

Kitap Arşivim Uygulaması - Proje Raporu

Proje Adı:

Kitap Arşivim Uygulaması

Proje Türü:

Web tabanlı kitap yönetim sistemi (HTML + Bootstrap5 + JavaScript + localStorage)

Projenin Amacı:

Kullanıcının okuduğu veya okumayı planladığı kitapları kolayca ekleyip yönetebileceği, sade ve kullanıcı dostu bir arayüz sunan bir web uygulaması geliştirmektir.

Uygulama sayesinde kullanıcılar kitap bilgilerini ekleyebilir, düzenleyebilir, silebilir ve verilerini kalıcı olarak saklayabilir.

Kullanılan Teknolojiler:

Teknoloji	Açıklama
HTML	Uygulama iskeletini ve formları oluşturmak için kullanıldı.
Bootstrap 5	Tüm stil ve responsive yapı için kullanıldı. Ekstra CSS yazılmadı.
JavaScript (ES6 Modules)	Uygulama mantığı modüler şekilde ayrıldı (main.js, ui.js, events.js, services/).
localStorage	Kullanıcı verilerinin tarayıcıda kalıcı olarak saklanması sağlandı.

Klasör Yapısı (Profesyonel Yaklaşım):

```
kitap-arsivim/  
├── index.html  
├── js/  
│   ├── main.js  
│   ├── ui.js  
│   ├── events.js  
│   └── services/  
│       └── kitapService.js
```

Uygulama Özellikleri:

- - Kitap Ekleme (FormData API ile veri toplama)
- - Kitap Listeleme (Tablo formatında, kapak görseli ile)
- - Silme İşlemi (Bootstrap Modal ile modern onay)
- - Düzenleme İşlemi (Bootstrap Modal üzerinden güncelleme)
- - Durum Seçimi (Okundu/Okunmadı, checkbox-radio mantığı)
- - Veri Saklama (localStorage ile kalıcı kayıt)

Teknik Detaylar:

FormData API sayesinde form verileri tek satırda okunur. Eski yöntemlerle `input.value` yerine daha dinamik bir yapı sunar.

Bootstrap Modal ile kullanıcıdan modern onay alınır. Bu, klasik JavaScript `confirm()` yerine daha profesyonel bir görünüm sağlar.

Checkbox'lar özel olarak radio mantığında çalışır; sadece bir tanesi seçilebilir.

Tüm JS dosyaları görevlerine göre modülerleştirilmiş ve okunabilirliği artırılmıştır.

Proje Çalışma Mantığı:

1. Sayfa açıldığında `main.js` çalışır.
2. `renderKitaplar()` çağrılır ve kitaplar tabloya yüklenir.
3. Formdan kitap eklendiğinde `localStorage`'a kaydedilir.
4. Düzenle ve Sil butonları ilgili modali açar.
5. Kullanıcı etkileşimleri `events.js` tarafından yönetilir.
6. Güncellemeler UI üzerinden anında görünür hale gelir.

Sonuç:

Bu proje, profesyonel yapıda geliştirilmiş, sade ama güçlü bir kitap arşiv yönetim sistemidir. Kod yapısı modüler, okunabilir ve sürdürülebilirdir. Uygulama, hem bireysel kullanımlar hem de eğitim amaçlı örneklemeler için idealdir.