

# Claude Code ile Agentic Yazılım Geliştirme Eğitimi

## Eğitim Açıklaması

Subagent ve MCP Entegrasyonlu 2 Günlük İleri Seviye Program

Bu eğitimin amacı; yazılım ekiplerinin Claude Code'u yalnızca bir terminal AI aracı olarak değil, subagent yapıları ve Model Context Protocol (MCP) entegrasyonu ile genişletilebilir bir agentic geliştirme platformu olarak kullanabilmesini sağlamaktır.

Eğitim sonunda katılımcılar; reasoning-first geliştirme yaklaşımını uygulayabilir, modüler subagent mimarileri tasarlayabilir, MCP üzerinden dış sistem entegrasyonu yapabilir ve kurumsal ölçekte güvenli AI agent mimarisi oluşturabilir.

### Eğitimin Çıktıları

- Subagent tabanlı geliştirme yaklaşımı
- MCP ile dış sistem entegrasyon yetkinliği
- Kurumsal AI agent mimarisi tasarım becerisi
- Token ve maliyet kontrol stratejisi
- Güvenli ve denetlenebilir AI kullanım modeli

## Eğitim İçeriği Nedir?

### 1. Gün - Claude Code Temelleri ve Subagent Mimarisi

## 1. Reasoning-First Geliştirme Yaklaşımı

- Plan-first development
- Step-by-step reasoning
- Uzun context avantajı
- Deterministik üretim teknikleri
- Kod üretmeden önce plan oluşturma yaklaşımı

## 2. Subagent Kavramı

- Ana agent (orchestrator) mantığı
- Görev bazlı subagent yapısı
- İzole bağlamlı çalışma modeli
- Role-based task delegation

## 3. Subagent Türleri

- Planner Agent
- Coder Agent
- Test Agent
- Reviewer Agent
- Refactor Agent

## 4. Subagent Orchestration Best Practices

- Task decomposition yaklaşımı
- Net input/output formatı tanımlama
- Structured output zorunluluğu
- Plan → Execute → Validate döngüsü
- Reasoning loop risklerini yönetme

## 2. Gün - MCP Entegrasyonu ve Kurumsal Agent Mimarisi

### 5. Model Context Protocol (MCP) Nedir?

- MCP mimarisi ve mantığı
- Tool exposure kavramı
- Context injection
- Claude Code + MCP entegrasyon yapısı

### 6. MCP ile Dış Sistem Entegrasyonu

- Git repository erişimi
- Dosya sistemi entegrasyonu
- API çağrıları
- Database sorgulama
- Log analizi

## 7. Agent + MCP Güvenlik Modeli

- Yetki sınırlandırma
- Tool scoping
- Prompt injection riskleri
- Output validation
- Audit ve log yaklaşımı

## 8. Kurumsal Agent Mimarisi Tasarımı

- Monolitik vs modüler agent yaklaşımı
- Central orchestrator tasarımı
- Subagent lifecycle yönetimi
- Token ve maliyet kontrolü
- On-prem vs cloud model stratejisi

## 9. Final Uçtan Uca Senaryo

- Requirement analizi
- Planner subagent tasarımı
- Kod ve test üretimi
- MCP ile veri entegrasyonu
- Security kontrolü
- Rapor ve dokümantasyon üretimi