



T.C.  
**KIRKLARELİ ÜNİVERSİTESİ**  
**TEKNİK BİLİMLER MESLEK YÜKSEKOKULU**  
**BİLGİSAYAR TEKNOLOJİLERİ BÖLÜMÜ**  
**BİLGİSAYAR PROGRAMCILIĞI PROGRAMI**

**ÖLÇEKLENİBİLİR PROJE MİMARİLERİ**

**EYLÜL KAY**

**1247008007**

**İLERİ WEB PROGRAMLAM0041**

**NADİR SUBAŞI**

**KIRKLARELİ**

**12.2025**



## **İÇİNDEKİLER.**

- 1. Giriş**
- 2. Projenin Amacı**
- 3. Kullanılan Teknolojiler**
- 4 .Sistem Mimarisi**
- 5. Veritabanı Yapısı**
- 6. Uygulama Modülleri**
  - 6.1 Kayıt Olma Modülü (register.php)**
  - 6.2. Giriş Yapma Modülü (login.php)**
  - 6.3. Kullanıcı Paneli (index.php)**
  - 6.4. Kimlik Doğrulama İşlemleri (auth.php)**
- 7. Güvenlik Yaklaşımları**
- 8. Projenin Çalışma Mantığı**
- 9. Sonuç**
- 10. Kaynakça**

## **1. Giriş**

Bu rapor, PHP tabanlı kullanıcı yönetim sistemi projesinin yapısını, çalışma mantığını ve kullanılan teknikleri kapsamlı şekilde açıklamak amacıyla hazırlanmıştır. Proje basit görünse de, web uygulamalarında sıkılıkla kullanılan kayıt, giriş, oturum yönetimi, veritabanı bağlantısı ve erişim kontrolü gibi temel yapı taşlarını barındırmaktadır. Bu nedenle proje, hem eğitim hem de gerçek yaşam projeleri için başlangıç düzeyinde iyi bir örnek niteliği taşır.

Uygulama, kullanıcıların sisteme üye olabileceği, daha sonra giriş yaparak kendilerine ayrılmış paneli görüntüleyebileceği temel bir üyelik sistemi sunar. Proje dosyaları sade, anlaşılır ve işlev odaklı olarak yapılandırılmıştır.

## **2. Projenin Amacı**

Projenin temel amacı, PHP kullanılarak bir kullanıcı yönetim sisteminin nasıl oluşturulduğunu uygulamalı olarak göstermek ve öğrencilerin şu konularda deneyim kazanmasını sağlamaktır:

- Form işlemleri
- Veritabanı bağlantısı
- Kullanıcı doğrulama (authentication)
- Oturum yönetimi (session)
- Güvenli şifre saklama yöntemleri
- Sayfalar arası yönlendirme ve erişim kontrolü

Bu yapı, daha gelişmiş sistemlerin alt temelini oluşturur. Öğrenciler, bu projenin üzerine kullanıcı rolleri, profil yönetimi, şifre sıfırlama gibi daha ileri özellikler ekleyerek sistemi genişletebilir.

### **3. Kullanılan Teknolojiler**

Proje geliştirilirken aşağıdaki teknolojilerden yararlanılmıştır:

PHP:

Sunucu taraflı iş mantığı, form işlemleri, giriş-kayıt kontrolü ve oturum yönetimi PHP ile yazılmıştır.

MySQL/MariaDB:

Kullanıcı bilgilerinin saklandığı veritabanıdır. PHP–MySQL bağlantısı mysqli üzerinden sağlanmıştır.

HTML ve CSS:

Form arayüzleri, kullanıcı paneli ve giriş ekranlarının temel tasarımını bu teknolojilerle oluşturulmuştur.

Session Mekanizması:

Kullanıcı oturumlarını yönetmek için PHP session sistemi kullanılmıştır. Kullanıcı giriş yaptığında session'a kullanıcı bilgisi yazılır ve korunan sayfalara erişim session kontrolüyle sağlanır.

Yerel Sunucu (XAMPP, WAMP vb.):

Projeyi çalıştırmak için gerekli PHP ve MySQL ortamı sağlar.

Bu teknolojiler birlikte kullanılarak basit fakat önemli özelliklere sahip bir web uygulaması ortaya çıkmıştır.

## 4. Sistem Mimarisi

Proje, sayfa tabanlı (page-driven) bir mimari kullanır. Her dosya belirli bir işlev sahiptir:

- index.php → Yalnızca giriş yapmış kullanıcıların görebileceği panel.
- login.php → Giriş formunun yer aldığı sayfa.
- register.php → Yeni kullanıcı kayıt sayfası.
- auth.php → Oturumun varlığını kontrol eden güvenlik dosyası.
- config.php → Sistem ayarları ve genel yapılandırmalar.
- db.php → Veritabanı bağlantısı ve bağlantının her yerde tekrar kullanılmasını sağlama.

Bu mimari, küçük projeler için oldukça pratiktir ve geliştiricinin dosya bazlı düşünmesini kolaylaştırır.

## 5. Veritabanı Yapısı

Veritabanında kullanıcıların bilgilerini tutmak için bir adet tablo kullanılır.

users tablosu şu alanlardan oluşur:

Alan	Tür	Açıklama
id	INT,Auto Increment	Her kullanıcı için benzersiz kimlik
username	VARCHAR	Kullanıcının görünen adı
email	VARCHAR	Kullanıcının girişte kullandığı e-posta
password	VARCHAR	Hash'lenmiş şifre

## **6. Uygulama modülleri**

### **6.1. Kayıt olma modülü (register.php)**

Bu modül, yeni kullanıcıların sisteme kayıt olmasını sağlar. Çalışma adımları:

1. Kullanıcı, kullanıcı adı, e-posta ve şifresini forma girer.
2. Form submit edildiğinde veriler php ile alınır.
3. Aynı e-postanın sistemde kayıtlı olup olmadığı kontrol edilir.
4. Kullanıcı yoksa şifre hash edilir ve veritabanına kaydedilir.
5. Başarılı kayıt sonrası kullanıcı giriş sayfasına yönlendirilir.

Bu modülde en önemli nokta, kullanıcı verilerinin kontrol edilmesi ve şifre hash'leme işlemidir. Bu sayede sahte veya tekrarlı kayıtların önüne geçilir ve güvenli bir kullanıcı yapısı oluşturulur.

### **6.2. Giriş Yapma Modülü (login.php)**

Giriş sistemi şu şekilde çalışır:

1. Kullanıcı e-posta ve şifresini girer.
2. Veritabanında e-posta aranır.
3. Kullanıcı bulunduysa, veritabanındaki hash ile girilen şifrenin doğruluğu password\_verify() ile karşılaştırılır.
4. Şifre doğruysa session başlatılır.
5. Kullanıcı index.php sayfasına yönlendirilir.

Oturum yönetiminin temeli tam bu noktada başlar. Başarılı girişten sonra kullanıcı bilgisi session'a kaydedilir ve giriş yapılmış kabul edilir.

### 6.3. Kullanıcı paneli (index.php)

Burası yalnızca oturumu açık kullanıcıların görebileceği bir alandır.

- Sayfanın en üstünde auth.php dosyası çağrılarak oturum kontrolü yapılır.
- Eğer kullanıcı giriş yapmamışsa otomatik olarak login sayfasına yönlendirilir.
- Panelde genelde şu bilgiler gösterilir:
  - Hoş geldiniz mesajı
  - Kullanıcı adı
  - Çıkış yap butonu

Bu sayfa, oturum yönetiminin doğru çalışıp çalışmadığını gözlemlemek için temel bir test ortamı işlevi görür.

### 6.4. Kimlik doğrulama işlemleri (auth.php)

Bu dosya projenin güvenlik katmanıdır.

- Session\_start() ile mevcut oturum kontrol edilir.
- Session'da kullanıcı bilgisi yoksa kullanıcı sisteme giriş yapmamış kabul edilir.
- Kullanıcı doğrudan index.php veya başka bir özel sayfaya erişmeye çalışırsa login sayfasına yönlendirilir.

Bu yapı, web uygulamalarında en çok kullanılan erişim kontrol yöntemlerinden biridir.

## **7. Güvenlik Yaklaşımları**

Proje küçük ölçekli olsa da aşağıdaki güvenlik yöntemlerini barındırır:

Şifre Hash'leme:

Kullanıcı şifreleri asla düz saklanmaz. Hash'lenmiş olarak depolanır.

Session Yönetimi:

Her kullanıcıya özel oturum anahtarları saklanır. Özel sayfalar yalnızca oturumu aktif kişilerce görüntülenebilir.

SQL Güvenliği için Temel Önlemler:

Parametreli sorgu veya mysqli kullanımı ile temel güvenlik sağlanır.

Doğrudan Erişimi Engellemeye:

Kritik sayfalar yalnızca auth.php kontrolünden geçerek açılabilir.

Bu yöntemler, daha büyük projelerde kullanılacak gelişmiş güvenlik adımlarının temelini oluşturur.

## **8. Projenin Çalışma Mantığı**

Proje genel çalışma akışı şu şekildedir:

1. Kullanıcı kayıt sayfasına gider ve bilgilerini girer.
2. Bilgiler veritabanına güvenli şekilde kaydedilir.
3. Kullanıcı login sayfasına giderek giriş yapar.
4. E-posta ve şifre doğrulanır, doğruysa session başlatılır.
5. Kullanıcı paneline yönlendirilir.
6. Korunan tüm sayfalar session üzerinden erişim kontrolü yapar.
7. Kullanıcı çıkış yapmak istediğiinde session sonlandırılır ve sistemden güvenli şekilde ayrılır.

Bu akış hemen hemen tüm web tabanlı kullanıcı sistemlerinde standart olarak kullanılan bir yöntemdir.

## **9. Sonuç**

Bu proje, PHP ile kullanıcı doğrulama, kayıt sistemi, oturum yönetimi ve veritabanı etkileşimi gibi temel konuların uygulamalı bir örneğini sunmaktadır. Özellikle öğrenciler için PHP'nin temel mantığını öğrenme açısından oldukça pratik bir projedir. Kod yapısı sade ve anlaşılır olduğu için geliştirilebilir niteliktedir. İlerleyen aşamalarda profil düzenleme, yetki seviyeleri, şifre yenileme gibi özellikler eklenerek proje daha profesyonel bir kullanıcı yönetim sistemine dönüştürülebilir.

## **10. Kaynakça**

1. PHP Official Documentation – [php.net](http://php.net)  
PHP'nin resmi dokümantasyonu. Fonksiyonlar, session yönetimi, veritabanı işlemleri ve şifreleme yöntemlerinin teknik detayları burada açıklanmaktadır.
2. MySQL Documentation – [dev.mysql.com](http://dev.mysql.com)  
MySQL veritabanı sorguları, tablo yapıları, güvenlik ilkeleri ve bağlantı yöntemleri için başvuru kaynağı.
3. W3Schools PHP Tutorials – [w3schools.com/php](http://w3schools.com/php)  
PHP'nin temel sözdizimi, form işlemleri, session kullanımı ve veritabanı bağlantı örneklerinin açıkladığı eğitim içerikleri.
4. Mozilla Developer Network (MDN) – [developer.mozilla.org](http://developer.mozilla.org)  
HTML ve web standartları ile ilgili teknik açıklamalar ve form yapılarının çalışma prensipleri.
5. OWASP Foundation – [owasp.org](http://owasp.org)  
Web uygulamalarında güvenlik, şifreleme, SQL Injection ve giriş doğrulama yöntemleri için referans niteliğinde güvenlik rehberleri.
6. Stack Overflow PHP Discussions – [stackoverflow.com](http://stackoverflow.com)  
PHP ile ilgili yaygın sorunların çözüm önerileri ve geliştirici tartışmalarının yer aldığı açık kaynak bilgi platformu.
7. PHP: The Right Way – [phptherightway.com](http://phptherightway.com)  
PHP için doğru, güvenli ve modern geliştirme yaklaşımlarını açıklayan çevrim içi rehber.