INFO4057: Software Architecture

Clean Architecture

1. Identify Layers

Exercise: Consider a simple task management application. Identify and list the different layers of Clean Architecture for this application. Provide a brief description of each layer and how they interact.

2. Dependency Rule

Exercise: Explain the dependency rule in Clean Architecture. Describe why it's important and how violating this rule can impact the maintainability of a software system. Provide an example to illustrate the concept.

3. Use Cases

Exercise: Identify three core use cases for a blogging platform. Implement these use cases as separate classes or functions. Discuss how these use cases encapsulate the business logic of the application.

4. SOLID Principles

Exercise: Explain the SOLID principles and how they relate to Clean Architecture. Provide an example for each principle and explain how adhering to these principles contributes to a more maintainable and scalable system.

5. Testing

Exercise: Discuss the importance of unit testing in Clean Architecture. Explain how Clean Architecture facilitates the testing of business logic independently of external dependencies.

6. Evolve the System

Exercise: Imagine adding a new feature to an e-commerce application. Discuss how Clean Architecture principles can help in seamlessly integrating this new feature while maintaining the separation of concerns.

7. Case Studies

Exercise: Research and summarize a real-world case study where Clean Architecture was applied. Discuss the challenges faced, benefits gained, and any notable outcomes. Relate the case study to Clean Architecture principles.

10. Case Studies

Give the Schema of Clean Architecture by Uncle Bob and explain the elements of the different layers