

Advanced .NET Core Web API Topics

1. API Versioning

- URL, query-string and header-based versioning
- Semantic versioning strategies

2. HATEOAS & Hypermedia

- Implementing HAL, JSON-LD or Siren formats
- Dynamic link generation for discoverability

3. gRPC & Protobuf Integration

- High-performance RPC services alongside REST
- Versioning and backward-compatibility in .proto files

4. Authentication & Authorization

- JWT bearer tokens, refresh tokens, and sliding expiration
- OpenID Connect with IdentityServer or Azure AD
- Claims-based policies and custom authorization handlers

5. Health Checks & Diagnostics

- `IHealthCheck` implementations, readiness vs. liveness probes
- Exposing health endpoints and UI dashboards

6. Resilience & Transient-Fault Handling

- Using Polly for retries, circuit breakers, bulkheads
- Timeout and fallback strategies

7. Caching Strategies

- In-memory vs. distributed caching (Redis, NCache)
- Cache-aside, write-through, and expiration policies

8. Rate Limiting & Throttling

- Per-client or per-endpoint limits
- Leaky bucket / token bucket algorithms

9. Logging & Monitoring

- Structured logging with Serilog or Microsoft.Extensions.Logging
- Correlation IDs, request scoping, and distributed tracing (Application Insights, OpenTelemetry)

10. Swagger/OpenAPI Deep Dive

- Custom operation filters, security definitions
- API documentation generation for multiple versions

11. Message-Based Communication

- Integration with RabbitMQ, Kafka, Azure Service Bus
- Request/response vs. publish/subscribe patterns

12. CQRS & MediatR

- Command vs. query segregation
- Pipeline behaviors and notification handlers

13. Performance Tuning

- Benchmarking endpoints with BenchmarkDotNet
- Reducing allocations, tuning Kestrel...

14. Testing Strategies

- Unit tests vs. integration tests with TestServer
- Contract testing using Pact

15. Deployment & CI/CD

- Dockerizing APIs, Kubernetes manifests
- Blue/green and Canary deployments in Azure DevOps or GitHub Actions

Advanced Clean Architecture Topics in .NET

1. Architectural Styles & Layering

- Onion, Hexagonal (Ports & Adapters), and Vertical-Slice architectures
- Trade-offs and composition root best practices

2. Domain-Driven Design (DDD) Patterns

- Entities, Value Objects, Aggregates, Repositories
- Domain Events, Event Sourcing, and CQRS interplay

3. Use Case & Application Services

- Defining Interactors/Use Cases as first-class citizens
- Mapping between DTOs, Domain Models, and Persistence Models

4. Dependency Inversion & IoC

- Advanced DI container features (module registration, decorator patterns)
- Avoiding the Service Locator anti-pattern

5. Specification & Repository Patterns

- Building composable query specifications
- Abstracting data access behind repositories vs. Query objects

6. Anti-Corruption Layer (ACL)

- Protecting your domain from external model changes
- Adapter implementations for legacy systems or third-party APIs

7. Cross-Cutting Concerns

- Centralized error handling (middleware, filters)
- Validation pipelines (FluentValidation, MediatR behaviors)
- Logging, auditing, and metrics

8. Modular Monoliths & Microservices Readiness

- Designing bounded contexts and module boundaries
- Preparing a modular codebase for eventual distributed deployment

9. Testing the Architecture

- Unit testing domain logic in isolation
- Integration testing of the full stack (in-memory DB, test containers)

10. Security in the Layers

- Enforcing authorization at the application layer
- Secure mapping and DTO hygiene

11. Automated Code Generation & Scaffolding

- T4 templates or Roslyn analyzers for repetitive boilerplate
- Ensuring consistency in layers

12. Continuous Refactoring & Metrics

- Using architecture-fitness functions (e.g., SonarQube rules)
 - Measuring coupling, cohesion, and cyclomatic complexity
-