

DWA_01.3 Knowledge Check_DWA1

1. Why is it important to manage complexity in Software?

-Maintainability: As software systems grow and evolve, they tend to become increasingly complex. Without proper management, this complexity can quickly spiral out of control, making it difficult to understand and modify the software.

-Team collaboration: Complex software can create communication and collaboration barriers among team members. When software is difficult to understand and modify, it becomes harder for developers to work together effectively. Managing complexity helps create a shared understanding among team members, promotes effective communication, and facilitates collaboration, leading to higher productivity and better software quality.

Debugging and troubleshooting: When the codebase becomes more efficient. Clear modularization and separation of concerns allow developers to isolate problems quickly and narrow down their search for bugs or errors.

2. What are the factors that create complexity in Software?

- Requirements
- Time and cost constraints
- User interaction
- Security

3. What are ways in which complexity can be managed in JavaScript?

Abstraction

Using abstraction techniques to hide complex implementation details and provide simpler interfaces. This can be achieved through functions, classes, or custom abstraction

Proper Naming conventions

Choose meaningful and descriptive names for variables, functions, and classes. Clear naming helps in understanding the purpose and behavior of different components, making the code more readability

Documentation

Helps new developers to understand the code more easily and reduces the learning curve.

4. Are there implications of not managing complexity on a small scale?

Yes, some of the potential implications will be:

Decreased efficiency- Without proper management complexity can create unnecessary obstacles and hinder productivity.

Reduced effectiveness- Complexity can obscure the core objectives and goals of small-scale endeavors. If complexity is not managed it becomes harder to understand and focus on the essential aspects of the task at hand. This can lead to a loss of effectiveness and hinder progress toward desired outcomes.

Higher risk of errors and mistakes- Unmanaged complexity increases the likelihood of errors and mistakes. When things are unnecessarily complex, it becomes easier for miscommunications, compromised quality, and potentially negative consequences for a small-scale project or operation.

5. List a couple of codified style guide rules, and explain them in detail.

Indentation and Formatting

Use consistent indentation to format code blocks and improve readability. It enhances code readability and distinguishes different levels of code hierarchy. Typically a standard indentation is used, such as two or four spaces to align blocks of code. This rule ensures that code is properly formatted and indented making it easier to understand

Naming Convention

Follow consistent naming conventions for variables, functions, and classes. To provide a standard way to name and identify different elements of code.

6. To date, what bug has taken you the longest to fix - why did it take so long?

There was a time I was unable to calculate the minimum and maximum values of different times in an object method. It took me about a whole week to discover it and it just wanted me to first convert the times in milliseconds so that I can be able to get time in whole numbers rather than the time then it was able to calculate the minimum and maximum values from the different time I was given. It just needed logic which I applied at a later stage.
