**İSTANBUL 29 MAYIS ÜNİVERSİTESİ**

**İKTİSADİ VE İDARİ BİLİMLER FAKÜLTESİ**

**YÖNETİM BİLİŞİM SİSTEMLERİ**

**WEB TABANLI UYGULAMA PROGRAMLAMA**

**PROJE KONU: ANKET OLUŞTURMA SİTESİ**

**DERS KODU: YBS204**

**ÖĞRENCİ AD: EMİRCAN**

**ÖĞRENCİ SOYAD: GÖKTAŞ**

**ÖĞRENCİ NO: 030422037**

**GITHUB LINK:** https://github.com/emirriuss/anket.sepeti

**SITE LINK:** https://emirriuss.github.io/anket.sepeti/

**İÇİNDEKİLER**

1. **PROJENIN ADI**
2. **PROJENIN AMACI VE KAPSAMI**
3. **TEKNOLOJİK ALTYAPI VE KULLANILAN DİLLER**
4. **KLASÖR VE DOSYA YAPISI**
5. **BENZER PLATFORMLARLA KARŞILAŞTIRMA**
6. **INDEX.HTML**
7. **BIRINCI.CSS**
8. **GIRIS.HTML**
9. **UYE-OL.HTML**
10. **PHP KULLANICI KAYIT VE GİRİŞ SİSTEMİ**
11. **SCRIPT.JS**
12. **MUSTERİ-ANKETİ.HTML**
13. **KURUMSAL-ANKET.HTML**
14. **KISISEL-ANKET.HTML**
15. **IS-ANKETI.HTML**
16. **KULLANILAN TEKNOLOJİLER**
17. **GITHUB DESKTOP İLE PROJEYİ GITHUB'A YÜKLEME VE GITHUB LINKLERIM**

**PROJENİN ADI**

**ANKETSEPETİ**  
Kullanıcıların farklı türde anketlere katılabileceği veya kendi anketlerini oluşturabileceği interaktif bir web platformudur.

**PROJENİN AMACI VE KAPSAMI**

**Amaç:**  
AnketSepeti, bireylerin ya da kurumların çeşitli konularda veri toplamak için hızlı ve kullanıcı dostu bir ortamda anket oluşturmasını ve katılım sağlamasını hedefler.

**Kapsam:**

* Üyelik sistemi: Kullanıcıların üye olarak anket oluşturabilmeleri veya katılabilmeleri sağlanır.
* Anket türleri: Kişisel, kurumsal, iş ve müşteri memnuniyeti anketleri gibi farklı türlerde anketler sunulur.
* Görsel destek: Anket sayfaları ilgili görsellerle desteklenmiştir.
* Form doğrulama ve kullanıcı etkileşimi için JavaScript desteği bulunmaktadır.
* Modern ve responsive bir kullanıcı arayüzü sunar.

**TEKNOLOJİK ALTYAPI VE KULLANILAN DİLLER**

* **HTML5:** Sayfa yapısının oluşturulmasında kullanılmıştır.
* **CSS3:** Tasarım ve stil dosyası olan birinci.css ile kullanıcı arayüzü modernleştirilmiştir.
* **JavaScript:** script.js dosyası ile sayfa etkileşimleri ve form kontrolleri gerçekleştirilmiştir.
* **Görseller:** .jpg, .png, .webp formatlarında çeşitli görseller ile kullanıcı deneyimi desteklenmiştir.

**KLASÖR VE DOSYA YAPISI**

AnketSepeti/

└── AnketSepeti/

├── birinci.css → Tüm sayfaların ortak stil dosyası

├── script.js → Form kontrol ve etkileşim scriptleri

├── index.html → Ana sayfa

├── giris.html → Giriş sayfası

├── uye-ol.html → Üyelik kayıt sayfası

├── kisisel-anket.html → Kişisel anket sayfası

├── kurumsal-anket.html → Kurumsal anket sayfası

├── is-anketi.html → İş anketi sayfası

├── musteri-anketi.html → Müşteri memnuniyet anketi sayfası

├── \*.jpg / \*.png / \*.webp → Sayfalarda kullanılan görseller

**BENZER PLATFORMLARLA KARŞILAŞTIRMA**

AnketSepeti, kullanıcıların farklı amaçlara yönelik (kişisel, kurumsal, iş, müşteri memnuniyeti vb.) anketler hazırlayabildiği, basit ve kullanıcı dostu bir arayüz sunan web tabanlı bir platformdur. Proje; piyasada yaygın olarak kullanılan **Google Forms** ve **SurveyMonkey** gibi hazır anket çözümleriyle karşılaştırıldığında belirli avantajlar ve bazı sınırlılıklar taşımaktadır.

**Karşılaştırmalı Özellik Tablosu**

| **Özellik** | **AnketSepeti** | **Google Forms** | **SurveyMonkey** |
| --- | --- | --- | --- |
| **Kullanım Ücreti** | Tamamen ücretsizdir. Kullanıcı kısıtlaması yoktur. | Tamamen ücretsiz (Google hesabı ile). | Ücretsiz sürüm sınırlı; gelişmiş özellikler ücretlidir. |
| **Anket Türleri** | Kategorize edilmiş: Kişisel, Kurumsal, İş, Müşteri anketleri ayrı ayrı sunulur. | Tür sınıflandırması yok; tek şablon üzerinden anket yapılır. | Tür sınıflandırması yok; ancak şablonlar mevcuttur. |
| **Kullanıcı Arayüzü** | Basit, görsel destekli ve sabit HTML yapısı. | Minimalist ve sade, Google ekosistemine uyumlu. | Profesyonel, zengin bileşenli ve özelleştirilebilir. |
| **Tasarım Özelleştirme** | CSS ile elle müdahale edilerek detaylı özelleştirme mümkündür. | Renk ve tema sınırlıdır; kişiselleştirme azdır. | Gelişmiş tema ve şablon özelleştirmeleri mümkündür. |
| **Etkileşim ve Animasyon** | JavaScript ile basit doğrulamalar ve kullanıcı uyarıları mevcuttur. | Otomatik geçişler ve uyarılar sağlar. | Gelişmiş yönlendirme, kural tabanlı akışlar sunar. |
| **Cihaz Uyumluluğu** | Mobil uyumluluk kısıtlı; responsive tasarım temel düzeydedir. | Mobil uyumlu, duyarlı yapıdadır. | Tüm cihazlara duyarlı ve performans odaklıdır. |
| **Görsel Zenginlik** | Anketlerde görsellerle destekleme imkânı vardır. | Görsel desteği sınırlıdır. | Görsel, video, logo ve marka özelleştirme desteklenir. |
| **Veri Analizi / Raporlama** | Temel seviyede; harici sistemlerle entegrasyon yoktur. | Google Sheets entegrasyonu ile temel analizler yapılabilir. | Gelişmiş grafikler, filtreleme, otomatik raporlar mümkündür. |
| **Veri Saklama** | HTML tabanlı; kullanıcı verisi tarayıcı düzeyindedir. | Google Drive üzerinde saklama yapılır. | Bulut tabanlı, kullanıcı hesabına özel sakl |

**INDEX.HTML**

Projenin ana sayfasını oluşturan temel yapıyı içermektedir. Dosya <!DOCTYPE html> ifadesi ile başlamakta olup, bu satır tarayıcıya sayfanın HTML5 standardında yazıldığını bildirir. Hemen ardından gelen <html lang="tr"> etiketiyle sayfanın dilinin Türkçe olduğu belirtilmiştir. Bu, hem SEO (arama motoru optimizasyonu) hem de erişilebilirlik açısından önemlidir.

<head> etiketleri arasında sayfa ile ilgili meta bilgiler, stil bağlantıları ve başlık yer almaktadır. Örneğin <meta charset="UTF-8"> etiketi, karakter kodlamasının Türkçe karakterleri destekleyen UTF-8 olduğunu belirtir. <meta name="viewport" content="width=device-width, initial-scale=1"> etiketi ise sayfanın mobil cihazlarda düzgün görünmesini sağlar. Bu satır sayesinde sayfa genişliği, cihazın ekran boyutuna göre ayarlanır ve responsive tasarım etkili bir şekilde çalışır.

<link rel="icon" href="aharfi.jpg" type="image/x-icon">

Bu satır, tarayıcı sekmesinde görünecek olan favicon'u tanımlar. Burada "aharfi.jpg" adlı bir simge kullanılmıştır. Ardından gelen <title>Anket Türleri</title> etiketi, tarayıcı sekmesinde görüntülenecek başlık metnini "Anket Türleri" olarak belirler.

CSS bağlantısı şu şekilde yapılmıştır:

<link rel="stylesheet" href="birinci.css">

Bu satır ile dış CSS dosyası (birinci.css) HTML sayfasına dahil edilmiştir. Bu sayede tüm stil ve görünüm ayarları bu dış dosyadan alınır. Ek olarak <style> etiketi içerisinde yer alan html { scroll-behavior: smooth; } ifadesi, sayfa içindeki bağlantıların yumuşak kaydırma (smooth scrolling) ile çalışmasını sağlar. Yani kullanıcı bir bağlantıya tıkladığında sayfa aniden değil, yavaşça o bölüme iner.

**Gövde (Body) Yapısı ve Navbar**

<body> etiketi ile sayfanın görünen kısmı başlar. İlk yapı öğesi olan <header class="navbar"> satırı, sitenin en üst bölümündeki navigasyon barını (üst menüyü) tanımlar. CSS içerisinde .navbar sınıfı tanımlanmış ve bu yapı flexbox kullanılarak yatay olarak hizalanmıştır.

Navbar içerisinde ilk olarak <div class="logo">AnketSepeti</div> yer alır. Bu, sitenin logosu veya marka adını temsil eder. Görsel bir logo kullanılmasa da marka adı metin olarak belirtilmiş ve CSS ile dikkat çekici hale getirilmiştir.

Ardından gelen <ul class="nav-links"> bloğu, menü bağlantılarını içeren sırasız bir listedir. İçerisindeki her <li> etiketi, bir bağlantı içerir. Örnek olarak:

<li><a href="#anasayfa">Ana Sayfa</a></li>

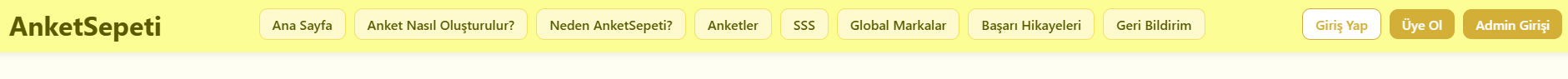
Bu yapı sayesinde kullanıcı sayfa içindeki farklı bölümlere (örneğin Ana Sayfa, Anketler, SSS gibi) kolaylıkla geçiş yapabilir. Bu bağlantılar id değerlerine referans verdiğinden, sayfa içinde hızlı gezinme sağlanır.

Son olarak navbar içerisinde <div class="auth-buttons"> bloğu yer alır. Bu bölümde "Giriş Yap" ve "Üye Ol" butonları bulunmaktadır. Her biri birer <a> etiketi ile başka sayfalara (giris.html ve uye-ol.html) yönlendirme yapmaktadır:

<a href="giris.html" class="login-btn" id="loginBtn">Giriş Yap</a>

<a href="uye-ol.html" class="signup-btn" id="signupBtn">Üye Ol</a>

Bu bağlantılarda kullanılan sınıf isimleri (login-btn, signup-btn) CSS dosyasında özel olarak tanımlanmış ve görsel olarak farklılaştırılmıştır. Ayrıca bu bölümde yer alan <span id="welcomeMessage" style="display: none;"> etiketi, kullanıcı giriş yaptığında gösterilecek kişiselleştirilmiş bir mesaj için hazırlanmıştır. Başlangıçta gizlidir ve JavaScript ile daha sonra aktif hale getirilebilir.



**Ana Sayfa Tanıtım Alanı (Hero Section)**

<section id="anasayfa" class="hero">

Sayfa, id="anasayfa" ve class="hero" şeklinde tanımlanan bir <section> etiketi ile başlar. Bu alan sitenin kullanıcıya ilk sunulan, dikkat çekici ve bilgilendirici tanıtım bölümüdür. CSS tarafında .hero sınıfı kullanılarak içerik yatayda ortalanmış, görseller ve metin esnek yapıda düzenlenmiştir.

<div class="hero-image left-image">

<img src="bir.jpg" alt="Sol Fotoğraf" />

</div>

Bu div, sayfanın sol tarafında bir görsel barındırır. img etiketi ile eklenen görsel, estetik katkı sağlamasının yanı sıra, kullanıcının dikkatini görsel içerikle destekleyerek artırmayı hedefler.

<div class="hero-content">

<h2>AnketSepeti ile 3 Adımda Anketinizi Oluşturun.</h2>

<p>...</p>

<a href="#anketler" class="btn">Anket Türlerine Göz At</a>

</div>

Ortadaki içerik alanı (hero-content) başlık, açıklama ve bir eylem çağrısı (Call-To-Action) butonu içerir. Buradaki <h2> başlığı kullanıcıya doğrudan uygulamanın ne işe yaradığını anlatır. Açıklayıcı paragraf ile sistemin nasıl işlediğine dair kısa bir bilgi verilir. Buton ise class="btn" şeklinde tanımlanmış ve sayfadaki “Anketler” bölümüne yönlendirme yapar. CSS ile bu buton altın sarısı renge sahip olup hover durumlarında rengi değişir.

<div class="hero-image right-image">

<img src="iki.jpg" alt="Sağ Fotoğraf" />

</div>

Son olarak sağ tarafta ikinci bir görsel yer alır. Bu şekilde simetrik bir yapı oluşturularak tasarım dengelenmiş olur.

metin, ekran görüntüsü, çevrimiçi reklamcılık, web sitesi içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

**“Anket Nasıl Oluşturulur?” Bölümü**

<section id="nasıloluşturulur?" class="how-to-create">

Bu bölüm, kullanıcıya anket oluşturma sürecini adım adım açıklayan bilgilendirici bir içerik alanıdır. id değeri sayesinde menüden buraya bağlantı kurulabilir. CSS tarafında .how-to-create sınıfı ile arka plan rengi açık sarı, yazı rengi ise koyu sarı olarak belirlenmiş; kullanıcıyı yormayan bir görsel bütünlük sağlanmıştır.

<h2 class="section-title">3 Adımda Anketinizi Oluşturun</h2>

<p class="section-subtitle"></p>

Başlık kısmı, kullanıcıyı bilgilendirmek üzere dikkat çekici font büyüklüğünde tanımlanmıştır. Alt başlık etiketi yer alsa da içeriği henüz yazılmamış durumdadır.

metin, ekran görüntüsü, tasarım içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.

**Adım 1 – Anket Amacı ve Hedefi**

<div class="step-box">

<div class="step-text">

<h3>1. Adım: Anketin Amacını ve Hedefini Tanımlayın</h3>

<p>...</p>

</div>

<div class="step-image">

<img src="bes.jpg" alt="Adım 1 görseli" />

</div>

</div>

Her bir adım bir .step-box sınıfı ile sarmalanmış ve CSS ile kart görünümüne kavuşturulmuştur. İlk adımda, anket sürecine başlamadan önce yapılması gereken hedef tanımı anlatılmaktadır. Açıklayıcı paragraf, kullanıcıyı yönlendirici nitelikte yazılmıştır. Alt kısımda yer alan görsel (bes.jpg), içeriği desteklemek için kullanılır.

**Adım 2 – Hedef Kitleyi Belirleme**

<h3>2. Adım: Hedef Katılımcı Profilini Belirleyin</h3>

<p>...</p>

<img src="uc.jpg" alt="Adım 2 görseli" />

İkinci adımda, anketin etkili sonuçlar vermesi için hedef katılımcıların belirlenmesinin önemi vurgulanmıştır. Kullanıcıya yaş, ilgi alanı, konum gibi demografik filtrelerin göz önünde bulundurulması gerektiği anlatılır. Görsel, anlatımı güçlendirmek için kartın altına eklenmiştir.

**Adım 3 – Anketi Paylaşmak**

<h3>3. Adım: Anketi Yayınlayın ve Paylaşım Kanallarını Belirleyin</h3>

<p>...</p>

<img src="dort.jpg" alt="Adım 3 görseli" />

Son adımda, anket tamamlandıktan sonra nasıl ve nerelerde paylaşılması gerektiği açıklanır. E-posta, sosyal medya ve web siteleri gibi paylaşım kanallarına değinilmiştir. Görsel yine içerikle paralel olacak şekilde yerleştirilmiştir.

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

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.

**“Neden AnketSepeti Kullanmalısınız?” Bölümü**

<section id="neden" class="why-choose">

Bu bölüm, kullanıcıya hizmetin güçlü yönlerini tanıtmak için hazırlanmıştır. id="neden" etiketi ile menü bağlantılarıyla eşleşme sağlanmış, class="why-choose" ise CSS üzerinden görsel biçimlendirme amacıyla atanmıştır.

<div class="container">

<h2 class="section-title">Neden AnketSepeti Kullanmalısınız?</h2>

<p class="section-subtitle">

AnketSepeti, her ihtiyaca uygun biçimde düzenleyebileceğiniz zengin anket şablonlarıyla size zaman kazandırır ve süreci kolaylaştırır.

</p>

</div>

Bu yapı içerisinde, bölüm başlığı <h2> etiketi ile belirtilmiş, CSS’te .section-title sınıfı ile font büyüklüğü ve rengi vurgulanmıştır. Altındaki <p> etiketi, kullanıcıya kısa ama öz bir açıklama sunar. Bu açıklama, kullanıcıyı ikna etmeye yönelik yazılmış bir metindir.

**“Özellikler” Bölümü**

<section id="ozellikler" class="features-section">

Bu bölümde, platformun anket oluşturma konusunda sunduğu teknik detaylar listelenir. id="ozellikler" menü bağlantıları için kullanılırken, features-section sınıfı bölümün arka planı, boşlukları ve yazı stili için CSS ile şekillendirilir.

<h2 class="section-title">Anketler Oluşturmak İçin Gereken Her Şey</h2>

<p class="section-subtitle">

AnketSepeti, ihtiyaçlarınıza göre anketleri tamamen özelleştirmenize olanak tanır...

</p>

Başlık ve açıklama kısmı, kullanıcıya platformun esnekliği ve gelişmiş araçları hakkında bilgi verir. Bu sayede kullanıcı platformun teknik gücüne dair bir fikir edinir.

**Özellik Açıklamaları ve Soru Tipleri Listesi**

<div class="feature-text">

<p>Kullanım amacınıza uygun profesyonel şablonlarla başlayabilir...</p>

Burada kullanıcıya platformun teknik kabiliyetleri sade bir dille anlatılmıştır. Hemen ardından bir <ul class="feature-list"> öğesi ile anketlerde kullanılabilecek soru türleri madde madde sıralanmıştır:

<ul class="feature-list">

<li>Tek seçenek</li>

<li>Çoktan seçmeli</li>

<li>Metin yanıtı</li>

<li>Görüntü seçimi</li>

<li>Yıldız puanı</li>

<li>Anlamsal diferansiyel</li>

<li>Net Promoter Skoru</li>

<li>Puanlama ölçeği</li>

<li>Sıralama (Ordering)</li>

<li>Matris - tekli seçim</li>

<li>Matris - çoklu seçim</li>

<li>Özel metin alanı</li>

</ul>

Her <li> etiketi, farklı bir anket sorusu türünü temsil eder. Bu yapı, kullanıcıya platformun ne kadar kapsamlı olduğunu görsel olarak sunar. CSS ile bu liste kutucuk haline getirilmiş, hover efektleriyle etkileşimli hale getirilmiştir.

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

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.**

Kullanıcılara AnketSepeti platformunda sunulan farklı anket kategorilerini tanıtarak onların ihtiyaçlarına uygun seçenekleri incelemelerini sağlar. Görsel içerikler ve açıklayıcı metinlerle desteklenen bu alan, hem bilgilendirici hem de yönlendirici bir işlev üstlenir.

**Yapı Başlangıcı**

<section id="anketler" class="survey-types">

Bu satır, “Anket Türleri” adlı bölümün başlangıcını belirtir. id="anketler" ifadesi sayesinde bu bölüm, üst menüdeki ilgili bağlantıyla entegre çalışabilir. class="survey-types" ise CSS dosyasında bu bölüme özel tanımlanan stil kurallarını uygular (örneğin: arka plan rengi, yazı stili, padding vs.).

<div class="container">

Bu div yapısı, içeriklerin sayfa ortasında hizalanmasını ve maksimum genişlik kontrolünü sağlar. Projede .container sınıfı sıkça kullanılmıştır ve düzenli hizalama için oldukça önemlidir.

**Başlık ve Açıklama**

<h2 class="section-title">Anket türleri</h2>

<p class="section-subtitle">LimeSurvey, müşterilerden, çalışanlardan... </p>

Başlık (h2) ve alt açıklama (p) etiketleri, kullanıcıya bu bölümün konusunu anlatır. .section-title ve .section-subtitle sınıfları sayesinde bu metinler CSS ile dikkat çekici hâle getirilmiştir. Alt başlık, platformun birçok sektöre ve hedefe yönelik geniş anket seçenekleri sunduğunu belirtir.

**Anket Kartları (survey-card)**

Bu bölümde dört adet “kart” (card) yapısı yer alır. Her biri bir anket türünü tanıtır. Kartlar, aşağıdaki gibi div class="survey-card" yapısı içinde düzenlenmiştir. Aşağıda örnek bir kartın açıklaması verilmiştir:

<div class="survey-card">

<img src="is-anketi.png" alt="İş Anketi">

<div class="card-icon"></div>

<h3>İş Anketleri</h3>

<p>İş operasyonları, çalışan memnuniyeti...</p>

<a href="is-anketi.html" class="btn-detail">Anketi İncele</a>

</div>

**Görsel**

<img src="..."> etiketi, anket türünü görsel olarak temsil eden bir resim içerir. Bu görsel kullanıcı ilgisini çekmek amacıyla eklenmiştir.

**Başlık ve Açıklama**

<h3> etiketiyle anketin türü belirtilir. Hemen altında <p> etiketi, bu anketin ne işe yaradığını kısa ve anlaşılır şekilde açıklar. Örneğin “İş Anketleri” kartında, çalışan memnuniyeti ve işyeri kültürü üzerine bilgi verilirken; “Müşteri Anketleri” kartında, müşteri memnuniyeti ve hizmet kalitesi ön plana çıkarılmıştır.

**Yönlendirme Butonu**

Her kartın sonunda yer alan:

<a href="is-anketi.html" class="btn-detail">Anketi İncele</a>

bağlantısı sayesinde kullanıcı detaylı bilgi almak için ilgili anket sayfasına yönlendirilebilir. CSS ile btn-detail sınıfı altın sarısı arka plan, yuvarlatılmış köşeler ve hover animasyonlarıyla zenginleştirilmiştir.

**Grid Yapısı (survey-grid)**

<div class="survey-grid">

...

</div>

Tüm kartlar, survey-grid adlı bir grid yapısı içinde yer alır. CSS tarafında bu yapı grid-template-columns: repeat(auto-fit, minmax(260px, 1fr)); şeklinde tanımlanarak ekran boyutuna göre otomatik olarak satır/sütun dizilimi sağlar. Bu sayede sayfa responsive olur ve mobil cihazlarda kartlar alt alta sıralanır.

**metin, ekran görüntüsü, web sayfası, web sitesi içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.**

**SSS (Sıkça Sorulan Sorular) Bölümü**

<section id="sss"> etiketiyle tanımlanan bu alanda, kullanıcıların anketlerle ilgili merak ettiği sorular accordion (aç/kapa) yapısıyla sunulmuştur. JavaScript ile kullanıcı bir başlığa tıkladığında ilgili açıklama görünür, diğerleri kapanır. Bu yapı hem yer tasarrufu sağlar hem de kullanıcı deneyimini artırır.

**Global Markalar Bölümü**

<section class="trusted-brands"> kısmında, Samsung, Meta, Adobe gibi büyük markaların logoları ve tanıtımları yer alır. Bu bölüm güven artırıcı bir referans alanı olarak tasarlanmıştır. Hover efektleri ile logolar renklendirilip vurgulanır.



**Başarı Hikayeleri**

<section class="success-stories"> içerisinde, kullanıcı yorumları kart yapısıyla sunulmuştur. Her biri bir şirket temsilcisine ait olumlu geri bildirim içerir. Bu alan, sosyal kanıt oluşturarak güveni pekiştirir.

**Geri Bildirim Formu**

<section class="feedback-form-section"> kısmında yer alan form, kullanıcıların deneyimlerini ve yorumlarını iletmesini sağlar. Ad, e-posta, pozisyon, yorum ve puan alanları içerir. Form verisi Formspree servisi ile gönderilir ve JavaScript ile doğrulanır.

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

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.

**Destek Bileşeni**

Sayfanın sağ alt köşesinde sabitlenen “Destek” butonuna tıklanınca açılan mini form ile kullanıcılar teknik destek talebi oluşturabilir. Aç/kapa işlemi ve form gönderimi JavaScript ile yönetilir.

**JavaScript İşlevleri**

* **Accordion Kontrolü:** SSS başlıklarına tıklanınca ilgili içerik açılır.
* **Geri Bildirim Formu:** Boş alan kontrolü ve Formspree’ye veri gönderimi yapılır.
* **Destek Formu:** Görünürlük kontrolü ve sunucuya veri aktarımı yapılır.
* **Kullanıcı Girişi:** localStorage ile oturum kontrolü yapılır, giriş yapan kullanıcının adı görüntülenir, çıkış yapılabilir.

metin, ekran görüntüsü, tasarım içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.

**BIRINCI.CSS**

Stil dosyası (birinci.css), sayfa düzeninden etkileşimli formlara kadar tüm görsel sunumu düzenleyen kapsamlı bir CSS yapısı sunmaktadır.

**Tasarımın Genel Özellikleri**

**Renk Paleti**

Tasarımda ağırlıklı olarak sarı, altın ve krem tonları tercih edilmiştir. Bu sayede sıcak, samimi ve güven verici bir kullanıcı arayüzü hedeflenmiştir.

background-color: #fffef2; /\* açık krem zemin \*/

color: #5a5a00; /\* koyu sarı metin \*/

background-color: #d4af37; /\* altın sarısı buton \*/

**Yapısal Bileşenler ve Kod Açıklamaları**

**1. Genel Sıfırlama ve Tipografi**

\* {

margin: 0;

padding: 0;

box-sizing: border-box;

}

body {

font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;

background-color: #fffef2;

color: #333;

}

* Tüm elementlerin dış boşlukları sıfırlanmış ve box-sizing: border-box kullanılarak kutu boyutlaması kontrol altına alınmıştır.
* Tüm metinlerde modern ve okunabilir bir yazı tipi ailesi seçilmiştir.

**2. Navigasyon Menüsü (.navbar)**

.navbar {

display: flex;

justify-content: space-between;

align-items: center;

background-color: #fdfd96;

position: sticky;

top: 0;

}

* Menü barı, flex yapısı ile yatayda düzenlenmiş ve sayfanın en üstünde sabitlenmiştir (position: sticky).
* Mobil cihazlar için aşağıdaki medya sorgusu kullanılmıştır:

@media (max-width: 768px) {

.navbar {

flex-direction: column;

}

}

**3. Giriş/Üye Ol Butonları**

.signup-btn {

background-color: #d4af37;

color: white;

border: none;

}

.signup-btn:hover {

background-color: #e1c84c;

}

* Kullanıcı etkileşiminde görsel geri bildirim (hover) sağlamak için geçiş efektleri kullanılmıştır.

**4. Hero Bölümü**

.hero {

display: flex;

justify-content: center;

padding: 60px 20px;

}

.hero-content h2 {

font-size: 2.4rem;

font-weight: 700;

}

* Sayfanın tanıtım kısmı olan hero bölümünde başlık, açıklama ve bir görsel yer alır.
* Flexbox ile ortalanmış ve responsive olması sağlanmıştır.

**5. Adım Adım Rehber (.how-to-create)**

css

KopyalaDüzenle

.step-box {

background-color: #f9f7d9;

border-radius: 16px;

transition: transform 0.3s ease;

}

.step-box:hover {

transform: translateY(-5px);

}

* Kullanıcıya süreçleri anlatmak için kutular kullanılmış ve hover efektleri ile animasyon sağlanmıştır.

**6. Form Alanları (.feedback-form, .auth-form)**

input[type="text"], textarea, select {

border: 2px solid #f6e05e;

border-radius: 12px;

transition: border-color 0.3s ease;

}

input:focus {

border-color: #d4af37;

background-color: #fffbe6;

}

* Form alanlarında odaklandığında (focus) kullanıcıya görsel geri bildirim verilmektedir.
* Responsive yapıda form elemanları mobil cihazlara uyarlanmıştır.

**7. Mobil Uyumlu Tasarım**

@media (max-width: 480px) {

.nav-links {

font-size: 0.95rem;

}

.auth-box {

padding: 30px 20px;

}

}

* Tüm bölümler, küçük ekranlara göre yeniden düzenlenmiş ve tipografi ile boşluklar küçültülmüştür.

**Ekstra Özellikler**

* **Accordion (SSS) yapısı:**  
  accordion-header sınıfı ile açılır kapanır bilgi kutuları.
* **Destek Formu:**  
  Sayfanın sağ altına sabitlenmiş bir butonla açılan mini destek formu (widget).
* **Marka Güven Alanı:**  
  trusted-brands alanı ile logolar üzerinden sosyal kanıt oluşturulmuş.
* **Başarı Hikayeleri:**  
  Kullanıcı deneyimlerini tanıtmak amacıyla özel kartlar şeklinde düzenlenmiştir.

Bu CSS dosyası; hem modern hem de kullanıcı dostu bir deneyim sunmak için baştan sona dikkatle tasarlanmış, mobil uyumlu ve görsel açıdan etkileyici bir yapıya sahiptir.  
Renk seçimi, yazı tipi uyumu, kutu gölgeleri, geçiş animasyonları ve responsive (duyarlı) yapı ile birlikte, güncel web tasarım standartlarına uygundur.

**GIRIS.HTML**

**1. Genel Bilgiler**

**Sayfa Amacı:**  
Kullanıcının e-posta ve şifre ile giriş yapmasını sağlar. Giriş bilgileri daha önce **localStorage**'a kayıt edilen kullanıcılarla karşılaştırılır. Giriş başarılı olursa kullanıcı "index.html" sayfasına yönlendirilir.

**Kullanılan Teknolojiler:**

* HTML5
* CSS3 (Responsive tasarım içerir)
* Vanilla JavaScript (localStorage ile kullanıcı doğrulama)

**2. Tasarım ve Kullanıcı Arayüzü**

**2.1. Görsel Tasarım Özellikleri**

* Arka plan: linear-gradient ile yumuşak sarı geçişli tasarım.
* Konteyner (auth-container): Gölgelendirme (box-shadow), kenar yuvarlama (border-radius) ve iç boşluk (padding) ile modern görünüm sağlanmış.
* Renk teması: **Altın sarısı (#d4af37)** ve açık pastel tonları. Uyumlu bir renk paleti tercih edilmiş.
* Form öğeleri büyük, okunaklı ve dokunmatik cihazlar için uygun boyutlarda.

**2.2. Responsive Tasarım**

* @media (max-width: 480px) kuralı ile mobil uyum sağlanmış.
* Giriş alanları ve butonlar küçültülerek küçük ekranlara adapte edilmiş.

**3. JavaScript Mantığı ve localStorage Kullanımı**

**3.1. Giriş Akışı**

1. Form gönderimi yakalanır (submit event listener).
2. E-posta ve şifre boşsa uyarı verilir.
3. localStorage içindeki tüm anahtarlar döngü ile kontrol edilir.
4. Anahtar e-posta ile eşleşirse karşılık gelen veri alınır.
5. Şifre karşılaştırması yapılır.
6. Giriş başarılıysa, currentUser olarak localStorage'a kaydedilir ve yönlendirme yapılır.

**3.2. Kod Kalitesi ve İyileştirme Notları**

| **Alan** | **Durum** | **Açıklama** |
| --- | --- | --- |
| Form Doğrulama | Yeterli | Boş alanlar kontrol ediliyor. |
| Şifre Güvenliği | Eksik | Şifreler açık metin olarak saklanıyor. localStorage kullanımı ile birlikte güvenlik zafiyeti oluşturur. |
| localStorage tarama | Başarılı | Anahtar bazlı arama ile e-posta eşleşmesi yapılmış. |
| JSON Hata Kontrolü | İyi | try...catch ile JSON parse hatası yakalanıyor. |
| Kullanıcı Yönlendirme | Var | Giriş sonrası index.html'e yönlendirme mevcut. |

**4. Depolama Mantığı (localStorage)**

**4.1. Kayıtlı Kullanıcı Formatı (örnek):**

{ "name": "Ahmet Yılmaz",

"email": "ahmet@gmail.com",

"password": "123456" }

**4.2. Depolama Stratejisi:**

* Kullanıcılar, e-posta adresleri anahtar olarak kullanılarak saklanıyor.
* Giriş yapılınca, başarılı kullanıcı verisi currentUser anahtarı ile ayrı olarak da kaydediliyor.

localStorage.setItem('currentUser', JSON.stringify(user));

**metin, ekran görüntüsü, yazı tipi, sayı, numara içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.**

**UYE-OL.HTML**

Kullanıcıların web tabanlı bir arayüz üzerinden yeni bir hesap oluşturmasını sağlayan, **kayıt (üye olma)** sürecini destekleyen bir sayfa tasarımıdır. Kullanıcıdan alınan veriler (isim, e-posta, şifre) doğrulanarak tarayıcıda güvenli bir şekilde depolanır. Sistem sunucuya bağlı olmadan çalışır, yani tamamen **istemci taraflı (client-side)** olarak işlem yapar.

Bu sistemin temel hedefleri:

* Modern ve kullanıcı dostu bir arayüz oluşturmak
* Form doğrulaması ve güvenliği sağlamak
* localStorage ile verilerin kalıcı saklanmasını sağlamak
* Kayıt sonrası yönlendirme yapmak

**Kullanılan Teknolojiler ve Kütüphaneler**

| **Teknoloji** | **Rolü** |
| --- | --- |
| **HTML5** | Sayfa yapısı ve form elementleri |
| **CSS3** | Responsive tasarım ve görsel stil |
| **JavaScript (Vanilla)** | Doğrulama ve veri kaydı işlemleri |
| **Web Storage API** | localStorage ile tarayıcıda veri saklama |

**HTML: Sayfa Yapısı**

HTML dosyası, standart yapıya uygun şekilde yapılandırılmıştır:

* <head> kısmında favicon, charset ve mobil uyumluluk (viewport) ayarlanmıştır.
* <form> içinde 4 adet input alanı yer alır: Ad Soyad, E-posta, Şifre ve Şifre Tekrar.
* Tüm inputlar required özelliğiyle temel doğrulamaya sahiptir.
* Form altındaki bağlantı, kayıtlı kullanıcıları giriş sayfasına yönlendirmek için kullanılır.

**JavaScript: Doğrulama ve Kayıt İşlemleri**

Kod, registerForm adlı form üzerine bir submit dinleyicisi yerleştirerek çalışır:

**✔ Form Validasyonu**

1. **Boş Alan Kontrolü:** Tüm alanların doldurulması istenir.
2. **Şifre Eşleşmesi:** Şifre ile tekrar şifresi uyuşmalı.
3. **E-posta Eşsizliği:** E-posta daha önce kayıtlı mı diye kontrol edilir.
4. **localStorage’a kayıt:** Bilgiler küçük harfli e-posta anahtarıyla JSON formatında kaydedilir.
5. **Yönlendirme:** Başarılı kayıt sonrası giris.html sayfasına geçiş yapılır.

metin, ekran görüntüsü, yazı tipi, sayı, numara içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir. metin, ekran görüntüsü, yazı tipi, sayı, numara içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.

**PHP KULLANICI KAYIT VE GİRİŞ SİSTEMİ**

**Genel Bakış**

Bu sistem, PHP kullanılarak geliştirilmiş basit bir kullanıcı kimlik doğrulama (authentication) altyapısı sunmaktadır. 3 temel dosyadan oluşmaktadır:

1. db\_connect.php — Veritabanı bağlantısı
2. login.php — Kullanıcı giriş işlemleri
3. register.php — Kullanıcı kayıt işlemleri

**1. db\_connect.php — Veritabanı Bağlantı Dosyası**

**Amaç:**

Bu dosya, mysqli sınıfı kullanarak bir MySQL veritabanına bağlantı oluşturur. Sistemdeki diğer dosyalar bu bağlantıyı kullanarak sorgularını yapar.

**Kod Açıklaması:**

$host = 'localhost';

$user = 'root';

$password = '';

$dbname = 'survey\_db';

$conn = new mysqli($host, $user, $password, $dbname);

* Yerel geliştirme ortamları için uygundur (XAMPP vb.).
* survey\_db adlı veritabanına bağlanılır.

if ($conn->connect\_error) {

die(json\_encode(['message' => 'Veritabanı bağlantı hatası: ' . $conn->connect\_error]));

}

* Bağlantı hatası varsa, JSON formatında bir mesajla program durdurulur.

$conn->set\_charset('utf8mb4');

* Unicode karakter desteği sağlanır. Doğru bir tercih.

**2. login.php — Kullanıcı Girisi Sistemi**

**Amaç:**

Kullanıcının giriş yapmasını sağlar. Önceden kayıt olmuş kullanıcı bilgileri (e-posta ve bilgileri (e-posta ve \u015ifre) ile doğrulama yapar.

**Özellikler:**

* JSON tabanlı API olarak POST istekleri kabul eder.
* İstemciden gelen JSON verisi çözülür, içinden email ve password değerleri alınır.

**Kod Açıklamaları:**

$data = json\_decode(file\_get\_contents('php://input'), true);

$email = $data['email'] ?? '';

$password = $data['password'] ?? '';

* Boş değerler kontrol edilir, yoksa hata mesajı verilir.

$stmt = $conn->prepare('SELECT id, name, email, password FROM users WHERE email = ?');

* Kullanıcı sorgusu e-posta ile yapılır.

if ($user['password'] !== $password) {

echo json\_encode(['message' => 'Şifre yanlış.']);

* Şifre doğrudan karşılaştırılıyor. Bu **güvenlik zafiyeti** oluşturur.

http\_response\_code(200);

* Başarılı girişte HTTP 200 ve kullanıcı bilgileri JSON ile dönülür.

**3. register.php — Kullanıcı Kayıt Sistemi**

**Amaç:**

Yeni kullanıcı kaydı oluşturmak için kullanılır. name, email ve password alanlarını alır.

**Kod Açıklamaları:**

$data = json\_decode(file\_get\_contents('php://input'), true);

$name = $data['name'] ?? '';

$email = $data['email'] ?? '';

$password = $data['password'] ?? '';

* JSON verisi çözülür. Eksik alanlar varsa işlem iptal edilir.

$stmt = $conn->prepare('SELECT \* FROM users WHERE email = ?');

* Daha önce bu e-posta ile kayıt yapılmış mı kontrol edilir.

$stmt = $conn->prepare('INSERT INTO users (name, email, password) VALUES (?, ?, ?)');

$stmt->bind\_param('sss', $name, $email, $password);

* Yeni kayıt eklenir. Ancak şifre yine **düz metin** olarak kaydedilir.

**Güvenlik Önerileri**

1. **Şifre Hashleme**
   * Kıyaslama yerine:
   * password\_verify($password, $user['password']);
   * Kayıt olurken:
   * $hashed = password\_hash($password, PASSWORD\_DEFAULT);
2. **Girdi Doğrulama**
   * E-posta: filter\_var($email, FILTER\_VALIDATE\_EMAIL)
   * Şifre uzunluğu, özel karakter kontrolü
3. **Rate Limiting ve CSRF Koruması**
   * IP bazlı limit
   * Token kullanımı
4. **JWT ya da Session Tabanlı Oturum**
   * Giriş başarılıysa JWT token üretilip istemciye dönülebilir.

**SCRIPT.JS**

HTML sayfalarıyla birlikte çalışır ve kullanıcı etkileşimlerine göre sayfa üzerindeki görsel öğeleri yönetir. Aynı zamanda backend tarafındaki PHP API dosyalarına da istek göndererek veri alışverişini sağlar.

**1. DOMContentLoaded Olayı**

document.addEventListener('DOMContentLoaded', () => {

Sayfa tamamen yüklendiğinde JavaScript kodlarının çalışmasını sağlar. Böylece sayfadaki HTML elementleri tam olarak yüklenmeden JavaScript erişimi yapılmaz.

**2. HTML Elemanlarının Tanımlanması**

const loginForm = document.getElementById('loginForm');

const registerForm = document.getElementById('registerForm');

const showRegister = document.getElementById('showRegister');

const showLogin = document.getElementById('showLogin');

HTML'deki form ve buton elementleri tanımlanır. Böylece kullanıcı giriş/kayıt formları üzerinde işlem yapılabilir.

**3. Formlar Arası Geçiş**

showRegister.addEventListener('click', () => { ... });

showLogin.addEventListener('click', () => { ... });

Kullanıcı "Kayıt Ol" veya "Giriş Yap" butonlarına tıkladığında form görünümleri arasında geçiş yapılmasını sağlar.

**4. Kayıt İşlemi (Register)**

registerForm.addEventListener('submit', async function(event) { ... });

Kullanıcı kayıt formunu gönderdiğinde aşağıdaki işlemler yapılır:

Form boş mu kontrol edilir.

Şifre ve şifre tekrarı uyuşuyor mu kontrol edilir.

register.php API dosyasına kullanıcı adı, e-posta ve şifre bilgisi gönderilir.

Başarılıysa form temizlenir ve giriş formuna geçilir.

**5. Giriş İşlemi (Login)**

loginForm.addEventListener('submit', async function(event) { ... });

Kullanıcı giriş formunu gönderdiğinde:

Alanlar dolu mu kontrol edilir.

login.php API'sine e-posta ve şifre bilgileri gönderilir.

Giriş başarılıysa kullanıcı bilgileri localStorage içine kaydedilir.

Kullanıcı ana sayfaya (index.html) yönlendirilir.

**6. Oturum Kontrolü (Session)**

const currentUser = localStorage.getItem('currentUser');

Eğer kullanıcı daha önce giriş yaptıysa ve bilgiler tarayıcıda saklandıysa:

current\_user.php API’sine tekrar istek gönderilir.

Kullanıcının adı arayüzde gösterilir.

"Çıkış Yap" butonu aktif hale gelir ve tıklandığında localStorage temizlenir, sayfa yenilenir.

**Güvenlik ve UX Detayları**

CORS (Cross-Origin Resource Sharing) uyumludur, API bağlantıları fetch ile yapılır.

Form Validasyonu yapılmıştır: Boş alan ve eşleşmeyen şifre kontrolü.

Kullanıcı deneyimi (UX) odaklı: Başarılı işlemlerde formlar gizlenir/gösterilir, kullanıcı adı selamlaması yapılır.

Yerel depolama (localStorage) kullanımıyla oturum bilgileri geçici olarak saklanır.

**Bağlantılı API Dosyaları**

Bu dosya aşağıdaki API uç noktalarıyla etkileşim içindedir:

register.php – Yeni kullanıcı kaydı için POST isteği gönderilir.

login.php – Giriş için kullanıcı doğrulaması yapılır.

current\_user.php – Oturum açık olan kullanıcının bilgilerini döner.

**MUSTERİ-ANKETİ.HTML**

Kullanıcı geri bildirimi toplamak amacıyla hazırlanmış bir **müşteri anketi formudur**. Kullanıcıların bir hizmet ya da ürünle ilgili deneyimlerini, görüşlerini ve önerilerini kolayca paylaşmalarını sağlar. Form; kullanıcıya adını yazma, hizmet deneyimini anlatma, ürün/hizmet hakkındaki düşüncelerini aktarma, iyileştirilmesi gereken alanları belirtme ve ek görüşlerini iletme gibi çeşitli metin alanları sunar. Kullanıcı adı zorunlu olmayıp, diğer alanlar da tamamen özgür şekilde doldurulabilir. Bu, kullanıcıların daha rahat ve içten geri bildirim vermelerini teşvik eder.

Formun görsel tasarımı oldukça sade ve kullanıcı dostudur. CSS ile hazırlanmış şık bir kutu içerisinde yer alan form, açık renk arka plan ve yumuşak kenarlı butonlarla desteklenmiştir. Böylece hem masaüstü hem de mobil cihazlarda okunabilirlik ve kullanılabilirlik sağlanmıştır. Gölgelendirme ve hover efektleri sayesinde kullanıcı arayüzü daha profesyonel bir görünüm kazanmıştır.

Sayfa yüklendiğinde form, JavaScript ile dinamik işlevler kazanır. Öncelikle, kullanıcı formu doldurup “Gönder” butonuna bastığında sayfa yenilenmeden bir teşekkür mesajı görüntülenir ve form alanları temizlenir. Bu sayede kullanıcı etkileşimi kesintiye uğramadan devam eder. preventDefault() fonksiyonu ile formun varsayılan gönderme davranışı engellenir.

* <head> kısmında responsive ve karakter kodlaması ayarlanmış, ayrıca **şık, altın tonlarında** bir CSS stili var.
* <body> kısmında:
  + Başlık (<h1>)
  + Form (<form>) var.
  + Formun içinde:
    - **Adınız** için animasyonlu, modern bir input alanı
    - 3 sabit soru bloğu (.question-group)
      * Her soru için soru metni
      * Soru tipi seçimi için açılır menü (select)
      * Cevap alanı (açık uçluysa textarea, çoktan seçmeli ise dinamik olarak şık ekleme alanı)
    - Butonlar: Gönder, JSON indir, Link oluştur
  + Form altında gönderme sonrası mesaj ve link gösterim alanı var.

**3. CSS Tasarımı**

* Ana renkler: **Krem-bej-beyaz zemin ve altın-sarı tonları.** (#d4af37, #b28100 gibi renkler)
* Modern, gölgeli, yuvarlak kenarlı kutular.
* Giriş alanları, butonlar, select menüler şık ve kullanımı kolay.
* "Adınız" alanında "floating label" (üstte kayan etiket) efekti var.
  + Kullanıcı yazı yazdığında veya focus olduğunda label küçülüp yukarı kayıyor.
* Butonlar sarı tonlarında, hover’da daha açık renk oluyor.
* Çoktan seçmeli şıklar için ayrı alan, şıklar eklenip çıkarılabiliyor.

**4. JavaScript İşlevleri**

**a. Soru Tipine Göre Cevap Alanını Güncelleme**

* Sayfa açılırken ve soru tipi değiştiğinde cevap alanı değiştirilir:
  + text seçilirse sadece bir textarea gösterilir.
  + multiple seçilirse:
    - Birden fazla şık eklenebilen alan açılır.
    - Başlangıçta 1 boş şık var.
    - “Şık Ekle” butonuyla yeni şık eklenir.
    - Her şıkkın yanında “×” butonu ile şık silinebilir.

**b. Form Verisini Alma (getFormData())**

* Formdaki “Adınız” ve tüm soruların bilgileri toplanır.
* Her soru için:
  + Soru metni
  + Soru tipi (text veya multiple)
  + Cevaplar:
    - text için textarea içeriği,
    - multiple için eklenen şıklar dizi olarak alınır.
* Veriler JSON formatına uygun olarak obje halinde döner.

**c. Form Gönderme**

* Submit olduğunda sayfanın yenilenmesi engellenir.
* Basit bir teşekkür mesajı gösterilir.
* Form sıfırlanır ve cevap alanları varsayılan açıl uçlu hale döner.

**d. JSON İndir**

* Anket verisi JSON olarak indirilir.
* Böylece kullanıcı verisini bilgisayarına kaydedebilir.

**e. Paylaşılabilir Link Oluşturma**

* Anket verisi JSON’a çevrilir.
* Bu JSON encode edilip URL’ye parametre olarak eklenir.
* Oluşan link kullanıcıya gösterilir.
* Bu link başkalarına gönderilebilir.

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

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

**KURUMSAL-ANKET.HTML**

HTML, CSS ve JavaScript kullanılarak geliştirilmiş, çalışanlardan geri bildirim toplamak amacıyla tasarlanmış kurumsal bir anket uygulamasıdır. Anket formunda kullanıcı adı haricinde; bağlılık, iletişim, yöneticiler, motivasyon, zorluklar, gelişim önerileri, değerler, eğitim ve denge konularını kapsayan 10 adet metin giriş alanı bulunmaktadır.

**Tasarım (CSS):**  
Form sade, mobil uyumlu ve merkezlenmiş yapıdadır. Açık renkler, yumuşak köşeler, gölgeler ve okunabilir yazı stiliyle kullanıcı dostu bir arayüz sunulmuştur. Hover efektleri ile etkileşim artırılmıştır.

**İşlevsellik (JavaScript):**  
Form gönderildiğinde sayfa yenilenmeden veri toplanır, “Teşekkürler!” mesajı gösterilir ve form sıfırlanır. getFormData() fonksiyonu ile veriler alınır. Ayrıca iki gelişmiş özellik entegre edilmiştir:

1. **JSON Olarak İndir:** Form verileri JSON formatında dışa aktarılabilir.
2. **Paylaşılabilir Link:** Veriler URL’ye kodlanarak paylaşılabilir hâle getirilir ve başka kullanıcılar formu otomatik dolu şekilde açabilir.

**HTML Yapısı:**

kurumsal geri bildirim amacıyla hazırlanmış dinamik ve kullanıcı etkileşimli bir anket formu sunar. Sayfanın başında “Kurumsal Geri Bildirim Anketi” başlığı yer alır ve ana içerik bölümünde, JavaScript kullanılarak önceden tanımlanmış 10 temel soru dinamik olarak oluşturulur. Her soru bir kutu içerisinde yer alır ve kullanıcı, o soruya açık uçlu yanıt (serbest metin alanı) veya çoktan seçmeli yanıt (birden fazla şık eklenip çıkarılabilen seçenekler) şeklinde cevap verebilir. Çoktan seçmeli seçeneği seçildiğinde, “Şık Ekle” butonuyla kullanıcı istediği kadar seçenek ekleyebilir, her şık yanında bulunan “Sil” butonuyla da istenmeyen seçenekleri kaldırabilir.

Formun alt kısmında “Gönder” butonu bulunur; kullanıcı bu butona bastığında JavaScript ile formun içeriği işlenir, ancak sayfa yenilenmez. Gönderim sonrası kullanıcıya sade bir “Gönderildi” uyarısı gösterilir. Ayrıca, formun hemen altında gizli olarak bulunan “Paylaş” butonu etkinleştirildiğinde, formdaki mevcut veriler JSON formatında kodlanarak URL’ye gömülür ve paylaşılabilir bir bağlantı oluşturulur. Bu link, sayfa altındaki özel bir alan içinde görüntülenir ve kullanıcı tek tıklamayla kolayca linki seçip kopyalayabilir. Link oluşturma işlemi sayfayı yenilemeden gerçekleşir. Bunun yanında, kullanıcı “JSON İndir” butonuna basarak anket verilerini JSON dosyası olarak indirebilir.

Sayfa, kullanıcı deneyimini artırmak için sade, profesyonel ve uyumlu bir renk paleti (sarı tonları ve açık krem arka plan) kullanır. Form alanları, metin kutuları, butonlar ve şık ekleme alanları modern ve erişilebilir şekilde tasarlanmıştır. JavaScript, form dinamikliği, veri işleme, URL’den veri alma (sayfa linkinde ankete ait JSON verisi varsa form otomatik doldurulur) ve veri indirme işlemlerini yönetir.

metin, ekran görüntüsü, tasarım içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

**KISISEL-ANKET.HTML**

**Anket Yapısı Genel Bakış**

* Sayfa, kullanıcıların dinamik şekilde **yeni sorular ekleyebileceği**, **çoktan seçmeli ya da açık uçlu** soru tipleri arasında seçim yapabileceği bir form içeriyor.
* Her soru kutusu içinde:
  + Soru metni için bir textarea,
  + Soru tipi için dropdown menü (Açık uçlu / Çoktan seçmeli),
  + Seçime göre açılan seçenek kutuları veya cevap alanı,
  + "Kaldır" butonu var.

**Soru Tipi Seçimi – Yeni Özellik**

**Açık Uçlu Soru Seçilince:**

* Katılımcının cevap yazabileceği örnek bir **textarea** alanı otomatik olarak görünür hale gelir.
* Bu kutu, yalnızca görsel olarak katılımcı cevabının nerede yazılacağını göstermek için yerleştirilmiştir (disabled olarak yani yazılamaz).
* İstenirse bu alan aktif hale getirilip gerçek cevap alınabilir.

**Çoktan Seçmeli Soru Seçilince:**

* “Şık Ekle” butonuyla çoktan seçmeli seçenekler eklenebilir.
* Her bir şık için ayrı bir input alanı gelir.
* "Sil" butonuyla şıklar tek tek kaldırılabilir.

**JavaScript ile Dinamik Özellikler**

**a. Yeni Soru Ekleme**

* "Soru Ekle" butonuna tıkladığında yeni bir question-group HTML bloğu oluşturuluyor.
* Bu blok içinde:
  + Soru metni (textarea),
  + Tip seçici (select),
  + Şıklar için alan (çoktan seçmeli seçilirse görünür),
  + Cevap alanı (açık uçlu seçilirse görünür) bulunuyor.

**b. toggleOptions Fonksiyonu**

function toggleOptions(select, index)

* Bu fonksiyon, seçilen soru tipine göre ilgili alanları (options-container, answer-area) **gösterir veya gizler.**

**4. JSON ve Link Özellikleri**

**a. downloadJSON()**

* Formda girilen tüm verileri bir JSON dosyasına dönüştürür ve indirmeni sağlar.

**b. generateLink()**

* Formdaki verileri URL'ye kodlar ve paylaşılabilir bir bağlantı üretir.

**c. window.onload()**

* Sayfa yüklenirken URL'de veri varsa (önceden oluşturulmuş linkten gelinmişse), bu verileri çözümler ve soruları sayfaya tekrar yükler.

**metin, ekran görüntüsü, yazı tipi, sayı, numara içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.**

**IS-ANKETI.HTML**

Bu sayfa, **"İş Anketi"** adıyla oluşturulmuş sade, işlevsel ve kullanıcı dostu bir geri bildirim formudur. HTML, CSS ve JavaScript kullanılarak hazırlanmış ve özellikle şirket çalışanlarının iş ortamları hakkında değerlendirme yapabilmesi için tasarlanmıştır.

**1. HTML Yapısı (İçerik ve Form)**

Form, çalışanlardan iş deneyimlerine yönelik kapsamlı geri bildirim almayı hedefler:

**Öne Çıkan Sorular:**

* İş memnuniyeti
* Karşılaşılan zorluklar
* Şirket kültürü
* Yönetim değerlendirmesi
* Stratejik planlar
* Kariyer gelişimi fırsatları

Her soru bir <label> ile tanımlanır ve altına bir <textarea> yerleştirilmiştir. Bu, kullanıcıların serbest metinle düşüncelerini yazmasını sağlar.

**2. CSS Stili (Görsel Tasarım)**

* **Renkler:** Açık zemin (#ffffff), gri kutular (#f9f9f9) ve sarı vurgular (#f1c40f) kullanılmış.
* **Modern görünüm:** Gölge (box-shadow), kenar yuvarlatma (border-radius) ve hover efektleri sayesinde profesyonel bir stil sunulmuş.
* **Responsive tasarım:** flex-wrap kullanımı mobil uyumluluğa katkı sağlar.

**Örnek:**

button {

background-color: #f1c40f;

border-radius: 10px;

transition: background-color 0.3s;

}

**3. JavaScript (İşlevsellik ve Dinamizm)**

**Ana Fonksiyonlar:**

**Form Verisi Toplama:**

function getFormData() {

return {

name: document.getElementById("name").value,

… }; }

**Form Gönderimi:**

* Sayfa yenilenmeden form verisi toplanır ve kullanıcıya teşekkür mesajı gösterilir.
* Form temizlenir (form.reset()).

**JSON Olarak İndirme:**

const blob = new Blob([JSON.stringify(data, null, 2)], {

type: "application/json",

});

Kullanıcı, anket verilerini .json dosyası olarak indirebilir.

**Link Oluşturma:**  
Kullanıcının doldurduğu veriler URL’ye kodlanır. Böylece link üzerinden geri çağrılabilir.

**Sayfa Açıldığında Veriyi Geri Yükleme:**

window.onload = function () {

const params = new URLSearchParams(window.location.search);

... }

Eğer URL'de veri varsa, form otomatik olarak bu bilgilerle doldurulur. Bu, **paylaşılabilir anket yanıtları** oluşturmak için faydalıdır.

metin, ekran görüntüsü, tasarım içeren bir resim

Yapay zeka tarafından oluşturulmuş içerik yanlış olabilir.

**KULLANILAN TEKNOLOJİLER**

Bu web projesi, modern ön yüz geliştirme için yaygın olarak tercih edilen temel teknolojilere dayanmaktadır:

* **HTML5 (HyperText Markup Language):** Uygulamanın iskelet yapısı HTML5 ile oluşturulmuştur. Sayfaların yapısı (örneğin: anasayfa, kayıt formu, sepet, yönetici paneli) bu dilin semantik öğeleri kullanılarak inşa edilmiştir.
* **CSS3 (Cascading Style Sheets):** Arayüzün görsel sunumu CSS3 ile şekillendirilmiştir. Hem genel stil dosyası (style.css) hem de bazı sayfalarda yer alan dahili <style> blokları sayesinde renkler, boyutlar, geçiş efektleri ve yerleşim düzeni oluşturulmuştur. Flexbox ve CSS Grid sistemleriyle responsive (mobil uyumlu) bir tasarım hedeflenmiştir.
* **JavaScript (ES6+):** Dinamik etkileşimler JavaScript ile sağlanmıştır. Örneğin: formların doğrulanması, kullanıcı giriş bilgileriyle çalışılması, sepet işlemleri ve localStorage kullanımı gibi işlemler bu dil aracılığıyla gerçekleştirilmiştir. Modern ES6 sözdizimi tercih edilmiştir.

**GITHUB DESKTOP İLE PROJEYİ YÜKLEME**

GitHub Desktop, komut satırına ihtiyaç duymadan projeleri kolayca GitHub’a yüklemek ve yönetmek için kullanılan görsel bir araçtır.

**1. GitHub Desktop’ı Kurmak**

GitHub Desktop resmi sitesinden (desktop.github.com) indirildikten sonra kurulumu yapılır. Uygulama açıldığında kullanıcıdan GitHub hesabı ile giriş yapması istenir. Henüz hesabı yoksa, aynı platform üzerinden kolayca hesap oluşturulabilir

**2. Yerel Projeyi GitHub’a Aktarmak**

Mevcut bir klasörü yüklemek için uygulama içinden File > Add Local Repository seçeneği kullanılır. Proje klasörü seçildikten sonra, “Publish repository” düğmesine tıklanarak GitHub’a yükleme işlemi tamamlanır. Depo adı ve açıklama eklendikten sonra, istenirse gizlilik durumu (public/private) da ayarlanabilir.

**GitHub Pages ile Yayınlama ve Domain Oluşturma**

Proje GitHub’a yüklendikten sonra, birkaç adımda GitHub Pages ile internet üzerinde yayınlanabilir:

1. GitHub üzerindeki projenin Settings sekmesine girilir.
2. Sol menüdeki Pages bölümünden yayınlanacak dal (örneğin main) seçilir.
3. Ayar kaydedildikten sonra birkaç dakika içinde site aktif hale gelir.
4. GitHub tarafından sağlanan bir bağlantı (örn: kullaniciadi.github.io/projeadi) ile siteye ulaşmak mümkün olur.