**MANUAL TESTING: WANTED DEAD OR ALIVE**

*TEST Magazine Journalist, Leah Alger, approaches a number of software testing assets to find that software testing is “not dead – it’s alive with new approaches”*

As we all know, the “death of manual testing” has been a key topic discussed at meet up’s, conferences, and in blogs and magazines for years’, and with Microsoft’s Director James Whittakes proclaiming “testing is dead,” you would think this type of testing really has popped its clogs.

Gone are the days when a software tester would ‘test’ his/her own code and send it to production; monitoring systems, delivery systems, artificial intelligence, test automation, continuous integration, microservices and rollback capabilities have taken the spotlight. But does this mean we don’t need, nor want, the helping hand of a manual tester anymore?

“Manual testing covers the most complex scenarios that involve a lot of human judgment and decision making. Machines add value where completing repetitive tasks such as regression, and humans excel in tasks involving experiment, judgment, creativity and wit,” said Dileep Singh Marway, QA Manager at The Economist.

According to a blog post by Mario Mathee, COO at DVT, it can be prohibitively expensive to design test automation scripts for new products that are still in development, therefore it’s constantly changing. Keeping automation scripts current could end up costing more than the software is worth, so the only way to get the application stable enough for automation is through manual testing.

**TECHNOLOGICAL REVOLUTION**

“Testing is completely changing, making us more efficient. We have the opportunity to take testing and bring new levels of automation to it. If you're a world-class performance tester you could spend your time writing the code, or just look for other tests and automate it,” agreed Vice President at Sonatype, Derek Weeks

With Google’s CEO Eric Schmidt saying the key for humans who want to succeed in the future will observe this “separation of powers by specialising in what we do best, while collaborating with computers”, it appears that we need to take into consideration how to deal with this technological revolution.

According to *Small Business*, to stay “ahead of the game”, testers must abide by the following:

* Stay current with technological trends by locating and reading any one of the many available publications written for workers in your field
* Tell your employer that you are interested in staying current with technological changes in the field and would like to attend any trainings or workshops offered by the company
* Locate trainings offered outside the company
* Streamline your technological experience; make a list of all the programmes you use and what you use them for
* Eliminate technological programmes that you do not need, making it much easier to stay informed about remaining applications
* Schedule a regular meeting with tech support.

**THE CREATIVITY OF A HUMAN**

“Modern software developments mean that testers need to be adaptable and willing to learn new skills. It is important testing is done from a different point of view as it’s going to get more and more technology driven as time goes on,” added Marway.

In the early stages of this tech-driven revolution, artificial intelligence requires the creativity of a human to drive the test strategy, according to Marway. “Humans are still far superior than computers when it comes to tasks that involve the utilisation of intelligence. A person can write a poem, sing, dance and play music; although a machine can be programmed to do such tasks, they do not possess the ability to create or have intuition like us humans – we are far more fallible,” he continued.

Another modern software development technique affecting software testers is DevOps. This is because of its goal to automate as much as possible. “DevOps has absolutely changed the way manual testers work. The use of open source components has made developers more sufficient because you don't need to write code from scratch anymore,” agreed Weeks.

**DEVOPS PRINCIPLES**

For a tester to be successful in a DevOps environment, Marway recommends the following:

* Provide training to DevOps engineers on testing methodologies
* Become more involved in ensuring quality in the test environment deployment, packaging and other components of the application lifecycle
* Discuss code quality for deployment tools, not just core applications.

The objective of DevOps is to make software delivery quick, while ensuring that it’s delivered to high-quality outputs. All processes touch upon aspects of building and developing software as part of the latter stage:

* Code – manage, develop and review
* Build – continuous integration
* Test – continuous testing
* Package – artefact repository
* Release – release automation
* Monitor – end-user experience
* Configure – infrastructure configuration

Marway believes that, despite DevOps principles and practices helping drive quality assurance and testing by improving communication and collaboration between all essential stakeholders, it is eating up part of the manual testing quota, due to the fact it encourages automation.

**ENTERPRISE IT**

According to Michael Quan, DevOps Manager at Worldpay, DevOps opens up a new way of working with new tools, technologies and social media making its way into enterprise IT. “We are at a unique point in time to improve productivity through collaboration and knowledge sharing,” he added.

It’s important for DevOps engineers and their teams to always be on the lookout for tools in order to improve productivity and efficiency. Some key DevOps tools include: Jenkins, SolarWindsLog & Event Manager, Vagrant, PagerDuty, Prometheus, Ganglia, Snort, Splunk, Nagios, Chef and SummoLogic, according to *Stackify.*

Nelson Ajoku, Scrum Master at HM Land Registry, stated: “As long as tests are still running, the scope of software testing is by far not dying because of manual testing being automated. If you’re deploying, as you should be, manual testing should still be needed. As for DevOps, you still need to confirm congregations; it’s just a way of testing your code after testing it manually.”

Furthermore, machines/artificial intelligence (AI) is becoming more agile due to its ability to adapt to real time, without the need to be programmed in advance to tasks. With the likes of Amazon, IBM and Google embracing artificial intelligence, some may say robots are more than capable to deliver faster, more accurate analyses than humans when analysing data sets.

“IBM and Google aren’t the only companies applying AI techniques. Within the past year, AI in software testing has become feasible. Software testing must evolve in response to the shift to agile and DevOps. No matter how many testers you employ, it’s simply not possible for manual testing to provide agile developers immediate feedback on whether any of their constant changes impacted the existing user experience,” wrote Tricentis in a blog post.

**REMAINING RELEVANT**

“It’s not possible for machines to become agile enough to not require any creative input from a human being. The stunted social skills of AI highlight to me that people's jobs are safe for now, though this is dependent on the type of role and notably the creativity of a role,” revealed Marway.

“From a testing point of view, machines and AI will compliment creativity and not eradicate the need for skilled creative test professionals. The testing world is an exclusive arena where people are willing to learn new skills to stay relevant; therefore, my advice would be to keep your skills up-to-date.”

AI algorithms can be immensely helpful in the testing industry in making smarter and more productive software for the end-user. It is, however, important to decipher how to use AI intelligently.

“So far, virtually all machine learning and AI algorithms are only as good as the humans who design, train and improve them,” commented Ilia Kolochenko, CEO at High-Tech Bridge

“The developments of AI technologies usually require expensive long term investments that people typically cannot afford. Therefore, I don’t see substantial risks or revolutions that may happen in the digital space because of AI – not within the next five years at least.”

It appears that, as long as testers are willing to make investment in remaining relevant and “ahead of the game”, long live software testing!