

ARCHITECTURE PATTERNS: A QUICK GUIDE

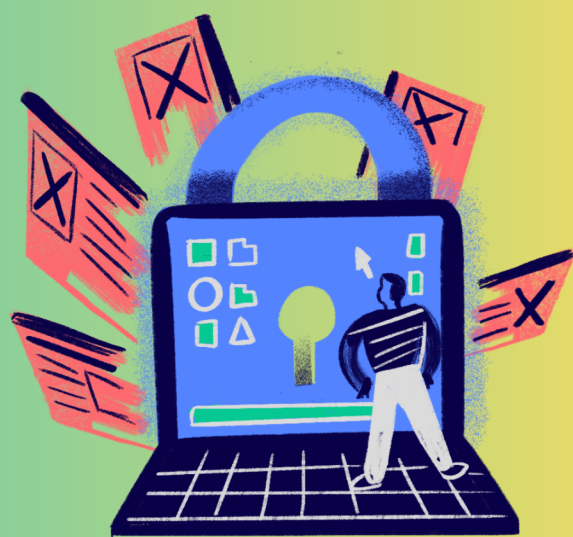


1. Multi-Layered Architecture

- **Structure:** Divides app into layers (Presentation, Business Logic, Data Access).
- **Advantages:**
 - Clear separation of concerns.
 - Easier to maintain.
- **Challenges:**
 - Can increase complexity.

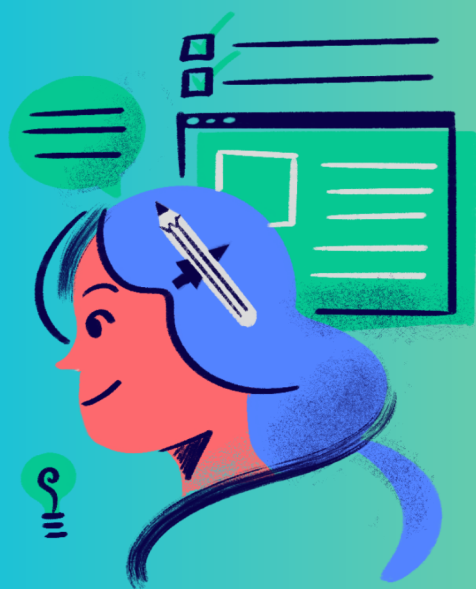
2. Repository Pattern

- **Purpose:** Simplifies data access by abstracting it.
- **Advantages:**
 - Streamlines testing.
 - Promotes modularity.
- **Challenges:**
 - Risk of bloated logic.



3. Client-Server Architecture

- **Concept:** Clients interact with a centralized server for services.
- **Advantages:**
 - Centralized resource management.
 - Scalable design.
- **Challenges:**
 - Server can be a performance bottleneck.



Pipe-And-Filter Architecture

- **Flow:** Data is processed through independent filters connected by pipes.
- **Advantages:**
 - Highly modular.
 - Components are reusable.
- **Challenges:**
 - Potential performance issues with data passing.

