

## **Automation in different test levels**

If we look at the test pyramid it says we should have a bigger focus on the API testing and Unit testing than on the UI testing. Upon UI automation it was realized many API faults and as these kind of testing are cheaper and quicker to run we could have a set of automated API testing running together with the UI testing. A suggestion would be to use the same BDD approach used for front end automation and using the framework Rest assured.

One more test level that could be implemented would be the Unit testing. I believe that these tests should be performed by the developers using some tool such as Junit. So if we have these two more automated test levels the UI testing would be much more reliable and other than save money since the front end tests are the most expensive ones.

We could think about to have some performance/load testing as well. These kind of tests would be executed against the API's in order to make sure that the services would support the application scalability. A good test tool to be used for this propose could be JMeter. However to implement these non-functional test we'll need a good non-functional requirements where these kind of information could be present in the user stories format.

## **Extra considerations**

After finish the test automation it's necessary to continue taking care of these tests. It's normal to have some broken tests for every new build that's why is important to have a good test architecture to make the maintenance easier .

As the software is getting mature the tests can run into the pesticide paradox testing and stops to find new bugs. So a good practice would be for every sprint the team run some exploratory tests against the features that got some changes and point the tests that are automation candidate.

Currently the application usability might have many problems. For instance it's not clear to the user how he can edit a task and for subtask it's not even possible to edit the "Due date". So in order to create a better user experience it could be applied an

usability testing analysing the efficiency, effectiveness and satisfaction by the user point of view.