

Open in app ↗

Sign up

Sign in

Medium

Search



MERN Stack Nedir ?

4 min read · Just now



Turkergamze

Follow



Share

Teknoloji Yığını (Tech Stack) Nedir?

Bir web sitesine girdiğinizde veya bir mobil uygulamayı açtığınızda arka planda çalışan bir sistem vardır. İşte bu sistemin tamamına yazılım dünyasında **Teknoloji Yığını** denir.

Teknoloji Yığını (Tech Stack); bir web sitesi veya uygulamanın hayata geçirilmesi, çalıştırılması ve ölçeklendirilmesi için bir araya getirilen programlama dilleri, kütüphaneler, çerçeveler (frameworks), veritabanları, API'ler ve sunucu araçlarının oluşturduğu yapısal bütündür. Kısacası, projenin dijital temelidir.



MERN Stack: Modern Web Geliştirmenin Güçlü Dörtlüsü

Bugün modern web projelerine baktığımızda, özellikle her katmanda JavaScript kullanma kolaylığı sunduğu için MERN Stack ön plana çıkıyor. MERN; MongoDB, Express.js, React ve Node.js teknolojilerinin bir araya gelmesiyle oluşan uçtan uca bir geliştirme mimarisidir.

- ***MongoDB (M): Verilerin güvenli ve esnek bir şekilde depolanacağı NoSQL veritabanı.***
- ***Express.js (E): Node.js ile birlikte web servis (API) geliştirilmesini kolaylaştıran framework.***
- ***React (R): kullanıcı arayüzleri (UI) tasarlamaya yardımcı olan popüler bir ön yüz kütüphanesidir.***

- *Node.js (N): Sunucu tarafının JavaScript dili ile geliştirilmesini sağlayan güçlü çalışma zamanı ortamı.*



MongoDB

MongoDB, 2009 yılından bu yana geliştiricilerin en sevdiği NoSQL veri tabanlarından biri. Klasik SQL veri tabanlarındaki katı tablo ve sütun yapısıyla uğraştıysanız MongoDB tam bir esneklik sağlıyor. Verileri tablolar yerine JSON benzeri dokümanlarda tuttuğu için, projenize yeni bir özellik eklediğinizde veri tabanını baştan tasarlamak zorunda kalmıyorsunuz. JavaScript objeleriyle doğrudan uyumlu çalışması da MERN tarafındaki veri işlemlerini oldukça pratikleştiriyor.

Express.js

Node.js tek başına harika bir güçtür fakat yönlendirmeleri (routing) ve HTTP isteklerini ham Node.js ile yönetmek bir süre sonra kod kalabalığı yaratır. Express.js burada imdada yetişerek sunucu tarafındaki mantığı oldukça sadeleştirir. Ön yüzden gelen istekleri yakalar, veri tabanı işlemlerini düzenler ve sonucu kullanıcıya geri döndürür.

React.js

React tarafında işin sırrı component (bileşen) mantığında yatıyor. Ekrandaki bir butonu, arama çubuğunu ya da kartı tek bir yerde yazıp projenin her yerinde defalarca

kullanabiliyorsunuz. Sayfayı her tıklamada baştan aşağı yenilemek yerine sadece değişen kısmı güncellediği için de kullanıcıya bir web sitesinden ziyade masaüstü uygulaması hızında bir akıcılık sunuyor. MERN mimarisinde de işin kullanıcıya dokunan, veriyi ekranda canlı canlı gösteren vitrin kısmını üstleniyor

Node.js

Node.js çıkmadan önce ön yüzde JavaScript, arka yüzde ise PHP veya C# yazıp iki farklı dille uğraşmak zorundaydık. Node.js aslında yeni bir programlama dili değil; Google Chrome'un V8 motorunu alıp JavaScript'i tarayıcının dışına, yani sunucuya taşıyan bir çalışma zamanı ortamı (runtime).

En büyük artısı asenkron (non-blocking) çalışması. Tek bir izlek (thread) üzerinden yürüse bile olay döngüsü (event loop) sayesinde aynı anda binlerce isteği kilitlenmeden yönetebiliyor. Üzerine bir de npm ekosistemini katınca, ihtiyacınız olan paketleri hazır çekip çok hızlı backend yazabiliyorsunuz. MERN tarafında da Express ile birlikte sunucu trafiğini ve MongoDB bağlantısını yöneten ana motor görevini görüyor.

Neden MERN Stack?

MERN'in bu kadar tutmasının ana sebebi aslında basit: Tek dille tüm projeyi bitirebiliyorsunuz. Front-end yazarken kafanızdaki JavaScript mantığını backend'e geçerken sıfırlamak zorunda kalmıyorsunuz (context switching olmuyor). Bu da hem öğrenme sürecini acayip hızlandırıyor hem de hızlıca MVP çıkarmak isteyen ekiplere ciddi zaman kazandırıyor.

Get Turkergamze'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

İşin en iyi yanı verinin akışında. Ön yüzden bastığınız JSON, arka yüzde hiç takılmadan gidip veritabanına yazılıyor. Kod yazarken bir yerde takılma ihtimaliniz

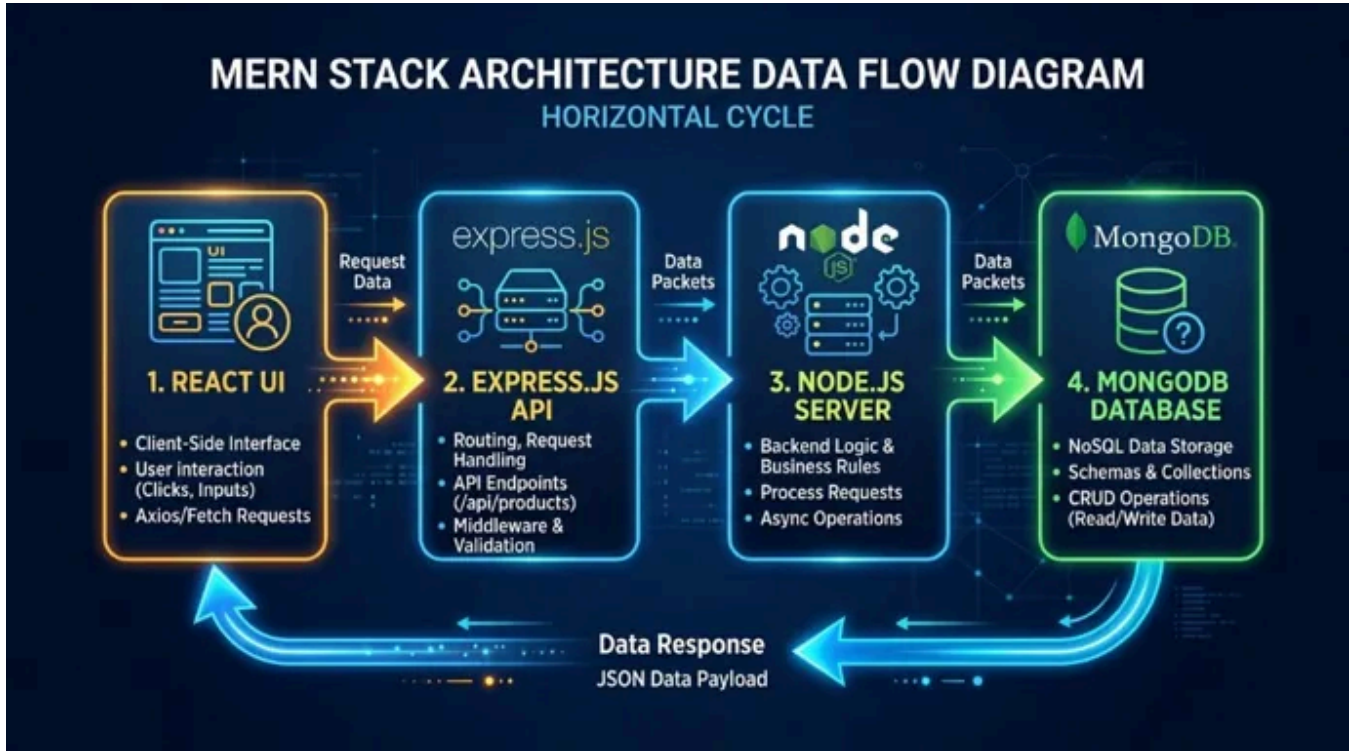
elbette var; fakat arkasında öyle bir topluluk desteği var ki cevapsız kalmanız imkansız gibi bir şey.

MERN Stack'te Veri Akışı Nasıl Gerçekleşir?

Peki veri MERN mimarisinde tam olarak nasıl dolaşıyor? Mantık oldukça basit bir zincire dayanıyor.

Süreç, kullanıcının ekranda bir butona basması ya da formu doldurmasıyla başlıyor. React bu isteği paketleyip arka taraftaki Express/Node.js sunucusuna fırlatıyor. Sunucu gerekli kontrolleri yapıp doğrudan MongoDB'ye uğruyor ve veriyi alıyor. Sonuç mu? Hazırlanan veri JSON olarak React'e geri dönüyor, React da sayfayı hiç yenilemeden ekranı anında güncelliyor. Kullanıcı tarafında her şey milisaniyeler içinde bitmiş oluyor .

Kısaca izlenen rota: React → Express.js → Node.js → MongoDB → Express.js → React



MERN'in Güçlü Yönleri ve Dikkat Edilmesi Gerekenler

Teknoloji seçimi yaparken işinizi kolaylaştırmak adına MERN mimarisinin artılarını ve eksilerini şu şekilde sıralayabiliriz:

Artıları

- *Tek Dil Konforu: Hem ön yüzde hem arka yüzde sadece JavaScript yazarsınız. Diller arası zihinsel geçiş yapmadığınız için geliştirme süreci oldukça akıcı ilerler.*
- *Bileşen Odaklı Ön Yüz: React sayesinde bir kez yazdığınız butonu veya kart yapısını projenin her yerinde tekrar tekrar kullanabilirsiniz.*
- *Esnek Veri Yapısı: MongoDB şemasız çalıştığı için “veri tabanı bozulur mu” korkusu olmadan yeni özellikleri projeye anında entegre edebilirsiniz.*
- *Devasa Topluluk: Karşılaşılabileceğiniz hataların %99'u internette zaten başkaları tarafından çözülmüştür; cevapsız kalmazsınız.*

Eksileri

- *Mimari Disiplin İhtiyacı: Esneklik harikadır ancak proje büyüdükçe sıkı bir klasör mimarisi kurmazsanız kodlar hızla karmaşıklaşabilir.*
- *JavaScript'in Esnek Yapısı: Kalabalık ekiplerde gözden kaçan tip hataları can sıkabilir. Bu yüzden büyük projelerde genelde TypeScript'e geçiş yapılır.*
- *İlişkisel Veri Sınırı: Çok karmaşık tablo ilişkileri veya finansal düzeyde katı işlem güvenliği gerekiyorsa, SQL veri tabanları daha doğru bir tercih olabilir.*

MERN Stack Nereelerde Kullanılır?

Verinin anlık aktığı ve kullanıcının kesintisiz işlem yaptığı projelerde MERN işini fazlasıyla iyi yapıyor.

- *Gerçek Zamanlı (Real-time) Uygulamalar: Node.js'in anlık bağlantı gücü sayesinde canlı sohbet uygulamaları, bildirim sistemleri veya canlı borsa/skor takipleri için mükemmeldir.*
- *Sürekli Etkileşimli Arayüzler (SPA): Trello veya Notion tarzı sürükle-bırak görev panoları, takvimler ve iş takip araçları React'in component mantığıyla çok rahat kurulur.*
- *E-Ticaret ve Sosyal Ağlar: Ürün tiplerinin veya kullanıcı profillerinin sürekli değiştiği yerlerde MongoDB'nin esnek doküman yapısı hayat kurtarır.*

Kısacası; ‘Kullanıcı sayfada sürekli bir şeye tıklasın ama ekran hiç beyazlayıp yenilenmesin’ dediğiniz her türlü dinamik projede MERN gözü kapalı tercih edilebilir.

Web Development

JavaScript

Full Stack

Software Engineering

Tech Stack



Follow

Written by Turkergamze

0 followers · 1 following

