

Next.js Projenizi (Hizmetia) Yayınlama Rehberi

Merhaba, yüklediğiniz `New-Project.zip` dosyasını detaylı olarak inceledim. Projeniz, geleneksel bir web sitesi değil, **Next.js** teknolojisi ile geliştirilmiş modern bir web uygulamasıdır. Bu nedenle, standart hosting (cPanel'e sadece dosya atarak yayınlama) yöntemleri bu proje için doğrudan çalışmayacaktır. Projenizin çalışması için **Node.js** adı verilen bir sunucu ortamına ihtiyacı vardır.

Dosya yapısını bozmadan projenizi en doğru ve kolay şekilde nasıl yayınlayabileceğinizi aşağıda adım adım anlattım.

Projenizin Teknik Özellikleri

- **Çatı (Framework):** Next.js 15
- **Programlama Dili:** TypeScript
- **Stil:** Tailwind CSS
- **Paket Yöneticisi:** Projede `bun.lock` dosyası mevcut, bu da `bun` paket yöneticisinin kullanıldığını gösteriyor. `npm` veya `yarn` da kullanılabilir.
- **Veritabanı:** Proje dosyaları içinde bir veritabanı bulunmuyor. Bu, uygulamanın büyük ihtimalle harici bir veritabanı hizmetine (Supabase, Firebase, MongoDB Atlas, PlanetScale vb.) bağlandığı anlamına gelir. Yayınlama sırasında bu veritabanı bilgilerini "ortam değişkenleri" (environment variables) olarak sağlamanız gerekecektir.

Yayınlama Seçenekleri

Projenizi yayınlamak için iki ana yöntem bulunmaktadır. **Seçenek 1 (Önerilen)** en kolay ve en verimli olanıdır.

Seçenek 1: Otomatik Derleme ve Yayınlama Platformları (Vercel & Netlify)

Bu platformlar, Next.js gibi modern JavaScript projeleri için özel olarak tasarlanmıştır. Kodunuzu bir Git deposuna (örneğin GitHub) yüklersiniz ve bu platformlar geri kalan her şeyi (sunucu kurulumu, projenin derlenmesi, güncellemeler) otomatik olarak halleder.

Neden Bu Yöntem Öneriliyor?

- **Kolay Kurulum:** Sadece birkaç tıklama ile projenizi yayınlayabilirsiniz.

- **Otomatik Güncelleme:** Kodunuzda bir deęişiklik yapıp Git deposuna gönderdiğinizde, siteniz otomatik olarak güncellenir.
- **Yüksek Performans:** Projeniz global bir sunucu ağı (CDN) üzerinden sunulur, bu da sitenizin dünyanın her yerinden hızlı açılmasını sağlar.
- **Ücretsiz Başlangıç Planı:** Birçok proje için yeterli olan cömert bir ücretsiz kullanım hakkı sunarlar.

Projenizin içinde zaten bir `netlify.toml` dosyası bulunuyor. Bu, projenin daha önce Netlify üzerinde yayınlanmak üzere yapılandırıldığını gösterir. Bu nedenle **Netlify** veya Next.js'in geliştiricisi olan **Vercel** sizin için en ideal seçeneklerdir.

Adım Adım Vercel/Netlify ile Yayınlama:

1. **GitHub Hesabı Oluşturun:** Eğer yoksa, github.com adresinden ücretsiz bir hesap oluşturun.
2. **Yeni Bir Depo (Repository) Oluşturun:** GitHub'da "New Repository" diyerek projeniz için yeni ve **private** (özel) bir depo oluşturun.
3. **Proje Dosyalarını GitHub'a Yükleyin:**
 - Bilgisayarınıza [GitHub Desktop](https://github.com/desktop) uygulamasını kurun.
 - Uygulama üzerinden yeni oluşturduğunuz depoyu klonlayın.
 - `New-Project.zip` dosyasından çıkan `Hizmetia projesi` klasörünün içindeki **tüm dosyaları** bu klonladığınız klasörün içine kopyalayın.
 - GitHub Desktop uygulamasından deęişiklikleri özetleyip ("Initial commit" yazabilirsiniz) ana dala ("main") gönderin (Commit & Push).
4. **Vercel veya Netlify'e Kaydolun:**
 - vercel.com veya netlify.com sitesine gidin ve GitHub hesabınızla kaydolun/giriş yapın.
5. **Projenizi Platforma Ekleyin:**
 - Dashboard (Kontrol Paneli) üzerinden "Add New Project" veya "Import from Git" butonuna tıklayın.
 - GitHub hesabınızı yetkilendirin ve 3. adımda oluşturduğunuz projeyi seçin.
6. **Yapılandırma ve Yayınlama:**
 - Platform, projenizin bir Next.js projesi olduğunu otomatik olarak algılayacaktır.
 - **Ortam Deęişkenleri (Environment Variables):** Bu adımda projenizin ihtiyaç duyduğu veritabanı bağlantı bilgileri gibi gizli anahtarları girmeniz gerekecektir. Bu

bilgileri projenin orijinal geliştiricisinden veya dökümantasyonundan öğrenmelisiniz.

- "Deploy" (Yayınla) butonuna basın.

Birkaç dakika içinde projeniz derlenecek ve size özel bir `.vercel.app` veya `.netlify.app` adresi ile yayına alınacaktır. Daha sonra kendi alan adınızı (domain) bu platforma kolayca yönlendirebilirsiniz.

Seenek 2: Geleneksel Sunucu (VPS veya Node.js Destekli Hosting)

Bu yöntem, bir sunucu kiralamanızı ve tüm kurulumu manuel olarak yapmanızı gerektirir. Teknik bilgi gerektirir ve yönetimi daha zordur.

Genel Adımlar:

1. DigitalOcean, Vultr gibi bir sağlayıcıdan bir **VPS (Sanal Sunucu)** kiralayın veya Node.js desteęi sunan bir cPanel hosting paketi bulun.
2. Sunucuya SSH ile bağlanın.
3. **Node.js** ve **npm/bun** kurun.
4. Proje dosyalarınızı sunucuya FTP veya `git clone` ile aktarın.
5. Proje klasöründe `npm install` (veya `bun install`) komutu ile bağımlılıkları kurun.
6. `npm run build` (veya `bun run build`) komutu ile projenizi derleyin.
7. `pm2` gibi bir proses yöneticisi ile `npm run start` komutunu kullanarak projenizi arka planda sürekli çalışacak şekilde başlatın.
8. Alan adınızdan gelen istekleri projenizin çalıştığı porta (genellikle 3000) yönlendirmek için **Nginx** veya **Apache** gibi bir web sunucusunu **reverse proxy** olarak yapılandırın.

Sonuç olarak, projenizin modern yapısı göz önüne alındığında, en hızlı, güvenilir ve sorunsuz yöntem kesinlikle 1. Seenek'tir.