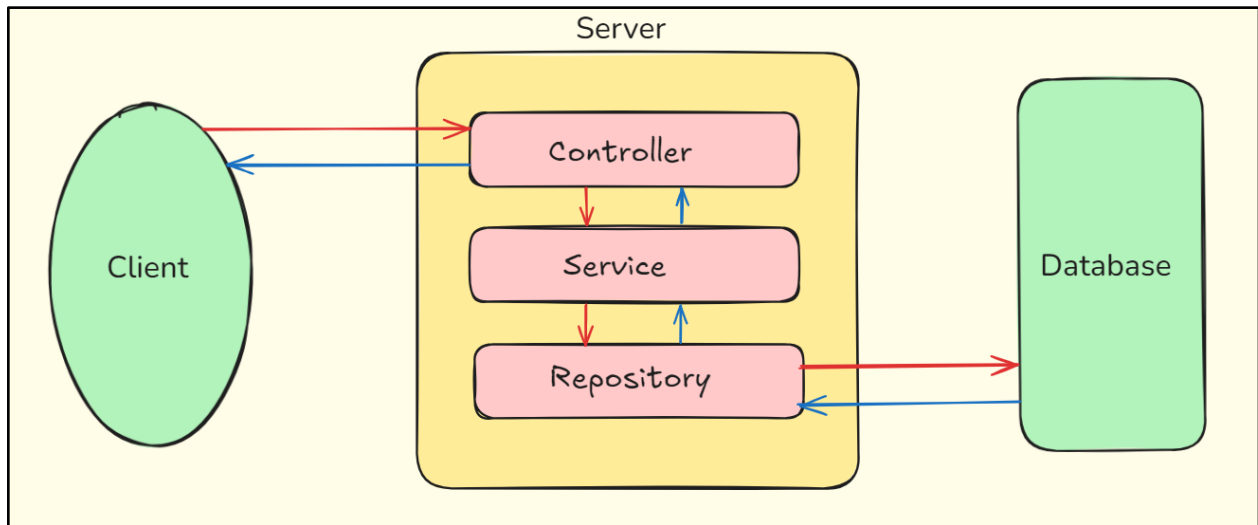


03 - Different Layers

Spring Boot applications are built on the MVC (Model-View-Controller) architectural pattern, which organizes the application into three separate layers, each with a specific role.



1. Controller:

- The Controller is the entry point of the application and acts as an intermediary between the user interface and the backend services.
- It accepts incoming HTTP requests, processes the data, and returns the response to the client.
- Controllers are responsible for handling user interactions and delegating the necessary business logic to the Service Layer.
- It returns the processed data (such as JSON or XML) to the user interface.

Key Annotation: `@RestController` or `@Controller`

2. Service:

- The Service Layer contains the business logic of the application.
- It performs the necessary processing of the data received from the Controller Layer and often interacts with the Repository Layer to retrieve or store data in the database.
- Service layer is also responsible for applying validation, calculations, and complex operations before returning the processed data back to the controller.

Key Annotation: [@Service](#)

3. Repository:

- The Repository Layer (also called the Data Access Object layer) is responsible for interacting with the database.
- It contains methods that perform CRUD (Create, Read, Update, Delete) operations on the data.
- Repository layer is decoupled from the business logic that allows for easier testing and maintenance of the codebase.

Key Annotation: [@Repository](#)

👉 Benefits of Using Layered Architecture:

- **Separation of Concerns:** Each layer is responsible for a specific task, making the code modular and easier to maintain.
- **Testability:** By decoupling the layers, individual layers can be tested separately.
- **Reusability:** Services and repositories can be reused across different controllers or applications.
- **Scalability:** Changes in one layer can be done without affecting the other layers, making the application easier to scale and modify.