

Befragung von Testern bei *FNT* zu automatisiertem Testen und automatischer Testdatengenerierung

	Could you briefly describe what exactly your job is at FNT?
Tester*in 1	QA Engineer
Tester*in 2	QA Automation for API & GUI tests in Command
Tester*in 3	I work in the QA doing the test designs, manual test execution and am also developing the automated tests for mostly the Requests domain

	How important would you say the job of a tester is for the development of high quality software?
Tester*in 1	When it comes to quality assurance, there are several quality gates that need to pass in order to deliver a high quality software. Some of these quality gates are fulfilled by developers and other by testers. As testers, we ensure that the application works as expected by conducting end-to-end tests. Developer's tests do not cover such complex tests that testers do. So in this sense, as testers we are executing tests that are as close as possible to what the customer will do and have the end user perspective, therefore the tester role is vital to the success of the product.
Tester*in 2	essential and it makes the life of developers much easier because they only need to make a rough test of their function and the QAs can make a full regression test of whatever else could be affected by a change.
Tester*in 3	who ever heard of "the rule of ten" knows that it's not only important, because of security reasons, but also from a monetary kind of perspective. So pretty important I'd say.

	Are you aware of any incidents when software had major bugs after insufficient testing?
Tester*in 1	Yes
Tester*in 2	Yes
Tester*in 3	No

	If yes, please elaborate.
Tester*in 1	Customer reported performance issues when trying to place several objects in Command. For notification-email service, the customer had a special configuration which was not part of the test suite.
Tester*in 2	Unfortunately this appears sometimes because manual release testing is very time consuming for multiple weeks and it is often not possible to make a code freeze at the same moment you start testing so some functions changed after an sufficient test and there is only a smoke test of the functionality after changing.
Tester*in 3	I'm sure there were instances, but nothing springs to my mind currently

	Are you aware of any incidents when software had major bugs after sufficient testing?
Tester*in 1	No
Tester*in 2	No
Tester*in 3	No

	If yes, please elaborate.
Tester*in 1	-
Tester*in 2	-
Tester*in 3	I'm sure there were instances, but nothing springs to my mind currently

	Approximately how many bugs are found per test run?
Tester*in 1	Because automated tests are implemented for the regression suite, the number of bugs found per test run is highly dependent on the automation coverage, the area in which changes occurred and how often the tests are run. So far, the highest number of bugs found per test run was 2 (several tests fail but the root cause was the same and it was traced back to 2 bugs which were not caught by manual testing, but by the automated tests). There are times when test runs finds no new bugs. In our team, each new feature is first tested manually - where we find on average 10-15 bugs, then it is automated (becomes part of the regression suite).
Tester*in 2	~ 3 per week, but we are running ~4000 automated tests each night
Tester*in 3	This really depends for us automated testers if there were any recent changes, so there's no definitive number I can tell. But I guess it's around two, when there were bigger changes in the code.

	How important do you consider the test data to be for high quality testing?
Tester*in 1	Test data is of great significance for production like scenarios.
Tester*in 2	Very, if possible it should be as real and (for automation) constant as possible to identify in a fast way if it is related to the change or data.
Tester*in 3	Because our test cases are use case driven it surely is better/important to work with data that is as close as possible to the data that our users will use.

	How does it work to create test data for new tests and new test instances of software?
Tester*in 1	Some test data is created manually and some test data for the automated tests is usually created by automated test scripts, typically via API tests.
Tester*in 2	For automation some tests are predefined and need to be created on the reference instance test-ref. For other tests we are trying to create as much data as possible in preconditions of each test.
Tester*in 3	In my case I'm usually working with self created data through the API or GUI of the test instances. In some cases there are also pre defined data sets that are created with each fresh start of the instance/domain.

	How time-consuming is this and also the maintenance of test data once it has been created?
Tester*in 1	Usually, little to no maintenance is needed for the test data, but this depends on the stability of the application. It can be the case, when the process of automatic test data creation can be affected by the application's flow, and just like the automated tests, the test data automation also needs to be maintained.
Tester*in 2	The maintenance of the predefined data is medium high, but to be honest very annoying, each 1-2 weeks someone breaks data because of unintentionally changes or the functionality has changed and tests are broken because data needs to be changed then.
Tester*in 3	<p>This also rather depends on if there were any changes to the models and objects that are used. So when there are no changes to the data it's low. If there are any changes it is relative to how big the changes were and if some new rules would apply. (Like you can have data sets with different attributes that will of course provoke different outcomes for the same case. So to cover each and every outcome you of course have to create/update the matching test data).</p> <p>I hope this made sense and I understood the intention of the question correctly. (:</p>

	How do you feel about automated testing?
Tester*in 1	Although the initial effort for test automation is high, the long term benefits overcome this disadvantage. The main advantage of automated testing is making sure we are able to release faster making sure all quality gates set in the testing process are fulfilled.
Tester*in 2	:)
Tester*in 3	It's a useful way to make regression testing less time consuming and more reliable.

	So far, FNT has created test data manually, even for automated tests. Do you think automating the test data could simplify the testing process?
Tester*in 1	yes
Tester*in 2	Yes!
Tester*in 3	Yes and no.

	If so, in what way?
Tester*in 1	Automating the test data is very important when it comes to portability.
Tester*in 2	Data can not be broken by unintentional manual changes. The functions to create the data should create valid data all the time, so the maintenance effort for data will be less and if the functions to create data are already changed while development it will be even faster!

	If so, in what way?
Tester*in 3	<p>I think this could be a great way for software in which the data models don't evolve that much anymore (because the automated data doesn't have to be updated every few days/weeks). However with software that is still in the early development stage I don't think that it'll simplify the testing process by a great deal, because someone has still to maintain the automated test data. The other problem I could imagine with automated test data that isn't created on the fly is when problems with the infrastructure are occurring (and that actually happens quite often in automated testing). All tests will fail, because of missing test data.</p> <p>Currently usually only the few tests that would be triggered during such occurrences will fail and the others would still be successful.</p>

	Some of the CIF tests are already successfully covered by an automated test data generation. Do you think this could eventually be adapted even outside the CIF test environment at FNT?
Tester*in 1	<p>This approach of test data generation could be applied for Command test data generation, but it might be too time consuming, so it needs to be analyzed if generating test data automatically is really needed as opposed to using a dump file with production like data. In some situation generating test data automatically could be useful and could complement the already existing test data on the instance.</p>
Tester*in 2	<p>Interesting idea to define the data in separate files, we use constants in the test files, very ugly in my opinion, so maybe we could adapt it for Command API and GUI tests as well.</p>
Tester*in 3	<p>Sure. The question is rather at what point of software maturity it would make sense and I can't give an answer to that, because I don't know much about this project yet.</p>

	Do you think manual testers compete with automated testing or can manual and automated testing complement each other?
Tester*in 1	<p>Surely manual and automated testing complement each other. Not every scenario can be covered by automated tests, so there is still a need for manual testing, especially for exploratory testing and testing workflows scenarios.</p>
Tester*in 2	<p>It can complement each other perfectly, not even for the testing itself as well for preparation or identifying the right test cases to automate or the type they should be automated (GUI or API).</p>
Tester*in 3	<p>Imho they complement each other, because sometimes automated testers can lose focus of user experience when doing only automated (and therefore rather technical) tests for a long time. However this of course really also depends on knowledge and experience of the software that needs to be tested. Very niche and special test cases and the related bugs are usually done and caught by manual testers. On the other hand the automated tests are catching bugs faster and can narrow down where the problem might come from more easily.</p>