Interviewer1

Okay, and the recording started. So I start with the demographic questions now. So the first question is, how old are you?

Interviewee

Right now I'm 42. Okay.

Interviewer1

And how do you identify your gender identity?

Interviewee

Male

Interviewer1

And what is your highest level of formal education?

Interviewee

That's overall 16 years of study. So that's BSC engineering.

Interviewer1

Oh, is that from CS

Interviewee

Yes.

Interviewer1

Okay. And how many years of experience? Do you have in Software Engineering field?

Interviewee

almost 17 years. Okay.

Interviewer1

And how many years of experience? Do you have in leading teams?

Interviewee

teams? That'll be 10/10 plus years. Okay.

Interviewer1

That's a long experience. Yeah, so these are all our demographic questions, so we will jump to the interview question. So first, we'll start about your perception. So what is your perceptions regarding a bag localization tool? If we ever use that word bag localization tool? What do you think about that

Interviewee

word? So what I think terms like that as it is, like, the first step in resolving a bug is to identify is to narrow down the area, right where it is originated from. So that should be the first step. I mean, we when we do it manually, we do it that way, right? We want to narrow down we want to identify the component first and then narrow down to classes and methods, something like this. So I will, considered as a tool that, that points me to the right direction, and to the right piece of code. And downward to why is it? I'm not sure? It depends. Yeah.

Interviewer1

So in this study, we define the bug localization tool, a tool that can identify potentially buggy files given the bug report or sectors or both. Suppose a repository contains 700 source code files. And based on the bug report, that tool will shortlist a few files potentially responsible for the bug. And then the developers can start further investigation or fix procedures from those shortlisted for that is our definition of bug localization tool for this study.

Interviewer2

And "" of the files are ranked, right. Yeah,

Interviewer1

they are ranked

Interviewee

going to ask that. You have something like, top five or top 10? Something like that? Yeah.

Interviewer1

Okay, so given the state of the art of such a tool, or would you use such a technique?

Interviewee

Yes, why not? If accuracy is good. Definitely, yes. Yeah. For unless use case wise, let's say I'm working in a "". Can you please stop the recording for a meet?

Interviewer1

So sounds very we are in the question that given the state of the art, would you use such a technique?

Interviewee

The answer is yes. Yeah. Okay, great.

Interviewer1

Okay. So what do you think will be the benefits of using such a bug localization tool?

Interviewee

Yeah, okay. So first of all, if the codebase is not very familiar to me, okay. And to the team, or let's say we are working on a legacy code base, so that we do not have full control of The or the code base, or the full knowledge of the code base. So in those cases, that will be really helpful if we, if the tool can narrow down the the identification, or an identification of the area, at least to some granularity. Okay, as you mentioned, please to some few few files or at the file level, that'll be great. From that perspective. And even if the code is familiar, when the family call familiar, so most of the bugs comes from the code areas that are I mean, the codes with poor quality. Okay, those are hard to understand and, and difficult to maintain maintain. So in those cases, these will we still have that, because, because those areas are hard to understand, and hard to identify for the developers. In those cases, those these will be helpful as well.

Interviewer1

Okay, and what do you think will be the harm for using those tools?

Interviewer2

Like, what are some challenges? Like, what are some problems that you see that may arise from using

Interviewee

this tool? Yeah, one thing might arise if the developers get dependent on and, and they don't. So it's like, like, make this one to zero. And it works. And or make this zero to one? And then it works. So fixing things like a hack, by not understanding the whole thing? If, yeah, so So let's say that the, the tool, identify the specific component. So it's actually short circuiting some knowledge to developer. Okay. So instead of going through the full identification process, let's say there are some communication involved. There are some messaging messaging pool involved, or some other things. So how these get diverted to that specific component. So that's knowledge developer won't have that knowledge. The tool will short circuit that knowledge. So that's one thing I can. Yeah, I can tell you right now. Maybe there are others?

Interviewer1

Yeah. So have you ever used any bug localization tool?

Interviewee

No.

Interviewer1

Okay, so then what kind of tool do you use for debugging? Like if you have a bug report? So what is your debugging process right now?

Interviewee

First state that the first whenever the bug comes, so we tried to find them or whatever the part of the proof we have, and whatever the other related components and parts that we have listed that we we tried to get the screenshot at specific time what clocks we generated? Okay, what if the determines is related to what what happened to specific that on that time? So those relevant stuff we try to gather and try to reproduce the stuff, okay. And, and based on those, those logs and other proofs, we try to narrow down the specific component. So let's say for example, if it's a software, or the services, if it's a micro services component, and there are a lot of services involved. So we first try to narrow down which service or which component is primarily involved. Okay, from there, and then we try, try to go to the deeper, let's say, find out which class level or which method is responsible for shooting the bug.

Interviewer1

Okay, so this is a fully manual process, you don't use any kind of automatic tool, like static analyzer or any kind of tool in this process,

Interviewee

right? No, no. Okay. But we do use like a debugger from the IDE, right?

Definitely. Definitely. Yeah. So

Interviewer2

including those kinds of tools like what kind of automation what kind of tools do you use when you are debugging like a debugger? What else would you use?

Interviewee

This agenda for, for our to maintain our code quality. So what we do is we pay I mean, we write the unit test. Okay. So so that's our general practice that. So that bug has been generated, because because those are not covered by any test, or the integration test, or the end to end tests. So that was a gap in our process, right? That's the reason that Mark has been entered. So so in that case, though, all through all those tests have failed to identify these, this specific back. So after fixing the back part, we the first thing we do is covered that specific work with an automated process. If it's it can be covered in the unit test, we already didn't just, it's not part of the unit test, and it has to be done as part of integration test. So it covered it with integration test. So that's the process we generally go through. Okay,

Interviewer2

and what kind of debugger would you be using? Like when you use debuggers? What kind of debugger would you use?

Interviewee

So okay, okay. Can we be like, more specific, like, like the news? Whatever the language is? Yeah.

Interviewer2

So like something like something that is built in into the IDE? Or do you use like, just like print statements? Or do you use to actually do things?

Interviewee

Yeah, so actually, these these cases depends. Okay. So let's say if, if, if, if this has been generated from the plus side, so front and back completely generated from from the browser, and we can debug it over there. So and if we have a source code, source code mapping over there, we can write to it debug, not to fix but to debug the issues over there. That's possible, right. And if it's a backend court, then our general preference is first to identify the the issue by going to the court debugging is generally our last option on those cases, okay, so, so even going to the court, if we can't understand the bug, then we'll use if you if you say between the code debugging, and about like to breakpoint or the print statement, the print statement will go first, or the logs. Okay, the logs goes first. So we try to generate more logs. And we'll try to see what they are. Because sometimes it happens that the thing the bug has been is, is produced in production environment, but it's not happening in staging or in our testing. And you know, that it's hard to reproduce this bugs in different because of the differences in environment it is hard reproduced. So, so after the code revision, the next thing will be a print statement. Okay, if you can, our our to our to, I mean, more aggressive logging, or change the mood of the logging, from from info to debug or auto trace mode, something like this. Okay. And, in our practices, some of for in our some of our tools does have the tracing and about. So, that actually helps, to some extent to identify so by tracing, I mean that in AWS perspectives, we have used x ray, or jagger for the totally the company's stackers. So those tools actually help identifying the, at least to some extent, to a component level. Okay.

Interviewer2

Okay. Thank you so much.

Interviewer1

Yeah. So our next question is, do you think there is a lack of feature of the current debugging tools or you think that if there's some features in the debugging tools that may increase the productivity of the developers

Interviewee

Um, honestly speaking, I truly am not a good user of debugger. Okay, I actually useless in debugging. So that's one of my shortcomings, maybe. But so I don't have any good suggestion over here. Because I, myself, don't use that debugging.

When you might even I mean, I mean by sorry, starting by debating, I mean, the I mean, debugging through through points or something like this. Okay. But other things that the practices that I've mentioned, that I've already gone through.

Interviewer2

So not a debugging tool, but you said one of the practices that you do is you go through the code to kind of see why it's behaving that way, would some kind of automation or some kind of tooling at those stages help? Like there's a lot of things that you described? If you do manually? Would any of those steps you feel you would benefit? With automation?

Okay. So you mentioned how to reproduce how to write unit tests, how to go through the code, if I'm remember, correct three things right. So first, is reproducing, the second is going through the code, and then actually fixing the cord. And fourth is then writing a unit test to see that it is fixed. So four activities that you have, is there value for automation in any of the four? Or should humans be the ones doing all you need to

Interviewee

know? Oh, that's, that's definitely tools will tools can definitely play roles over here that identify the as I said, at the beginning, that identifying the specific component, and based on whatever the criteria that what, what, what could be the source of the the actual bug?

I suppose sounds might help if, based on a bug, and automatic test is generated and which is fading. So look at that test, maybe it will be easier to find out the bug, will it help?

Interviewer2

Or automated reproduction? Like looking at the stack trace or something and then automatically we reproduce the bug?

Interviewee

is definitely that will definitely help. Yes. That's kind of your implementation part. So it was kind of I was skipping that specific thing, actually. Yeah, definitely. That will help. Yep.

Interviewer1

Okay, great. So our next question is, according to your perception, what are the minimum required functionalities of a bug localization tool should have, like it can just rank it can currently just rank the potentially bug files, do you think, there are other kinds of functionalities that are required for developers?

Interviewee

Generally up to Rosburg. So

Interviewer1

certainly it currently bug location tool can just rank potentially buggy according to their severity, like there is 80% probability with a first file in the rank list is the main reason for thr bug. So that is the return output of a bug localization tool. So do you think that is enough or any other kinds of features is required for developers

Interviewee

developers? Again, it depends on the code base. Sometimes, if the code is not well written, methods can become very big. And the classes and it depends that how So since you're saying it's in the file level. So again, how big the file is. So I want to say if for those extreme cases, if it goes to the class level, or the method level, and identifies their probability as well, that will definitely help.

Interviewer2

And is there anything else? Besides the granularity? Is there any other information that we can provide about the bug or about the file that would help?

Interviewee

Okay, one example can be like, which was the last commit when this bug was introduced?

Interviewer2

Would you like to see like, similar bugs or similar? Files? Would that help as well?

Interviewee

Similar, you're seeing a previous history of

Interviewer2

Yeah. With similar bugs, we also return a set of similar bugs that were fixed in the past and where they were fixed. Would that help? Would it help? If

Interviewee

yes,

Interviewer2

we tell you why we think this file is the relevant file. The reason? The reason?

Interviewee

Yeah, yes. Yes. Oh, this will help the future. Okay,

Interviewer1

so what do you think is the lowest acceptable performance of a bug localization tool? Like, it can return that top five, top five most potentially buggy files, but it all it also can return the top 10 files? Like what is the maximum number of files you think is just too much? Like the tools accuracy currently decreases, if you like, decrease the number of top five, like in top five, it can predict in, say 90% accuracy, but in top 10 it can predict in 100% accuracy. So do think, what is the maximum number of files it should return?

Interviewee

I think five is a good number. Beyond that, no. five is five, five is good.

Interviewer1

Okay, and in terms of like, latency of the tools, like these tools actually send matters or your source code like it can be very large source code based with the bug report. So it takes some time. So what is expected latency of such a tool ? Should it return the list in like 10 minutes?

Interviewee

Few minutes? Two minutes?

Three minutes. Okay, but however, again, I mean, there are no actually straightforward answer. If it's a code base of millions of code, or hundreds of 1000s of lines of code, then some bigger number I will definitely accept. And if we don't have any clue that why it's originated from the definitely, even if it takes more time. 10 minutes, 15 minutes successful. Interest a huge reach from this.

Interviewer2

Yeah, but something like a day would not be acceptable.

Interviewee

Right? Not at all. We on to that will never wait for that long. Okay, but what happened that I'm, I'm seeing from that perspective that maybe it can run parallely the manual process will start I mean, the manual, we can't hold the manual effort for that.

Interviewer2

Can I ask a side question here, which is, on average, or the last bug that you fixed? How long do you take to find the not actually fix the problem, but find where the problem is? How long would you normally take?

Interviewee

It's very difficult to come up with the number. So even the last forever average number, so

if we're still over the last first there was one yml file I was working with to actually 15 minutes. 15 minutes 10 To 10 to 15 minutes. Okay. But I have a very yeah, there's just the statement is going to say that I have struggled with bug that took more than a couple of days. Okay, just to clarify.

So today's

that involved infrastructure. Code, not the code actually. The infrastructure, okay. Yeah.

Interviewer2

So at the high end, maybe a couple of days, but normally 15 minutes to an hour is

Interviewee

an hour,

Interviewer2

an hour. Okay.

Interviewer1

Okay, so what Do you think that a bug localization tool will be useful for like it is best for newcomers or experienced people or in role based like it is for software developers or it is for it is it will be best suited for QA?

Interviewee

QA engineers firstly, level it can help the support team. Okay. First of all, they can take advantage of team that, if so, it depends how the system is run, if the initial support is given by a support team, not the actual core developers, and if it's something that that the tool can identify and the support interface, it means lesser cost, right. So, so most fruitfully to the support team and, and then to junior developers, or to say, a new developer who is in the team and has less knowledge of the codebase So, yeah,

Interviewer1

so, what types of bugs would this be useful for, like it should focus on the easy or the difficult ones

Interviewer2

before that just a follow up to the previous for new QA developers when they use do you foresee any challenges that they would face because of using this tool? Now challenges in using but challenges because of using Do you think there'll be some problems they'll face because of using this tool?

Interviewee

One thing I mentioned that that's the after effect that in my salon throughout my career, I have learned the systems I mean like frameworks you can go learn those from tutorials right. But you have to know a system you have to use it and fixing and tracing bug all the by tracing, I mean the manual tracing, tracing and well from its widest origin I mean from the from the user to the root root cause it gives you the total understanding of the system. Okay, so before you go to the actual code development, it gives you the confidence and better knowledge of the system that will that has one after effect they will face

Interviewer2

Do you think that with that kind of an effect would this be better for more experienced developers then because they already know the system

Interviewee

Yes, from their perspective, it's true

Interviewer2

that then experienced people may not

Interviewee

be but men have needed that that but even if the experienced one I mean if that the better patterns that you're saying that if it can be narrowed down and it shows some the similar thing happened before and that's the way to fix it. Something has to happen so I'm the similar things happen in many cases. One bug produces another bug, the fix of on bug produces. So sometimes there are patterns in those things. So in those way that that can help this experience developer as well as the other features that you mentioned.

Interviewer1

Okay, so I was asking about what types of bug would this tool will be useful for, like, do you think it will be useful for easy bugs or difficult bugs? That includes like multiple systems or multiple deployments or those kinds of difficult bugs?

Interviewee

I think it can be useful for for both of the cases are both based on the use because it can be useful for both cases. One have some some different functionality and easy ones have some other features. So based on the who is fixing it, and what is the situation that type of it can be useful in both cases.

Interviewer1

So Where do you want the output of a bug localization tools, so do you think it should be better if it is in in kind of issue management system, or in the IDE of the developers?

Interviewee

Look for I'm just I'm just thinking about, like for code quality, we are accustomed to see those code quality things. Right into the the code repository management tools. Okay. And also IDE integrated is an integral part of that. So I'll say that. Yeah, I mean, issue management, let's say JIRA, and Bitbucket or Jira, and GitHub is connected. You can trace one on from there. Okay, it's linked from JIRA to GitHub. But I'd like to see that on GitHub, or Bitbucket. And, and there is a, there's a IDE components as part of the IDE. So that actually syncs that data, the same thing that happens with code quality. So you see before, before committing the code, you see the issues on IDE, because we have a you have a plugin for IDE. Okay. And then when you commit that code, it goes to the central repository, the CI is run, and you also see the output over there. That's how I would imagine. Yeah, right.

Interviewer1

Yeah. So previously have said that, like, there should be an explanation why the bug localization tool thinks that a specific file is buggy. So why do you think that that kind of explanation is important? So, do you think that kind of explanation can teach the developer or this anything?

Interviewee

One thing my reasoning helps is the first thing is whether the tool reasoned it the right way. Okay, maybe the tool is looking at the different perspective, which is not correct. Okay. So, why it has identified this, I mean, let's say many things can happen are you going to use the machine learning AI and ML and it can take the decision based on many features. Okay. And, I mean, if the deep learning is in use, the reasoning becomes more difficult. As as I understand, so. So that's one thing whether the perception of the tool and the perception of a human matches. Okay. And And then. And secondly, it will help in fixing the quote actual actual, actual fixing. That's two things I can do right now. Okay.

Interviewer1

So that is the final question. So those, would you ever use a bug localization tool? Like I have as described in this study?

Interviewee

I'm a kind of a person who would like to experiment with many things. So whenever new technology was there, somebody used to, like, take it right away and put into the team that we need to use it right away. Okay. So, I also take those those it's safe to say, general opinion that I'm saying, but from my perspective, I'm 100%. Right. Okay. So,

Interviewer1

would you pay for that kind of tool? Like if that is not free? Or, freemium type of model?

Interviewee

Okay, so, again, since I'm sure you have the record that you were you were discussing a developer in "some country", not in Europe, not the US. And you have to understand that what how much cost they are doing? Let's say that, for our code quality tool, we are only spending 10 to 100$ per month. product. Okay. And that, I mean, the the bill varies based on our users. So I'd like to see a similar pricing for that. For the particular position, maybe depending on the tool, maybe maybe a bit bit higher, depending on the answer, depending on the solution I'm using it for maybe, maybe I'll pay for the for the bug localization tool, if it works better, then I'll pay a little bit higher than the code quality tool.

Interviewer2

So, like, instead of, like you said, but based on the use, right, so maybe based on the number of bug reports that we correctly classify if we suggest a file where it's going to be fixed in and the fixed is actually in that file, then we charge

Interviewee

but how do you know that whether it's done you have to be I mean, you have to rely on the manual input from the

Interviewer2

no because we have access we will need access to the repository and the issues anyway. So we will because otherwise, how would we know where the bug report is in what where the source code is and what the source code has and everything right.

Interviewee

I will say that it truly be based on accuracy is one thing okay. I will say that tool is widely adopted, if it's I mean if the accuracy is good, okay. Otherwise no one will try to use it. Yeah. Yeah. Yeah. So, I will say that said that it has some kind of units Okay, some kind of just given comparison, what is the similar thing? Probably Bitbucket Bitbucket are some CI pipelines, they have some concept like this, and how many minutes have been consumed? Okay. So, that can be one parameter that because you have to have to run something right they need an infrastructure and and that that is directly proportional to your, your cost that how many bugs and how many our hours of or how many minutes of analyzer analysis you have to run. So yeah, so that's, that can be one criteria. Okay, let's see it. Is and

Interviewer2

one more thing "name" In the right last answer, so it's a given. Right. So the tool will require access to the repository. And also the bug reports it's a given. So I do think that should be a problem for these kinds of tools like, adaptation. We are using sonar and everything all the time. So do you think it will be a problem? Given the access?

Interviewee

No, I don't think so. So but in that case, you're the issue report. And so so this get to the specifics, like so that we're on the screen, you can better. So your issue management JIRA

has to be connected with GitHub? Because that specific issue? How do you track that? I mean, so you have to develop too many too many tools, too many clients.

Someone is tracking direct tracking the issues on GitHub, some are using JIRA, some using some other tools. And you have to make integrations for every option, every other option.

Interviewer2

We might have, we might just like some of the popular ones. So the source code repository, there's, most of them have to one of two options, right? It's either Bitbucket or GitHub. So likely just those two to begin with, right? And the same way for issue management, GitHub issues and JIRA, right, which is, again, the two popular ones. Right, and then sure, there are going to be others that we won't be able to help with. But this would capture a lot of people working on this. The question is that, would you be okay with giving us access to your JIRA repository and your GitHub repository? If you use that?

Interviewee

That's absolutely fine. I don't see an issue.

Interviewer2

And this access, let's say that the computation occurs in your computer. That's one thing. But if you give access to us read access, no write access, read access,

Interviewee

yes, we this is fine here.

Interviewer2

And then the computation happens remotely on the cloud. Would that be okay? So?

Interviewee

Yes. Okay.

I see it as a cloud too. Okay.

Interviewer2

Okay, perfect. Thank you so much.

Interviewer1

Thank you. Sounds great. So that was our final question. Do you have any question for us?

Interviewee

Nothing right now. But I mean, the questions that you asked her so that answers that have given us I mean, kind of instantaneous. So if I have any more input, can I send it to you?

Interviewer2

Yeah. Absolutely. So one of the things that I think we can stop recording and kind of explain the process