

Medium

Search



Frontend Geliştirmeye Başlarken Öğrendiğim En Önemli Şeyler
HTML, CSS, JavaScript, React, Tailwind CSS ve gerçek proje deneyimleriyle frontend geliştirme yolculuğumdan öğrendiklerim.
İnternette kullandığımız uygulamalara bakınca her şey çok kolay görünüyordu. Bir butona tıklıyoruz, sayfalar değişiyor, bilgiler anında geliyor. Ama frontend geliştirmeye başladıktan sonra bunun arkasında ne kadar fazla emek ve planlama olduğunu fark ettim.
İlk başlarda amacım sadece web sitesi yapabilmektir. Fakat zaman geçtikçe aslında işin sadece ekrana bir şeyler yerleştirmekten ibaret olmadığını gördüm. Yazılan kodun okunabilir olması, tekrar kullanılabilirliği ve kullanıcıya iyi bir deneyim sunması da en az çalışılan bir proje geliştirmek kadar önemliymiş.
Bu yazıda, frontend öğrenirken bana en çok katkı sağlayan konulara ve bu süreçte

edindiğim deneyimlere kısaca değinmek istiyorum.

4 min read · Jul 13, 2026



Aslinurgunduzalp

Follow



Share

HTML, CSS ve JavaScript ile Temeli Oluşturmak

Frontend öğrenmeye başladığımda ilk tanıştığım teknolojiler HTML ve CSS oldu. HTML sayesinde sayfanın iskeletini oluşturmayı, CSS ile de bu yapıyı daha düzenli ve estetik hale getirmeyi öğrendim.

İlk yaptığım projeler oldukça basitti. Başlıklar, kartlar, menüler ve birkaç butondan oluşan sayfalar hazırlıyordum. Ancak bu küçük projeler sayesinde aslında sağlam bir temel oluşturduğumu sonradan fark ettim.

Daha sonra JavaScript öğrenmeye başladığımda web sayfalarının gerçekten “canlanmaya” başladığını gördüm. Kullanıcı etkileşimleri, formlar, filtreleme işlemleri ve dinamik içerikler artık mümkün oluyordu.

Başlangıçta özellikle asenkron işlemler, API istekleri ve DOM manipülasyonu biraz karmaşık gelmişti. Fakat farklı uygulamalar geliştirdikçe bu konular daha anlaşılır hale geldi.

TypeScript ile Daha Düzenli Kod Yazmak

JavaScript’i öğrendikten sonra TypeScript kullanmaya başladım.

İlk zamanlarda “Neden ekstra kod yazıyoruz?” diye düşündüğüm olmuştu. Ama proje büyüdükçe veri tiplerinin kontrol edilmesinin aslında ne kadar faydalı olduğunu gördüm.

Yanlış veri gönderildiğinde daha uygulama çalışmadan hata alınabilmesi özellikle büyük projelerde büyük avantaj sağlıyor. Ayrıca ekip halinde çalışırken herkesin aynı kod düzenini takip etmesi de çok daha kolay oluyor.

React ile Component Mantığını Öğrenmek

React öğrenmeye başladığımda en hoşuma giden konu component yapısı oldu.

Eskiden uzun HTML dosyaları hazırlarken şimdi her bölümü ayrı bir component olarak geliştirebiliyordum. Header, Footer, Navbar veya Product Card gibi yapıları

tekrar tekrar kullanabilmek hem kod tekrarını azaltıyor hem de projeyi daha düzenli hale getiriyor.

React kullanırken en çok öğrendiğim konulardan bazıları şunlar oldu:

Component mantığı

Props kullanımı

State yönetimi

Event işlemleri

Hook yapıları

Sayfalar arası veri aktarımı

İlk başta state mantığını anlamakta biraz zorlandım. Ama birkaç proje geliştirdikten sonra verinin componentler arasında nasıl aktığını daha rahat anlamaya başladım.

Tailwind CSS ile Daha Hızlı Arayüz Geliştirmek

CSS yazmayı sevsem de bazen uzun dosyalar oluşabiliyordu.

Tailwind CSS kullanmaya başladıktan sonra birçok işlemi daha kısa sürede yapabildiğimi fark ettim. Hazır utility class yapısı sayesinde ayrı CSS dosyaları yazmadan bile düzenli tasarımlar oluşturabiliyorum.

Bir diğer önemli konu ise responsive tasarım.

Bugün kullanıcıların büyük kısmı uygulamaları telefon üzerinden kullanıyor. Bu yüzden geliştirdiğim projelerde sadece masaüstünü değil, tablet ve mobil ekranları da düşünmeye çalışıyorum.

Responsive tasarım artık ekstra bir özellik değil, olması gereken temel bir gereklilik gibi geliyor.

Get Aslinurgunduzalp's stories in your inbox

Join Medium for free to get updates from this writer.

Enter your email

Subscribe

☐

Remember me for faster sign in

API Kullanımı ve State Yönetimi

Gerçek projelerde en çok karşıma çıkan konulardan biri API kullanımı oldu.

Önceden hazırladığım örnek projelerde veriler sabitti. Daha sonra API kullanmaya başladığımda verilerin sunucudan geldiğini ve sürekli değişebildiğini öğrendim. Genellikle Axios veya Fetch API kullanarak veri çekme işlemleri yaptım. Bu süreçte sadece veriyi ekrana yazdırmanın yeterli olmadığını da öğrendim. Örneğin;

- Yükleniyor ekranı göstermek,
- Hata durumlarını yönetmek,
- Boş veri geldiğinde kullanıcıyı bilgilendirmek

gibi detaylar kullanıcı deneyimini doğrudan etkiliyor.

Aynı zamanda proje büyüdükçe state yönetiminin neden önemli olduğu da daha net anlaşılmaya başlıyor.

Testing, Performans ve Deployment

Bir proje çalışıyor olabilir ama bu onun tamamen hazır olduğu anlamına gelmiyor. Testing konusu sayesinde geliştirilen özelliklerin kontrol edilmesinin ne kadar önemli olduğunu öğrendim. Henüz bu alanda kendimi geliştirmeye devam ediyorum ancak test yazmanın özellikle büyük projelerde büyük kolaylık sağladığını düşünüyorum.

Performans tarafında ise gereksiz render işlemlerini azaltmak, görselleri optimize etmek ve uygulamanın daha hızlı açılmasını sağlamak kullanıcı deneyimini ciddi şekilde etkiliyor.

Proje tamamlandıktan sonra deployment süreci de işin ayrı bir aşaması oluyor.

Özellikle React projelerinde Vercel kullanarak uygulamaları internet ortamında yayınlamak oldukça pratik geliyor. Bilgisayarımda çalışan bir projeyi birkaç dakika içinde herkesin erişebileceği hale getirmek benim için en keyifli anlardan biri oluyor.

Takım Çalışması Gerçekten Çok Şey Öğretiyor

Bence frontend geliştirmede sadece kod yazmak yeterli değil.

Takım halinde çalışırken iletişim kurabilmek, görev paylaşımı yapmak ve yapılan değişiklikleri doğru şekilde yönetmek de sürecin önemli bir parçası.

Bu süreçte Git ve GitHub kullanmayı öğrendim.

Özellikle branch yapısıyla çalışmak, merge işlemleri yapmak ve ekip arkadaşlarının geliştirdiği kodlarla birlikte ilerlemek gerçek proje deneyimi kazanmamı sağladı. Zaman zaman merge conflict gibi problemlerle de karşılaştım. İlk başta çözmesi zor görünse de mantığını öğrendikten sonra bu süreçlerin aslında yazılım geliştirmenin doğal bir parçası olduğunu anladım.

Sonuç

Frontend geliştirmeye başladığım günden bugüne kadar öğrendiğim en önemli şey, bu alanın sürekli geliştiği oldu.

Her yeni proje bana yeni bir konu öğretiyor. Bazen küçük bir hata çözmek bile saatler sürebiliyor ama sonunda çözüme ulaştığımda çok şey öğrenmiş oluyorum. Şu ana kadar HTML, CSS, JavaScript, TypeScript, React, Tailwind CSS, API entegrasyonları ve deployment süreçleri üzerine çalışma fırsatı buldum. Bunların hepsi bana sadece teknik bilgi değil, aynı zamanda problem çözme becerisi de kazandırdı.

Önümüzdeki dönemde testing, performans optimizasyonu ve modern frontend mimarileri konusunda kendimi daha da geliştirmeyi hedefliyorum. Çünkü yazılım dünyasında öğrenilecek şeylerin hiçbir zaman bitmediğini düşünüyorum. Teşekkürler.

Etiket Önerileri

#Frontend #React #JavaScript #WebDevelopment #Programming

Html Css

JavaScript

Frontend Development



Follow

Written by Aslinurgunduzalp

0 followers · 1 following

