

Yazılım Mimarisi Eğitim Notları

Yazılım Mimarisi ve Geliştirme

[Yazılım Mimarisi Nedir?](#)
[Neden Sürece İhtiyaç Var?](#)
[Yazılım Geliştirme Fazları](#)
[Yazılım Geliştirme Modelleri](#)
[Agile Frameworkler \(Kanban, XP, Scum\)](#)
[Yazılım Geliştirme Prensipleri](#)
[AWS Well-Architected Framework Nedir ?](#)

Prensipler ve Kavramlar

[Conway Yasası](#)
[Amazon 2 PT \(Two Pizza Teams\)](#)
[Amazon 2 PT 'ler arası İletişim Kuralları](#)
[MVP \(Minimum Viable Product\) Nedir?](#)
[SaaS Müşterilerinin Yaşam Döngüsü](#)
[Örnek Bir MindMap Çalışması](#)
[Sketch, Wireframe, Mockup, Prototype](#)
[S.O.L.I.D Prensipleri](#)
[TDD\(Test\) vs BDD\(Behaviour\)](#)
[Reactive Manifesto](#)
[12 Factor App Uygun Geliştirme Nedir ?](#)
[JIRA'daki Scrum Board'una Bakış](#)
[Pomodoro Tekniği \(Zaman Yönetimi\)](#)
[ACID](#)
[CAP Teoremi](#)
[ACID, BASE, CAP](#)
[RAID](#)
[Blockchain](#)

Mimari Örüntüler

[Mimari Örüntülere Giriş](#)
[Microservices Mimarisi](#)
[Microfrontends Mimarisi](#)
[SOA Mimarisi](#)
[Serverless Architecture \(FaaS\)](#)
[Katmanlı Uygulama Mimarisi](#)
[Flux Patten \(Redux, vb...\)](#)
[Saga Pattern](#)
[MVC \(Model View Controller\)](#)
[Yazılım Mimarisinde Brokerlik](#)
[Event-Driven Architecture](#)
[Blackboard System](#)
[Pipe and filter architecture](#)
[Entegrasyon Örüntüler](#)

Tasarım Örüntüleri

[Tasarım Örüntüleri](#) **Creational Patterns**

[Abstact Factory](#)
[Builder Pattern](#)
[Factory Method](#)
[Prototype](#)
[Singleton](#)

Structural Patterns

[Adapter](#)
[Bridge](#)
[Composite](#)
[Decorator](#)
[Facade](#)
[Flyweight](#)
[Proxy](#)

Behavioral Patterns

[Chain Of Responsibility](#)
[Command](#)
[Interpreter](#)
[Mediator](#)
[Memento](#)
[Observer](#)
[State](#)
[Strategy](#)
[Template Method](#)
[Visitor](#)