



(*) önemli not: yazının sonunda.

"Yapay Zeka "sizin" yerinizi almayacak!" Bunu sizi rahatlatmak için veya YZ karşıtı olduğumuz için de söylemiyoruz. Aksine, biz yapay zekayı her gün kullanıyoruz. Claude, ChatGPT, Gemini gibi araçlar günlük çalışma akışımızın bir parçası ve YZ destekli yazılım geliştirmeyi yaygın biçimde kullanıyoruz. Neleri yapabildiklerini biliyoruz. **Yapamadıkları en önemli şey ise, siz olmadan çalışabilmeleri.**

Geliştiricilerin neden endişelendiğini anlıyoruz çünkü sürekli aynı başlıkları görüyoruz: "YZ tüm kodu yazacak", "CoPilot junior geliştiricilerin yerini alıyor", "ChatGPT artık uygulama geliştiriyor". Bunların belirli ölçüde doğruluk payı var. Yapay zeka gerçekten kod yazabiliyor, uygulama iskeleti oluşturabiliyor ve tekrar eden işleri insanlardan çok daha hızlı yapabiliyor. **Ancak "kod üretebilmek" ile "bir yazılım geliştiricisinin yerini alabilmek" arasında çok büyük bir fark var. İşte sizin gerçek değerinizi oluşturacak nokta da bu fark.**

İnsanların YZ hakkında en çok yanıldığı konu, onu tamamen bağımsız çalışan bir sistem olarak görmeleri. Sanki problemi veriyorsunuz, kahvenizi içmeye gidiyorsunuz ve döndüğünüzde kusursuz bir ürün sizi bekliyor. Gerçekte işler kesinlikle böyle değil. YZ'nın sürekli insan yönlendirmesine ihtiyacı var. Doğru bağlamlara ihtiyacı var. Sistem ne yapacak? mevcut mimari nasıl? kısıtlar neler? güvenlik, performans ve ölçeklenebilirlik gereksinimleri neler? Bu bağlamları sağlamazsanız doğru sonuç alamazsınız.

Üstelik gerekli tüm bilgileri eksiksiz verseniz bile YZ yine de hata yapabilir. Gereksinimleri tam olarak anlamış gibi görünür, kodu derler (compile eder), ilk bakışta doğru görünür, fakat tasarım ilkelerini ihlal eden küçük kararlar alabilir. Daha kötüsü, bunu size söylemeyebilir. Son derece kendinden emin görünür :) **Sonuçta eğer çıktıyı dikkatle incelemeszeniz, bu küçük hatalar haftalar sonra üretim ortamında ciddi problemlere dönüşebilir. Bunun sonucunda yaptığınız işleri geri almak zorunda kalırsınız ki bu da zaman kaybı, maliyet ve risktir.**

Bu arada bu durum Claude'a ya da ChatGPT'ye özgü değildir. Bu, büyük dil modellerinin çalışma biçiminin doğal sonucudur. **Bunlar deterministik sistemler değildir; olasılıksal tahmin modelleridir.** Bir sonraki kelimeyi tahmin ederler. Bazen de bu tahminler yanlış olur. Dolayısıyla YZ'nın ürettiği kodu doğrulamadan güvenemezsiniz. **Doğru olanın ne olduğunu anlayabilecek bilgiye hala sizin sahip olmanız gerekir. Mimari bakış açısı, test etme becerisi ve teknik muhakeme hala geliştiricinin sorumluluğudur.**

İkinci önemli nokta: yazılım geliştirme yalnızca kod yazmak değildir. Esas iş, problemi doğru anlamaktır. Paydaşlarla konuşur, eksik noktaları ortaya çıkarır, yanlış gereksinimleri düzeltir, teknik yaklaşımları değerlendirir ve gerektiğinde "bu yaklaşım doğru değil" diyebilirsiniz. YZ bunların hiçbirini gerçekten yapamaz. O yalnızca sorulan soruya cevap verir, sorulması gereken soruyu kendisi keşfedemez.

NEDEN YZ KULLANMALI

Buna rağmen **YZ'nın olağanüstü başarılı olduğu alanlar var.** Hata ayıklamada saatlerce sürecektir araştırmaları dakikalara indirebilir. Kod incelemesinde performans, güvenlik ve okunabilirlik açısından değerli öneriler sunabilir. Tekrarlayan kodları çok hızlı oluşturabilir. Kısacası **YZ, iyi geliştiricileri çok daha üretken hale getirir. Basit işleri hızlandırır ve geliştiricilerin karmaşık problemlere daha fazla odaklanmasını sağlar.**

Dolayısıyla bizce değişen şey, geliştiricilerin ortadan kalkması değil, ekiplerin verimliliğidir. YZ kullanan ekipler eskiden yirmi kişinin yaptığı işi beş on kişiyle yapabiliyor. Bunun nedeni insanların gereksiz hale gelmesi değil, her geliştiricinin daha üretken olmasıdır. **Bu nedenle geleceğin en değerli geliştiricileri, YZ'yi en iyi kullananlar olacaktır.** Etkili prompt yazabilen, YZ çıktısını eleştirel gözle değerlendirebilen ve kendi teknik bilgisini geliştirmeyi sürdüren kişiler öne çıkacaktır.

Sonuç olarak YZ geliştiricilerin yerini almayacak. Ancak YZ'yi etkili kullanan geliştiriciler, kullanmayanların önüne geçecek. **Yani asıl soru "YZ işimi elimden alacak mı?" değil, tam tersi "ben YZ ile birlikte çalışan daha iyi bir geliştirici nasıl olabilirim?" sorusudur.**

(*) CTIS bölümü olarak derslerde ve projelerde YZ kullanımını uzun zamandır destekliyoruz. Üniversitemiz de 2026-2027 Akademik Yılı'ndan itibaren bütün üniversite öğrencilerinin alabileceği bir Yapay Zeka Yetkinliği dersi açıyor.