

Cursor ile AI Native Yazılım Geliştirme Eğitimi

Süre: 2 Gün

Cursor ile AI Native Yazılım Geliştirme Eğitimi Hakkında

Yazılım ekipleri için 2 günlük kurumsal Cursor eğitimi

Cursor ile AI Native Yazılım Geliştirme Eğitimi, yazılım geliştirme ekiplerinin yapay zekâyı yalnızca kod tamamlama amacıyla değil; proje analizi, özellik geliştirme, test, hata ayıklama ve kod iyileştirme süreçlerinde kontrollü biçimde kullanmasına odaklanır.

Katılımcılar Cursor arayüzünü ve Agent Mode kullanımını öğrenir; proje bağlamını yönetme, birden fazla dosyayı etkileyen değişiklikleri değerlendirme, test üretme ve yapay zekâ tarafından önerilen kodu doğrulama üzerine uygulamalar yapar. Eğitim, bu pratiklerin ekip standartlarına ve kurumsal yazılım geliştirme süreçlerine nasıl uyarlanacağını da ele alır.

Programda Model Context Protocol (MCP) entegrasyonu da işlenir. Bu konuyu ayrıca derinleştirmek isteyen ekipler Model Context Protocol Eğitimi içeriğini inceleyebilir. Kod tabanlı agent geliştirme yaklaşımına odaklanan ekipler için Claude Code ile Agentic Yazılım Geliştirme Eğitimi de ilgili bir programdır.

Eğitim sonunda katılımcılar:

- AI Native yazılım geliştirme yaklaşımını ve Cursor'ın geliştirme sürecindeki kullanım alanlarını açıklayabilir.
- Cursor Chat, satır içi düzenleme ve Agent Mode özelliklerini günlük geliştirme görevlerinde kullanabilir.
- Proje bağlamını seçerek büyük kod tabanlarında daha kontrollü çalışmalar yürütebilir.
- Çok dosyalı değişiklikleri, refactoring önerilerini ve üretilen testleri gözden geçirebilir.
- Yapay zekâ destekli kod üretiminde güvenlik, kalite ve ekip içi kullanım kurallarını değerlendirebilir.

Eğitim İçeriği

1. Gün: Cursor Temelleri ve Günlük Geliştirme Pratikleri

1. AI Native Development Nedir?

- Geleneksel IDE ile AI-first IDE yaklaşımının karşılaştırılması

- AI Driven Coding modeli
- Prompt → Plan → Kod → Test döngüsü
- Cursor'ın yazılım geliştirme sürecindeki yeri

2. Cursor Arayüzü ve Temel Özellikler

- Chat panelinin kullanımı
- Satır içi kod üretimi ve düzenleme
- Mevcut kodu açıklama ve hata düzeltme
- Değişiklikleri diff üzerinden inceleme ve yönetme

3. Agent Mode ve Context Yönetimi

- Agent Mode ile görev yürütme
- Proje yapısını analiz etme
- Birden fazla dosyayı kapsayan refactoring çalışmaları
- İlgili dosyaları ve proje bağlamını seçme
- Bağlam ve token kullanımını yönetme

4. Günlük Kullanım Senaryoları

- Yeni bir özellik geliştirme
- Mevcut kod tabanını anlama
- Birim testi oluşturma
- Kod dokümantasyonu hazırlama
- Refactoring önerilerini değerlendirme
- MCP entegrasyonunun geliştirme akışındaki kullanımını inceleme

2. Gün: İleri Seviye Cursor Kullanımı ve Kurumsal Adaptasyon

5. Büyük Kod Tabanlarında Cursor Kullanımı

- Monorepo yapılarında çalışma
- Bağlam sınırlarını dikkate alarak görev tanımlama
- Kırılma riski taşıyan değişiklikleri belirleme
- Büyük kod tabanları için refactoring stratejileri

6. Test, Debug ve Kod Kalitesi

- Yapay zekâ desteğiyle birim testi hazırlama
- Uç durumları belirleme ve test kapsamını değerlendirme
- Hata ayıklama sürecinde Cursor kullanımı
- Performans iyileştirme önerilerini inceleme
- Clean Code ilkeleri doğrultusunda kodu gözden geçirme

7. Güvenlik ve Kurumsal Kullanım

- Yapay zekâ tarafından üretilen kodu doğrulama
- Olası güvenlik açıklarını ve riskli değişiklikleri inceleme
- Regülasyona tabi ortamlarda kullanım gereksinimlerini değerlendirme
- Bulut ve kurum içi model stratejilerini veri akışı ve kurum politikaları açısından karşılaştırma

8. Takım Bazlı Cursor Kullanım Modeli

- Ekip için Cursor kullanım politikası taslağı oluşturma
- Tekrarlanabilir görevler için prompt şablonları hazırlama
- Kod inceleme sürecinde yapay zekâ desteğini kullanma
- Sprint planlamasında yapay zekâ destekli analizden yararlanma

9. Final: Uçtan Uca Geliştirme Senaryosu

- Gereksinimleri analiz etme
- Teknik çözümü planlama
- Kod üretme ve değişiklikleri inceleme
- MCP entegrasyonu senaryosunu değerlendirme
- Testleri hazırlama ve çalışmaları doğrulama
- Refactoring ve güvenlik kontrollerini uygulama
- Geliştirilen çözümü dokümante etme

Bu Kurumsal Eğitime Kimler Katılmalı?

Eğitim; Cursor kullanımını geliştirme süreçlerine dahil etmek isteyen yazılım geliştiriciler, teknik liderler, yazılım mimarları, test otomasyonu uzmanları ve yazılım ekipleri için uygundur. Kurumlar programı, ekiplerinin geliştirme pratikleri ve kullanım politikaları üzerine ortak bir yaklaşım oluşturmak amacıyla değerlendirebilir.

Eğitim Katılım Ön Koşulları

Katılımcıların temel seviyede yazılım geliştirme bilgisine sahip olması önerilir.