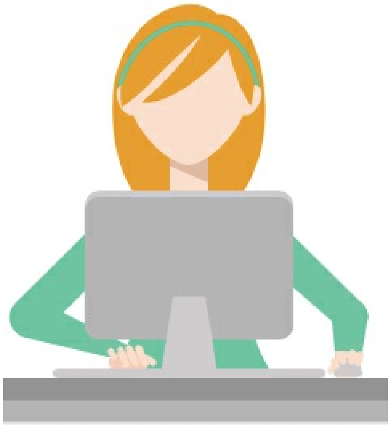
Role of a Quality Assurance Engineer (QA)

by Devmountain Tutorials

WHAT'S COVERED

This section will explore the role of Quality Assurance Engineer (QA) by discussing:

1. QUALITY MINDSET
2. HOW A QA ENGINEER DOES THE WORK
3. DAILY WORK OF A QA ENGINEER
4. QA IN THE STAGES OF THE SOFTWARE DEVELOPMENT LIFE CYCLE



Quality Assurance Engineer

Hi, I'm Amanda. I'm the Quality Assurance (QA) Engineer for Poodle Jumper and I've been working here for a year and a half. I started as a customer service representative and worked closely with the engineers to solve issues before I was promoted. Quality Assurance is a great entry-level role in IT. The core of quality assurance is a mindset more than anything else, and it can be a great place to learn more about coding and development. When I was young, I always pointed out the broken or weird things in our house, like the door jamb that had a gap or bottle that didn't sit flat. It made my mom crazy, but that's just how I've always been. I love that I get to play detective by digging for problems or inconsistencies in our product to help make it better for our customers. I'm a Poodle Jumper customer too, and I love being able to use our service for my dog, Watson.

1. QUALITY MINDSET

Understanding the role of a QA Engineer might seem straightforward. We are responsible for testing software to ensure a good experience for our users and raising the team's confidence in the quality of our software. We don't just test or point out all the problems at the end; we get to contribute throughout the whole process by asking questions, raising risks, and making sure everything lines up.

At Poodle Jumper, or really at any software development company, things don't usually go right the first time. It's my job as the QA Engineer to poke around the software and make sure things work as expected. I investigate risky areas, find problems, log issues, and collaborate with the developers to uncover what is *really* causing the problem.

☆ BIG IDEA

Sometimes problems are simple; other times, there is a reason that isn't so obvious. Finding the root cause of a problem with software is called root cause analysis.

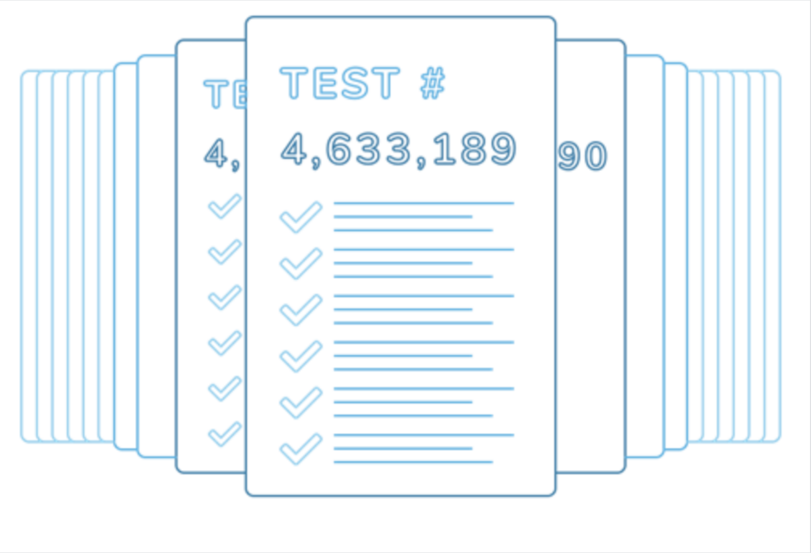
The other day I found a link in our web application that wasn't working. Reuben, my team's web developer, is really practical and was happy that I let him know. After I found a second broken link, I did a little digging and realized that our design didn't explain some of the user flow very well. I asked Mori, UX) to walk me through it and we found a gap that hasn't been resolved. We added a few notes to the design document so that we didn't lose track. This collaboration is helpful for everyone, including Ruben (Web Dev), who will also have a better understanding of what need needs to develop.

HINT

In software, the terms bug, defect, issue, or problem can be used interchangeably to mean something that is unexpected or not working correctly. The first versions of computers were so large they had real insects get caught in the electronics and cause problems. The term has been attributed to programming pioneer, Grace Hopper, who described an incident where a technician pulled moth from a computer.

2. HOW A QA ENGINEER DOES THE WORK

Working as a QA Engineer you'll spend a good amount of time using the software to see how it behaves. It's important to understand how our customers use the product so I test different variations. I spend time reviewing documentation, asking questions, or talking to our users. The earlier we find or prevent problems the better and my role is key.



☆ BIG IDEA

Exhaustive testing (testing every possible combination of inputs and outputs of software) is impossible, or at least highly impractical. Instead, a QA Engineer need to work smarter rather than harder to test as efficiently as they can. Productivity is crucial, so you have to decide what are key issues and what are distractions.

Did you know that in2014 Forbes Magazine rated QA Engineering as #2 happiest job? And in2018 CareerBliss also named QA Analyst as the #2 happiest job? The factors they considered included work environment, growth opportunities, compensation, company culture, daily tasks, reputation, and relationships!

3. DAILY WORK OF A QA ENGINEER

Something that I really appreciate about my day-to-day work is how much it varies. Sure, there are some patterns to it, but the work I'm doing on any day frequently depends on where we are in the software development lifecycle.

One of the first things I do every day is a check-in with my team. This is a meeting where we talk about what we worked on the day before, what we're planning to do that day, and then any roadblocks we've run into. This is a great way for me to know what everyone is doing and where I can help.

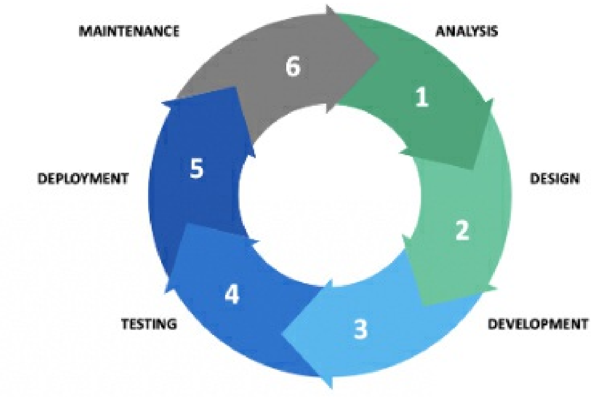
HINT

If a developer is running into problems with a particular task, I know there's probably more risk of lingering issues in that part of the project. If someone is about to finish a major feature set, then I know that I should make sure I'm ready to test it.

? DID YOU KNOW

Asking questions when it seems like the plans are missing important details is the best thing any team member can do while planning new features.

4. QA IN THE STAGES OF THE SOFTWARE DEVELOPMENT LIFE CYCLE



Software Development Lifecycle

Early in an iteration, the developers start working on new code and there are fewer things demanding my immediate attention, but that doesn't mean I don't have work to do. I write test cases based on the requirements and details from the developer. The test case specifies the inputs, conditions, procedures to test, and the expected results for a feature. This helps me structure the test plan that I'll execute when the code is complete. I've started to write code to automate portions of my test. Because I've just started out, I'm not very fast at automation but I'm getting better the more I practice. Learning automation is one of my favorite parts of the job!

In the middle of an iteration, development tasks start trickling to my queue as developers complete them. Testing something for the first time is called **initial testing**. It is likely I'll come across some problems. Software is complex, and it doesn't always work the first time. Depending on the problem, I might send a task back to the developer with notes on what to fix or report a new bug for them to fix in the future. Whatever the problem I find, it's important to make sure we're having the right conversations to address it appropriately.

As developers are wrapping up their development for the iteration, it's time to get ready for delivery! Most of our initial testing passed. I'll retest any tasks that were sent back to the developer; this is known as **confirmation testing**. It gives me a chance to make sure that everything was fixed and that it was fixed completely. It's not unusual for miscommunication to occur around bugs, so it's important to be thorough here. When features don't work as expected, it is an opportunity to build relationships with the developers. I need to share what I've found in a non-judgmental way. Sometimes the issue is on my end with how I configured my test environment, but sometimes there is an issue with the code that the developer needs to fix it. Either way, assuming the best intentions and approaching the problem with an open mind will help us fix the problem more effectively.

HINT

Remember, the worst thing you can do in quality assurance work is to keep your mouth shut–ask questions!

Last but not least, is **regression testing**. Regression testing is when I verify the major features or functionality of the product, not just what we developed in this iteration. This is important to make sure we didn't accidentally break something. This is where my automation comes in handy because it reduces the amount of manual testing that I need to complete every iteration. Automation can't catch everything, but it helps speed up the process.

TERMS TO KNOW

Initial testing
Testing something for the first time.

Confirmation testing
Retesting any tasks that were sent back to the developer to make sure that everything is fixed completely.

Regression testing
Verifying the major features or functionality of the whole product, not just the development on one iteration. It is important to make sure that nothing was accidentally broken.

TERMS TO KNOW

Confirmation testing
Retesting any tasks that were sent back to the developer to make sure that everything is fixed completely.

Initial testing
Testing something for the first time.

Regression testing
Verifying the major features or functionality of the whole product, not just the development on one iteration. It is important to make sure that nothing was accidentally broken.