PRACTICE ASSIGNMENT 8

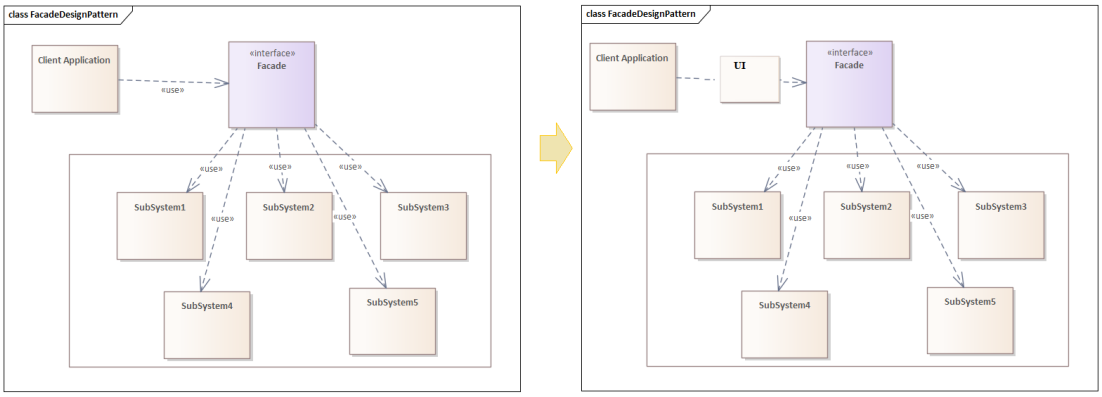
SCREAMING ARCHITECTURE

In this project, we will learn about Screaming architecture by looking at Scenario 2 in your Lab1:

Facade Design Pattern Scenario 2. You're developing an application that needs to write data to files in various formats (CSV, JSON, and XML), optionally compressing and encrypting the data for security and storage efficiency. You want to provide a simple interface for client code, hiding the complexities of format conversions, compression, and encryption.

The term "screaming architecture" is used when we can, just by looking at a new project at a glance, get the core idea of what the project does and what it is about. Then let practice "Screaming architecture" by following some Project Architecture Rules below:

Rule #0: Before getting started let make your project structure more complex by adding UI component.



Rule #1: Refactoring your structure by introducing the following layout:

- Make the directory **src/main/java/features** Application/Library sources code
- src/main/ resources: Library/ Static resources that '/main/java' referencing to.
- src/test/java : Test sources
- src/test/resources: : Test resources that '/test/java' use

Rule #2: Create a separate file for each module.

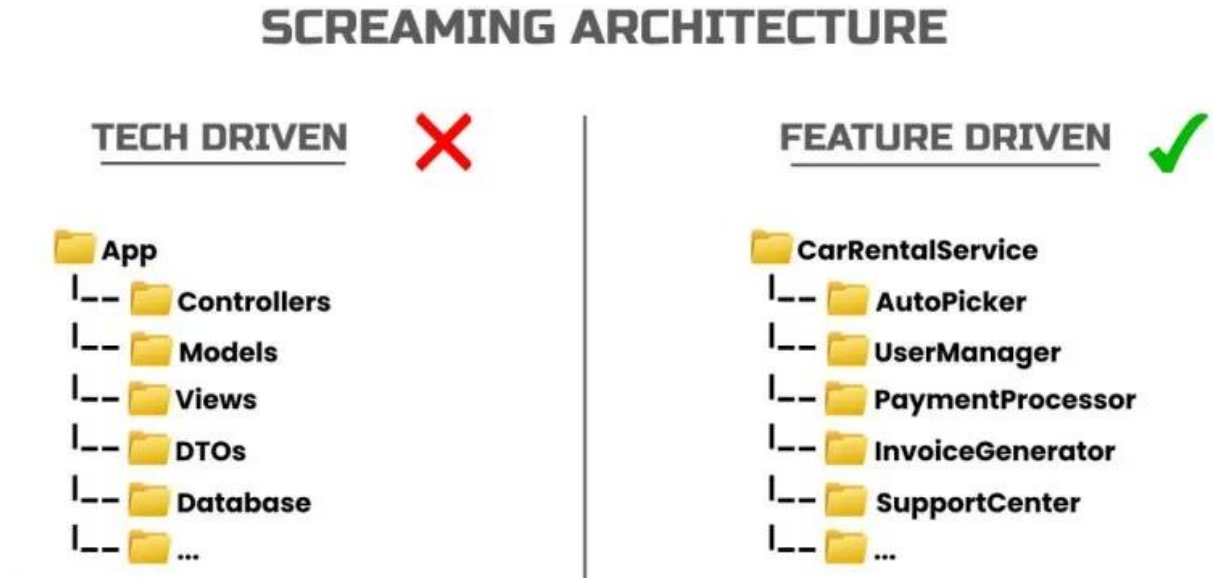
There are a lot of potential drawbacks to having multiple modules in a single file. Having multiple modules in one file can lead to increased complexity. It becomes harder to locate and modify specific sections of code.

Rule #3: Choose your 'style.'

Before moving to next rule, read about File-Naming Convention and pick a style. (Most popular ones are: **kebab-case, snake_case, CamelCase…)**

Rule #4: Use 'your choice of naming' for file/folder names your project folder and modules to respect their Use Case.

Use Cases is what drives the application and is the parts that are prominent when studying both the code and the documentation.



Rule #5: Place UI components inside features/ui directory

Treat every component as a feature, even if that component doesn't do anything besides displaying its UI on the webpage.

~The end~

Your project/ solution should be submitted to Moodle before deadline.