

# Minimap of Topics

- Clean Code
- Programming Paradigms
- Object Oriented Programming
- Design Principles
- Design Patterns
- Architectural Principles
- Architectural Styles
- Architectural Patterns
- Enterprise Patterns

- Structured Programming
- Functional Programming
- Object Oriented Programming

- Model-Driven Design
  - Domain Models
  - Anemic Models
  - Layered Architectures
  - Domain Language
  - Class Invariants
- Paradigm Features
  - Abstract Classes
  - Concrete Classes
  - Scope / Visibility
  - Interfaces
- Primary Principles
  - Inheritance
  - Polymorphism
  - Abstraction
  - Encapsulation

- Component Principles
  - Policy vs Detail
  - Coupling and Cohesion
  - Boundaries
- Domain-Driven Design
  - Model-View Controller
  - Microservices
  - Blackboard Pattern
  - Microkernel
  - Serverless Architecture
  - Message Queues / Streams
  - Event Sourcing

**Note on Listed Topics**  
The roadmap is not an exhaustive list of topics. It's just some of the most important Topics under each category.

# Software Design & Architecture

Visit Backend Developer Roadmap

Clean Code Principles

Programming Paradigms

Object Oriented Programming

Design Principles

Design Patterns

Architectural Principles

Architectural Styles

Architectural Patterns

Structural

Component-Based

Monolithic

Layered

Enterprise Patterns

Visit Backend Developer Roadmap

Find the detailed version of this roadmap along with resources, progress tracking and other resources.  
<https://roadmap.sh>

**References**  
✔ Khalil's Software Design & Architecture Map

- Be Consistent
- Meaningful names over Comments
- Indentation and Code Style
- Keep methods, classes, files small
- Pure functions
- Minimize cyclomatic complexity
- Avoid passing nulls, booleans
- Keep framework code distant
- Use correct constructs
- Tests should be fast and independent
- Use meaningful names
- Organize code by the actor it belongs to
- Command Query Separation
- Keep it simple and refactor often

- Composition over Inheritance
- Encapsulate what varies
- Program against abstarctions
- Hollywood principle
- SOLID
- DRY
- YAGNI

- GoF Design Patterns
- PoSA Patterns

- Event-Driven
- Publish-Subscribe

- Messaging
- Distributed
  - Client-Server
  - Peer-to-Peer

- DTOs
- Identity Maps
- Use Cases
- Repositories
- Mappers
- Transaction Script
- Commands / Queries
- Value Objects
- Domain Models
- Entities
- ORMs