

Implementation (Adhere to MVC, OOP Principles, and Use Design Patterns)

1. MVC Structure

- **Models** - Define the data structure and relationships (e.g., User, Book, Borrow).
- **Controllers** - Handle the request logically (e.g., Book Controller, User Controller).
- **Views** - Present data to the user using Frontend Technologies.

2. OOP Principles

- **Encapsulation** - Keep data and methods in models and controllers.
- **Inheritance** - Extend core Laravel classes (e.g., Controller, Model).
- **Polymorphism** - Use interfaces or abstract classes for different user roles.

3. Design Patterns

- **Repository Pattern** - Abstract the data access layer to promote code reusability and maintainability.
- **Observer Pattern** - Use Laravel's event listeners for actions like sending notifications when a book is borrowed.
- **Factory Pattern** - Create objects (e.g., users, books) in a standardized way using Laravel's factories.