at, taslak, çizim içeren bir resim

Açıklama otomatik olarak oluşturuldu

T.C

DOKUZ EYLÜL ÜNİVERSİTESİ

FEN FAKÜLTESİ

**STAJ RAPORU**

**Gamze KOÇAK**

**Eylül/2025**

**İzmir**

İÇİNDEKİLER

Sayfa

[İçindekiler 1](#_TOC_250004)

[Şekiller Listesi](#_TOC_250002) 3

BÖLÜM BİR-GİRİŞ 4

* 1. Stajın Amacı 4
  2. Projede Kullanılan Teknolojiler 4

BÖLÜM İKİ- WEB SİTESİ YAPISININ OLUŞTURULMASI 5

* 1. Temel Sayfa Şablonu ve Bileşenleri 5
     1. Header ve Footer Tasarımı 5
     2. Favicon Entegrasyonu 6
  2. Sayfaların Geliştirilmesi 6
     1. Anasayfa Geliştirmesi 7
     2. Hakkımızda Sayfası 8
     3. Projeler Sayfası 9
     4. İletişim Sayfası 10
     5. 404 Hata Sayfası 11
  3. Ek Geliştirmeler 12
     1. Müşteri Yorumu ve Hizmetler Bölümü 12
     2. Sıkça Sorulan Sorular (SSS) Bölümü 13

BÖLÜM ÜÇ- KULLANICI DENEYİMİ VE TEKNİK GELİŞTİRME 14

3.1 Kullanıcı Deneyimi 14

* + 1. Sayfa Başına Dön (Scroll Top) Butonu 14
    2. Çerez (Cookie) Bilgilendirme Banner’ı 15
    3. Sayfa Yükleme Efekti (Loader) 16
    4. Karanlık Mod Özelliği 17

3.2 Mobil Uyumluluk 19

3.2.1 Responsive Tasarım 19

3.2.2 Hamburger Menü 20

3.3 Animasyonlar ve Görsel Efektler 21

3.3.1 Kaydırma Animasyonları (AOS Kütüphanesi) 21

3.3.2 CSS Geçiş (Transition) Efektleri 21

3.3.3 Resim Görüntüleyici (Lightbox) 22

3.4 Erişilebilirlik ve Diğer Teknik Geliştirmeler 22

3.4.1 Temel Erişilebilirlik (Accessibility) Standartları 23

3.4.2 Yazıcı Dostu Sayfa Düzeni (Print Layout) 23

3.4.3 Form Doğrulama (Validation) 24

BÖLÜM DÖRT-OPTİMİZASYON VE PROJE YÖNETİMİ 25

* 1. Lightouse ve Kod Optimizasyonu 25
  2. Git Kullanımı ve Projenin Canlıya Alınması 26

BÖLÜM BEŞ-SONUÇ 27

* 1. Staj Süresince Elde Edilen Kazanımlar 27
  2. Proje Değerlendirmesi 27

ŞEKİLLER LİSTESİ

Sayfa

[Şekil 2.2.1 Anasayfa Görünümü](#_TOC_250004) 8

[Şekil 2.2.2 Hakkımızda Sayfası](#_TOC_250002) 9

[Şekil 2.2.3 Projeler Sayfası](#_TOC_250002) 10

[Şekil 2.2.4 İletişim Sayfası](#_TOC_250002) 11

[Şekil 2.2.5 404 Hata Sayfası](#_TOC_250002) 12

[Şekil 3.1.1 Sayfa Başına Dön Butonu JavaScript Kodu](#_TOC_250002) 15

[Şekil 3.1.2 Çerez Bilgilendirme Banner’ı JavaScript Kodu](#_TOC_250002) 16

[Şekil 3.1.4 Karanlık Mod JavaScript Kodu ve Görünümü](#_TOC_250002) 18

[Şekil 3.2.1 Responsive Tasarım CSS Kodu](#_TOC_250002) 20

[Şekil 3.2.2 Hamburger Menü JavaScript Kodu](#_TOC_250002) 20

[Şekil 3.4.2 Yazıcı Dostu Sayfa Düzeni CSS Kodu](#_TOC_250002) 24

BÖLÜM BİR

GİRİŞ

**1.1 Stajın Amacı**

Bu staj çalışmasının temel amacı, lisans öğrenimi boyunca edindiğim teorik yazılım geliştirme bilgisini, kurumsal bir proje üzerinde uygulayarak pratik deneyime dönüştürmektir. Ostaş İnşaat firması bünyesinde yürütülen bu stajda, firmanın ihtiyaçları doğrultusunda modern bir web sitesi projesini baştan sona hayata geçirmek hedeflenmiştir.

Stajın kapsamı, Ostaş İnşaat için kurumsal bir web sitesinin sıfırdan tasarlanıp geliştirilmesini içermektedir. Bu kapsamda, web sitesinin geliştirilmesinin yanı sıra versiyon kontrolü altında tutulması, canlıya alınması ve performans denetimlerinin yapılmasını da içermiştir.

**1.2 Projede Kullanılan Teknolojiler**

Projede, web sayfalarının iskeleti için HTML5, modern ve responsive arayüz tasarımı için CSS3, kullanıcı etkileşimlerini yönetmek için ise JavaScript olmak üzere temel web teknolojilerinden etkin bir şekilde yararlanılmıştır. Görsel derinlik ve dinamizm, AOS (Animate On Scroll) kütüphanesiyle eklenen animasyonlarla artırılmıştır. Projenin kod yönetimi ve versiyon takibi Git ve GitHub aracılığıyla sağlanmış, kesintisiz entegrasyon ve dağıtım süreçleri ise Netlify platformu üzerinden yürütülmüştür. Projenin kalite güvencesi kapsamında, Google Lighthouse kullanılarak performans, erişilebilirlik ve en iyi pratikler açısından detaylı analizler yapılmıştır.

BÖLÜM İKİ

WEB SİTESİ YAPISININ OLUŞTURULMASI

**2.1 Temel Sayfa Şablonu ve Bileşenleri**

Bir web sitesi geliştirme projesine başlarken atılan ilk ve en önemli adımlardan biri, tüm sayfalarda tekrar edecek olan standart bir sayfa şablonu oluşturmaktır. Bu yaklaşım, "Kendini Tekrar Etme" prensibine dayanır ve projenin ilerleyen aşamalarında büyük bir verimlilik ve bakım kolaylığı sağlar. Her sayfa için aynı kodları tekrar yazmak yerine, ortak bir iskelet oluşturulmuş ve sadece sayfalara özgü içerik alanları değişken bırakılmıştır.

Bu kapsamda, projenin ana HTML dosyası (index.html) bir şablon görevi görecek şekilde hazırlanmıştır. Bu temel şablonda, tüm sayfalar için geçerli olacak olan <html>, <head> ve <body> etiketleri ile temel yapı kurulmuştur. <head> bölümü içerisinde, karakter kodlamasını belirten meta charset="UTF-8" etiketi, mobil uyumluluk için kritik olan meta name="viewport" etiketi, sitenin genel stil dosyasını (style.css) çağıran <link> etiketi ve tarayıcı sekmesinde görünecek olan sayfa başlığı (<title>) gibi standart elemanlar tanımlanmıştır.

**2.1.1 Header ve Footer Tasarımı**

Web sitesi şablonunun en önemli iki bileşeni olan Header (üst kısım) ve Footer (alt kısım), sitenin tüm sayfalarında tutarlı bir marka kimliği ve kullanıcı deneyimi sunmak amacıyla tasarlanmış ve kodlanmıştır.

Sayfanın en üstünde yer alan ve kullanıcının ilk karşılaştığı alan olan Header, anlamsal bütünlük için <header> etiketiyle oluşturulmuştur. İçerisinde firma logosu ve <nav> etiketiyle sarmalanan ana navigasyon menüsü bulunur. Bu iki elemanın farklı ekran boyutlarında dahi düzenli bir şekilde konumlanması, CSS Flexbox yapısı ile sağlanmıştır. Mobil cihazlarda ise, standart menü gizlenerek yerine JavaScript ile yönetilen bir "Hamburger Menü" getirilmiş, bu sayede küçük ekranlarda temiz ve kullanışlı bir gezinme deneyimi hedeflenmiştir.

Sayfa şablonunu tamamlayan ve en altta yer alan Footer alanı ise ikincil bilgilere, iletişim detaylarına ve sosyal medya linklerine hızlı erişim sağlamak amacıyla tasarlanmıştır. Anlamsal <footer> etiketi içinde, içerik genellikle CSS Flexbox kullanılarak çok sütunlu bir yapıda düzenlenmiştir. Bu yapı, siteye özgü bir arka plan rengiyle görsel olarak ana içerikten ayrıştırılmıştır. Mobil uyumluluk kapsamında, bu sütunlu yapı küçük ekranlarda alt alta sıralanacak şekilde yeniden düzenlenerek okunabilirlik ve kullanım kolaylığı korunmuştur.

**2.1.2 Favicon Entegrasyonu**

Favicon entegrasyonu, web sitesinin marka kimliğini güçlendirmek ve kullanıcı deneyimini iyileştirmek amacıyla gerçekleştirilmiştir. Tarayıcı sekmelerinde ve yer imlerinde görünen bu küçük simge, kullanıcının siteyi çok sayıda sekme arasında kolayca tanımasını sağlar ve profesyonel bir görünüm sunar.

Bu projede favicon olarak, şirketin kurumsal logosunun sadeleştirilmiş bir versiyonu, sitenin tüm sayfalarında geçerli olan ana şablonun <head> bölümüne <link> etiketleri aracılığıyla eklenmiştir. Bu entegrasyon, standart tarayıcıların yanı sıra mobil cihazların ana ekranları gibi farklı platformları da kapsayacak şekilde çeşitli boyutlarda yapılmıştır.

**2.2 Sayfaların Geliştirilmesi**

Bu bölümde, firmanın hedeflerine hizmet edecek şekilde tasarlanan her bir sayfanın yapısı, amacı ve teknik geliştirme detayları ele alınacaktır.Oluşturulan temel şablon, geliştirilen tüm sayfalar için bir iskelet görevi görmüştür. Bu sayede, her sayfanın aynı tutarlı görsel kimliğe ve kullanıcı deneyimine sahip olması sağlanmıştır. Proje kapsamında Ana Sayfa, Hakkımızda, Projeler ve İletişim olmak üzere dört temel sayfa kodlanmıştır

**2.2.1 Ana Sayfa Geliştirmesi**

Ana sayfa, bir web sitesinin dijital vitrini ve kullanıcıların firma ile ilk temas kurduğu en kritik alan olması sebebiyle, ziyaretçilere ilk andan itibaren güven veren, firmanın kimliğini ve vizyonunu net bir şekilde özetleyen bir yapıda tasarlanmıştır.

Sayfa, ziyaretçiyi ilk olarak, firmanın vizyonunu yansıtan güçlü bir başlık, etkileyici bir arka plan görseli ve kullanıcıyı projelere yönlendiren bir eylem çağrısı butonu içeren tam ekran bir ‘Hero’ bölümüyle karşılar. Bu karşılama alanının hemen altında, firmanın misyonunu ve deneyimini özetleyen kısa bir ‘Hakkımızda’ tanıtım bölümüne yer verilmiştir. Bu bölümde, ziyaretçilere daha detaylı bilgi alabilecekleri ‘Hakkımızda’ sayfasına geçiş yapma imkânı da sunulmuştur. Ziyaretçide güven oluşturmanın en önemli adımı olan somut iş örneklerini sergilemek amacıyla, ana sayfanın devamında firmanın öne çıkan projelerinden oluşan bir galeri bölümü bulunmaktadır. Kart yapısı içinde sunulan her bir proje, kullanıcıyı daha fazla detayın yer aldığı ana ‘Projeler’ sayfasına yönlendirir.

Proje galerisinin hemen ardından, somut iş örnekleriyle oluşturulan güveni pekiştirmek ve gerçek müşteri memnuniyetini yansıtmak amacıyla bir ‘Müşteri Yorumları’ bölümüne yer verilmiştir. Bu alanda, müşterilerin deneyimlerini paylaştığı kısa ve etkili alıntılar sunularak potansiyel müşteriler için güçlü bir sosyal kanıt oluşturulur ve firma imajı desteklenir.

Tüm bu modüler yapı, HTML5’in anlamsal etiketleri kullanılarak oluşturulmuş ve bölümlerin sayfa içindeki yerleşimi CSS Flexbox ve Grid gibi modern tekniklerle esnek ve mobil uyumlu bir şekilde kodlanmıştır.

metin, ekran görüntüsü, giyim, kişi, şahıs içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

**2.2.1 Ana Sayfa Görünümü**

**2.2.2 Hakkımızda Sayfası**

"Hakkımızda" sayfası, bir firmanın sadece ne yaptığını değil, aynı zamanda kim olduğunu, misyonunu, vizyonunu ve temel değerlerini anlattığı kritik bir bölümdür. Şirket için bu sayfa, potansiyel müşteriler nezdinde derin bir güven ve profesyonellik algısı oluşturmak amacıyla tasarlanmıştır.

Sayfa, firmanın tarihçesi ve kurumsal kimliğini detaylandıran bölümlerin yanı sıra, şirketin sunduğu temel hizmetlerin açıklandığı bir alanı da içermektedir. Bu sayede ziyaretçiler, firmanın kimliği ile yetkinlikleri arasında doğrudan bir bağ kurabilmektedir. Ziyaretçi deneyimini daha da iyileştirmek ve olası soruları proaktif bir şekilde yanıtlamak amacıyla, sayfanın sonuna ‘Sıkça Sorulan Sorular’ (SSS) bölümü eklenmiştir. Bu bölüm, kullanıcıların merak ettikleri konulara hızlıca yanıt bulmalarını sağlayan, tıklanınca açılıp kapanan bir akordiyon yapısında tasarlanmıştır. Tüm bu bölümlerin geliştirilmesinde, daha önce oluşturulan temel sayfa şablonu kullanılarak görsel bütünlük korunmuş ve sayfa içi yerleşimlerde CSS Flexbox'tan yararlanılarak esnek ve mobil uyumlu bir yapı kodlanmıştır.

metin, ekran görüntüsü, web sitesi, tasarım içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

**2.2.2 Hakkımızda Sayfası**

**2.2.3 Projeler Sayfası**

"Projeler" sayfası, firmanın teknik yetkinliğini, deneyimini ve vizyonunu somut kanıtlarla sergilediği dijital portfolyosudur. Bu sayfanın temel amacı, potansiyel müşterilere ve iş ortaklarına firmanın tamamladığı ve devam etmekte olan işlerin kalitesini, ölçeğini ve çeşitliliğini göstererek güven vermektir. Sayfa, ziyaretçilerin projeleri kolayca inceleyebilmesi için kullanıcı dostu ve görsel odaklı bir yapıda tasarlanmıştır.

Sayfanın en başında, kullanıcıların ilgilendikleri proje türüne göre arama yapabilmelerini sağlayan bir filtreleme menüsü bulunmaktadır. Bu menü sayesinde projeler "Konut Projeleri", "Ticari Yapılar", "Altyapı Projeleri" gibi kategorilere ayrılarak kullanıcıların aradıkları örneklere hızlıca ulaşması hedeflenmiştir.

Filtreleme menüsünün altında, firmanın tüm projeleri, her biri bir projeyi temsil eden "kart" yapıları içerisinde, grid (ızgara) bir düzende sergilenir. Her proje kartında projeye ait dikkat çekici bir görsel, projenin adı ve kategorisi yer alır. Bu grid yapı, CSS Grid Layout kullanılarak oluşturulmuştur. Bu sayede, farklı ekran boyutlarında kartların sayısı ve düzeni otomatik olarak ayarlanarak hem masaüstü hem de mobil cihazlarda tutarlı ve estetik bir görünüm elde edilmiştir.

Kullanıcı deneyimini zenginleştirmek amacıyla, proje kartlarının üzerine gelindiğinde hafif bir büyüme (transform: scale) efekti gibi görsel geri bildirimler eklenmiştir. Daha da önemlisi, bir proje görseline tıklandığında, sayfadan ayrılmadan projenin diğer fotoğraflarını büyük boyutta ve kolayca inceleyebilmeyi sağlayan bir Lightbox galeri özelliği entegre edilmiştir. Bu interaktif galeri, JavaScript ile yönetilmekte olup, ziyaretçilerin projeleri detaylı bir şekilde keşfetmesine olanak tanır.

metin, bina, ekran görüntüsü, dış mekan içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

**2.2.3 Projeler Sayfası**

**2.2.4 İletişim Sayfası**

"İletişim" sayfası, kullanıcı ile firma arasında doğrudan bir köprü kurarak erişilebilirliği en üst düzeye çıkarmak amacıyla tasarlanmıştır. Bu sayfanın merkezinde, ziyaretçilerin web sitesi üzerinden kolayca mesaj göndermelerini sağlayan interaktif bir iletişim formu yer alır. Kullanıcıların hatalı veya eksik bilgi göndermesini önlemek için, bu form JavaScript tabanlı bir doğrulama (validation) mekanizması ile donatılmış olup, gerekli alanların doldurulmasını ve e-posta formatının doğruluğunu kontrol ederek anında geri bildirim sağlar. Formu kullanmayı tercih etmeyenler için ise firmanın adresi, telefonu ve e-posta adresi gibi temel iletişim bilgileri açıkça belirtilmiştir. Bu fiziki bilgileri desteklemek ve ziyaretçilere yol tarifi kolaylığı sunmak amacıyla, sayfanın bir bölümüne şirketin konumunu gösteren interaktif bir Google Maps haritası entegre edilmiştir. Tüm bu bileşenler, CSS Flexbox kullanılarak mobil cihazlarla tam uyumlu, fonksiyonel ve kullanıcı dostu bir düzende bir araya getirilerek firmaya ulaşmak için kapsamlı bir merkez oluşturulmuştur.

metin, ekran görüntüsü, web sayfası içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

**2.2.4 İletişim Sayfası**

**2.2.5 404 Hata Sayfası**

Web sitesi geliştirme sürecinde, kullanıcıların karşılaşabileceği olumsuz deneyimleri dahi profesyonel bir şekilde yönetmek, marka imajı için oldukça önemlidir. Bu bağlamda, ziyaretçilerin var olmayan bir linke tıklaması veya adresi yanlış yazması durumunda karşılaştıkları standart ve teknik hata mesajı yerine, özel bir "404 Hata Sayfası" tasarlanmıştır. Bu sayfanın temel amacı, kullanıcıyı kaybolmuş hissettirmek yerine, ona yardımcı olmak ve site içinde kolayca yolunu bulmasını sağlamaktır.

Şirket için geliştirilen 404 sayfası, teknik bir jargon kullanmaktan kaçınarak, kullanıcı dostu ve anlaşılır bir dille "Aradığınız Sayfa Bulunamadı" mesajını iletir. Bu mesaj, genellikle konuyla ilgili sade bir grafik veya ikon ile desteklenerek sayfanın görsel olarak daha az rahatsız edici olması hedeflenmiştir. Sayfanın en kritik işlevi, kullanıcıyı sitede tutmaktır. Bu nedenle, sayfanın orta kısmına belirgin bir şekilde yerleştirilmiş "Ana Sayfaya Dön" butonu veya linki eklenmiştir. Bu sayede, kullanıcılar tek bir tıklama ile sitenin ana akışına kolayca geri dönebilirler.

Teknik olarak, bu özel hata sayfası da sitenin genel şablonu (Header ve Footer dahil) üzerine inşa edilmiştir. Bu yaklaşım, kullanıcı bir hata ile karşılaştığında bile sitenin tutarlı ve kurumsal kimliğinden ayrılmamasını sağlar. Sayfa, CSS ile sitenin genel renk ve tipografi paletine uygun olarak stillendirilerek, olası bir kullanıcı hatasının profesyonel ve yapıcı bir deneyimle çözümlenmesi sağlanmıştır.

metin, yazı tipi, logo, ekran görüntüsü içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

**2.2.5 404 Hata Sayfası**

**2.3 Ek Geliştirmeler**

Web sitesinin ana sayfaları tamamlandıktan sonra, kullanıcı deneyimini zenginleştirmek, marka imajını güçlendirmek ve sitenin genel işlevselliğini artırmak amacıyla çeşitli ek modüller ve bölümler geliştirilmiştir. Bu geliştirmeler, genellikle ana sayfa veya hakkımızda gibi stratejik sayfalara entegre edilen, yeniden kullanılabilir bileşenler olarak tasarlanmıştır.

**2.3.1 Müşteri Yorumları ve Hizmetler Bölümü**

Web sitesinin kurumsal kimliğini ve güvenilirliğini pekiştirmek amacıyla, firmanın yetkinliklerini ve bu yetkinliklerin müşteri memnuniyetine nasıl dönüştüğünü gösteren iki önemli bölüm geliştirilmiştir. Ziyaretçilere şirketin sunduğu değeri net bir şekilde anlatmak için tasarlanan Hizmetler bölümünde, şirketin ana faaliyet alanları, her biri bir ikon, başlık ve kısa açıklama içeren kartlar halinde, CSS Grid kullanılarak düzenli bir ızgara yapısında sunulmuştur. Bu hizmetlerin somut bir kanıtı olarak, potansiyel müşterilere sosyal kanıt (social proof) sunan Müşteri Yorumları bölümü eklenmiştir. Bu bölümde, memnun müşteri ifadeleri, her biri ayrı bir kart içerisinde olacak şekilde, CSS Flexbox kullanılarak yan yana düzenli ve statik bir yapıda sergilenerek firmanın itibarı güçlendirilmiştir.

**2.3.2 Sıkça Sorulan Sorular (SSS) Bölümü**

Sıkça Sorulan Sorular (SSS) bölümü, modern bir kullanıcı arayüzü (UI) deseni olan akordiyon bileşeni olarak implemente edilmiştir.

Bileşenin temelini, erişilebilirlik ve anlamsallık ilkelerine uygun olarak <details> ve <summary> etiketleri veya alternatif olarak aria-expanded ve aria-controls gibi ARIA (Accessible Rich Internet Applications) rolleriyle zenginleştirilmiş <div> ve <button> elemanları oluşturur. Başlangıçta tüm cevap panelleri, CSS'te max-height: 0 ve overflow: hidden özellikleri kullanılarak görsel olarak gizlenir. Kullanıcı bir soru başlığına tıkladığında, querySelectorAll ile seçilen tüm ilgili elemanlara atanmış olan bir JavaScript click olay dinleyicisi (event listener) tetiklenir. Bu olay dinleyicisinin callback fonksiyonu, öncelikle açık olan diğer tüm panellerdeki .is-active sınıfını kaldırarak mevcut akordiyonu kapatır ve ardından tıklanan hedefin üst elementine bu sınıfı ekler (toggle). .is-active sınıfı aktif olduğunda, CSS tarafında hedeflenen cevap panelinin max-height değeri, içeriği tamamen gösterecek yeterli bir değere (örneğin 500px veya dinamik olarak hesaplanan element.scrollHeight) transition efektiyle getirilir. Bu max-height tekniği, display: none özelliğinin aksine akıcı ve animasyonlu bir açılıp kapanma efekti yaratılmasına olanak tanır ve böylece hem estetik hem de fonksiyonel olarak üst düzey bir kullanıcı deneyimi sunar.

BÖLÜM ÜÇ

KULLANICI DENEYİMİ VE TEKNİK GELİŞTİRME

Bu bölümde, firmanın web sitesinin temel yapısı ve sayfaları oluşturulduktan sonra, projeyi modern standartlara taşımak ve kullanıcı memnuniyetini en üst düzeye çıkarmak için yapılan ek geliştirmeler ele alınacaktır. Bu geliştirmeler, doğrudan kullanıcı etkileşimini iyileştiren Kullanıcı Deneyimi (UX) özellikleri ve sitenin farklı platformlarda sorunsuz çalışmasını sağlayan mobil uyumluluk, animasyonlar gibi teknik geliştirmeler olmak üzere üç başlık altında incelenecektir.

**3.1 Kullanıcı Deneyimi**

Kullanıcı deneyimi (UX), bir ziyaretçinin web sitesini kullanırken ne kadar kolay, verimli ve keyifli bir süreç yaşadığıyla ilgilenir. Bu bölümde, sitenin sadece işlevsel olması değil, aynı zamanda kullanıcı dostu ve konforlu bir etkileşim sunması için eklenen özel geliştirmeler açıklanacaktır. Bu geliştirmeler, kullanıcının site içindeki gezintisini kolaylaştıran, şeffaflığı artıran ve görsel konfor sağlayan küçük ama etkili dokunuşlardır.

**3.1.1 Sayfa Başına Dön (Scroll Top) Butonu**

Özellikle içeriği uzun olan sayfalarda kullanıcı deneyimini iyileştirmek amacıyla, sayfanın en başına tek bir tıklama ile dönülmesini sağlayan bir "Sayfa Başına Dön" (Scroll to Top) butonu geliştirilmiştir. Bu butonun temel işlevi, kullanıcıları uzun ve yorucu bir geri kaydırma zahmetinden kurtararak site içi gezinmeyi daha hızlı ve konforlu hale getirmektir.

Buton, kullanıcı deneyimini olumsuz etkilememek için akıllı bir görünürlük mantığıyla çalışır. Sayfa ilk yüklendiğinde veya en üstte iken gizli durumdadır. Kullanıcı, sayfayı belirli bir miktar aşağı kaydırdığında, buton ekranın sağ alt köşesinde yumuşak bir animasyonla belirir. Bu davranış, JavaScript ile sayfanın dikey kaydırma (window.scrollY) pozisyonunun dinlenmesiyle kontrol edilir. Belirlenen eşik aşıldığında butona bir CSS class'ı eklenerek görünür hale getirilmesi sağlanır.

Kullanıcı butona tıkladığında ise yine JavaScript devreye girer ve window.scrollTo() metodunu behavior: 'smooth' parametresi ile kullanarak sayfanın akıcı bir animasyonla en üste kaymasını sağlar. Bu küçük ama etkili geliştirme, sitenin kullanılabilirliğini ve profesyonel algısını artıran önemli bir kullanıcı deneyimi detayıdır.

metin, ekran görüntüsü, yazılım, ekran, görüntüleme içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

**3.1.1 Sayfa Başına Dön Butonu JavaScript Kodu**

**3.1.2 Çerez (Cookie) Bilgilendirme Banner’ı**

Web sitelerinin, kullanıcı deneyimini iyileştirmek ve site trafiğini analiz etmek gibi amaçlarla çerez (cookie) kullanması yaygın bir uygulamadır. 6698 sayılı Kişisel Verilerin Korunması Kanunu (KVKK) ve Avrupa Genel Veri Koruma Tüzüğü (GDPR) gibi yasal düzenlemeler uyarınca, site ziyaretçilerini bu konuda bilgilendirmek ve rızalarını almak yasal bir zorunluluktur. Bu yasal uyumluluğu sağlamak ve kullanıcılarla şeffaf bir ilişki kurmak amacıyla, siteye bir "Çerez Bilgilendirme Banner'ı" entegre edilmiştir.

Bu bileşen, kullanıcı siteyi ilk kez ziyaret ettiğinde ekranın alt kısmında beliren, bilgilendirici bir bant şeklinde tasarlanmıştır. Banner üzerinde, sitenin çerez kullandığına dair kısa bir bilgilendirme metni ve kullanıcının bu durumu kabul ettiğini belirteceği bir "Anladım" veya "Kabul Et" butonu yer alır. Tasarım, kullanıcının site içeriğine erişimini engellemeyecek fakat fark edilir olacak şekilde, modern ve sade bir tarzda oluşturulmuştur.

Banner'ın işlevselliği JavaScript ile yönetilmektedir. Bir kullanıcı siteye girdiğinde, script öncelikle tarayıcının yerel depolama alanında (localStorage) veya çerezlerinde daha önceden onay verilip verilmediğini kontrol eder. Eğer daha önce bir onay kaydedilmemişse, banner görünür hale getirilir. Kullanıcı "Kabul Et" butonuna tıkladığında ise, bu onay bilgisi kullanıcının tarayıcısına kaydedilir. Bu sayede, aynı kullanıcı siteyi tekrar ziyaret ettiğinde banner yeniden gösterilmez ve kesintisiz bir gezinme deneyimi sağlanır. Bu geliştirme, sitenin yasal yükümlülüklerini yerine getirirken aynı zamanda kullanıcı güvenini artıran önemli bir adımdır.

metin, ekran görüntüsü, ekran, görüntüleme, yazılım içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

**3.1.2 Çerez Bilgilendirme Banner’ı JavaScript Kodu**

**3.1.3 Sayfa Yükleme Efekti (Loader)**

Web sayfalarının, özellikle yüksek çözünürlüklü görseller veya çok sayıda bileşen içerdiğinde, tamamen yüklenmesi birkaç saniye sürebilir. Bu yükleme sürecinde, sayfa elemanlarının parça parça ve düzensiz bir şekilde ekrana gelmesi, kullanıcıda sitenin yavaş veya bozuk çalıştığı algısı yaratabilir. Bu algıyı yönetmek ve daha profesyonel bir ilk izlenim sunmak amacıyla, sayfa içeriği tam olarak hazır olana kadar ekranda gösterilen bir "Sayfa Yükleme Efekti" (Loader) geliştirilmiştir.

Loader, sayfa açıldığı anda tüm ekranı kaplayan bir katman ve bu katmanın merkezinde yer alan, markayla uyumlu, sade bir animasyondan (örneğin, dönen bir halka veya Ostaş İnşaat logosunun belirmesi) oluşur. Bu efektin temel amacı, arka planda tüm içerik (resimler, scriptler, fontlar vb.) yüklenirken kullanıcıya görsel bir geri bildirim sunmak ve sitenin aktif olarak çalıştığını belirtmektir. Bu, kullanıcının bekleme süresini daha az fark etmesini sağlar ve sitenin yavaş olduğu düşüncesiyle sayfadan ayrılmasını engeller.

Teknik olarak bu özellik, HTML'de basit bir <div> elemanı ve bu elemanı ekranı kaplayacak şekilde biçimlendiren CSS kodları ile oluşturulmuştur. Varsayılan olarak görünür olan bu yükleme ekranının kaybolma anı JavaScript ile yönetilir. JavaScript, tarayıcının window.onload olayını dinler; bu olay, sayfadaki tüm varlıkların tamamen yüklendiği anda tetiklenir. Olay tetiklendiğinde, loader elemanına bir CSS class'ı eklenerek yumuşak bir "fade-out" (karararak kaybolma) animasyonuyla ekrandan kaldırılır ve altındaki tam olarak yüklenmiş sayfa kullanıcıya sunulur. Bu geliştirme, sitenin algılanan performansını ve genel kullanıcı deneyimini önemli ölçüde iyileştirir.

**3.1.4 Karanlık Mod Özelliği**

Modern web uygulamalarında giderek yaygınlaşan Karanlık Mod (Dark Mode), kullanıcıya standart aydınlık tema yerine düşük ışıklı bir renk paleti sunan bir kullanıcı deneyimi özelliğidir. Bu özelliğin Ostaş İnşaat web sitesine entegre edilmesindeki temel amaçlar; özellikle düşük ışıklı ortamlarda kullanıcıların göz yorgunluğunu azaltmak, OLED ekranlı cihazlarda pil tüketimini düşürmeye yardımcı olmak ve kullanıcılara estetik bir tercih sunarak site deneyimini kişiselleştirmektir.

Bu özellik, kullanıcının site arayüzünde kolayca erişebileceği, genellikle header (üst bilgi) bölümünde konumlandırılmış bir açma/kapama butonu (toggle switch) aracılığıyla kontrol edilir. Kullanıcı bu butona tıkladığında, sitenin renk şeması anında aydınlık ve karanlık temalar arasında geçiş yapar. Kullanıcının yaptığı bu tercih, tarayıcının yerel depolama alanına (localStorage) kaydedilir. Bu sayede, kullanıcı siteyi daha sonra tekrar ziyaret ettiğinde, en son seçtiği tema otomatik olarak uygulanır ve her seferinde ayar yapmasına gerek kalmaz.

Teknik altyapısı, modern CSS ve JavaScript yeteneklerine dayanmaktadır. İlk olarak, sitenin tüm renk paleti (arka plan, metin, başlık renkleri vb.) CSS Değişkenleri (Custom Properties) kullanılarak tanımlanmıştır. Örneğin, :root seçicisi içinde --background-color: #fff; ve --text-color: #333; gibi değişkenler belirlenmiştir. Ardından, .dark-mode adında bir class oluşturulmuş ve bu class aktif olduğunda aynı değişkenlerin renk kodları koyu tema renkleriyle (--background-color: #121212;, --text-color: #eee;) ezilmiştir.

Kullanıcının tema değiştirme butonuna tıklama eylemi JavaScript ile dinlenir. Tıklama gerçekleştiğinde, JavaScript sayfanın <body> etiketine .dark-mode class'ını ekler veya kaldırır. Bu tek class değişikliği, CSS değişkenleri sayesinde tüm sitenin renk şemasının anında değişmesini tetikler. Bu verimli ve sürdürülebilir yöntem, siteye hem modern bir işlevsellik kazandırmış hem de kullanıcı konforunu ön planda tutan bir tasarım anlayışını ortaya koymuştur.



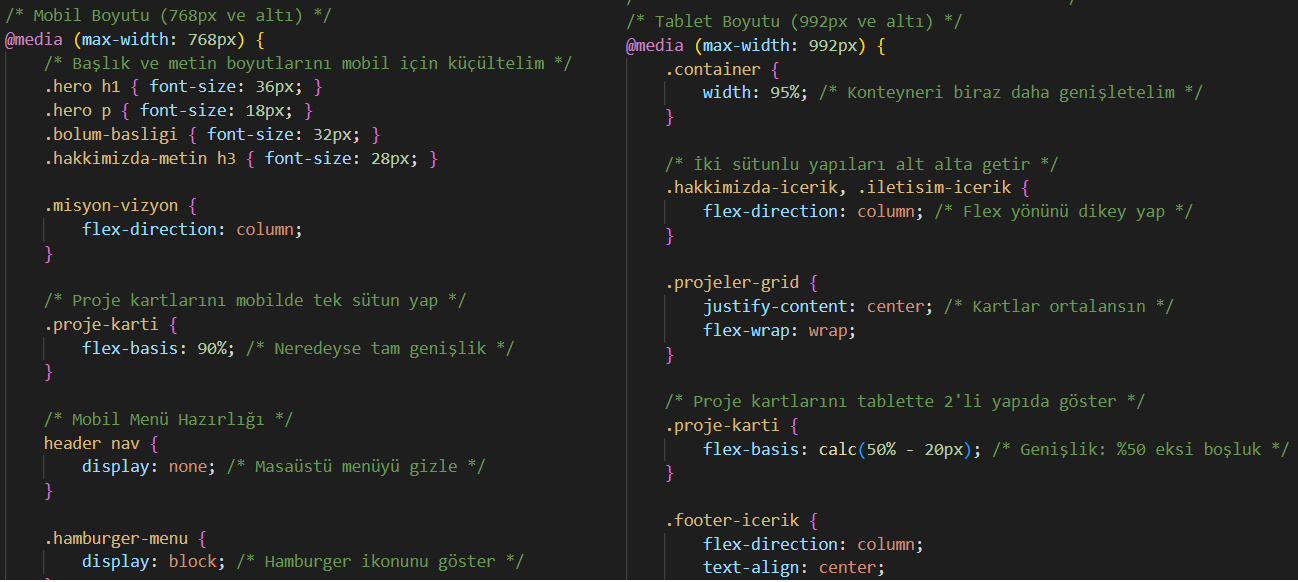
**3.1.4 Karanlık Mod JavaScript Kodu ve Görünümü**

**3.2 Mobil Uyumluluk**

Bu projede Duyarlı Web Tasarımı (Responsive Web Design) yaklaşımıyla ulaşılmıştır. CSS Medya Sorguları (Media Queries) kullanılarak, masaüstünde geniş bir düzende görünen sayfa yapısı, tablet ve telefon gibi daha küçük ekranlarda okunabilirliği ve kullanılabilirliği koruyacak şekilde otomatik olarak yeniden düzenlenmiştir. Bu uyumun en belirgin örneklerinden biri, masaüstünde açık olarak görünen ana navigasyon menüsünün, mobil cihazlarda yer tasarrufu sağlayan ve tıklandığında açılan kompakt bir Hamburger Menü'ye dönüşmesidir. Bu sayede web sitesi, tüm platformlarda tutarlı, erişilebilir ve kullanıcı dostu bir deneyim sunmaktadır.

**3.2.1 Responsive Tasarım**

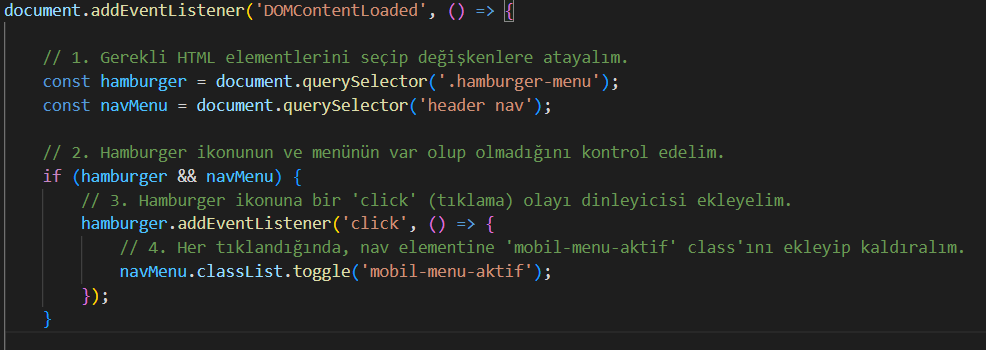
Responsive Tasarım, Ostaş İnşaat web sitesinin tek bir kod tabanı üzerinden, tüm cihazlarda optimal bir deneyim sunmasını sağlayan temel mimari yaklaşımdır. Projenin CSS yapısı, öncelikle mobil cihazları hedefleyen ve sayfa yükleme performansını artıran "Önce Mobil" (Mobile-First) stratejisi üzerine kurulmuştur. Bu stratejinin doğru çalışması için, tarayıcıya sayfa genişliğini cihazınkiyle eşitlemesini ve başlangıç ölçeğini ayarlamasını bildiren temel <meta name="viewport"> etiketi HTML şablonuna eklenmiştir. Sayfa yerleşimleri, sabit piksel değerleri yerine, CSS Flexbox ve Grid gibi modern modüllerle oluşturulan esnek ızgara sistemlerine dayanmaktadır. Flexbox tek eksenli hizalamalarda ve mobil için içerik yığınlamada kullanılırken, Grid ise projeler gibi çok sütunlu galerilerin farklı ekran boyutlarına akıllıca adapte olmasını sağlayan karmaşık düzenler için tercih edilmiştir. Bu esnek yapının farklı ekran genişliklerindeki davranışını kontrol etmek amacıyla, içeriğin bozulmaya başladığı noktalarda (768px, 1024px gibi) belirlenen Medya Sorguları (Media Queries) kullanılmıştır. Bu sorgular, ekran büyüdükçe tek sütunlu mobil yerleşimini aşamalı olarak çok sütunlu masaüstü görünümlerine dönüştürerek, her platformda tutarlı, erişilebilir ve profesyonel bir kullanıcı deneyimi sunulmasını sağlamıştır.



**3.2.1 Responsive Tasarım Css Kodu**

**3.2.2 Hamburger Menü**

Hamburger menü, mobil cihazlardaki kısıtlı ekran alanını verimli kullanmak amacıyla tasarlanmış, responsive tasarımın en kritik kullanıcı arayüzü bileşenidir. Web sitesinde, CSS Medya Sorguları (@media) kullanılarak ekran belirli bir genişliğin altına düştüğünde, masaüstünde görünen standart yatay navigasyon çubuğu gizlenir ve yerini evrensel olarak tanınan üç çizgili hamburger ikonu alır. Kullanıcı bu ikona tıkladığında, bir JavaScript tıklama olayı (click event) tetiklenir. Bu olay, navigasyon menüsüne .menu-aktif gibi bir CSS class'ı ekleyip kaldırarak, menünün CSS transition efektleriyle ekranın kenarından yumuşak bir şekilde kayarak açılmasını veya kapanmasını sağlar. Bu HTML, CSS ve JavaScript kombinasyonu sayesinde, mobil cihazlarda hem değerli ekran alanından tasarruf edilmiş hem de kullanıcıya sezgisel ve düzenli bir navigasyon çözümü sunulmuştur.



**3.2.2 Hamburger Menü JavaScript Kodu**

**3.3 Animasyonlar ve Görsel Efektler**

Web sitesinde, kullanıcı deneyimini zenginleştirmek ve siteye dinamizm katmak amacıyla, modern ve işlevsel animasyon tekniklerinden yararlanılmıştır. Bu geliştirmeler, estetik bir amaç gütmenin ötesinde, kullanıcı etkileşimini daha sezgisel ve akıcı hale getirmeyi hedefler.

**3.3.1 Kaydırma Animasyonları (AOS Kütüphanesi)**

Web sitesindeki uzun sayfalara dinamizm katmak ve kullanıcı etkileşimini artırmak amacıyla, ziyaretçi sayfayı aşağı kaydırdıkça içeriklerin estetik animasyonlarla belirmesini sağlayan "Animate On Scroll" (AOS) kütüphanesi kullanılmıştır. Bu harici JavaScript kütüphanesi, karmaşık kodlar yazmaya gerek kalmadan, verimli bir şekilde kaydırma animasyonları oluşturma imkânı sunar. Projeye dahil edilip AOS.init(); komutuyla aktif hale getirildikten sonra, animasyon eklenmek istenen HTML elemanlarına data-aos="fade-up" veya data-aos="zoom-in" gibi basit HTML öznitelikleri eklemek yeterli olmuştur. Bu yöntemle, sitenin farklı bölümlerine süre ve gecikme gibi ayarları da özelleştirilebilen, akıcı ve modern animasyonlar entegre edilerek sayfa gezinme deneyimi zenginleştirilmiştir.

**3.3.2 CSS Geçiş (Transition) Efektleri**

Kullanıcı arayüzündeki interaktif elemanların (butonlar, linkler, kartlar vb.) durum değişikliklerini yumuşatmak ve daha akıcı bir deneyim sunmak amacıyla, proje genelinde CSS Geçiş (Transition) efektleri etkin bir şekilde kullanılmıştır. Bu teknik, bir elemanın bir stil durumundan diğerine (örneğin, normal halinden :hover durumuna) geçerken gerçekleşen değişikliğin aniden değil, belirli bir süre içinde kademeli olarak yaşanmasını sağlar.

Örneğin, bir butonun üzerine fare ile gelindiğinde arka plan renginin ve gölgesinin anında değil de 0.3 saniye gibi kısa bir sürede pürüzsüzce değişmesi bu yöntemle sağlanmıştır. Teknik olarak bu, elemanın temel CSS kuralına transition özelliğinin eklenmesiyle gerçekleştirilir. Bu özellik, hangi CSS özelliğinin (background-color, transform vb.), ne kadar sürede (transition-duration) ve hangi hızlanma eğrisiyle (transition-timing-function) değişeceğini tarayıcıya bildirir. Bu küçük ama etkili dokunuşlar, kullanıcı arayüzünün daha "canlı" ve tepkisel hissedilmesine katkıda bulunur.

**3.3.3 Resim Görüntüleyici (Lightbox)**

Proje galerilerinde kullanıcıya kesintisiz bir medya inceleme deneyimi sunmak için, modal pencere (modal window) kullanıcı arayüzü deseni temel alınarak bir Lightbox bileşeni entegre edilmiştir. Bu bileşen, kullanıcının bir ön izleme resmine tıklamasıyla tetiklenir ve sayfanın DOM (Document Object Model) yapısını dinamik olarak manipüle ederek çalışır. Teknik implementasyon, harici bir JavaScript kütüphanesi aracılığıyla yönetilmiştir. Kütüphane, öncelikle data-lightbox özniteliğine sahip <a> etiketlerine bir olay dinleyicisi (event listener) atar. Kullanıcı bu linklerden birine tıkladığında, event.preventDefault() metodu çağrılarak tarayıcının varsayılan linke gitme davranışı engellenir. Ardından script, DOM'a dinamik olarak bir overlay <div> elemanı (CSS'te position: fixed ve yüksek z-index ile konumlandırılmış) ve resmin yükleneceği bir container <div> elemanı enjekte eder. Tıklanan <a> etiketinin href özniteliğinden alınan yüksek çözünürlüklü resim yolu, bu container içindeki <img> elemanının src özniteliğine atanır. Aynı data-lightbox değerine sahip tüm linkler bir dizi (array) veya NodeList olarak gruplanır; bu sayede "ileri/geri" navigasyon okları, bu dizinin index'ini artırıp azaltarak galeri içinde gezinmeyi sağlar. Kapatma butonuna tıklandığında ise bu dinamik olarak oluşturulmuş DOM elemanları kaldırılır veya CSS ile gizlenerek orijinal sayfa akışına geri dönülür. Bu yöntem, sayfa yenilemesine gerek kalmadan, state (durum) yönetimini tamamen istemci tarafında (client-side) çözerek son derece performanslı ve modern bir galeri deneyimi sunar.

**3.4 Erişilebilirlik ve Diğer Teknik Geliştirmeler**

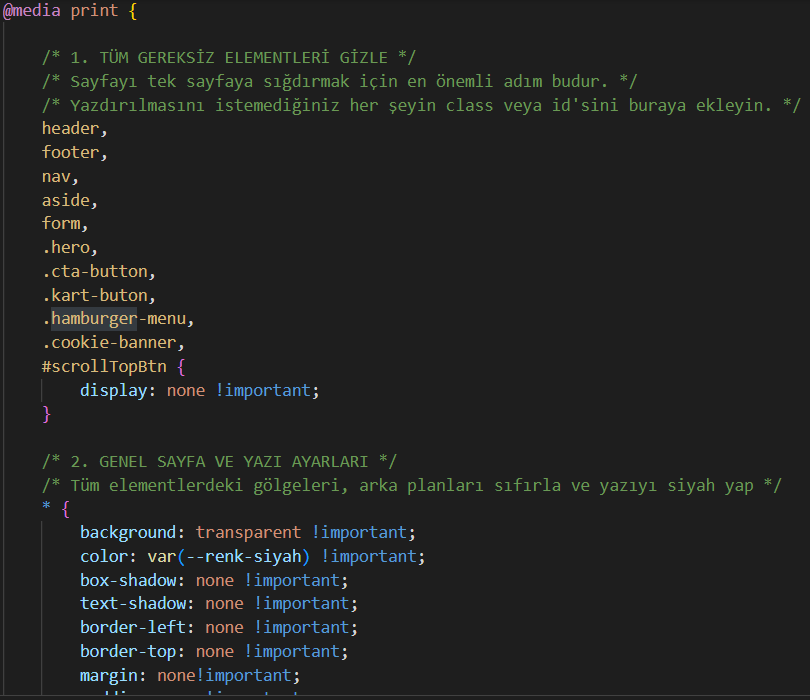
Projenin teknik kalitesini ve kapsayıcılığını artırmak amacıyla bir dizi önemli iyileştirme yapılmıştır. Ekran okuyucular için anlamsal HTML ve görsellerde alt metinleri kullanılarak temel erişilebilirlik standartları sağlanmıştır. Ayrıca, @media print kuralı ile gereksiz arayüz elemanlarını gizleyen bir yazıcı dostu sayfa düzeni oluşturulmuştur. Son olarak, iletişim formuna eklenen istemci taraflı form doğrulama (validation) ile kullanıcı tarafından gönderilen verilerin bütünlüğü güvence altına alınmıştır.

**3.4.1 Temel Erişilebilirlik (Accessibility) Standartları**

Erişilebilirlik, web sitesinin engelli kullanıcılar da dahil olmak üzere herkes tarafından kullanılabilir olmasını sağlayan temel prensipleri ifade eder. Bu projede, daha kapsayıcı bir dijital deneyim sunmak amacıyla, anlamsal HTML5 etiketleri (<main>, <nav> vb.) kullanılarak sayfa yapısının ekran okuyucular tarafından mantıksal bir sırada yorumlanması sağlanmıştır. Sitedeki tüm anlamlı görsellere, içeriği açıklayan alternatif metinler (alt text) eklenerek görme engelli kullanıcılara bilgi sunulmuştur. Aynı zamanda, sitenin fareye ihtiyaç duyulmadan tamamen klavye ile gezilebilir olması ve odaklanılan interaktif elemanların görsel olarak belirginleştirilmesi sağlanarak motor becerileri kısıtlı kullanıcılar için erişim kolaylaştırılmıştır. Bu temel standartlar, metin ve arka plan arasında yeterli renk kontrastı sağlanması gibi detaylarla desteklenerek sitenin daha geniş bir kitle tarafından sorunsuzca kullanılması hedeflenmiştir.

**3.4.2 Yazıcı Dostu Sayfa Düzeni (Print Layout)**

Web sitesi ziyaretçilerinin, proje detayları veya iletişim bilgileri gibi içerikleri fiziksel bir kopya olarak saklama ihtiyacına yönelik, özel bir "Yazıcı Dostu Sayfa Düzeni" geliştirilmiştir. Bu özellik, kullanıcının "Yazdır" komutunu verdiğinde devreye giren ve @media print olarak bilinen özel bir CSS Medya Sorgusu (Media Query) ile hayata geçirilmiştir. Bu kural, normal ekran görünümünü etkilemeden, sadece yazdırma işlemi için geçerli olan bir dizi stil uygular. Bu stiller; navigasyon menüsü, butonlar ve footer gibi kağıt üzerinde işlevi olmayan tüm dijital arayüz elemanlarını display: none; ile gizler. Aynı zamanda, mürekkep tasarrufu sağlamak ve okunabilirliği artırmak için metin rengini siyah, arka planı ise beyaz olarak ayarlar ve ana içeriği kağıda tam sığacak şekilde yeniden düzenler. Bu sayede, kullanıcılar bir sayfayı yazdırmak istediklerinde, karşılarına web sitesinin karmaşık arayüzü yerine, doğrudan ana içeriğe odaklanan, temiz ve profesyonel bir doküman çıkar.



**3.4.2 Yazıcı Dostu Sayfa Düzeni Css Kodu**

**3.4.3 Form Doğrulama (Validation)**

İletişim formundan gönderilen verilerin doğru ve eksiksiz olmasını sağlamak amacıyla, istemci tarafında (client-side) çalışan iki katmanlı bir doğrulama mekanizması geliştirilmiştir. İlk katman olarak, tarayıcının yerleşik yeteneklerinden faydalanan HTML5 öznitelikleri (required, type="email") kullanılarak temel kontroller sağlanmıştır. Daha gelişmiş ve kullanıcı dostu bir deneyim için ikinci katmanda ise özel bir JavaScript script'i devreye girer. Bu script, kullanıcı formu göndermeye çalıştığında tetiklenerek, tüm alanları önceden belirlenmiş kurallara göre (geçerli e-posta formatı, boş bırakılmama durumu vb.) kontrol eder. Herhangi bir hata tespit edildiğinde, formun gönderilmesini event.preventDefault() ile engeller ve hatalı alanın yanına kullanıcıyı bilgilendiren, anlaşılır bir hata mesajı dinamik olarak ekleyerek neyin düzeltilmesi gerektiğini belirtir. Bu iki katmanlı yaklaşım, hem kullanıcı hatalarını en aza indirerek deneyimi iyileştirir hem de sadece geçerli verilerin alınmasını sağlayarak veri bütünlüğünü güvence altına alır.

BÖLÜM DÖRT

OPTİMİZASYON VE PROJE YÖNETİMİ

**4.1 Lightouse ve Kod Optimizasyonu**

Projenin teknik kalitesini ve modern web standartlarına uygunluğunu objektif metriklerle doğrulamak amacıyla, Google tarafından geliştirilen Lighthouse denetim aracı kullanılarak iteratif bir optimizasyon süreci yürütülmüştür.

Bu süreçte web uygulamasının Performans, Erişilebilirlik, En İyi Pratikler ve SEO kategorilerindeki puanları analiz edilmiş ve raporlarda belirtilen iyileştirme önerileri doğrultusunda doğrudan kod tabanında optimizasyonlar yapılmıştır. Performans metriklerini iyileştirmek için, öncelikle Kritik Oluşturma Yolu (Critical Rendering Path) optimize edilmiştir; bu kapsamda, sayfanın ilk render'ını bloke etmeyen JavaScript dosyaları <script> etiketlerine defer özniteliği eklenerek eşzamansız (asynchronous) yüklenmesi sağlanmıştır. Sayfa yükleme süresini doğrudan etkileyen varlıkların (assets) boyutunu düşürmek amacıyla, tüm görseller kayıplı (lossy) sıkıştırma algoritmalarıyla optimize edilmiş ve CSS/JavaScript dosyaları gereksiz boşluk ve yorum satırlarından arındırılarak küçültülmüştür (minification). Erişilebilirlik denetimlerinde tespit edilen eksik alt öznitelikleri görsellere eklenmiş ve WCAG standartlarını karşılamayan düşük renk kontrast oranları düzeltilmiştir. "En İyi Pratikler" ve "SEO" puanlarını artırmak için ise, tüm sayfalara benzersiz meta açıklamaları eklenmiş ve güvenlik açığı oluşturabilecek eski kütüphane kullanımlarından kaçınılmıştır. Bu döngüsel denetim ve iyileştirme süreci sonucunda, sitenin Lighthouse puanlarının 90 ve üzeri seviyelere çıkarılması sağlanarak, projenin sadece işlevsel değil, aynı zamanda performanslı, erişilebilir ve teknik olarak sağlam bir yapıya sahip olduğu doğrulanmıştır.

**4.2 Git Kullanımı ve Projenin Canlıya Alınması**

Proje, yerel geliştirme ortamında git init komutuyla bir Git deposu (repository) olarak başlatılmıştır. Geliştirme süreci boyunca yapılan her mantıksal değişiklik (örneğin, "Header bileşeni tamamlandı", "Karanlık mod özelliği eklendi" vb.), anlamlı ve açıklayıcı mesajlarla birer commit olarak kaydedilmiştir. Ana kod tabanının kararlılığını korumak amacıyla, yeni özellikler doğrudan main ana dalına (branch) işlenmek yerine, özellik dalları (feature branches) üzerinde geliştirilmiştir. Örneğin, "Müşteri Yorumları" bölümü feature/testimonials adında bir dalda tamamlanmış, test edildikten sonra git merge komutu ile main dalına entegre edilmiştir. Tüm bu yerel depo, kodun merkezi bir yedeği ve tek doğruluk kaynağı (single source of truth) olması amacıyla GitHub üzerinde oluşturulan bir uzak depoya (remote repository) bağlanmış ve git push komutlarıyla düzenli olarak senkronize edilmiştir.

Web sitesinin internet üzerinde yayınlanması için geleneksel FTP yöntemleri yerine, modern bir Sürekli Dağıtım (Continuous Deployment - CI/CD) iş akışı benimsenmiştir. Bu süreç için, statik sitelerle kusursuz entegrasyonu ve otomasyon yetenekleriyle öne çıkan Netlify platformu tercih edilmiştir. Netlify projesi, doğrudan projenin GitHub deposunun main dalına bağlanmıştır. Bu bağlantı sayesinde, main dalına her yeni bir kod itildiğinde (git push), Netlify bu değişikliği otomatik olarak algılamış, projeyi kendi sunucularında yeniden oluşturmuş ve saniyeler içinde canlı web sitesini güncellemiştir. Bu otomatize edilmiş iş akışı, manuel dağıtım süreçlerini ortadan kaldırarak hem zamandan tasarruf sağlamış hem de insan hatası riskini en aza indirerek projenin hızlı ve güvenilir bir şekilde güncellenmesine olanak tanımıştır.

BÖLÜM BEŞ

SONUÇ

**5.1 Staj Süresince Elde Edilen Kazanımlar**

Bu staj süreci, üniversitede edindiğim teorik bilgileri Ostaş İnşaat için geliştirilen somut bir proje üzerinde uygulayarak pratik deneyime dönüştürme ve profesyonel iş hayatını tanıma açısından son derece verimli geçmiştir. Bu dönemde, HTML5, CSS3 ve JavaScript gibi temel web teknolojileriyle sıfırdan bir web sitesi inşa etme yetkinliği kazanılmış; özellikle Flexbox, Grid ve Medya Sorguları gibi modern CSS teknikleriyle Responsive Tasarım prensipleri etkin bir şekilde hayata geçirilmiştir. JavaScript ile DOM manipülasyonu yapılarak Karanlık Mod ve Form Doğrulama gibi interaktif bileşenler geliştirme konusunda önemli bir tecrübe edinilmiştir. Teknik becerilerin ötesinde, Git ve GitHub kullanarak versiyon kontrolü yapma ve bir projenin kod tabanını profesyonel standartlarda yönetme disiplini kazanılmıştır. Netlify platformu aracılığıyla bir projenin Sürekli Dağıtım (CI/CD) mantığıyla nasıl otomatize bir şekilde canlıya alındığı ve Lighthouse gibi denetim araçlarıyla performans ve erişilebilirlik optimizasyonunun nasıl yapıldığı pratik olarak öğrenilmiştir. Bu süreç, bir projenin başlangıcından sonuna dek tüm yaşam döngüsünü deneyimleme, karşılaşılan teknik problemlere karşı araştırma yaparak çözüm üretme ve zaman yönetimi gibi değerli mesleki yetkinlikler kazandırmıştır.

**5.2 Proje Değerlendirmesi**

Firma için modern ve kullanıcı dostu bir kurumsal web sitesi geliştirme hedefi, proje kapsamında başarıyla gerçekleştirilmiştir. Proje sürecinde karşılaşılan zorluklar, etkili problem çözme yöntemleriyle aşılarak önemli bir tecrübe kazandırmıştır. Sonuç olarak, tamamlanan bu proje, firmanın dijital kimliğini başarılı bir şekilde yansıtmakla kalmamış, aynı zamanda gelecekte yapılabilecek CMS entegrasyonu gibi geliştirmeler için de ölçeklenebilir ve sağlam bir temel oluşturmuştur.