You can use the same strategy as one used in your previous teams and currently so you can talk from your experience if they squash or not.

**Matti:**

Yeah. So, at Microsoft, they squash. So then, it's not a lot of commits. I think that people do maybe one commit a week, or maybe two or three commits every week, like 1 developer and then you can scale it up to like by number of people in the team, but I don't know.

**Dawid Wozniak:**

Yeah. OK. And the same question, but the about files. So how many files in average project that we created. How many are there? Yeah. So here you can see that we have like 105 files, but some of them are just GitHub flows. Some of them are *public*.

**Matti:**

So you're asking how many files are in the repositories I am working normally.

**Dawid Wozniak:**

Yeah.

**Matti:**

I think it could be around 1000 files in the biggest report, but probably more around 100 on average.

**Dawid Wozniak:**

Yeah, OK, that's great.

**Matti:**

Except if you have NPM installed, right?

**Dawid Wozniak:**

Yeah. Then yeah, but we don't count them here. We just count the default of the files and the dependency list is ignored.

**Matti:**

It's probably around one hundred I guess.

**Dawid Wozniak:**

OK, so I will send the invitation probably for the next feedback wave. So, then you can see what change you can talk with me about the changes, what was implemented from your feedback. I can talk what other people said about this. I can tell you about their ideas, what you think about their ideas. That's the end for today. Thank you one more time for showing up and that's it. We'll see each other hopefully next time.