

P7-1

Sun, Jul 21, 2024 2:13AM • 36:14

SUMMARY KEYWORDS

test, automate, requirements, software, data, agile, test cases, project, frameworks, testing, company, api, quality, services, depending, server, methodologies, business, metrics, meetings

00:00

Hmm

00:08

Hi, smile. I'm good to see you. How are you doing?

00:15

We're doing in Colombia.

00:16

Thank you for inviting me for survey. I'm doing great.

00:21

And I'm also doing great. As I mentioned, during our first communication, I was researching. Currently, I'm researching software quality and software development process of financial industry, and securing data quality, basically, in the software development processes. Now, I would like to start recording, then we go through the consent form, which will take like three or four minutes and ask your verbal consent for starting interior. Is it okay for you? Yes, we'll

00:56

discuss I am fine with that.

00:58

Let's go. Like if you have any questions regarding to the interview, please feel free to ask them now or in future as well, you will have an opportunity to ask, like no worries about it. So as I understand,

01:11

as you're explaining, so basically those videos will be deleted later, right?

01:17

Yep, the videos will not be stored, we just record the videos in order to take out the subscription subscription of the text. Basically, that's why we need it. So the consent form basically says that we are the group of researchers of the University of Toronto, which is in Estonia, and I am the student that gets

master's degree there in software engineering. And currently, we are researching the how the software quality is being implemented and modern financial products. That's why I actually reached out to you via LinkedIn. So, during the interview, the basically we have a few requirements for the interview that you should be 18 or older, you already said that you are older, you have experience in financial sector, more than one year have experienced with implementing software quality. So, you do have experience with software quality. Yeah, I

02:27

in general, I have like eight years of experience overall. So as the engineer and out of them basically like two and half years in the banking sector and fintech sector.

02:46

That sounds great. And I see that you're fluent in English more than me even regarding the privacy and confidentiality, will you talk about but I will tell about again. So, we are we the original recordings will not be accessible by the others it will be accessible only by me or my research supervisor and audio recordings will be transcribed and we will use this data to we will aggregate this data and basically

03:22

then record the

03:25

outcomes of the interviews

03:30

by the by participating in, are you okay? If this interview material will be shared with the University of Tartu and they can use it in their further researchers. Yeah,

03:44

I agree everything for science. Let's go

03:50

I agree with you as well.

03:53

So let's go through the interior questions. I will start with like, we have like major seven questions and there are sub questions if we have enough time we can go through them as well. So my first question is, how is the software quality ensured in the project during like early design phases or planning phases? In your experience?

04:24

Well, for for past, like I would say five years I'm working in agile and based on agile and also based on testing principles, there is one principle called early testing. So you need to participate as a test engineer as early as possible. Because the more you wait and when when you find bugs in later stages, let's say in development stage and release stage, the cost of the bug increase over the time. So

that's why you need to participate, ideally in IDEA validation phase. In my case, basically, I'm participating in grooming meetings, giving my recommendations regarding edge cases, based on my previous experience, and like, I'm grooming user stories, I'm adding acceptance criterias. And on the spot, I'm thinking of what could be edge cases based on my previous experience with this and that feature. So they were about the personal things.

05:52

As far as I understood the grooming meetings are the part of the user stories. What about their like requirements gathering? Who is responsible for doing the requirements gathering? Like, are you part of this place or not?

06:09

I'm not responsible for gathering requirements. However, I'm responsible for grooming it, meaning adding more edge cases and some requirements related to some data types, like data sizes, some non functional requirements, like we have a lot of performance testing running around. And so basically, product owner and business hours are responsible for gathering requirements communicating with, with stakeholders, with business people, basically, and deliver and get to us. Sometimes we have some prototype, if there is a UI part. And on, like, tangible prototype, where like, let's say in figma, where I can see what will be UI? And what fields will there be, let's say it's a registration page or some other page. And sometimes

07:19

the acceptance criteria is based on the prototypes as well already.

07:25

Yeah, not me.

07:26

I'm adding. So again, like let's say, a product owner and later business and always create a document basically creates a user story with a prototype and add some user story that okay, we need we need some registration page where like, Okay, we will get first name, second name, address, password, and email, etc. And I will just ask questions that okay, how long? How many characters? Will a name accept? Will it accept only English letters? Or there will be some other letters are long as well like Japanese letters? Or? Or will they? What size of password page? Will it be encrypted? What will be the algorithm for encryption? So I'm giving those questions and by giving those questions I'm helping to add more criterias to acceptance criteria.

08:40

Got it? By the way, do you have like any specific methodologies or frameworks from the beginning of project like Do you have it in your mind or if like, already gets defined from the beginning of the project or how it basically happens well

09:03

so in this company that I'm working right now, I'm working at Xcel which is like one of the biggest payment processors in US it helps to move money from one bank to another and across America, it operates in us and basically,

09:24

we have

09:27

one week sprints and

09:32

so basically, it's a fragile Scrum and however, like in previous company, I also worked in bank which is online banking company founded in Estonia operating in Baltics and Poland. Recently, they added Spain and Germany I guess, as I heard, so there they had on methodology, which was based on a journal. However, basically it was changed methodology like we had additional meetings. Sometimes we had user acceptance testing with business people, which is not usually not happening in, in, in Scrum. So yeah, basically mostly it's scrum Kanban for smaller projects, and also some, because AGL methodologies, they are very flexible. So and different companies has different at different things like configurations to agile, like some companies has strictly 15 minutes, they will stand up. Some other companies has one hour meetings. So it depends on preference, and it's quite agile. That's why it's called HR is more about

10:56

software quality frameworks, like you name it, like, tons of frameworks, like unit testing, what you plan to do beforehand, a project, or is it defined after you start with the project?

11:17

No, usually.

11:20

So we do, I do test

11:22

planning. And basically, before grooming meeting, I have a time to look, look it up, what tickets are going to be discussed on grooming meetings. And so I understand, depending on criterias, I understand which maybe some serve some internal services, some util called frameworks, or I need some other testing frameworks, some new framework. So basically, it's on me. So I definitely I definitely get advices and recommendations from others as well listen to them. And we discuss it what exactly we need, maybe some utilities, maybe like, maybe we need to test our application on different platforms. And we don't need, they don't have our internal tools. So maybe we need to buy let's say browsers that sound for browser compatibility. So depending on need, and depending on requirements, I'm deciding which tools to use with the testing team.

12:43

What about, like, what's the general functional and non functional quality metrics that generally used by your team? Like most used by your team, we can seem that way as well.

13:00

Well, most use team, I would say that it's not team based. It's more like project based. Each service has on SLOs and isolate. And

13:16

basically,

13:19

like, each service, each API has response time. And, like, concurrent thread, like requirements, I say, like, one, let's say there is some endpoint API REST API, and it should handle I'm just guessing, as an example, I'm getting like let's say it's like 1000 calls per second, let's say. So basically, depending

13:57

on the

14:02

server, yeah.

14:06

Yes, it's basically how many requests can be

14:12

initiated per second.

14:15

Can you tell me like how many topics differ from project to project maybe you have in experience some projects that require one set of metrics and other another set and how they were different? Like

14:31

oh, we use

14:33

alpha tools which emulates our codebase and do white, white box testing, like sonar cube, gives you test coverage, code coverage and all those metrics. Okay.

14:52

Basically, ideally,

14:54

we try to have 100% like very close to one one There's percent test coverage, meaning like each role is covered, at least with the unit test. Regarding non functional requirements, it's a bit harder, because usually it's coming from business people. And

15:18

so it can you show me how to do it as well?

15:22

Yeah, it depends. Like,

15:25

it depends on hardware. And also,

15:29

in most cases,

15:32

those requirements are not matching real world scenarios, because first, are usually those metrics, like non functional metrics are changing over time. Because we launched some product, release it, and then we see that we have more users. And requirements should be changing, or maybe less users. So it's mostly done by Yes. DevOps and reliability guys. Yeah, basically, we get requirements from business people as well, based on analysis of real production servers. So it's a bit complicated thing. Like, we gather data over time, there are a lot of analytical metrics, how our software is used, and which services are more important. So that's why a lot we get a lot of real, real time data and this, those requirements formed over time and matching real life scenarios.

16:43

So I know that the financial sector is heavily regulated. And basically, you need to provide concise services without fraud, or at least with like, super low person to approach what critical aspects of software quality are generally prioritized by your company in order to minimize the error?

17:13

In in the software?

17:17

That's right. So yeah. Well, that's your right, we have a lot of compliance, industry compliance, governmental compliance.

17:33

We need to follow our software,

17:36

like our our social need to follow PCI DSS standards as well, like, passwords should be encrypted, and there should be some random data in an encryption algorithm. It's like hacker, let's say, will not be able

to create a rainbow. Like basically based on an encrypted version, it should all be able to decrypt and match. And also all like data. Basically, we have we have our own servers, and in most cases, we try to keep our data

18:22

regarding customers internally.

18:25

And also, if, let's say, Okay, right now in Zell it's, I'm not sure how is it in sell because I'm in another team. But in in bank basically, all bets are supposed to be like in Europe, PCI standards are different. So all data supposed to be only in Europe, servers. And also we have our dates. And in addition, if you're if someone purchase bikes, some item, like let's say someone getting loan and basically, customers need to be able to remove their data if they need so, like let's say there is a card like credit card. And if there is a third party shop, let's say some ecommerce shop like Amazon, which use in bank as authentication and if someone made a purchase with their car, but within bank, we should not store data about transaction more than 15 days. That was the case and also if if there is

19:50

let's say.

19:53

Basically, if there is a request to remove data from from customer, you should Be able to remove old data, even metadata. Yeah, there is definitely a vault error and false. There's differently, like different scalability methods. Like we have duplicate, duplicate databases, duplicate servers, we have our own CDs.

20:31

depending on region,

20:34

some increase the SLO or meet the SLO goals.

20:39

Yeah, that's right. And also, we are caching we have cache servers, caching for API's.

20:49

So what you're caching the response request response lifecycle? Or, yeah, basically,

20:59

like, basically, some of the endpoints which are used the most are stored in cache servers. So that our, like, let's say, a main server is not loaded.

21:15

So

21:17

yeah. Have

21:19

you ever seen the software quality aspect of the project to conflict with the business objectives? And do you have such kind of experience in a previous projects?

21:37

Well, sorry, I didn't get question. Can you explain one more

21:42

example, business objective is setting something in the way that is not feasible for the solution. And it's not possible to ensure the software quality of the solution. And you have conflict with the basically business objectives of the project.

22:08

Oh, in Agile?

22:16

So, basically, ideally,

22:20

I'm sharing my test scenarios beforehand. So let's say in during grooming process, so during grooming process, I already know what tests I'm going to run. So I share my test scenarios with the team. And they also know what exactly I'm going to test and call it beforehand. Yeah, in our case, in most cases, sometimes, yeah, new test cases arise. Yeah, we are reviewing test cases. And the like, we have separate user stories for each test. Because we, I do planning and we negotiate within a team as well, like which test cases should be tested manually and which test cases should be automated? The template, it's all done the we are doing prioritization, depending on how important is this specific feature in the software overall? So we calculate severity, for for specific case. So let's say if, if some specific feature doesn't work, how it's going to affect our customers, right? Is it a huge deal will lead to some failures or like very, very few people will notice it. So we are checking as I am checking severity and probability of some specific bug to occur by test cases and the higher severity and probability of occurrence some specific bug if my test case will find the bug, the more probable that I will automate it and test it. And basically, yeah, we try to not to launch our products. If priority one test cases are not tested and automated. Yeah, so we have we also have our good enough, depending on how testing is going. depending on what's the result, we might get flexible. So there might be scenario Real when like testing is delayed, and we need to deliver is still agile. And I can't say that. Okay, I have tested all priority one and priority two test cases, but I haven't done automation. So it can be negligible, we can launch the product. And then next steps. next sprint, I can start automating it faster. So usually in ideal situation, like priority one, and priority two should be tested, and prior to watch should be automated.

25:39

What about like,

25:41

currently, you're in financial industry? I don't know about the framework tools that you use there. What's the most vital frameworks, tools, methodologies in software quality and not in management in general, you are using in finance right now? What are the workforce? Let's say?

26:03

Well, right now, I must have testing API's. So I have salt, and rest API's. And sorry, so I'm mostly dealing with API testing. And we have testing frameworks. I am modernizing testing framework. So my testing framework, my current testing for our is Spring Boot test. It's Spring Boot application. And there is like integration and unit test by using failsafe library and J Unit. Our I'm transferring our tests to rest assured and test NG. I'm adding I'm unifying tests, and trying to do it more end to end. Because in our scenarios, we have a lot of services, like at least 20 services, and each service has own integration and unit tests. And there are a lot of mocking Mockito is used. And a lot of data manipulation with database, which I don't like, in my opinion, you should do less manipulation as the least manipulation as possible. And more like, do things through endpoint rather than connecting to database and everything. So

27:44

your tests, rest assured or services.

27:48

Yeah. API's rest assured and test Ng, and also fail safe J unit and Spring Boot test. But we're also we have a lot of there are a lot of cufflink performance tests. So, yeah, we have different teams. In many teams, there are they have UI part as well, they use Selenium Selenium Grid. And in in bank, in private company, in previous company in in bank. I also like, basically, I used to cross browser testing.com. It's the same as Browser Stack. And I had the Selenium grid for local cross browser testing. Basically, again, with Selenium, Java test Ng and again, rest assured for API's. Good.

28:52

So there is no special magic background by was the financial industry. It's the software, basically, quality testing frameworks that we use day to day in the tech industry in different other industries. They are still used, but they're basically configured denial I will say

29:15

I will say that yes, but also

29:19

like, for instance, we are using soap UI, which is very soft protocol is very popular for telecommunication and banking, based on my experience, mostly banking and, like telecom companies you recall, more like startups, they mostly use REST API's. So at zaal we have soap and wrists, boss SOAP and REST. So mistaken

29:49

soap is more reliable than the rest. That's why it's used in finance. Or that's not true.

29:58

Yeah, oh, A it's more secure, more and more layers of security. And yeah, however, I prefer rest, I think salt is very old. Yeah. And

30:14

super old school thing. Yeah.

30:19

What about if you're dealing with tests and quality assurance every day? What key advantages or disadvantages you can tell me about automating this metrics, automating the testing?

30:37

Well, basically advantages. So you can use, you can test different permutation. Let's say, I gave you an example with login page. So I can, instead of typing different variation of passwords, in order to check incorrect password, or let's say, if I'm doing SQL injection, trying to do SQL injection, so instead of doing it manually, I can just create a Java class, make it as a data provider for my test. I can create a CSV file, and add a lot of different a lot of variations of username and password. And just use this as a data provider for my test, which will help me to have a small chunk of code and test different variations. So basically, it helps you to test multiple variations. So it brought us manual, enhance manual testing. And what else? Well, it helps you with regression testing. So like, every time you have changed in code, you're not going to test everything manually over and over. It's physically not possible. So that's why automation comes to help. And it helps you to run different tests in pipeline, or locally. And to test like, every aspect of your software, if there is some code changes to be sure that your code change didn't break existing called.

32:33

Got it? What about the limitations that you encountered in automation? Because I'm also a software engineer, I'm seeing so many cases that it's very hard to automate things, or you have different kinds of limitation that stops you from automating?

32:54

Well, yeah, there are mostly based on my experience, the biggest limitation is time. So

33:04

meaning in Agile,

33:09

the project is pushed to launch as fast as possible. And basically, so I need to prioritize which tests to automate, and which not. Ideally, I would love to test all like the most possible scenarios that came to my mind and automate all of them. But in, in practice, it doesn't happen. And other limitation is definitely, there might be performance issues. When you have a lot of tests running in a pipeline, you have more and more tests, and increase the build time. In order to avoid it, you just basically add

different profiles, you create different pipelines for different testing. Let's say there is only pipeline for regression that you're running on the when you are doing regression. And you're just basically group your tests in different tests yields, you have smoke tests, which are necessary to run it every time. And maybe you have regression tests yield that you are running. So basically, and also, like when you are running performance test, and definitely you need strong server performance server capabilities

34:46

to run your tests from CPU.

34:51

Part most of the companies they can afford they just pay for tools.

35:00

We're about like, the companies can afford this kind of tools are they like more planning to invest in or they prefer to save money and energy in that regard.

35:18

In the financial sector, the in most cases, they have money in telecom and banking and content. And they, they want to be sure that their customers get high quality and performance testing is important. So they are ready to pay for more servers and maybe some paid tools as well.

35:49

So they have money and usually they are

35:51

paying for

35:54

service now I have like one minute left, left in zoom. I didn't know that they have time limit. I will stop the meeting and I will add you again. Is it okay for you?

36:06

Yeah, no problem. Yeah.

36:07

Okay. Let's thank you. Thank you.