I’m a quality engineer and I’m not sure how I feel about that

JOEP SCHUURKES

In this experience report I’ll cover my first five months at a scale-up, working as a quality engineer for the first time. My main finding is that for a quality engineering role to work well, certain structures need to be in place. The most important one is that the impact the quality engineer is expected to have, is clear to both the quality engineer and the team(s) they are supporting. However, regardless of which structures you put in place, a quality engineer will also need to work around those structures to be fully effective.

1. INTRODUCTION

In August of ‘22 I took on the role of quality engineer for the first time. The role wasn’t very clearly defined, as it had been introduced just recently to the company. It also wasn’t established yet in the two teams I would be supporting. In the first three months I focused on fulfilling the role as it was described to me. This left me frustrated. In the two months after that I took a different approach. I let go of those expectations and started contributing wherever I could. That period was cut short, as my role changed significantly after layoffs and a reorg. After the reorg, we did take one significant step in creating more clarity around the quality engineering role. I describe that step in the Epilogue of this report.

The main themes of this experience report are:

- trying to bring value to a new and somewhat ill-defined role;
- being asked to support a team, while they don’t feel a clear need for the role;
- positioning testing and QA in an agile context.

Although this report focuses on my experiences as a quality engineer, I believe it’s relevant to all roles with a supporting and coaching focus, such as scrum master and staff engineer.

2. BACKGROUND

Before starting this job as a quality engineer, I’d spent about 16 years working in software development. In 2006 I was someone with a degree in philosophy looking for a job, any job. Someone was willing to hire me as a software tester and that turned out to be a great fit for me. So since then I've been working as a software tester, scrum master, team lead, principal engineer, and consultant.

In March of 2022 I started looking for a new job. I was looking for a hands-on role: work directly with teams, close to the technology. Preferably for a smaller company rather than a corporation and preferably a fully remote position. I found a job matching all those criteria at a company in the Employer of Record (EOR) business. Luckily they were as enthusiastic about hiring me as I was about joining them.

3. MY STORY

3.1 The situation when I joined

The company was a typical scale-up. It was founded in late 2019 and then grew rapidly. I joined in late August of 2022 and was assigned to be the quality engineer of two teams that made up one of several departments in Engineering. Both teams were created a few months earlier, with mostly newly hired team members. One team (let’s call them team A) had four developers; the other team (team B) had three. Each team had their own product manager. The engineering manager of both teams, who was also my manager, had started only one week before me. None of these people I’d be working with, were involved in my hiring

Joep Schuurkes, email: xp2023@joep.slmail.me

process. Neither did they have any experience in working with quality engineers. And I had no experience working as one.

Luckily I was not the first quality engineer who had been hired. There was a principal quality engineer, who had joined the company in 2021. There were also four other quality engineers, all of them had joined earlier in 2022.

3.2 The first three months: September to December

The first 2-3 weeks my main focus was getting onboarded. Quite soon it became clear that the teams, our manager, and I would need to figure out among ourselves what my role of quality engineer would entail. Our principal quality engineer had created some good documentation on quality engineering, including a practical guide on how to get started. The focus of this guide was on facilitating risk sessions and exploratory testing. However, these descriptions didn't really address where a quality engineer was expected to fit in a team's process. Later it became clear that this was the core of the challenge the company had with quality engineers: there was a solid vision, but the execution of that vision was lacking.

3.2.1 Process workshop with team A

To get a better understanding of how I could support team A as a quality engineer, I facilitated a workshop for the team in my third week on the job. My plan was that the team would map out their development process (both activities and artifacts), how they felt about the different parts of the process (attitudes), after which we'd decide what I should start doing (actions).

As the workshop progressed, it became clear that my plan was too ambitious. Mapping out the process took longer than I had anticipated. We didn't get to the attitudes part. We did list quite a few possible actions, which we narrowed down to the following three:

1) perform usability testing on an internal feature, supported by the UX designer;
2) lead the testing of an upcoming epic making changes to the integration between Salesforce and our own platform;
3) pair test with developers.

When the principal quality engineer heard about this workshop, he commented "This feels like something your team might like to share more widely at some point?" To which I replied: "Am not sure how useful that session ended up being, so it might have to be part of a larger story." This captures my feeling about the workshop well. I was disappointed we only scratched the surface.

Of the three actions from the workshop, I only really took action on the second one. For the usability testing, I talked with our UX designer. He explained the process. I filled out a template. Neither of us followed up after that. And the pair testing was forgotten about fairly quickly by everyone. So the one thing I did end up doing, was mapping out the Salesforce - platform integration.

3.2.2 The diagram about the Salesforce - platform integration

Since during the process workshop I was asked to get involved with the Salesforce - platform integration, that's what I did. I had the impression that we missed an overview of which data got synced at which point between Salesforce, our own platform, and our customer support system. So I mapped this out, listed things to test, relevant variations in the data, and the different user roles involved. I also facilitated a session with team A and the team owning the next step in the process, to see if any of our changes would impact the work of the other team. After the session I talked with the QE of the other team. I said I was mostly positive about the workshop, but also had some doubts. Engagement during the workshop was not very high and I felt I was still struggling to connect with the team. She 'welcomed me to the club' and said after six months she was still figuring out the job to a degree.

Apart from the initial sessions of me sharing the diagram with my team and later with both the teams, I'm not aware of any purpose it served afterwards.

3.2.3 The Quality Engineering menu and the Visual QA Checklist

To provide a clearer and more concrete picture of what I could do as a quality engineer, I created a "Quality Engineering menu". It listed a number of activities I could do, collaborate on or support across refinement, development & testing, spikes, and investigating issues. The idea was that people could look through the menu and see what they could ask of me. Or that they would ask: "You say you can do this thing, can you also do this other related thing?" In addition to sharing the menu with teams A and B, I shared it with the other quality engineers.

Responses from both the teams and the quality engineers were positive. However, no one ever used it as a starting point of a conversation or to request me to do a thing listed on the menu.

Around the same time I created a Visual QA Checklist for team B, but also shared it with team A. Again, initial responses were positive, especially from our UX designer. However, thanks to the page analytics feature, I can say it hasn't seen much use since I created it.

3.2.4 Talking with colleagues about their expectations

3.2.4.My manager

During my first 1:1 with my manager, he mentioned he wasn't sure if he was the right person to manage me and that the principal engineer might be a better fit. This foreshadowed a reorg and a quality strategy that would be hinted at in the next five months, but that never materialized. I wasn't bothered by his remark. He had a valid point and I like to think of myself as someone who does not need a lot of management.

During our 1:1s he also regularly asked about how quality engineers align across the Engineering department. If there is an overall strategy or approach we were following. I never was able to give him a better answer than that we align informally and I'm supposed to focus on facilitating risk sessions and exploratory testing. I also shared the Notion pages and Miro boards our principal quality engineer had created. These were very vision-focused, so had limited value.

Something that did change during our 1:1s, is that my manager started to appreciate me and my observations more and more. We noticed a lot of the same things, for example that a project impacting three systems was being run more like three separate projects.

3.2.4.My manager's manager

About two months (mid-November) in, I had a meeting with my manager's manager, as she wanted to check in on how I was doing. To me that seemed like a perfect opportunity to ask her what impact she was expecting of me as a quality engineer. Unfortunately I didn't get a very clear answer. She didn't have very specific expectations. She did mention our stakeholders value sustainable deliveries (sustainable both in pace and in quality) and predictability. I also mentioned I felt I was still very much searching and trying to find my way into the teams. And that I felt I should already be doing more. She replied that I was being humble and doing well. About two months later we had another call, where she mentioned it was important for me to speak up about things I noticed and to push for things I wanted.

3.2.4.The principal quality engineer

Every two weeks I'd have a 1-on-1 with our principal quality engineer. He was very enthusiastic about my approach of trying to figure out with the teams what a quality engineer could do for them. We had great conversations about quality engineering and he was very supportive. That also allowed me to express my frustration about finding my place within the two teams. He said things would get better once the new quality strategy would be in place. In the meantime he encouraged all the quality engineers to share and collaborate, to inspire and learn from each other. He did a great job at this and it did make me feel I was part of an informal team of quality engineers.

3.2.4.Being challenged by one of the product managers

Near the end of November, after I had found a small bug on production, the product manager of the team suggested on Slack that she, the UX designer and I should test features before they were released. I replied that I didn't think we should become the developers' test team. That would result in the developers depending on us for some of the testing. She responded that she appreciates the coaching stance of the quality engineers, but that it's also important to get your hands dirty and contribute directly.

We continued the conversation over a call, clarified our positions with each other, and realized they weren't that far apart after all. We agreed to bring it up with the team in the next retro. There everyone agreed developers should request feedback from the designer, product manager and myself, when they submit a PR. That may have happened a few times, but it never became part of the team's way of working.

3.2.5 Looking back on the first three months

At the end of November I wrote a blog post “Three lessons after three months of quality engineering”, reflecting on my experiences so far. The three lessons were three things I had learned to be important: visibility, connections, and patience. I was struggling with each of the three. And that didn’t really change over the course of December.

So by the end of the year, I was quite frustrated and demotivated. I had tried different approaches to figure out my role of quality engineer. I had asked people directly. I had asked through a process workshop. I had shown I could make diagrams to inform design and testing. I had shared a list of things for people to ask me to do. And I had also told myself to be patient, to temper my too high expectations of myself. And I was surprised how my colleagues seemed to be positive about what I was doing, because I wasn’t.

3.3 The next two months: January to February

After a week off because of the end-of-year holidays, I noticed that I was able to let go a bit more. I was feeling less frustrated. And my goal shifted from being a good quality engineer, doing the things that were expected of the role, to contributing in any way I could.

It would have been interesting to see how that change in my attitude would play out over the first few months of the year, but that was not to be. On 1 February layoffs and a reorg were announced - with the reorg going into effect on 6 March. In this section I’ll cover some of the work I did with the two teams in January and February. In the next section I’ll talk more about the effect the announcement and the reorg had.

3.3.1 Spike on test automation for Salesforce integration

The work on the Salesforce integration made me realize that some additional automation could help the team. Testing required you to create several entities in Salesforce (client, contacts, product, opportunity), then the opportunity moved through a workflow. Since both Salesforce and our own platform have APIs, I figured it should be possible to cover the basic scenarios with automated tests using those APIs.

It took some effort, but I managed to build an end-to-end API test proof of concept, covering the process from a client becoming a prospect to closing the deal. Unfortunately I got this done right before the layoff announcements. With the impact of those and the news that after the reorg the teams I was supporting, would be split up, things never progressed beyond my proof of concept.

3.3.2 Pointing out the obvious in team B’s retro

Ever since I started, my focus had been on team A. They had been working on the important Salesforce integration epic and they were still forming as a team. Team B, on the other hand, didn’t have a strong need for me. They had been doing well, working on a smaller and less risky epic. So apart from sharing my Quality Engineering Menu and the Visual QA Checklist, I hadn’t been very involved with team B.

I still wanted to contribute to the team, so during a retro I mentioned how little the team was utilizing me. I’m quite sure the team picked up on some of the underlying emotions on my side. This resulted in two responses, one from one of the senior developers, one from the product manager. The senior developer said it would be great if I could come up with some team quality and health metrics, preferable quantitative ones. We had a good conversation about those, but it felt like too big and complex a task to me. The product manager made an active effort to involve me in testing new invoice reporting functionality. That went quite well, but it was also a one-off, not a step towards involving me more structurally.

3.3.3 Helping team A with test scenarios

In January it became clear that team A would benefit from some more thorough testing. That was partially due to the issues I found by doing some brief exploratory testing on some recently released features. So my manager decided he would set up a meeting with the team to discuss how to use test scenarios. Unfortunately it took him several weeks to schedule this meeting. I let that happen, reasoning that if the session was important, it was up to him to make it happen. By the time the meeting did take place, it was the day after the announcement about the layoffs and the reorg.

It was still a good session. We agreed that I would make sure test scenarios would be created, and that the developers were responsible for executing them and for sharing a brief report. And we explicitly mentioned pairing as a good option for both the creation as the execution of the test scenarios.

Unfortunately nothing much happened afterwards. I created test scenarios for two user stories and shared them with the team. And that was it. No one mentioned using them, no one asked about them during standup, no one gave me feedback about the test scenarios not being useful. I kept quiet too. I didn't have the energy anymore.

3.3.4 Looking back on this period

Not focusing as much on getting the risk sessions and exploratory testing going, and instead focusing on making myself useful, did certainly help me. The layoffs and reorg made for a weird ending of this period. For instance, I completed my test automation proof of concept, but any follow-up was on hold. I also didn't make any real progress in finding my place in the two teams. I had pointed out areas where I could help team A improve, but the lack of response gave me the impression everyone was ok with how things were.

I'm not sure what would have happened without the layoffs and reorg. I suspect I would have continued trying to make myself useful, until someone - either someone else or myself - would point out that I was working more in the vicinity of the teams than with the teams.

3.4 The layoffs and the reorg

3.4.1 The announcements

On 1 February layoffs were announced. Sales in the last quarter of the previous year had not been as high as anticipated, so layoffs were one of several cost-cutting measures. I got to stay, but one other quality engineer and our principal quality engineer were not as lucky.

With the layoffs also came a reorg. It would go into effect on 6 March. Both of the teams I was supporting, would be broken up and spread across different teams in Engineering - although none of us had any idea yet which teams that would be.

This changed things dramatically. Everyone needed some time to recover. Working on my role in the teams didn't seem very useful, since those two teams would cease to exist within a month. The main thing I did in this period, was trying to fix some tests that had been disabled. It ended up being quite a learning experience. What initially seemed a simple task, became rather complicated, as I found myself in the middle of quite some technical debt. In the end, with a lot of help from a senior engineer, I managed to merge my changes at the end of March.

3.4.2 My new role after the reorg

A week after the announcements, the Director of Engineering scheduled a call with me. He asked if I'd want to join the Engineering Effectiveness team. It would be a slimmed down version of the team that in the original reorg plan would be led by our principal quality engineer. Apart from me the team would consist of a senior and a junior infrastructure engineer. Our scope would be quite broad, as illustrated by the team name: Engineering Effectiveness. Additionally I'd support the other teams in the Platform department as a quality engineer. Thirdly, as I'd be the most senior quality engineer, I should spend some of my time supporting the other quality engineers.

After sleeping on it, I decided to accept his offer. It was a choice based more on intuition than anything else, but it turned out to be a good decision. Once the new team started, I noticed I had more energy than in the months before.

About a week later, during a 1-to-1 my manager said I was high on everyone's list to be the quality engineer for their team and that he'd like to keep working with me. I replied that I'd enjoy working more with him as well, but that I had decided to join the Engineering Effectiveness team. It made for a weird ending of this period. On the one hand, I had the feeling I hadn't achieved much with the two teams I had been working with. On the other hand, people seemed very positive about the things I had been doing.

3.5 Epilogue

In April, so one month after the reorg came into effect, my manager and I started talking about how to deploy the quality engineers more effectively. At one point in our second conversation he paused for a second and said: "It's a leadership problem, isn't it?" I wasn't sure what to say, so I didn't say anything. It did feel like a breakthrough. The one thing I had been missing, was a clear statement about what impact I was supposed to have as a quality engineer. And my manager seemed to have just realized this too.

So he asked me to come up with three to five things we need in Engineering and for which quality engineers are best suited to make an impact. Together with the other quality engineers, we came up with four challenges and which activities by quality engineers could make a difference in those. I then presented this to the heads and directors of engineering. The idea was that they could use this in the conversations with their engineering managers.

Unfortunately, a bit more than a week after my presentation, a second round of layoffs was announced. I was laid off and so were the other quality engineers. So while I felt that the presentation was a significant step in the right direction, it was also the last step.

4. WHAT I'VE LEARNED

4.1 Personal reflections

My overall feeling about my first six months as a quality engineer is that I failed. I failed to do what I was asked to do, which was to facilitate risk sessions and exploratory testing for the two teams. I wasn't able to make quality engineering happen for those teams. I struggled to create opportunities where I could show my value and to then fully utilize those opportunities.

On the other hand, part of me does realize the circumstances were very challenging. I was introduced to the teams based on a vision on quality engineering that was not theirs. There was no clear need I was meant to address and I had no intention to push for things if they wouldn't meet a need actually experienced by the team. What got to me the most was that I was getting very little energy back compared to the energy I was putting in, not even negative energy.

So another way of looking at it, was that I took cues from my environment and adapted accordingly. I let go of that initial mission of risk sessions and exploratory testing and instead found other ways to make myself useful. And people clearly appreciated me doing so.

Looking back, the main thing that struck me are all the conversations that never took place. Conversations with the teams, my manager, my manager's manager, the product managers. We all had good intentions, we were all trying our best, but we never got a real dialogue going about quality engineering, about where I fit in. If there's one lesson I take away from this, it's that one.

4.2 Quality engineer reflections

Positioning the quality engineering role in an organization is not easy. It's a position of influence, not authority. Its scope is the whole software development process. This means there are no specific process steps you can point to and say "This is where quality engineering happens." A quality engineer needs to be close enough to the work to have a positive effect on it, without making the team dependent on them for part of the work. In that sense, "quality engineer" can be seen as a rebranding of "agile tester", trying to work in an actually agile way, instead of being the testing bottleneck in a sprint of mini-waterfalls.

The open-ended nature of what a quality engineer does and where in the process, makes it challenging for teams to start working with one. If they have a need that's easily coupled with what a quality engineer brings, they're off to a good start. If they don't have such a need identified, you end up in a situation where the quality engineer says "Tell me where I can help." and the team replies: "Tell us where you could help." The quality engineer will first need to 'sell the problem' (you're actually not doing well enough), to then be able 'sell the solution', i.e. their services. I don't think that's a fair expectation towards quality engineers.

So this is where management plays a crucial role. If the team itself hasn't set any goals the quality engineer can contribute to, the team's manager, or the manager's manager will have to step in. They have the appropriate authority to decide what impact the quality engineer is expected to have on how the team works. Of course, that should be done through dialogue and collaboration, but if you send a quality engineer to a team with no more than "do good things with the team", results will vary dramatically between teams and quality engineers.

4.3 Reflections on testing-related roles

My experiences as a quality engineer also made me reflect on testing-related roles in general. Throughout my career I've been through quite a few of them: tester in a separate testing team, tester in an Agile team, principal test engineer, and as described in this experience report, quality engineer.

I've come to the conclusion that they are all inherently unstable. They require a constant balancing act across three axes:

- doing testing yourself versus supporting testing by others;

- being in a separate team versus being embedded in a team;
- doing your job versus improving the system.

It's very hard to support others testing without doing any testing yourself. Yet any testing you do, might become your domain, with others leaving it to you to test. It's great to be in the same team as the developers and designers and business analysts. It also gets lonely, so you form an informal team with people in a similar role as yours. It's important to do your job, but to do a good job you find you have to change the organization a little first.

So even though your official job description puts you clearly on one end of each axis - for example my quality engineering role was supporting testing by others, being embedded in a team, and doing my job -, you can only make that work by constantly observing and adjusting your actual position on these three axes. You need to work as much inside the existing structures, as around them. That makes these roles challenging and interesting, but often enough also frustrating. And yet, I wouldn't want it any other way.

5. ACKNOWLEDGEMENTS

First of all I want to thank Jutta Eckstein, my XP2023 shepherd, for guiding me through the writing process with helpful comments and regular encouragement.

I also want to thank Elizabeth Zagroba and Chris Chant for their thoughtful reviews.

Finally a big shoutout to all my colleagues during my period as quality engineer. This experience report may mostly focus on my frustrations with the role, but I truly enjoyed working with you all and was sad that time was cut short.