QA Engineer Assignments

Instructions

By completing these assignments, you are providing a small showcase of your skills and experience. There are some bonus tasks to each assignment optional to complete, but it's recommended that you complete them if you have enough time.

Assignments should be completed within 7 days. We believe this is more than enough considering work time and personal time and other obligations that might affect the candidate's time for completing these assignments. If you're not able to complete all the assignments, simply send what you have managed to complete within those 7 days.

For each assignment that requires using additional tools and doing some coding, you can either open a public repository on your personal GitHub/Bitbucket account and share the solution that way or you may simply send the code with instructions via email. It's always necessary to prepare instructions on how to run each one of your solutions so we can review them and try them out on our side as well.

For any additional questions about these assignments, feel free to reach out to: iva@embed.xyz

Assignment #1

Go to the following website: https://react-shopping-cart-67954.firebaseapp.com

This website has only one page and is a mock-up of a simple online shopping website for buying T-Shirts. You can filter the shirts by size and you can add shirts to the shopping cart. For every shirt, you can change the quantity you're willing to buy. At the checkout, the website notifies you about the total amount you're about to spend.

Create test cases for this website that you want to use for manual regression testing.

Share the document where you have created those test cases as your solution to this assignment.

Bonus task: Select 2-3 of the test cases and automate them using any open source test framework or tool you're familiar with. Even though this is optional, if you have any test automation skills, it's highly recommended that you complete this bonus task. Share the solution using your private GitHub/Bitbucket account or share the solution via email by archiving it. Make sure to prepare the instructions on how to run the tests.

Assignment #2

Focus on the same website as the one from the previous task: https://react-shopping-cart-67954.firebaseapp.com

Describe how would you perform a load test to ensure that at least 1,000 users can simultaneously browse this website, add items to the cart and checkout their orders. Write down which tool(s) would you use (and why), how would you configure the test, how would you run the test and how would you interpret the results.

Bonus task: Create and configure an actual load test using the tool(s) you proposed. Share the solution using your private GitHub/Bitbucket account or share the solution via email by archiving it. Make sure to prepare the instructions on how to run the tests.

Assignment #3

You will use the following Bitcoin (BTC) price API created by CoinDesk to complete this task: https://api.coindesk.com/v1/bpi/currentprice.json

Create an automated test that will detect any fluctuations in the BTC price. Repeat requests every 30 seconds within a predefined amount of time. The amount of time to run this test should be configurable. For example, if the test would run for 5 minutes, you should send 10 requests in total and calculate the results.

The output/test results to produce should be the following:

- Number of collected BTC values
- Average BTC price in USD
- Median BTC price in USD
- Standard deviation of BTC price in USD
- Bonus task: mark any outliers that fluctuate more than 5% compared to the median value
- Average response time for all requests
- Median response time for all requests
- Minimum response time
- Maximum response time

Share the solution using your private GitHub/Bitbucket account or share the solution via email by archiving it. Make sure to prepare the instructions on how to run the tests.

Assignment #4

Go to the website: https://www.boredpanda.com/

This is a website about basically nothing in particular and people waste time here when they are bored. But this is not why you’re here right now. :)

What you should do instead is perform usability testing on this website. Spend 10-15 minutes on this website, look around, get a feeling of how it can be used, and interact with its content.

Write a report about things you like about this website and things you don’t like about this website (and what you recommend to change/improve it). It doesn’t have to be anything fancy and structured in a specific way.

The point of this task is not about writing tests, but about getting an insight into how you think about the design, website functionality and how it can potentially affect the users.