**T.C**

**SAKARYA ÜNİVERSİTESİ**

**BİLGİSAYAR VE BİLİŞİM BİLİMLERİ FAKÜLTESİ**

**BİLİŞİM SİSTEMLERİ MÜHENDİSLİĞİ BÖLÜMÜ**

**WEB TEKNOLOJİLERİ PROJE RAPORU**

**HAZIRLAYAN**

**Ceyda Sebile Kahveci**

**B221200035**

Proje Konusu = Çiçek ve Çikolata satışı

Giriş: Yaşamın en özel anları, sevdiklerimize duyduğumuz sevgiyi paylaşmanın ve özel günleri daha anlamlı kılmanın vazgeçilmez bir parçasıdır. Bu özel anları renklendirmenin ve sevgimizi ifade etmenin en tatlı yollarından biri de çiçek ve çikolata hediye etmektir. Ancak, geçmişte bu güzelliklere ulaşmak bazen büyük bir zahmet gerektirirdi.

Geçmişte: Çiçekçilere veya çikolata dükkanlarına ulaşmak, özel günlerde sevdiklerimize anlam dolu hediyeler seçmek için bazen gerçek bir maceraydı. Mağazalar arasında dolaşmak, çeşitli seçenekleri değerlendirmek ve en uygununu bulmak oldukça zaman alıcı olabilirdi. Özellikle yoğun dönemlerde, talep karşılanamadığı için bazen hayal kırıklıkları yaşanabiliyordu.

Günümüzde: Teknolojinin gelişmesiyle birlikte, çiçek ve çikolata alışverişi büyük bir dönüşüm geçirdi. Online çiçek ve çikolata satış siteleri, kullanıcılara geniş ürün seçenekleri sunmanın yanı sıra, kolay ve hızlı bir alışveriş deneyimi yaşatıyor. Artık sevdiklerimize özel bir hediye seçmek, sadece bilgisayar veya cep telefonu üzerinden birkaç tıkla mümkün.

Gelecekte: Çiçek ve çikolata satışı konusundaki gelecek hedefim, bu özel deneyimi daha da kişiselleştirmek ve modern teknolojiyi kullanarak duygusal bağları daha güçlü kılmaktır. Çiçek ve çikolata satışı sitesi kurarak, müşterilere sadece ürünleri sunmakla kalmayacak, aynı zamanda duygusal bir deneyim yaşatacağım. Gelecekteki platformum, kullanıcıların özel günlerde sevdiklerine unutulmaz anlar yaşatmak için geniş bir ürün yelpazesi sunacak. Ayrıca, sanal çiçek tasarım stüdyoları ve kişiselleştirilmiş not ekleme gibi interaktif özelliklerle müşterilere benzersiz bir alışveriş deneyimi sunmayı hedefliyorum. Bu platform, sadece çiçek ve çikolata satışı yapmanın ötesinde, insanların duygusal anlarını daha da özel kılma misyonunu taşıyacak. Gelecekteki çiçek ve çikolata satış sitesi, duygusal bağların sanal dünyada da güçlenmesine aracılık ederek, sevgi ve mutluluğu daha da yayacak bir adım olacak.



Toplam da 9 html dosyası vardır. Bu sayfalar: anasayfa.html , çiçekler.html , bitkiler.html , teraryum.html , çikolata.html , hakkımızda.html , iletişim.html , sepet.html , yarışmasoruları.html

1 tane style.css dosyası , 1 tane script.js dosyası vardır

HTML Yapısı:

Navigasyon Bölümü: ( anasayfa.html dosyasında 34. Satır ile 40.satır arası)

Boş bir "alt" niteliği olan bir logo içerir (boş "alt" niteliği içeren bir resim bağlantısı).

"click" ID'sine sahip bir giriş kutusu içerir ve içinde "menu-btn" sınıfına sahip bir etiket bulunur, bu etiket çubuk simgesini içerir.

#header ( FLEXBOX ): ( style.css dosyasında 1.satır ile 5.satır arası)

display: flex;: Header'ı bir flex container olarak ayarlar.

align-items: center;: Header içindeki öğeleri dikeyde ortalar.

justify-content: space-between;: Header içindeki öğeleri yatayda boşluk bırakarak iki kenara yayar.

Navbar (Menü): ( anasayfa.html dosyasında 42.satır ile 51.satır arası)

"navbar" ID'sine sahip bir sırasız liste (<ul>) içerir.

Liste öğeleri (<li>), farklı menü öğelerini temsil eder (Anasayfa, Çiçekler, Bitkiler, Teraryum, Çikolata, Hakkımızda, İletişim).

Her menü öğesi, çeşitli "href" niteliklerine sahip bir bağlantı (<a>) içerir.

Sepet İkonu: ( anasayfa.html dosyasında 52.satır ile 57.satır arası)

"navbar1" ID'sine sahip başka bir sırasız liste (<ul>) içerir.

"sepet.html" ile bağlantılı bir sepet simgesini temsil eden tek bir liste öğesi (<li>) içerir.

CSS Stilleri: ( style.css dosyasında 1.satır ile 49.satır arası)

Navbar Stilleri (#navbar):

Flexbox teknolojisi kullanılmıştır. ( display: flex ) , ( style.css dosyasında 19.satır )

align-items: center;: öğeleri dikeyde ortalar. ( style.css dosyasında 20.satır )

Liste öğeleri üzerinde özel bir stil yok, biraz dolguya ve göreceli bir konuma sahiptir.

Bağlantılar (<a>), metin dekorasyonuna sahip değildir, bir geçiş efekti vardır ve üzerine gelme/etkin durumlarına sahiptir.

Aktif ve üzerine gelme durumlarında bir alt çizgi efekti bulunur.

Sepet İkonu Stilleri (#navbar1): ( style.css dosyasında 52.satır ile 82.satır arası)

Flexbox teknolojisi kullanılmıştır. ( display: flex ) , ( style.css dosyasında 54.satır)

align-items: center;: Menü içindeki öğeleri dikeyde ortalar. ( style.css dosyasında 55.satır )

Navbar stiline benzer.

Sepet simgesi bağlantıları üzerinde üzerine gelme ve etkin durumlar alt çizgi efektiyle birlikte bulunur.

Mobil Duyarlı Stiller(Responsive): ( style.css dosyasında 98.satır ile 140.satır arası)

960 pikselden daha geniş ekranlarda çubuk simgesi başlangıçta gizlenmiştir (display: none).

Ekran genişliği 960 piksel veya daha az olduğunda çubuk simgesi görünür olur.

Onay kutusu gizlenir.

Ekran genişliği 960 piksel veya daha az olduğunda, navbar üstte sabitlenir, başlangıçta ekranın dışında konumlandırılır ve onay kutusu işaretlendiğinde görünüme geçer.

Her menü öğesi, onay kutusu işaretlendiğinde kayan bir efekte sahiptir.

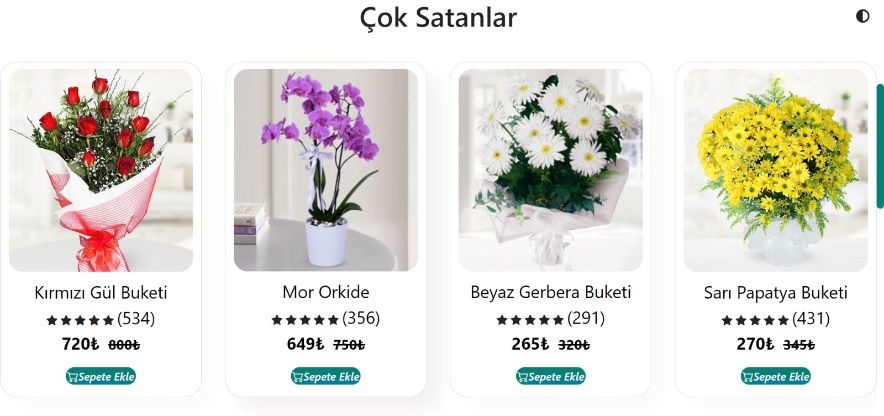
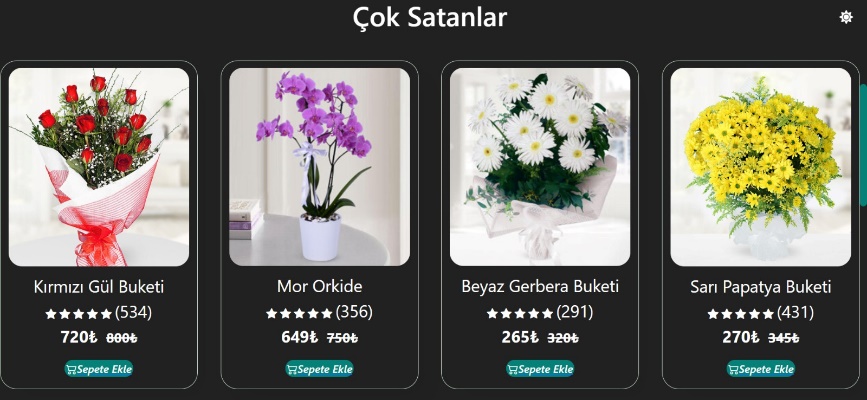
Kullanılan Teknolojiler:

HTML: İçeriğin yapısını oluşturmak için kullanılır.

CSS: HTML öğelerini stillemek için kullanılır.

Flexbox: Esnek düzenler oluşturmak için kullanılır.

Bootstrap: Sepet ikonu için bootstrap kütüphanesi kullanılır.

Sayfada sağ üst köşede tıklandığında sayfanın karanlık mod ve aydınlık mod olması için özellik ekledim.

HTML: (anasayfa.html dosyasında 59.satır ile 62.satır arası)

<div class="dark-mode-toggle" onclick="toggleDarkMode()">: Karanlık modu açıp kapatan bir buton ekledim . Butona tıklandığında toggleDarkMode fonksiyonu çağrılacak.

CSS: (style.css dosyasında 694.satır ile 724.satır arası)

.dark-mode: Karanlık modun aktif olduğu durumu temsil eden bir sınıf. Arka plan rengi ve metin rengi belirttim.

.dark-mode-toggle: Karanlık modu açıp kapatan butonun stilini belirleyen sınıf. Bu buton sabit bir konumda (position: fixed) sağ üst köşede bulunacak şekilde tasarladım.

dark-mode .pro span, .dark-mode .pro .bi-star-fill, .dark-mode .pro .text1, .dark-mode .pro .fs-4, .dark-mode .pro .fs-5: Bu kurallar, karanlık modda (.dark-mode sınıfı altında) belirli bir sınıfa (pro) ait olan yazıları, yıldızları ve belirli font büyüklüklerini beyaz renk yapar.

JavaScript: ( script.js dosyasında 99.satır ile 117.satır arası)

toggleDarkMode() fonksiyonu: Bu fonksiyon, butona tıklandığında çalışacak ve sayfanın arka plan rengini ve metin rengini değiştirecek. Eğer karanlık mod açıksa, kapatılacak; eğer kapalıysa, açılacak.

Bu örnekte, HTML ile yapılan düğme ve CSS ile stil belirlemeleriyle birlikte JavaScript kullanılarak sayfa üzerinde karanlık modun açılması ve kapanmasını sağladım.

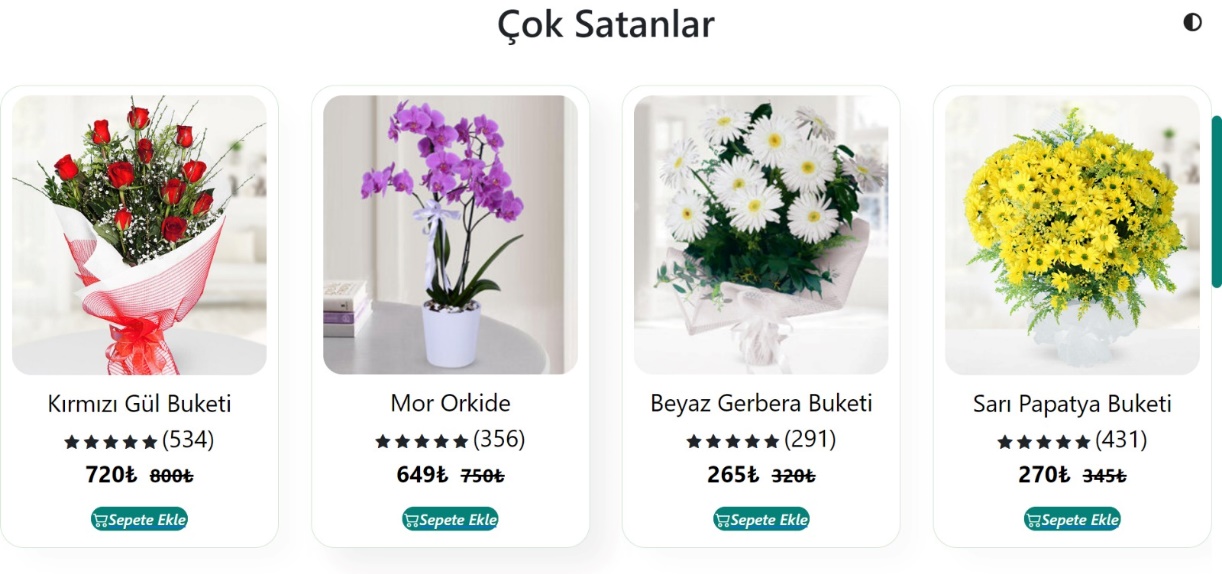


CSS: ( style.css dosyasında 726.satır ile 741.satır arası)

.dark-mode #buton1: Bu kural, karanlık modda belirli bir ID'ye (buton1) sahip bir butonun metin rengini ve kenarlık rengini belirler.

.dark-mode #buton1:before: Bu kural, belirli bir ID'ye (buton1) sahip butonun önceki (before) öğesinin arka plan rengini belirler.

.dark-mode #buton1:hover, .dark-mode #buton1:focus: Bu kurallar, belirli bir ID'ye (buton1) sahip butona fare üzerine gelindiğinde ve odaklandığında arka plan rengi ve metin rengini belirler.



HTML: ( anasayfa.html dosyasında 75.satır ile 235.satır arası)

Kullanılan HTML kodları, ürünlerin tanımlandığı bir bölümü (#Ürün1) içerir. Her bir ürün, bir div elementi içinde yer alır ve ürün detaylarını içeren alt öğeleri içerir (resim, başlık, yıldızlar, fiyatlar, sepete ekle butonu).

CSS: ( style.css dosyasında 145.satır ile 205. satır arası)

#Ürün1: Bu kurallar, Ürün1 ID'sine sahip bölümü stilize eder.

#Ürün1 .pro-container: Bu kurallar, ürünleri içeren konteyneri stilize eder. Ürünlerin arasında boşluk bırakma ve sıralama işlemleri burada yapıldı. ( display: flex ) özelliği kullanıldı. (style.css dosyasında 152.satır)

#Ürün1 .pro: Bu kurallar, her bir ürünü stilize eder. Ürün kutularının genişlik, yükseklik, kenarlık, köşe yuvarlatma, gölge ve geçiş efektleri gibi özellikleri belirlenmiştir.

#Ürün1 .pro:hover: Bu kural, ürün kutularına fare üzerine gelindiğinde uygulanan stil değişikliklerini tanımlar.

#Ürün1 .pro img: Bu kural, ürün resimlerini genişlik ve köşe yuvarlatma özellikleriyle stilize eder.

#Ürün1 .pro .des span: Bu kural, ürün başlıklarını stilize eder.

#Ürün1 .pro .cart: Bu kural, sepete ekle butonunu stilize eder.

.text: Bu kural, metni çizili (strike-through) olarak biçimlendirir.

FLEXBOX:

#Ürün1 .pro-container: Bu kural, ürünleri içeren (.pro-container) bir flex container olarak tanımlar. display: flex; özelliği ile içindeki ürünleri yatay bir düzende sıralar. justify-content: space-between; ile içindeki öğeleri konteynerin genişliği boyunca eşit aralıklarla yerleştirir ve flex-wrap: wrap; ile öğeleri sardırarak bir satırda sığmıyorsa alt satıra geçmelerini sağlar.

#Ürün1 .pro: Bu kural, her bir ürünü temsil eden kutuları stilize eder ve display: flex; kullanarak içindeki öğeleri yatay bir düzende sıralar.



HTML: ( anasayfa.html dosyasında 241.satır ile 243.satır arası)

HTML, bir anket bölümü oluşturuyor. Bu bölümde bir buton (buton1) bulunuyor. Buton, kullanıcıyı "yarışmasoruları.html" sayfasına yönlendiren bir bağlantı içeriyor.

CSS: ( style.css dosyasında 210.satır ile 274.satır arası)

CSS, anket bölümünün stilini belirliyor.

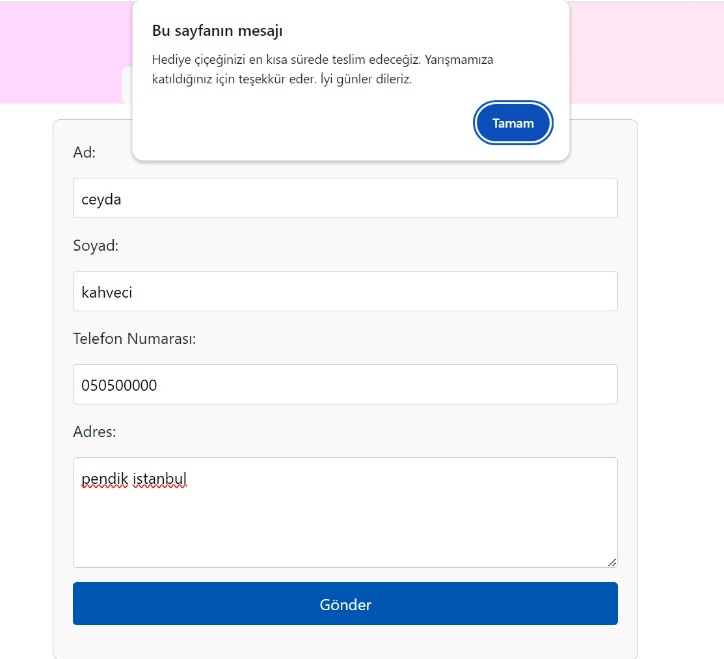
.anket: Anket bölümünün genel stilini belirleyen bir sınıf.

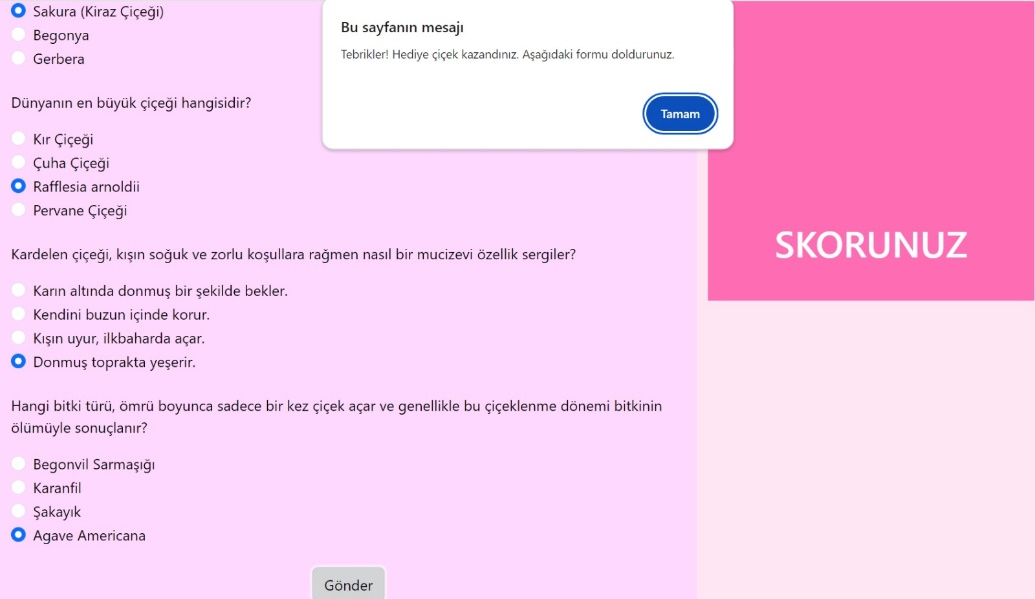
#buton1: Yarışma soruları butonunun stilini belirleyen bir ID. Butonun genel özellikleri, renk geçişleri, gölgeler ve diğer stil özellikleri bu bölümde tanımlandı.

RESPONSİVE:

@media (max-width: 724px): Ekran genişliği 724 piksele eşit veya daha küçükse uygulanacak özel stiller.

Bu kod örneği, HTML ve CSS kullanarak bir anket bölümü oluşturuyor ve bu bölümde bir buton yer alıyor. Buton, kullanıcının yarışma sorularına yönlendirilmesini sağlar ve Responsive tasarım için de bir media query kullandım. bu, ekran genişliği belirli bir değerin altında olduğunda stil değişikliklerinin yapılmasını sağlıyor.





HTML: ( yarışmasoruları.html dosyası içerisinde 64.satır ile 226.satır arası)

Soruların yer aldığı ana bölüm <section id="questions"> içerisinde tanımlandı.

Form ve Sorular:

Soruların yer aldığı form elemanları (<form>, <p>, <div>, <input>, <label>) mevcut.

Her soru için radio butonları kullanıldı.

Responsive Tasarım: ( style.css dosyası içerisinde 636 satır ile 641.satır arası)

Media query (@media screen and (max-width: 600px)) ile küçük ekranlara uygun stil düzenlemeleri yapıldı.

CSS Kullanımı:

Genel Stiller: ( style.css dosyası içerisinde 594 satır ile 634.satır arası)

.question-form, .result, ve .container gibi sınıflara genel stil verildi.

Form elemanlarına (input, textarea, button) genel stil tanımlamaları bulunuyor.

Duyarlılık Eklemeleri: (RESPONSİVE) (style.css dosyası içerisinde 643. satır ile 662.satır arası)

Media query ile küçük ekranlara uygun stil düzenlemeleri yapıldı.

Küçük ekranlarda formun genişliğini maksimuma çıkarır.

Küçük ekranlarda öğeler arasındaki boşluğu azaltır.

Küçük ekranlarda metin ve telefon giriş alanlarına daha küçük padding ekler.

Küçük ekranlarda gönder butonuna daha küçük padding ekler.

Javascript Kullanımı: ( script.js dosyası içerisinde 12.satır ile 68.satır arası )

Soruların Kontrolü:

Kullanıcının verdiği cevaplar, doğru cevaplarla karşılaştırılarak puan hesaplanıyor.

Puan yüzde cinsinden animasyonlu bir şekilde gösteriliyor.Kullanıcı 100 puan alırsa, ödül kazandığını belirten bir tebrik mesajı gösteriliyor.

Kullanıcı Bilgi Formu: ( script.js dosyası içerisinde 70.satır ile 92.satır arası )

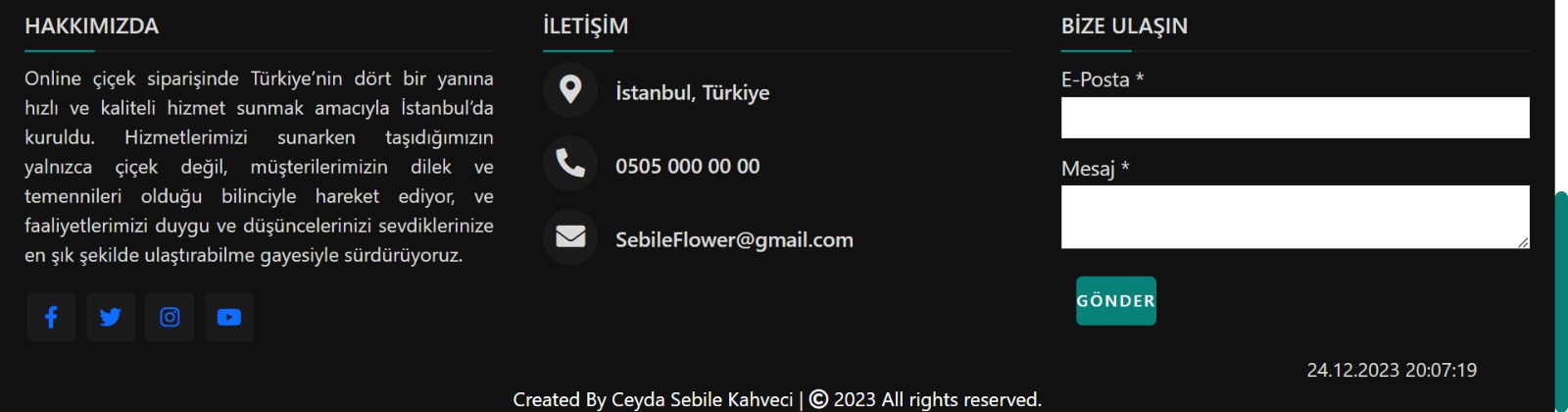
Kullanıcı ödül kazandığında, ad, soyad, telefon numarası ve adres bilgilerini girmesi için bir form görüntüleniyor.

Bu form gönderildiğinde, teşekkür mesajı görüntüleniyor.

Bootstrap Kullanımı:

Bootstrap sınıfları, örneğin btn, btn-light, container, row, col-8, col-4, py-3, my-4, vb., sayesinde tasarım ve düzenleme kolaylığı sağlandı.

Bu kod örneği, HTML, CSS ve Javascript kullanarak bir yarışma soruları formu oluştururken, doğru cevapları kontrol edip kullanıcıya puan veren ve ödül kazanılması durumunda bilgi girişi için bir form gösteren bir uygulama içeriyor. Bootstrap sınıfları da tasarımı kolaylaştırmak için kullanıldı.



HTML : ( anasayfa.html dosyasında 257.satır ile 308. satır arası )

Footer Yapısı:

<footer> etiketi ile sayfanın alt kısmında yer alan footer bölümü oluşturuldu.

"Hakkımızda", "İletişim", ve "Bize Ulaşın" başlıklı üç ana kutucuktan oluşan bir yapı kullanıldı.

Sosyal medya bağlantıları içeren ikonlar (<span class="fab fa-facebook-f"></span>, <span class="fab fa-twitter"></span>, <span class="fab fa-instagram"></span>, <span class="fab fa-youtube"></span>) eklendi.

İletişim bilgileri ve bir iletişim formu içermektedir.

CSS Kullanımı: (style.css dosyasında 279.satır ile 432.satır arası)

Stil Tanımlamaları: (style.css dosyasında 279.satır ile 418. satır arası)

Genel stil tanımlamaları, renkler, font büyüklükleri gibi özellikler belirlendi.

main-content, left, center, ve right kutucuklarına özel stil verildi.

İletişim bilgileri ve sosyal medya ikonları için özel stiller tanımlandı.

.main-content sınıfına (display: flex özelliği verildi) (style.css dosyasında 296.satır)

Duyarlılık Eklemeleri ( RESPONSİVE): (style.css dosyasında 420.satır ile 432. satır arası)

@media screen and (max-width: 900px)

(max-width: 900px) koşulu, tarayıcı ekranının maksimum genişliğinin 900 piksel olduğunu belirtir.

Bu medya sorgusu, eğer ekran genişliği 900 pikselden küçükse içeriğin stilini değiştirecek.

FLEXBOX:

.main-content (style.css dosyasında 425.satır ile 428. satır arası)

Sayfadaki ana içeriği temsil eden bir sınıfı seçer.

flex-wrap: wrap; ile, içeriğin flex kutularının alt satıra geçmesini sağlar.

flex-direction: column; ile, içeriğin sütun yönlü (column) düzenlenmesini belirtir.

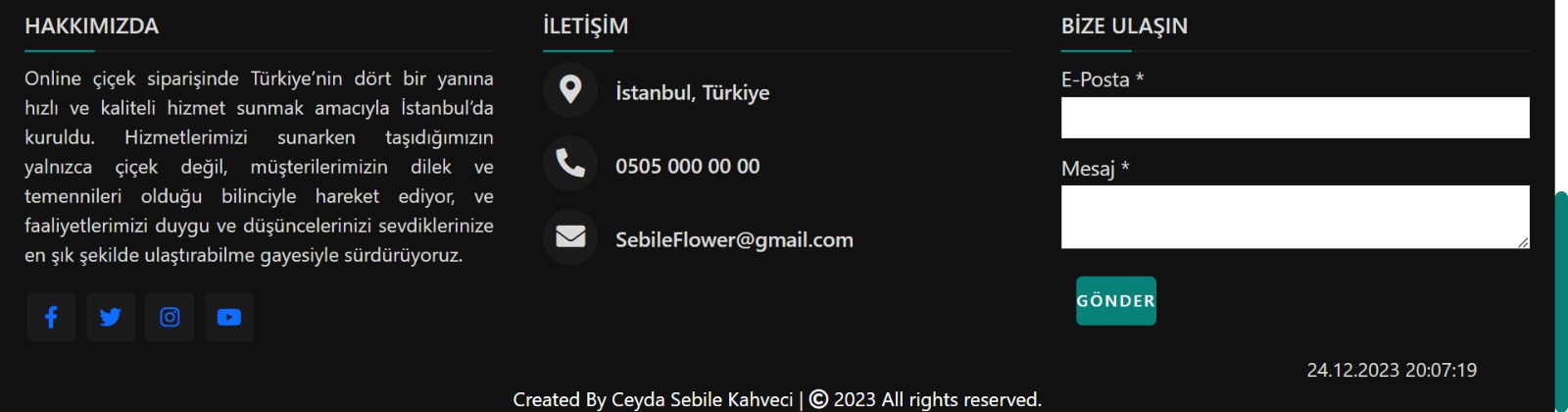
Bootstrap Kullanımı: (anasayfa.html dosyasında 6.satır ile 16. satır arası)

Sosyal medya ikonları ve iletişim ikonları için bootstrap kütüphaneleri kullanıldı.

Flexbox Kullanımı: (style.css dosyasında 295.satır ile 301. satır arası ve 425. Satır ile 428.satır arası)

display: flex ve flex-basis gibi Flexbox özellikleri, sayfa düzeninin oluşturulmasında kullanıldı. Örneğin, main-content içindeki kutucuklar flex konteyneri olarak düzenlendi.

Bu kod örneği, sayfa altındaki bir footer yapısını oluştururken Hakkımızda, İletişim ve Bize Ulaşın başlıklı üç ana bölüm içeriyor. İletişim bilgileri, sosyal medya bağlantıları ve iletişim formu ile zenginleştirilmiş bir tasarım içermektedir. Duyarlılık eklemeleri sayesinde mobil cihazlara uyum sağlamaktadır.



HTML ve CSS: (anasayfa.html dosyasında 311.satır ile 313. satır arası)

<div style="float: right; margin-right: 5%;" id="zaman"></div>: Saat ve tarihi gösterecek olan <div> elementi Sağa yaslanmış (float: right;) ve sağdan 5% boşluk bırakılmış bir şekilde stilendirilmiştir.

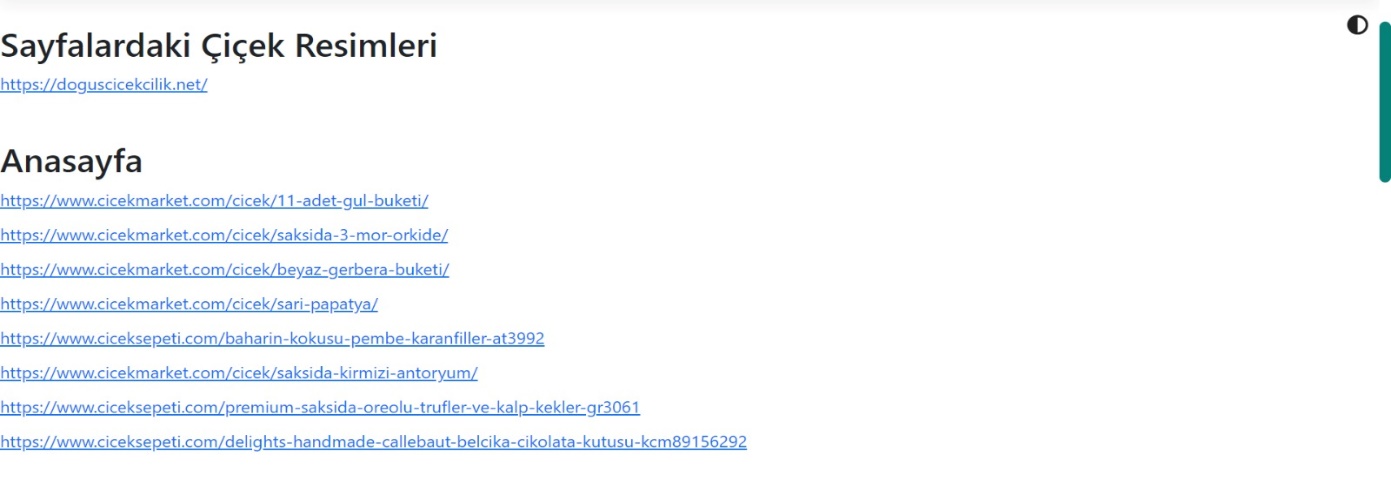
JavaScript: (script.js dosyasında 1.satır ile 7.satır arası)

function tarihSaat() { ... }: tarihSaat adlı bir JavaScript fonksiyonu oluşturulmuştur. Bu fonksiyon, mevcut tarih ve saat bilgisini alır ve HTML içindeki belirli bir <div> elementinin içine yerleştirir.

var date = new Date().toLocaleString('tr-TR');: JavaScript'te Date objesi oluşturularak, toLocaleString fonksiyonu kullanılarak tarayıcı tarafından kullanıcının yerel diline ('tr-TR' Türkçe için) uygun bir şekilde tarih ve saat bilgisi elde edilir.

document.getElementById("zaman").innerHTML = date;: zaman id'sine sahip HTML elementine, elde edilen tarih ve saat bilgisini içeren date değişkenini yerleştirir.

setInterval(tarihSaat, 1000);: tarihSaat fonksiyonunu 1000 milisaniyede bir (1 saniye) çağıran ve böylece sürekli olarak güncel tarih ve saat bilgisini gösteren bir zamanlayıcı (timer) işlemi oluşturur.



HTML: (hakkımızda.html dosyasında 64.satır ile 267. satır arası)

Sayfada farklı kategorilere ait ürün linklerini içeren <div> containerları ve bu containerlarda başlık (<h2>) ve linklerin bulunduğu alt containerlar bulunmaktadır.

Her kategori için ayrı bir <div class="link-container"> kullanılmıştır. Her ürün linki ise <div class="link-item"> içinde bulunmaktadır.

<a> elementleri ile her bir ürünün linki verilmiştir.

CSS: (style.css dosyasında 668.satır ile 676. satır arası)

.link-container ve .link-item sınıfları ile belirli stil özellikleri tanımlanmıştır.

margin-bottom özelliği ile link containerlar ve link itemler arasında boşluk bırakılmıştır.

.link-item sınıfındaki margin-bottom özelliği ile her bir ürün linki arasında boşluk bırakılmıştır.

Responsive Tasarım (Media Query): (style.css dosyasında 678.satır ile 688. satır arası)

@media screen and (max-width: 600px): Ekran genişliği 600 pikselden küçük olduğunda uygulanacak stil özelliklerini belirtir.

.link-container sınıfındaki text-align: center; özelliği ile link containerların içindeki metinlerin ortalanması sağlanır.

.link-item sınıfındaki display: block; margin: 0 auto; özellikleri ile link itemlerin blok olarak düzenlenmesi ve ortalanması sağlanır.



HTML: (iletişim.html dosyasında 68.satır ile 85. satır arası)

İletişim bilgilerini ve iletişim formunu içeren bir <div class="contact-section"> bulunmaktadır.

İletişim bilgileri <div class="contact-info"> içinde bulunmaktadır. Her bir bilgi ayrı bir <div> içinde ve bir <i> (ikon) ile temsil edilmektedir.

İletişim formu <div class="contact-form"> içinde bulunmaktadır. Form, ad, e-posta ve mesajı içeren input ve textarea elemanlarını içerir.

CSS: (style.css dosyasında 439.satır ile 542. satır arası)

Genel stil özellikleri \* selektörü ile tanımlanmıştır. Box model, padding ve margin sıfırlanmıştır.

.iletişim ve .contact-section sınıfları ile sayfa ve iletişim bölümünün stil özellikleri belirlenmiştir.

.contact-info ve .contact-form sınıfları ile iletişim bilgileri ve iletişim formunun genel özellikleri belirlenmiştir.

Form elemanlarına özel stiller (örneğin, background, color, border, border-radius vb.) tanımlanmıştır.

Flexbox: (style.css dosyasında 444.satır ile 457. satır arası)

.iletişim{ min-height: 100vh; background-size: cover; display: flex; justify-content: center; align-items: center; }: Bu stil özellikleri, iletişim sayfasının arka planını kaplamak için kullanılır. Sayfanın yüksekliğini (min-height) ekran yüksekliği kadar alır, arka plan resmini kaplamak için 'cover' özelliği kullanılır ve içerikleri merkeze hizalamak için flexbox kullanılır.

.contact-section{ width: 100%; display: flex; justify-content: center; align-items: center; }: İletişim formunu içeren bölümün genişliği ve içeriklerin merkezlenmesi için kullanılır.

Responsive tasarım için (style.css dosyasında 544.satır ile 560 satır arası)

@media sorgusu, ekranın maksimum genişliği 950 piksel olduğunda aşağıdaki stil özelliklerini uygular:

.contact-section elemanının flex-direction özelliği column olarak değiştirilir. Bu, içindeki elemanları dikey (yatay yerine dikey) bir düzen içinde sıralar.

.contact-info ve .contact-form elemanlarına margin özelliği uygulanır. Bu, bu elemanlara bir kenar boşluğu (margin) ekler.

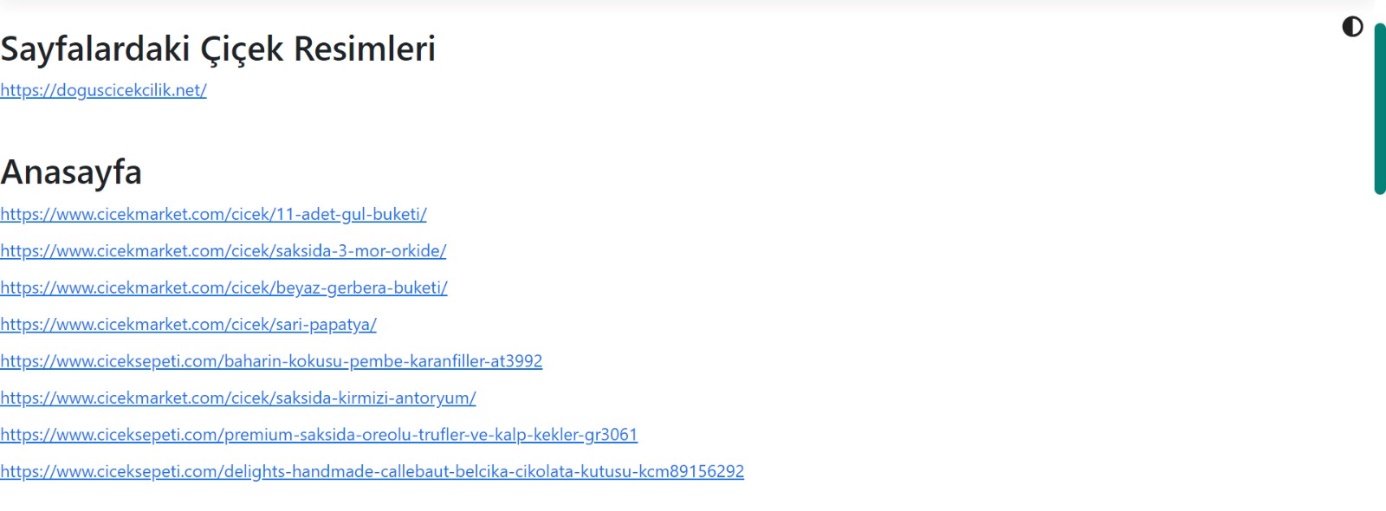
.contact-form elemanının başlık (h2 etiketi içinde) olan yazının font büyüklüğü 30 pikseline düşürülür.

.text-box sınıfına sahip input ve textarea elemanlarının genişliği yüzde 100 yapılır, yani ekranın tam genişliğini kaplar.

Bu değişiklikler, ekran genişliği 950 pikselden küçüldüğünde sayfanın daha iyi bir şekilde görüntülenmesini sağlamak için yapılır. Özellikle küçük ekranlı cihazlar (örneğin, tabletler veya telefonlar) için daha uygun bir düzen ve görünüm sağlar.

Alert Messages CSS: (style.css dosyasında 562.satır ile 588. satır arası)

.alert-success ve .alert-error sınıfları ile başarı ve hata durumları için alert mesajlarının stil özellikleri belirlenmiştir.



CSS: (style.css dosyasında 746.satır ile 753. satır arası)

body::-webkit-scrollbar: Bu stil özelliği, web sayfasının body (sayfa gövdesi) elementine ait scrollbar'ın genel özelliklerini tanımlar. -webkit-scrollbar öneki, bu özelliklerin WebKit tabanlı tarayıcılarda (Chrome, Safari) kullanılacağını belirtir.

width: 11px;: Scrollbar'ın genişliğini 11 piksel olarak belirler. Bu, scrollbar'ın ne kadar geniş olacağını kontrol eder.

body::-webkit-scrollbar-thumb: Bu stil özelliği, scrollbar'ın "thumb" adı verilen, sürüklenebilen bölgesinin özelliklerini tanımlar.

background: #088178;: Scrollbar thumb'ın arka plan rengini belirler. Bu durumda, rengi #088178 olarak ayarlanmıştır.

border-radius: 8px;: Scrollbar thumb'ın köşelerine yuvarlak bir kenarlık (border-radius) ekler.