Peer-graded Assignment: Assessing Quality through Scenarios

ⓘ It looks like this is your first peer-graded assignment. Learn more   ✕

Review your peers' assignments
Congratulations on submitting your assignment! Now your peers can review it. In order to get your grade, you must also review some of your peers' assignments. Your grade should be ready by **Sep 2, 11:59 PM PDT**.

[Review assignments]

Instructions   **My submission**   Discussions

Amazon.com
Submitted on August 30, 2020   Shareable Link

PROMPT

Select a public website that you use enough to be familiar with what a typical user may want to do. This website should not require the peer reviewer to sign up for an account or pay to use the site in any way. The website should also not, to the best of your knowledge, serve malware, use JavaScript to complete cryptocurrency mining, or any other negative practice that might harm the peer reviewer.

Select three quality attributes that are likely to be important when deciding a website architecture for the website you chose. You can use usability, security, performance, reliability, or any other reasonable quality attribute as the basis of your selection. Briefly explain the importance of each quality attribute as it relates to the software/service you selected.

Then, for each selected quality attribute, write one scenario that would help quantitatively assess whether the software solution meets its goal. You shall write your scenarios using a format that was presented in the lectures.

Write your first selection, discussion of importance and scenario here.

Scenarios - Amazon.com

A large e-commerce website with thousand or million views per day per item.

Millions of services running in the background or even a click on the page.

1st selection -> Performance

The users shop on the website and try to order multiple items. The users do not want to wait for so long for each order. And many users are shopping at the same time.

Source: Customers use a web browser or mobile devices

Stimulus: Users add an item to the cart.

Environment: The system is operating normally. Latency is less than 100ms for every query by URL. Should be able to auto-scale from 10 to 1 million users.

Artifact: MENG stack

Response: Round trip from "add to cart" to the browser update the cart

Response Measure: 99.99% within 3 seconds

PROMPT

Write your second selection, discussion of importance and scenario here.

2nd selection -> Usability

The users add the item to the cart. First, the users need to find the "Add to Cart" button. The information of the items must be clear and easy so the user can determine to do it. And the button is easy to find and click.

Source: Customers use a web browser or mobile devices

Stimulus: Users add an item to the cart.

Environment: User find the "Add to Cart" button easily and press on it naturally

Artifact: MENG stack

Response: A confirmation the users clicked on it

Response Measure: 99.99% within the confirmation message appears without the error message.

PROMPT

Write your third selection, discussion of importance and scenario here.

3rd selection -> Reliability

The users add items to the cart. But they wonder if the cart actually grabs the item they wanted to add or is there something wrong in between and in the back0end of the system. So the system must be visible or do some confirmation to make sure the item is the correct one.

Source: Customers use a web browser or mobile devices

Stimulus: Users add an item to the cart.

Environment: Users add an item to the cart and the cart will save it and keep it.

Artifact: MENG stack

Response: Front-end checks by users that the items linger in the cart. Back-end check by query the item is visible in the cart.

Response Measure: 99.99% items are in the right cart

[Edit submission]

Comments
Comments left for the learner are visible only to that learner and the person who left the comment.

[Share your thoughts...]