Jody Eyre
Full Stack Assessment Theory Questions

1. **In design Heuristics, what does the term "advantages of matching between system and the real world" mean? What are the advantages?**

Design heuristics are known as design principles which are a set of guidelines which can be broadly applied or used by designers to inform their decision making during development. It's important to remember that these principles are not super strict rules but they are to be used to make informed choices and solve problems with the objective of making the user experience as good as possible.
Advantages of matching between the system and the real world means that the user interface should be in sync with the users experience or scenario of how a software, application or website should be worked. The software/application should speak what the user already knows, language and actions should be relevant, applicable and familiar. This basically means that the software (or whatever is applicable) should be intuitive to the user and behave or function in a way which is familiar, like how things work in the real world. An example of this could be when you user your phone and if you want to zoom in on a picture you use a pinching action with your fingers and the phone recognises the movement and zooms in on the image. This matches an action which is performed in the real world. This example shows you how this makes an interface more intuitive and user friendly.
Advantages of this principle are:

1. As the experiences are more intuitive and user friendly - matching real world actions - users don't need to learn new habits or actions they can use their prior knowledge of how things work in real life to interact with digital systems. This enhances the user experience.
2. Easier to learn how to use digital devices if you have a system which behaves in a way which is expected.
3. The reduction of cognitive load is an advantage as it takes less mental capacity to understand a system so users don't need to think about it as much so it feels easier.
4. Users can complete tasks more efficiently when they intuitively know how to user a system rather than having to think about it.
5. Less errors are made when systems or applications match the real world as users can predict behaviours.
6. It is more satisfying for a user to interact with a digital application or software when they feel like they know how a system works.

2. What do you understand to be the "single source of truth" and how does it relate to redux? What are the advantages?

The phrase "the single source of truth" is a concept or idea which is used in software development and data management which refers to the practice of structuring information and data in a way that ensures there is a single source of a particular piece of data. This means that a specific piece of information should only be stored and maintained in one place within a

system - there should not be any duplicates in several different locations. The goal of a single source of truth is to determine whether a company is relying on one source of information for critical decisions. The single source of truth will detect, collect and integrate data from a whole range of sources into one location by locating the primary location of the data storage.

Redux is a state management library where the state of an entire application is stored in one JavaScript object called the store. Redux is based on three principles where one of them is the single source of truth.This is very similar to the single source of truth because the store serves as a single source of truth for the entire state of the application. All of the components within the application can access this state through the store and changes can to the state can be made through actions.

The advantages are:

1. The main benefit could be said to be that it helps to avoid data inconsistencies and duplication of data which improves data integrity.
2. If companies use the single source of truth then it can improve efficiency within an organisation by ensuring that data is up to date and relevant. There is no duplicate data entry or version control issue. With lots of organisations collecting and analysing data the single source of truth is essential in order for these processes to function properly.
3. As redux stores data in a single location rather than a collection - to represent data one object tree can be used. This allows you to be able to see what your application knows at any given time allowing you to serialise it.
4. As redux enforces the single source of truth principle you can predictably determine where to find and update the status of an application which makes it much easier and efficiency for state management.
5. Debuffing is also easier as you can follow where the flow of data goes through a single source. This makes it much easier to diagnose issues effectively.
6. As traffic may be increased through an application Redux' single state of truth allows scalability. It makes managing state in one place easier and it is easier to analyse how data flows through an application.

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

State is data we import which is usually to show the user - that is subject to change. Stateful and stateless components can be called many things - container vs presentational components is an example of one. These names or categories refer to how a component manages and interacts with data or "state". The literal difference between them is that one has state and the other doesn't.

Stateful components

1. These are commonly known as class components
2. They have their own internal state which can be changed using setState
3. These components are commonly used when a component needs to maintain and manage its own data that can change or alter over time.
4. They have access to life cycle methods which allows you to hook into various stages of components life such as mounting, updating and unmounting.

Stateless components

1. They are defined as a function commonly using arrow functions

2. They dont have their own internal state - they receive data and behaviour via props.
3. They are usually used to represent user interface elements which dont need to manage their state.
4. They don't have life cycle methods or access to internal state management

(insert code here)

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

Exploratory testing is an approach to the testing of software that is simultaneous learning, test design and execution. It has a large focus on discovery and relies on the guidance of the tester to uncover defects that are not easily covered in the scope of other testing models. It is often used in scenarios where a team needs to learn about a product or application on a short time frame with rapid feedback.
Most software testing approaches are structured and test cases are defined based on pre defined user stories and maps. Test coverage is measured using metrics but what can be missed quite often are edge cases. Scripted testing is an approach where the testing team follow a stepped procedure to check the application under testing. This ensures all tests are executed during the testing phase.

Advantages of exploratory

1. An advantage of exploratory testing is that as it is more unstructured in nature it can reveal bugs that would go undiscovered during a structured phase of testing.
2. Test execution can be implemented without formally follow test steps but tools allow findings to be formalised and documented which allows for visual feedback and allows teams to react and adapt to changes quickly and collaboratively - facilitating an agile workflow.
3. Another advantage is that it creates an instant feedback loop.
4. It allows a tester to review the product from a user perspective rather than following rigid testing steps.
5. This type of testing can be opened up to all key stake holders enabling a faster and more efficient review by people beyond a traditional tester - they may think differently or interact differently.

Disadvantages of exploratory

1. Exploratory testing cant provide comprehensive testing alone so there must be a balance between scripted testing and exploratory.
2. This type of testing cannot be relied on solely to find all bugs and can be reliant on a users knowledge and experience.
3. If there is regulated or compliance based testing need, scripted testing is more suitable as it allows certain checklists to be followed for legal reasons.
4. Without documents, can be difficult to track the testing
5. Cant be automated

Advantages of scripted testing

1. Test cases can be designed to specific requirements and can give consistent coverage
2. Test case is easy to track and follows user requirement documents
3. It is easy to access scripts of other testers or team members - encouraging collaboration
4. It can help to understand complex test scenarios where planning is needed
5. It can determine usability
6. Can be automated

Disadvantages of scripted testing

1. Feedback can be slow
2. Can be difficult to catch edge cases
3. Can overlook user perspective and miss bugs due to lack of creativity
4. Cannot be done by all stakeholders - usually a specified software tester