

PV168

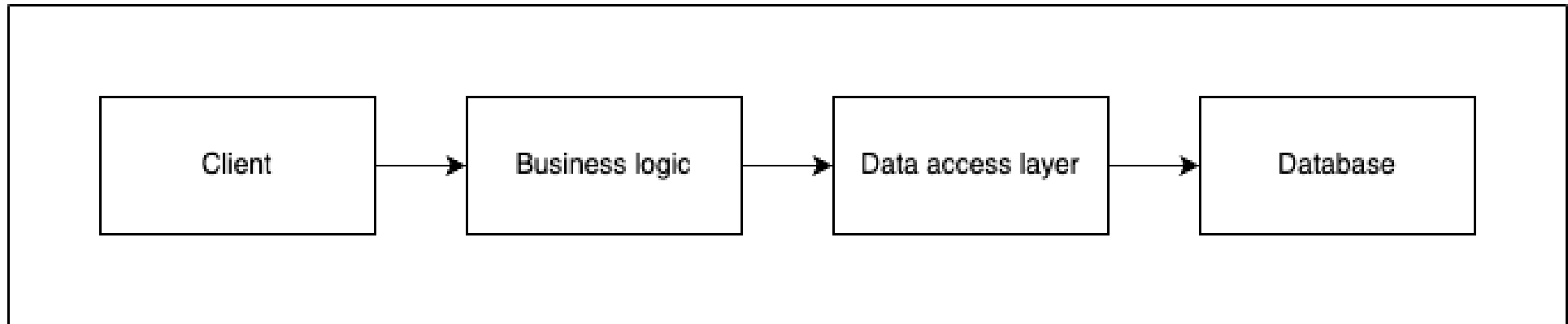
Architectural Tuning

Monolithic Architecture

- Monolithic architecture represents a traditional approach to software design where an entire application is treated as a single, tightly-integrated unit.
- All components and modules are interconnected and interdependent within a single codebase.
- The entire application is typically deployed as a single unit.

Monolithic Architecture

Monolithic architecture diagram



Challenges of Monolithic Architecture

- Limited Scalability
- Technology Stack Limitations
- Deployment Bottlenecks
- Limited Fault Isolation

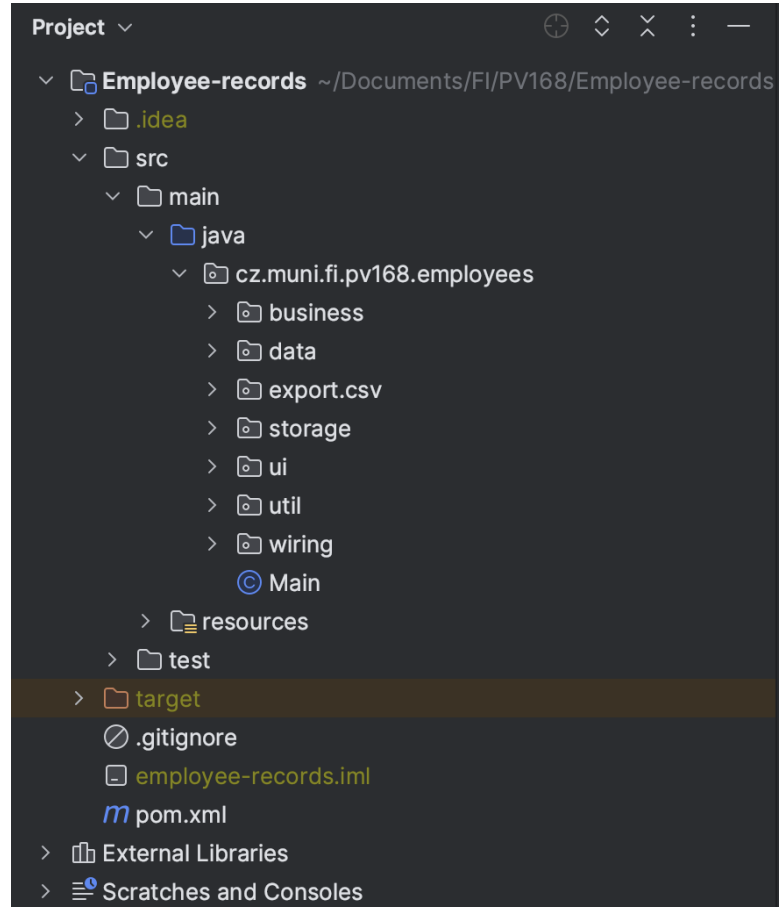
Software Modularization



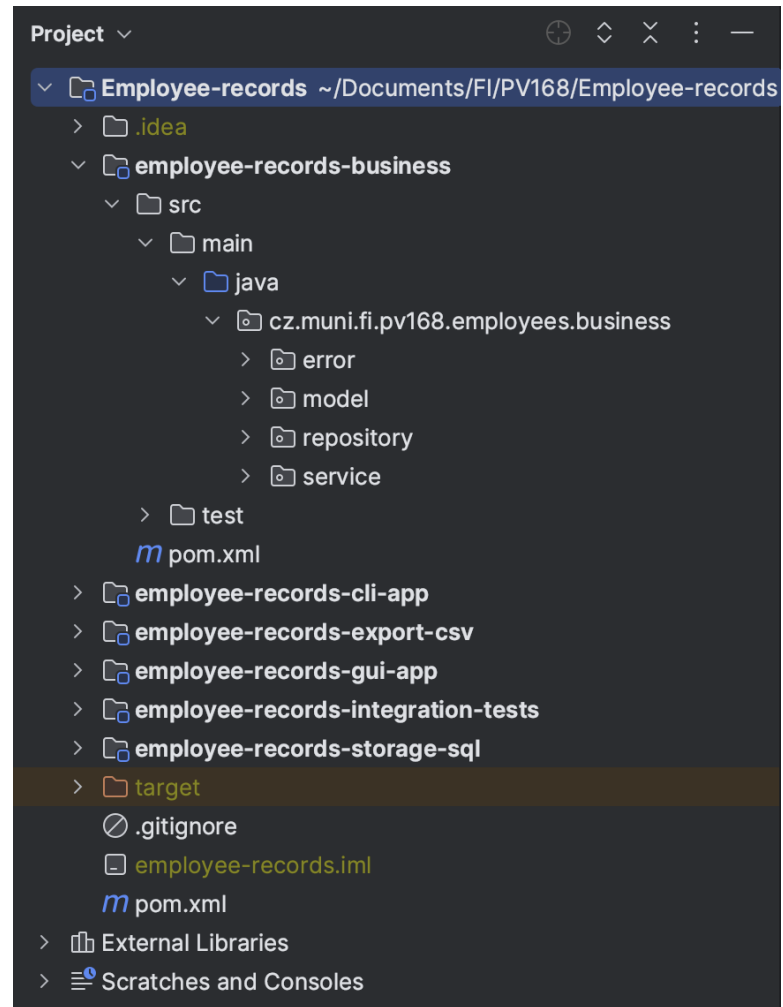
Software Modularization

- Modularization is the process of breaking down the overall functionality of software into smaller, autonomous, and reusable parts called modules.
- The goal is to achieve better organization, maintainability, and code clarity.
- The modules are then combined to form a complete application.

From...



... to



Modularization Advantages

- Enhances code clarity and readability.
- Facilitates code reuse and reduces duplication.
- Eases software management and maintenance.

Modularization Disadvantages

- Increased Complexity in Inter-module Communication
- Potential for Overhead in Module Management
- Dependency Management
- Initial Overhead in Design

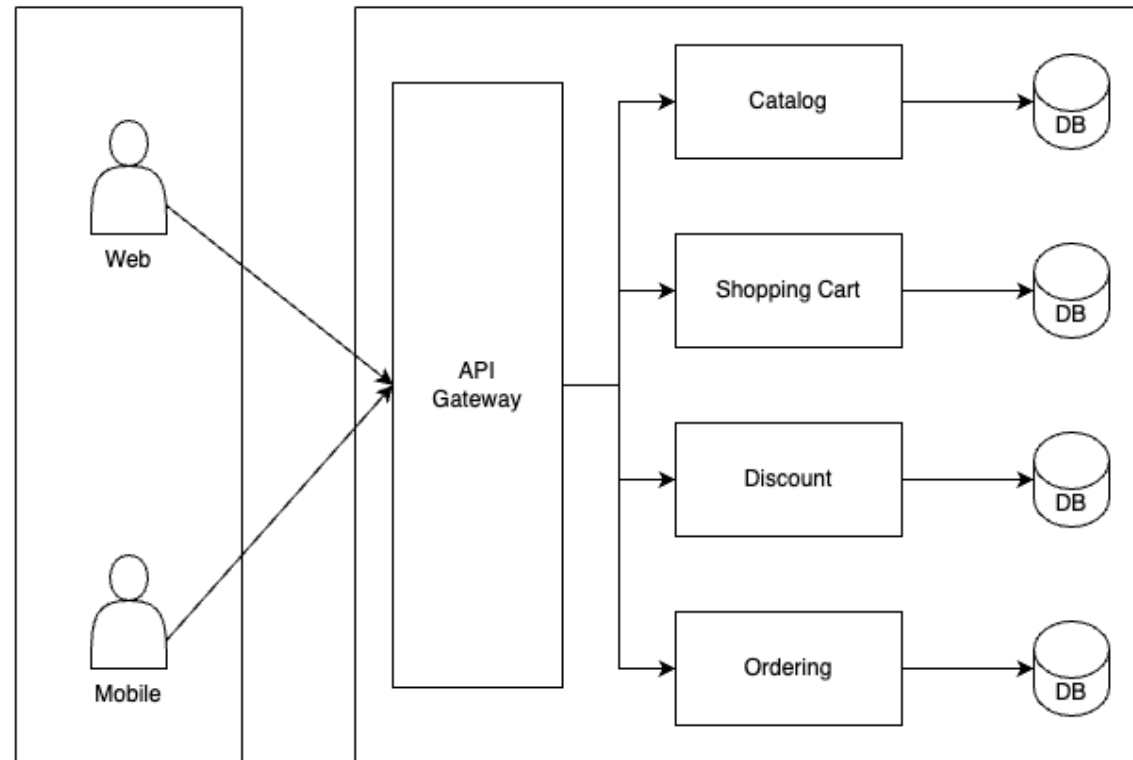
Microservices Architecture

- Service Independence
- Decentralized Data Management
- Scalability
- Decentralized Communication

Microservices Architecture

Client Apps

Microservices



Advantages of Microservices Architecture

- Scalability
- Flexibility and Technology Diversity
- Rapid Development and Deployment
- Fault Isolation
- Improved Team Autonomy

Disadvantages of Microservices Architecture

- Increased Operational Complexity
- Distributed System Challenges
 - Data Consistency
 - Inter-Service Communication
- Testing and Debugging
 - Testing Across Services
 - Debugging Across Services

Communication in Microservices

- Challenges
 - Complex Communication Patterns
 - Latency Management
- Strategies
 - API Gateway
 - Event-Driven Architecture
- Tools
 - Message Brokers

Transaction Management

- Challenges
 - Distributed Transactions
 - Data Consistency
- Strategies
 - Saga Pattern
 - Event Sourcing
- Tools
 - Distributed Database
 - Transaction Monitoring Tools

Testing in Microservices

- Challenges
 - Integration Testing Complexity
 - Data Consistency Testing
- Strategies
 - Contract Testing
 - Chaos Engineering
- Tools
 - Testing Frameworks
 - Containerized Testing

Security in Microservices

- Challenges
 - Identity and Access Management
 - Secure Communication
- Strategies
 - OAuth 2.0
 - JWT (JSON Web Tokens)
- Tools
 - API Security Tools
 - Security Auditing

State Management in Microservices

- Challenges
 - Consistency Across Services
 - Stateful Service Challenge
- Strategies
 - Event Sourcing
 - CQRS (Command Query Responsibility Segregation)
- Tools
 - Stateful Service Frameworks
 - Distributed Cache

Logging in Microservices Architecture

- Challenges
 - Distributed Nature
 - Data Volume
- Strategies
 - Centralized Logging
 - Structured Logging
- Tools
 - Prometheus and Grafana
 - Application Performance Monitoring (APM)