

Jayson Kiggundu

Professor Woolfolk

LSP Spring 2025

4/24/25

#### Question 4

Using a design pattern may not be beneficial when it unnecessarily complicates a solution that could be handled more directly. For instance, in a small-scale project with a narrow scope and few anticipated changes, applying a pattern like Abstract Factory can introduce multiple layers of abstraction—interfaces, concrete classes, factories—that aren't truly needed. This can lead to code that is harder to follow, debug, and maintain, especially for developers who weren't involved in the initial design.

Instead of increasing flexibility, it can make the system rigid because changes might require navigating through extra layers that serve no real purpose. Overuse of patterns can also result in “design bloat,” where the architecture becomes heavier than the problem it's trying to solve. In these cases, a simpler, more straightforward approach is often more effective and easier to manage. Patterns are tools—not rules—and should only be used when the complexity they introduce is justified by clear benefits.