In the field of software engineering “full-stack developer” is a known term used for a software engineer who is dealing with various layers of software and performs diverse technical tasks. Designing and developing backend, front-end, build & deployment, DevOps & monitoring are few of them to mention.

<img src="/img/NadeeshaQE1.jpg" width="640" height="480" />

This kind of concept is required for QA Engineers as well…...!

Once Upon a time we had a segregation like functional testers, performance testers, security testers & test automation specialists etc. Gone are those days, the fast moving industry trends require QA Engineers who expand their abilities and play different roles to deliver a quality product. Ultimately deal with various stacks related to quality engineering and help to release product faster with quality. In full stack quality engineering, the tester should be able to perform functional testing, performance testing, back-end(API)testing etc.

Simply full stack quality engineers should have the capability of working on all aspects of quality across all the application layers using different testing methods. They should think about different aspects of software quality such as usability, functionality, performance, security and automation strategies.

<img src="/img/NadeeshaQE2.jpg" width="640" height="480" />

\*\*Why full stack quality engineering? \*\*

With the adoption of Agile & DevOps culture, a story is not complete without testing. Moreover, the full stack quality engineering focus on proactively looking for defects throughout planning, design & development stages of the software. Ultimately testing becomes a process which focuses on defect prevention rather than defect detection.

Apart from the product quality, it provides with efficiency which means there’ll be less wait time between testing cycles. Reason is, a full stack quality engineers take care of every aspect of the quality scattered across different layers of the software.

\*\*Full stack quality engineering concerns\*\*

\* Test design

\* Systems thinking

\* End user thinking

\* Performance Engineering

-Service level performance

-Client side profiling

-DB performance

\* CI/CD (DevOps mindset)

\* Product expertise

\* Exploratory testing

\* Understand system architecture & design

\* Unit testing & test driven development

\* Code quality & coverage

\* UI & Service level functional & end to end testing

\* Effective communication & leadership

Full stack quality engineering comes with the diversification depend on the application to be tested. Assume the application to be tested has been developed based on the microservices architecture. In order to assure the quality, the tester needs to identify the architecture of the system, dependencies between different services, APIs etc.

\*\*Looking out of the window\*\*

In a recent article I have seen a quote “Testers aren’t judged by the progress of quality of the digital product, instead they are judged by the amount of bug issues they produce.” I don’t want to argue about the above fact but if someone asks me how would you like to be judged as a tester, obviously it should be based on the quality of the digital product. Then the full stack quality engineering is the way to go. We live in the year 2019, agile methodologies have been around for quite some time, many companies have understood by now, that a product is a holistic thing. Security, functionality, service design, performance, localization etc.- these are integral parts of a product which should be considered from the beginning. This concept matters for a full stack tester. Not only from the initial product design to deployment but also after the deployment, the full stack tester has the responsibility towards the system quality.

\*\*Anything that can go wrong will go wrong ???\*\*

If you ask ‘who is the natural enemy of a full stack tester’? the answer would be the Murphy & his infamous law “Anything that can go wrong will go wrong”. Someone who is truly dedicated and responsible for the quality of a software product doesn’t want everything that can go wrong to actually go wrong!!! This is exactly what he/she tries to prevent. Each and every stage of the product life cycle & and each and every layer of the application as well as the system infrastructure, (simply every nook and corners of the project), the full stack tester keeps assuring the quality. You’d say, “Definitely a nosy guy…!!” :D

\*\*Priorities of a full stack tester\*\*

Full stack QA Engineers should have straight priorities.

\* Prevention > Finding

\* Advising > Criticizing

\* Draft checks > Result checks

\* Early on > Later on

\* Perfection > Good enough

Now it's time to explore new ventures of quality engineering by embracing the concept of full stack quality engineering. Please remember this is not about mastering only the breadth but also explore the depth. Go exploring and have fun.

And happy testing!! :)

<img src="/img/NadeeshaQE3.jpg" width="640" height="480" />