Capstone Assignment 3.3 – Analyze and Evaluate an Architecture

Objective: You are also given several Quality Attribute Scenarios, written from various perspectives, to evaluate the architecture of the app.

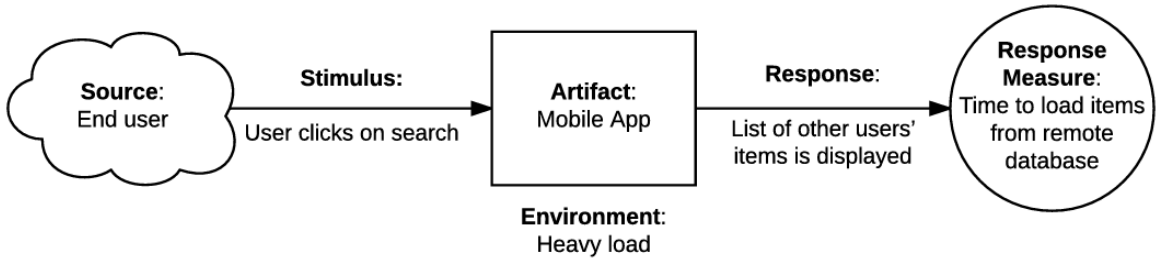
For each Quality Attribute Scenario provided, identify if it is a risk, non-risk, tradeoff, or sensitivity point in the applications architecture and give a brief explanation (3 sentences or less) for each. This can be put into point form, but each Scenario should be covered.

Once you have finished your evaluation, update the Utility Tree to reflect the Scenarios. You should be able to make 4-5 changes, including adding a Quality Attribute and adding more Attribute Refinements and ASRs to reflect the risks. You may also update the current ASRs to reflect the Scenarios or (if justified in the evaluation) change their priorities.

Once you have finished your evaluation, use the Quality Attribute Scenarios to create an updated Utility Tree that considers the Quality Attribute Scenarios based on the primary ASRs from the previo us Utility Tree.

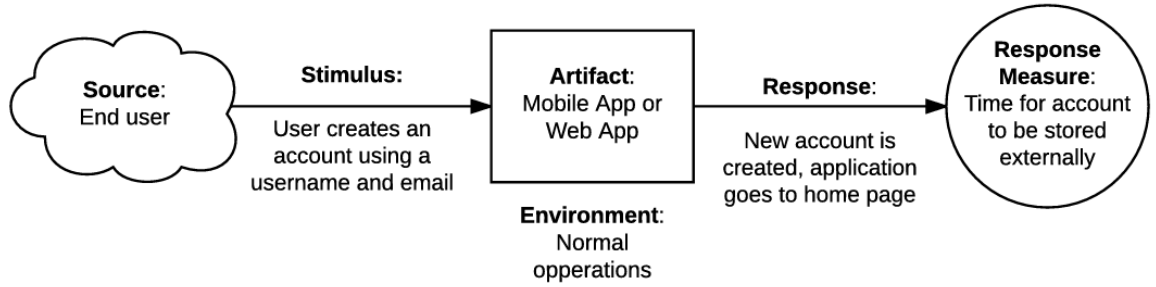
Concrete Quality Attribute Scenarios Comments

1.



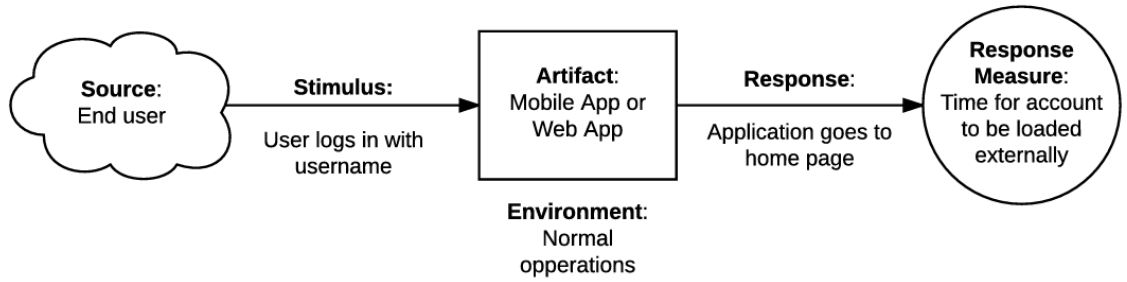
In this first scenario, we can see the first **risk.** The system may not comply with the ASR requirement of l;ess than 1 one second to complete a task, so we can assume that it will take more than that.

2.



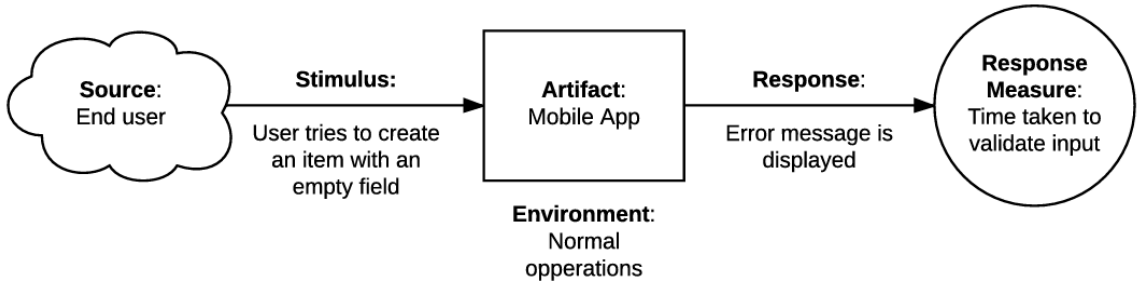
This create a **risk** scenario, because the user can login without a password and only using username and email

3.



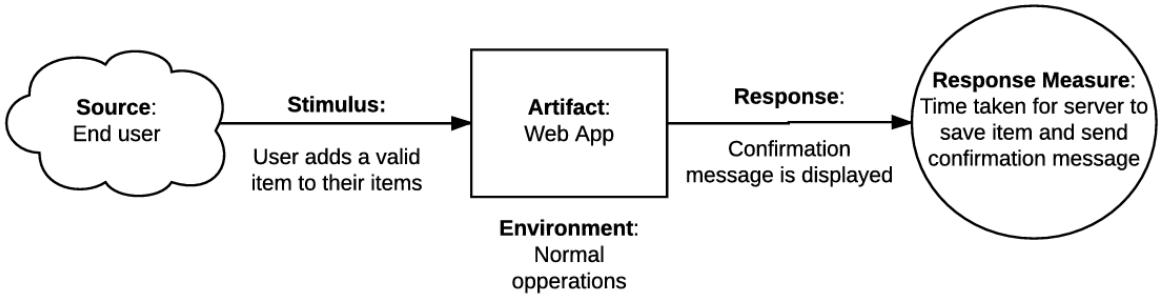
This creates a **tradeoff** since anyone that knows the username can login and the account.

4.



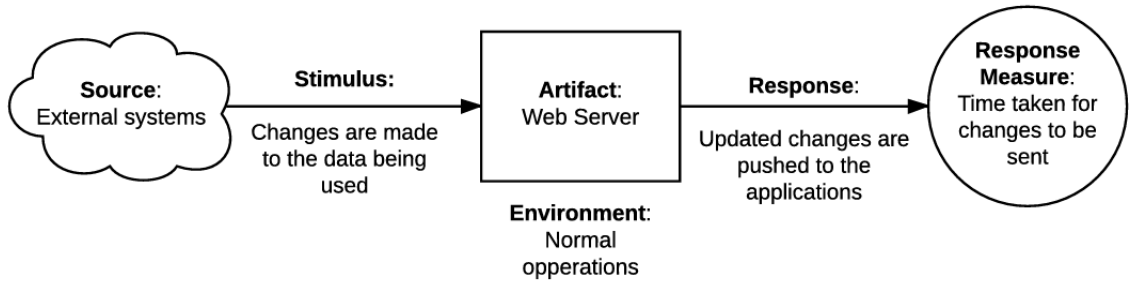
This is a **Non-Risk** scenario because the software is performing as expected and popping the message error.

5.



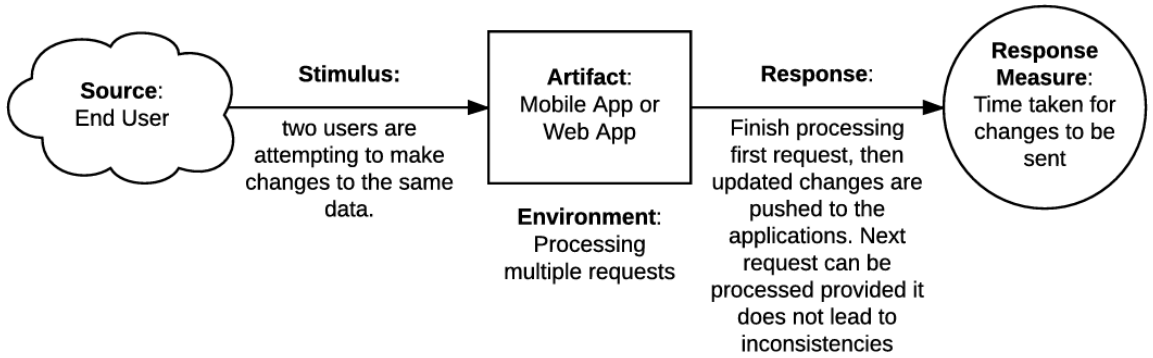
Another **non-risk** scenario, The program is covering this error as expected.

6.



This scenario represents a sensity point. If we invert the push strategy to pull on the data sync we can minimize the problem

7.



Lastly, it's a **non-risk** since the system needs to handle multiple modifications of the data at the same time and needs this security lock.

Utility Tree