Microservices Layer Overview

Microservices Layer Overview

The Microservices layer encapsulates the business logic into loosely coupled, independently deployable services. Each service handles a specific domain within the larger application.

Key Components:

1. **Core.Api**:
   - Handles HTTP requests and routes them to the appropriate service.
   - Manages request validation and response formatting.

2. **Pipeline.Api**:
   - Manages data pipelines for processing large datasets.
   - Implements streaming and batch processing capabilities.

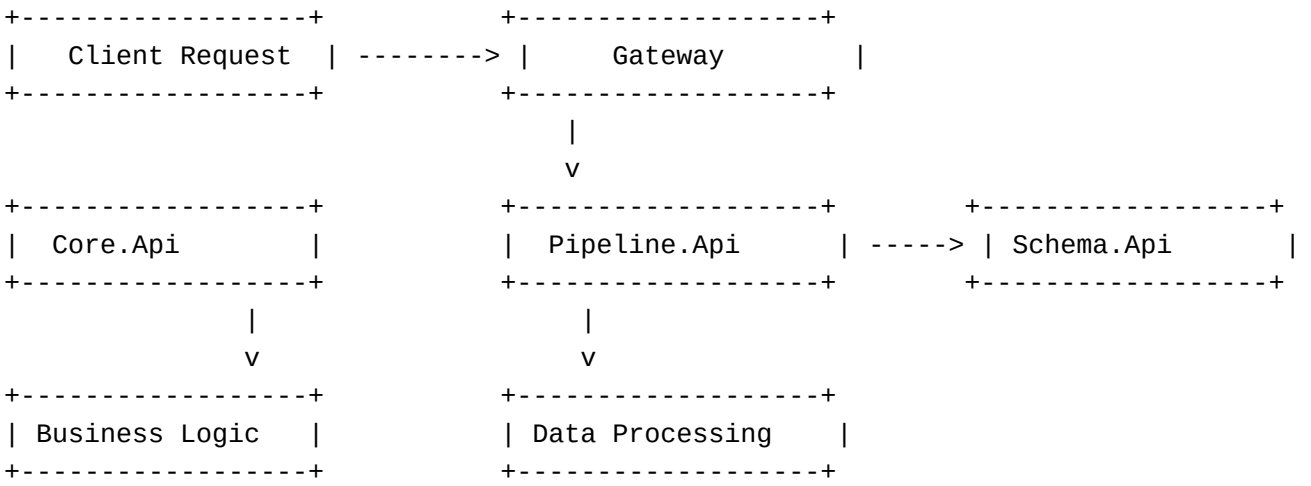
3. **Schema.Api**:
   - Manages schema-related operations, including validation and transformation.
   - Interacts with other services to ensure data integrity.

Design Patterns:

- **Microservices Pattern**: Divides the application into small, independent services.
- **API Gateway Pattern**: Used in conjunction with the Gateway layer to route requests to the appropriate microservice.

ASCII Diagram:

Microservices Layer Overview



Q: What is the role of the Microservices layer?

A: The Microservices layer divides the application into small, independent services, each handling a specific domain.

Q: How do Microservices interact with the Gateway layer?

A: The Gateway layer routes client requests to the appropriate microservice, ensuring efficient request handling and response.

Q: What design patterns are used in the Microservices layer?

A: The Microservices pattern divides the application into smaller, deployable units, while the API Gateway pattern routes requests.