quiz solution w3

Question 1

Identify the attribute refinements of a system's performance. Select the 2 correct answers.

1 point

□ **boot time**

This should not be selected

Incorrect. Although this is undoubtedly important, it is too specific.

☑ ~~**latency**~~

Correct

Correct! Latency is the time it takes to produce an output after receiving an input.

□ **simplicity**

This should not be selected

Incorrect. A system may be simple but perform poorly, or vice versa. Simplicity is not a refinement of performance.

□ **maintenance downtime**

This should not be selected

Incorrect. The maintenance downtime is not part of the system's performance. This attribute refinement is usually considered part of **availability**.

☑ ~~**throughput**~~

Correct

Correct! Throughput is the amount of output produced over a period of time.

Question 2

Guidelines such as "the system should be easy and intuitive to learn", "the system should minimize user errors," and "the system should make it easy for users to complete tasks" fall under which category of quality attribute?

1 point

☑ ~~usability~~

Correct

Correct! The usability is how easy it is for an end user to interact with the system.

□ **intuitiveness**

Incorrect

Incorrect. Intuitiveness is important but only covers one of these phrases. Try again!

□ **complexity**

Incorrect

Incorrect. These guidelines refer to the user's experience, while "complexity" usually refers to the complexity of the system, which the user may not even see. A very complex system could have a very simple interface, for example.

□ **astonishment**

Incorrect

Incorrect. Quality attributes are typically positive qualities that we strive to attain with our software. We try to avoid astonishment; remember the "Principle of Least Astonishment!"

Question 3

Calum is leading a team of developers and would like to promote conceptual integrity. Which of these is **NOT** a way he could promote conceptual integrity:

1 point

□ **Establish conventions, such as structural rules or naming conventions**

Incorrect

Incorrect. Establishing naming conventions, structural rules, and design principles can help a team establish better conceptual integrity.

☑ ~~**Split the development team into subteams for each component of the architecture**~~

Correct

Correct! Although any large project will have subteams, this is a factor that tends to work against conceptual integrity.

□ **Do regular code reviews with the development team**

Incorrect

Incorrect. Code reviews will allow your team to fix and discuss examples in which conceptual integrity was not achieved, improving existing code and future practice.

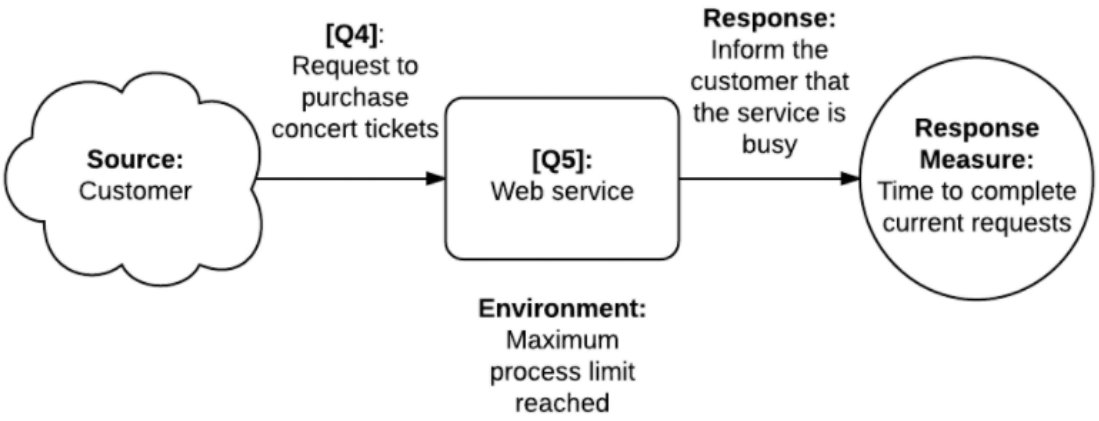
□ **Adopt good documentation practices**

Incorrect

Incorrect. Good documentation practices will help your team get on the same page, help to onboard new developers more easily, and make it easier to go back and maintain the code while preserving conceptual integrity.

Question 4

Have a look at this diagram of a quality attribute scenario:


Source: Customer
[Q4]: Request to purchase concert tickets
[Q5]: Web service
Environment: Maximum process limit reached
Response: Inform the customer that the service is busy
Response Measure: Time to complete current requests

[Q4] is a condition that will cause the system to respond. What is this called?

1 point

□ **perturbation**

Incorrect

Incorrect. Perturbation is sometimes used in process control, but that's not the term here!

□ **request**

Incorrect

Although the example is a request, there can be other things that trigger a response!

☑ ~~**stimulus**~~

Correct

Correct! This is a stimulus.

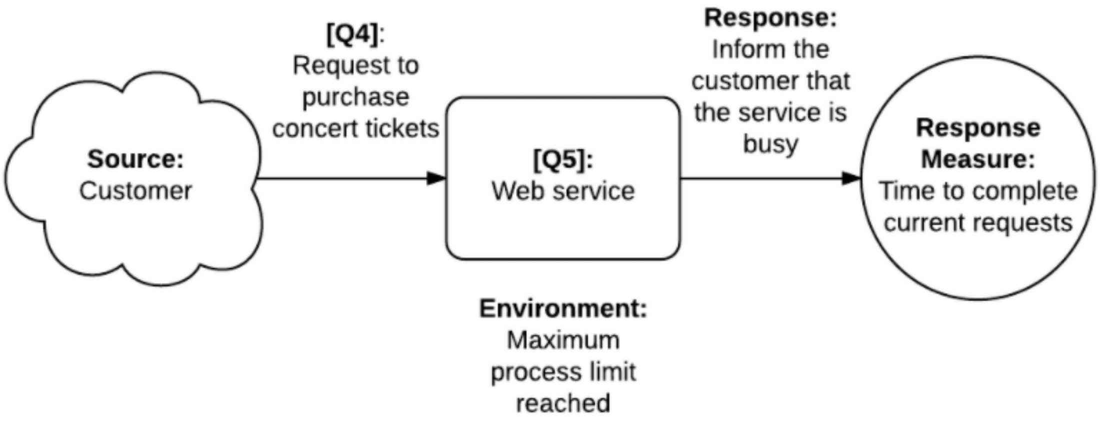
□ **error**

Incorrect

Incorrect. This implies that the signal is always something "wrong". Even normal conditions produce a response, as shown here.

Question 5

Let's look at the same diagram of a quality attribute scenario:


[Q4]:
Request to purchase concert tickets
Response:
Inform the customer that the service is busy
Source:
Customer
[Q5]:
Web service
Response Measure:
Time to complete current requests
Environment:
Maximum process limit reached

[Q5] is the part of the system affected by the stimulus. What is this called?

1 point

☐ **context**

Incorrect

Incorrect. Context is very similar to environment.

☐ **component**

Incorrect

Incorrect. This may be a software component in the general sense, but we have another name for it within a quality attribute scenario.

☐ **process**

Incorrect

Incorrect. This is not the term used in this situation!

☑ ~~**artifact**~~

Correct

Correct! Don't confuse this artifact with the artifacts we talked about in deployment diagrams.

Question 6

Which of these could be considered under the **environment** in a quality attribute scenario?

1 point

☐ **unrecognized system request**

Incorrect

Incorrect. This is an example of a stimulus.

☑ ~~recovering from error~~

Correct

Correct! Recovering from an error is a possible environment in which the software can receive stimuli.

☐ **internal subsystems**

Incorrect

Incorrect. Internal subsystems are a possible stimulus source.

☐ **send error to external system**

Incorrect

Incorrect. This is an example of a response that the system may make.

Question 7

Who are the three main groups of people involved in the architecture tradeoff analysis method?

1 point

☐ **evaluation team, outsiders, project team**

Incorrect

Incorrect. Outsiders are a subgroup of the evaluation team!

☐ **peers, outsiders, designers**

Incorrect

Incorrect. These are subgroups of one of the main groups!

□ **clients, designers, stakeholders**

Incorrect

Incorrect. Some of these are subgroups.

☑ ~~evaluation team, project decision makers, architecture stakeholders~~

Correct

Correct! Remember that each of these groups also has subgroups. For example, the evaluation team could include those responsible for architecture design, as well as completely independent evaluators.

Question 8

Leon is analyzing the architecture and notices that under conditions of high numbers of users signing on at the same time, there is a potential that one of the architecturally significant requirements (ASRs) will not be met. What is this called?

1 point

□ **non-risk scenario**

Incorrect

Incorrect. A scenario in which an ASR may be violated is definitely a risk.

☑ ~~risk scenario~~

Correct

Correct! This is a risk scenario and should be evaluated.

□ **utility shortfall**

Incorrect

Incorrect. This is not a term that is used in ATAM.

□ **tradeoff**

Incorrect

Incorrect. A tradeoff is a decision that improves the level of one quality attribute in exchange for a lower level in another.

Question 9

Maddie's software team is split into two subteams working on two different components that work together. These teams share one large room and are in constant contact. According to Conway's Law, what could happen if Maddie does not physically separate the two subteams?

1 point

□ **The component interfaces will not be reusable**

Incorrect

Incorrect. This could be a further consequence of the correct answer, but it is a stretch to conclude this directly from Conway's Law.

☑ ~~**The components they produce will be very tightly coupled**~~

Correct

Correct! Conway's Law says that software tends to mirror real world organization, so a team in constant contact will produce more tightly coupled software.

□ **They will not leave sufficient comments, thinking that the other team already knows the relevant details**

Incorrect

Incorrect. Good developers should comment appropriately whether or not they are separated.

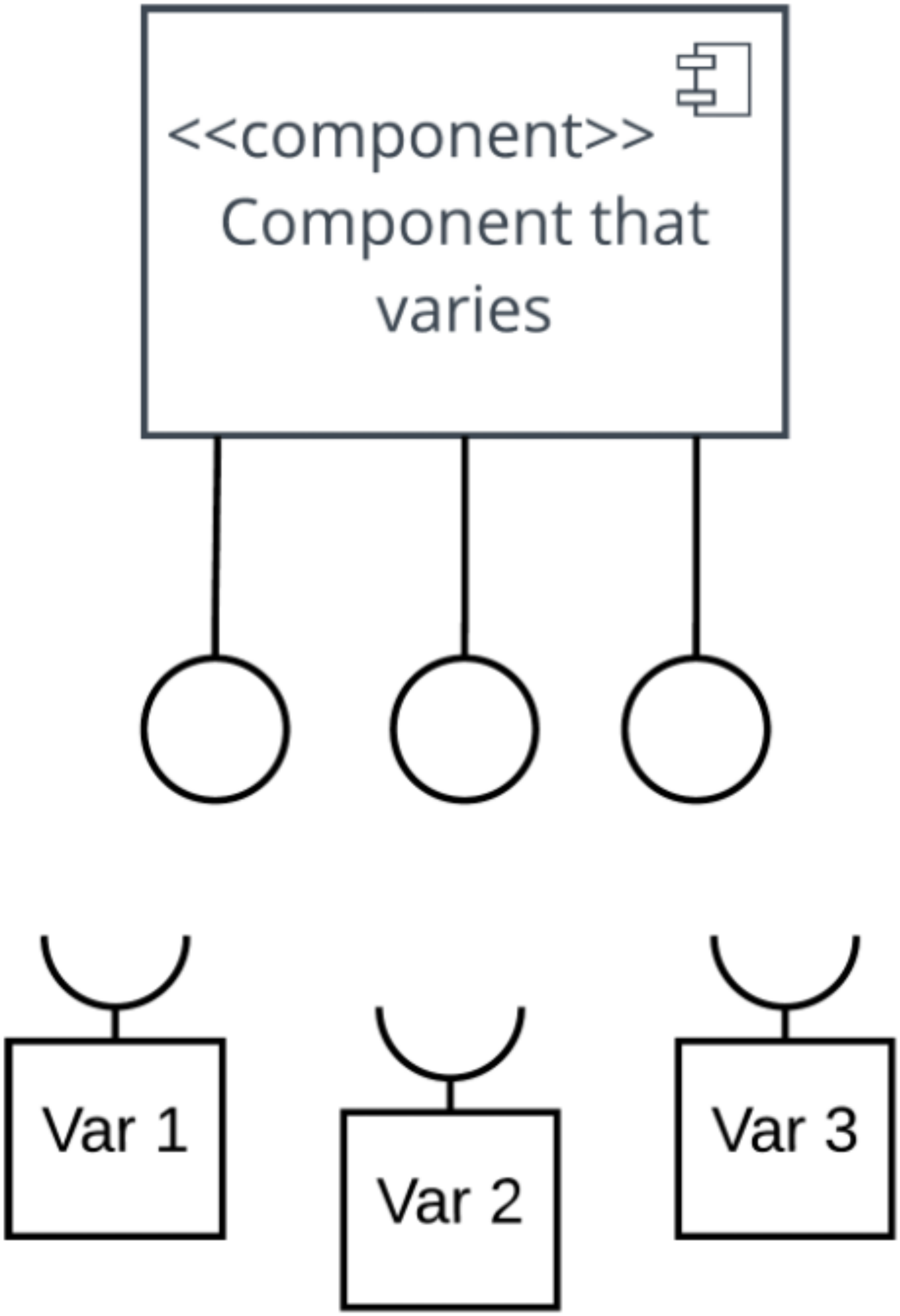
□ **Their components will repeat code unnecessarily**

Incorrect

Incorrect. This may or may not be true, but Conway's Law doesn't tell us either way!

Question 10

This is one of the styles of implementing variations. What is it called?



1 point

☑ ~~adaptation~~

Correct

Correct! Adaptations change certain parts of the component.

□ **extension**

Incorrect

Incorrect. Extensions are optional and add new functionality, usually through a common interface.

□ **composition**

Incorrect

Incorrect. This is not one of the forms of variation we discussed!

□ **replacement**

Incorrect

Incorrect. Replacement involves taking a component out of a system and replacing it with a different component.

Question 11

Which of these is **NOT** a typical advantage of developing a product line?

1 point

□ **overall cost reduction**

Incorrect

Incorrect. This is usually an advantage of a product line, as long as there are enough products.

□ **consistency over the product line**

Incorrect

Incorrect. This is often an advantage of a product line, which can be a boon for developers that use your platform as well as end-users.

☑ ~~**greatly reduced up-front development**~~

Correct

Correct! This is NOT an advantage of a product line, which typically needs more up-front development.

□ **reducing time-to-market of later products**

Incorrect

Incorrect. This is often an advantage of a product line, since much of the code is already in place after the up-front development.

Question 12

Andy is planning the development of a product line of eBook readers and he has to categorize the components that will be handled by each team. There is one product in the line which has a backlight for reading during the night. In the development process, where would you categorize this feature?

1 point

☑ ~~**Product-Specific**~~

Correct

Correct! Product-specific components are developed by the application engineering team, because they only apply to one product.

□ **One-Off**

Incorrect

Incorrect. This not the term used in product line development.

□ **Variation**

Incorrect

Incorrect. Variations apply to more than one product, so they are handled by the domain engineering team.

□ **Commonality**

Incorrect

Incorrect. Only one product in the line uses this feature!