

Hi! My name is Arseny

And now I'm going to show you my insights



I really hate those moments of realizing that it's actually not a png after copy



...sorry

First of all let me ask several questions about “The awesome Q/A tool”

Consider the situation like it's a real app.

Just in order to plan my activity or even to collect and organise some test data.

- 1) What is the target audience of this product?
- 2) Is it going to be a first launch of the product or it's just an ordinary release?
- 3) Do you have some experience of testing it within the team?
May be you have some manual testing scenarios, cases and etc.
- 4) Do you have some experience of collecting analytics or predictions like the average load or maybe the usage of browsers for this app?



Like this

Then let us skip some checks

[illegible]

**By the way, are there any limits for the length of an input text?
Or may be there are some additional rules for text filtering?**

Because we actually consider *user acceptance tests*

It's time to choose some tools for making automation



What about browser?

I chose Chrome-driver (because ~~I'm a lazy person~~)

But in a real life I hope some research will help me to choose browser (like from the first slide). And it will be really helpful to use selenoid tool for multi browser testing



Now let's make some tests

- 1) test_show_headers – just for testing main headers of the page. I don't have a special explanation, it was just the first test of the app;
- 2) test_show_tips – test for checking tips for input questions and representation of them with the answers;
- 3) test_delete_question
- 4) test_create_a_new_question
- 5) test_counting_questions – it's interesting here, because the form may show you “no questions”, “1 question” or “2 questions”. So, that is the point;
- 6) test_sort_questions – my favourite one;
- 7) test_decline_adding_empty_fields – just for testing filters of input text for questions and answers



Here you can find 2 questions. Feel free to create your own questions!

Since I'm lazy I used pytest parametrization in tests with the same logic

So, that's why there are two additional tests in my test run

▼ ✓ Test Results	2 s 539 ms
▼ ✓ tests	2 s 539 ms
▼ ✓ test_user_acceptance	2 s 539 ms
✓ test_show_headers	57 ms
▼ ✓ test_show_tips	642 ms
✓ (header0-tip_element0-Here you can find the created questions and their answers_)	332 ms
✓ (header1-tip_element1-Here you can create new questions and their answers_)	310 ms
✓ test_delete_question	101 ms
✓ test_create_a_new_question	322 ms
✓ test_counting_questions	222 ms
✓ test_sort_questions	837 ms
▼ ✓ test_decline_adding_empty_fields	358 ms
✓ (-some answer)	176 ms
✓ (some question-)	182 ms

As you see, everything is ok with the run (I hope)

That's all

Thank you!
Hope it wasn't boring

