

Dosya Yükleme ve Yönetimi Uygulaması

Hazırlayan: Semih Gökmen

Tarih: 22.05.2025

1. Giriş

Bu rapor, DFA Bilişim tarafından staj görevleri kapsamında istenilen “Dosya Yükleme ve Yönetimi Uygulaması” adlı projenin tasarımını, uygulamasını ve kullanılan teknolojileri açıklamaktadır. Proje, kullanıcıların güvenli bir şekilde dosya yükleyip yönetebildiği, kayıt olabildiği ve giriş yapabildiği web tabanlı bir platformdur. Uygulama, basitlik, kullanılabilirlik ve güvenlik öncelikli olarak tasarlanmıştır.

2. Proje Amaçları

- Kullanıcıların dosyalarını çevrimiçi olarak güvenli ve kullanıcı dostu bir ortamda yönetebilmesini sağlamak.
- Temel kimlik doğrulama işlemlerini sunmak (kayıt, giriş, şifre sıfırlama).
- Dosya yükleme, listeleme, indirme ve silme işlemlerini desteklemek.
- Her kullanıcının yalnızca kendi dosyalarına erişebilmesini sağlamak.
- Modern ve duyarlı (responsive) bir arayüz sunmak.

3. Sistem Mimarisi

Uygulama, basit bir istemci-sunucu mimarisi ile geliştirilmiştir:

- **Frontend:** HTML, CSS, JavaScript
- **Backend:** PHP
- **Sunucu:** PHP built-in server
- **Veri Depolama:** JSON dosyaları (users.json, uploads.json)
- **Dosya Depolama:** Yerel dizin (uploads/)
- **Geliştirme Ortamı:** Visual Studio Code

4. Kullanılan Teknolojiler ve Sebepleri

4.1 PHP

Sebepler: PHP, yaygın olarak desteklenen, kolay dağıtılabilen ve küçük-orta ölçekli web uygulamaları için uygun bir dildir. Prosedürel PHP, hızlı prototipleme için yeterlidir.

Kullanım: Kimlik doğrulama, dosya yükleme/indirme/silme ve veri yönetimi işlemleri için.

4.2 HTML & CSS

Sebepler: Web sayfalarının yapılandırılması ve stil verilmesi için standart teknolojilerdir. CSS, duyarlı ve modern tasarım sağlar.

Kullanım: Tüm kullanıcı arayüzleri (giriş, kayıt, panel, dosya listesi vb.) HTML ve CSS ile oluşturulmuştur.

4.3 JavaScript

Sebepler: Kullanıcı etkileşimlerinin dinamik hale getirilmesini sağlar; sürükle bırak dosya yükleme, sayfa yenilenmeden dosya listesini güncelleme ve tablo sıralama gibi işlevler için tercih edilmiştir.

Kullanım: Dosya seçimi, AJAX ile dosya yükleme ve silme işlemleri, kullanıcı arayüzü geri bildirimi.

4.4 JSON Dosyaları

Sebep: Hafif ve kolay yönetilebilen bir veri saklama yöntemidir. Küçük ölçekli uygulamalarda veri tabanı ihtiyacı duymadan kullanılabilir.

Kullanım: users.json dosyası kullanıcı bilgilerini (şifreler hashlenmiş şekilde), uploads.json ise dosya yükleme bilgilerini saklar.

4.5 Font Awesome (CDN)

Sebep: Yaygın kullanılan modern bir ikon setidir; kullanıcı deneyimi ve görsel sadelik açısından avantaj sağlar.

Kullanım: Dosya yükleme, indirme, silme, çıkış gibi eylemler için ikon desteği sunar.

4.6 Sunucu: PHP Built-in Server

Sebep: PHP'nin dahili sunucusu, küçük projeler ve lokal geliştirme ortamları için yapılandırma gerektirmeden hızlı test ve geliştirme imkânı sunar.

Kullanım: Uygulamanın lokal geliştirme ve test süreci sırasında çalıştırılması için tercih edilmiştir. Harici bir web sunucusu kurulumuna gerek kalmadan doğrudan çalıştırılabilir.

4.7 Geliştirme Ortamı: Visual Studio Code

Sebep: Hafif yapısı, geniş eklenti desteği ve entegre terminal özelliği ile modern web geliştirme için tercih edilen popüler bir editördür.

Kullanım: Kod yazımı, proje dosyalarının yönetimi, terminal üzerinden PHP server başlatma ve hata ayıklama işlemleri için kullanılmıştır.

5. Güvenlik Önlemleri

- **Şifre Hashleme:** Kullanıcı şifreleri PHP'nin password_hash() fonksiyonu ile güvenli bir şekilde saklanır.
- **Oturum Yönetimi:** Kimlik doğrulama PHP oturumları (session) ile kontrol edilir. Dosya işlemlerine yalnızca giriş yapmış kullanıcılar erişebilir.
- **Dosya Doğrulama:** Sadece PDF, JPG ve PNG dosya türlerine izin verilir. Dosya isimleri özel karakterlerden arındırılır, dizin geçiş saldırılarına karşı korunur.
- **Kullanıcı İzolasyonu:** Her kullanıcı yalnızca kendi dosyalarını görüntüleyebilir ve yönetebilir.

6. Uygulama Yapısı

- dashboard.php – Giriş sonrası ana panel
- login_form.php, register_form.php, forgot_password_form.php – Kimlik doğrulama ekranları
- file_list.php – Kullanıcı dosyalarının listelendiği ve işlem yapılabilirdiği sayfa
- helpers.php – Oturum ve JSON işlemleri için yardımcı fonksiyonlar
- upload_process.php – Dosya yükleme işlemleri
- delete_file.php – Dosya silme işlemleri
- login_process.php, register_process.php, reset_password_process.php – Kimlik doğrulama mantığı
- logout_process.php – Oturumu sonlandırma
- uploads/ – Yüklenen dosyaların saklandığı dizin
- users.json, uploads.json – Veri saklama dosyaları

7. Kullanıcı Deneyimi

- **Kayıt/Giriş:** Basit, doğrulamalı ve kullanıcıya geri bildirim sunan formlar.
- **Dosya Yükleme:** Sürükle bırak desteği ve çoklu dosya seçimi imkânı.
- **Dosya Yönetimi:** Kullanıcılar yalnızca kendi dosyalarını görüntüleyebilir, indirebilir ve silebilir.
- **Şifre Sıfırlama:** Güvenli ve kullanıcı dostu bir sıfırlama süreci sunar.

8. Kısıtlar ve Gelecek Çalışmalar

- **Ölçeklenebilirlik:** JSON dosyaları büyük veri veya eş zamanlı kullanıcı senaryoları için uygun değildir. MySQL gibi bir veri tabanına geçiş önerilir.
- **Güvenlik:** Üretim ortamı için CSRF koruması, hız sınırlama (rate limiting) ve virüs tarama gibi ek güvenlik önlemleri alınmalıdır.
- **Ek Özellikler:** Kullanıcı rolleri, dosya paylaşımı, gelişmiş arama ve filtreleme gibi özellikler ileride eklenebilir.

9. Sonuç

Dosya Yükleme ve Yönetimi Uygulaması, web ortamında temel dosya yönetimini güvenli, kullanıcı dostu ve modern bir şekilde gerçekleştirmektedir. Seçilen teknolojiler, basitlik ve işlevselliği dengelerken, projenin eğitim veya kişisel kullanım için kolayca dağıtılıp sürdürülebilir olmasını sağlamaktadır.