

## Fullstack Assessment Theory - Caitlin Oddy

### Question 1:

**In design Heuristics, what does the term “Matching between system and the real world” mean? What are the advantages?**

-----

The term ‘Matching between system and the real world’ comes from Jakob Nielsen’s 10 Usability Heuristics for User Interface Design. This principle is that design should mirror the users’ real world experience. This includes using phrasing and ideas the user is accustomed to and having controls match their real world counterparts.

Matching can be achieved by conducting research to find out what terminology users are familiar with and assessing what actions they associate with any icons etc to be used to make sure there is no ambiguity.

There are several benefits to matching between system and the real world, including:

- **Decreased error rate and user dissatisfaction:** As users are using design elements they are familiar with from the real world, they are less likely to make mistakes. This in turn leads to them having a more satisfactory experience.
- **Increased speed:** With users not needing to read lots of instructions and familiarise themselves with new controls they can begin to use the system much faster.
- **Ease of use:** There is less arbitrary new information for users to absorb and so they can focus on the new elements the designer wants them to take in, while not being overwhelmed.

### Question 2:

**What do you understand by “Single source of truth”? and how does it relate to redux? What are the advantages ?**

-----

Single source of truth is the concept in software and data that a piece of information or data should only be stored in one place. There can be references to it throughout but it is only stored in one place. This reduces the risk of duplicate information that may not match.

Redux is an application of the concept of 'single source of truth' for React. It is a state management tool, with the application state stored as an object, called the store.

The key advantage of redux is that all components of the app are accessing the same state so that complex applications can be managed more easily as there is only one place to store. This feature makes it easy to debug issues as well, as the developer only needs to check one place to see the state.

### **Question 3:**

**What is the difference between a stateless component and a stateful component in React?**

-----

Stateless and stateful components in React are also referred to as functional and class components.

Stateless (functional) components do not store a state, or track changes to them. They are functions and are used to display information, using *props* to take in data and change what to display accordingly using *return*. They do not have a life cycle.

Stateful, or class, components however do have a state. They hold information and update it based on modifications. They have a life cycle (mount, update and unmount).

However, React now features *hooks* which allow functional components to have states and since functional components are generally less complicated this is seen as a good solution.

### **Question 4:**

**List out the advantages and disadvantages of exploratory testing (used in Agile) and scripted testing?**

-----

#### Exploratory Testing:

Exploratory testing does not follow a set path, rather it involves testers 'exploring' the system, creating and adapting tests as they explore. The aim is to better match users' experience using the system.

Advantages:

- Better suited to finding issues of usability and UX
- More closely matches real life use of a system

#### Disadvantages:

- Relies on testers properly exploring and creating tests as they explore which may be biased or simply not extensive enough
- Slow and requires a person to administer the tests
- Inconsistency across different testers

#### Scripted Testing:

Scripted testing follows a pre-set path, with testers given specific test cases and steps, called a test script. This testing does not allow deviation by the tester, they are simply to complete the tests required and give the results.

#### Advantages:

- More objective, less influenced by the individual tester
- Greater possibility for automation
- Well suited to finding functional issues
- Easier to repeat tests with different testers

#### Disadvantages:

- Testers only test what is in script and so issues in other areas may be missed
- Time must be spent creating test scripts
- No way of adapting quickly to different needs