**Coding Principles – NotePlan**

# 1. Coding Guidelines

**Naming Conventions**

* Classes use CamelCase (ex., NoteController, UserService).
* Variables and methods use camelCase (ex., noteTitle, getUserByEmail()).
* Constants use ALL\_CAPS\_WITH\_UNDERSCORES (ex., DEFAULT\_NOTE\_COLOR).

**Code Structure**

* One public class per Java file.
* Clear separation between:
  + Controller (API entry point)
  + Service (business logic)
  + Repository (data access layer)
* Each class should have a single, well-defined responsibility.

**Comments**

* Comments are used only when the logic isn’t immediately obvious.

**Formatting**

* Use of 4 spaces for indentation, not tabs.
* Lines kept under 120 characters.
* Spacing between logical code blocks.

**Error Handling**

* Avoid the overusing of try-catch blocks.
* Use custom exceptions when needed (e.g., UserNotFoundException).

### 2. SOLID Principles

**Single Responsibility Principle (SRP)**  
A class should only have one responsibility.  
Example: NoteService handles only note-related logic, not authentication or user validation.

**Open/Closed Principle (OCP)**  
Software should be open for extension but closed for modification.  
Example: Add new filtering behavior for notes via interfaces like NoteFilterStrategy without changing existing code.

**Liskov Substitution Principle (LSP)**  
Subclasses should be usable in place of their parent classes without causing errors.  
Example: An Admin class that would extend User should work anywhere Note is used.

**Interface Segregation Principle (ISP)**  
Prefer many small, specific interfaces over one large, general one.  
Example: Use separate interfaces like CreateNote, EditNote, and ShareNote instead of one big NoteRepository interface.

**Dependency Inversion Principle (DIP)**  
High-level modules should not depend on low-level modules; both should depend on abstractions.  
Example: NoteController depends on an interface like NoteService, which Spring injects, instead of creating a concrete class directly.