



KOCAELİ ÜNİVERSİTESİ

TEKNOLOJİ FAKÜLTESİ

BİLİŞİM SİSTEMLERİ MÜHENDİSLİĞİ

VERİTABANI YÖNETİM SİSTEMLERİ

Dr. Öğr. Üyesi Samet DİRİ

OTOPARK OTOMASYON PROJESİ

201307057

Abdullah İŞLER

Proje Raporu

1. Giriş

Günümüzde kentleşmenin artmasıyla birlikte otopark alanlarının etkin ve verimli bir şekilde yönetilmesi büyük önem taşımaktadır. Manuel sistemlerin kullanıldığı otoparklarda araç giriş-çıkışlarının takibi, ödeme işlemleri ve yer durumu gibi verilerin doğru şekilde yönetilememesi, hem kullanıcılar hem de işletmeciler açısından zaman ve kaynak kaybına yol açabilmektedir.

Bu proje kapsamında geliştirilen "Otopark Otomasyon Sistemi", otopark yönetimini dijitalleştirerek bu sorunlara çözüm sunmayı amaçlamaktadır. Web tabanlı bir yapıda geliştirilen bu sistem, otoparka giriş yapan araçların kaydının tutulmasını, çıkış işlemlerinin gerçekleştirilmesini, ödeme takibinin yapılmasını ve otopark doluluk oranlarının grafiksel olarak analiz edilmesini sağlar.

Sistemde iki temel kullanıcı rolü bulunmaktadır: Yönetici (Admin) ve Personel. Yönetici kullanıcılar sisteme tam yetkiyle erişebilirken, personel kullanıcılar yalnızca belirli işlemleri gerçekleştirebilmektedir. Bu ayrım sayesinde veri güvenliği ve görev dağılımı sağlanmıştır.

Bu proje, React ile geliştirilen kullanıcı dostu bir arayüz, Node.js tabanlı Express.js ile oluşturulmuş RESTful API ve MySQL veritabanı ile desteklenen bir arka uç yapısına sahiptir. Proje, gerçek dünyada kullanılabilecek, sürdürülebilir ve geliştirilebilir bir temel sunmaktadır.

2. Proje Tanıtımı ve Amaç

Bu proje, otopark işletmelerinin günlük süreçlerini dijitalleştirerek daha sistematik ve verimli bir şekilde yönetilmesini sağlamak amacıyla geliştirilmiştir. Otopark Otomasyon Sistemi, araç giriş-çıkış takibi, anlık doluluk kontrolü, ödeme alma ve haftalık performans raporlama gibi işlemleri merkezi bir sistem üzerinden gerçekleştirmeye olanak tanır.

Sistem, hem otopark personelinin iş yükünü azaltmayı hem de yöneticilerin iş süreçlerini daha sağlıklı ve kontrollü bir şekilde yürütebilmesini hedeflemektedir. Kullanıcı dostu arayüzü ve yetki seviyelerine göre ayrılmış erişim yapısıyla farklı rollerin sisteme erişimi sınırlandırılmıştır.

Proje kapsamında;

- Otopark alanlarının tanımlanması ve duruma göre güncellenmesi,
- Araç bilgilerinin sisteme girilmesi ve yönetilmesi,
- Giriş yapan araçların kaydedilmesi,
- Çıkış esnasında ödeme alınması ve faturalandırma,
- Günlük ve haftalık bazda giriş-çıkış, kazanç analizlerinin grafiklerle raporlanması gibi temel işlevler sunulmaktadır.

Uygulama web tabanlı olduğundan, kullanıcılar herhangi bir cihazdan sisteme kolayca erişim sağlayabilir. Bu sayede anlık veriler görüntülenebilir, geçmiş işlemler incelenebilir ve raporlama süreçleri hızlıca yönetilebilir.

3. Kullanılan Teknolojiler

Bu proje, web tabanlı bir otopark otomasyon sistemi geliştirmek amacıyla modern ve açık kaynak teknolojiler kullanılarak hazırlanmıştır. Sistemin hem yönetici hem de personel tarafında işlevsel, kullanıcı dostu ve güvenli bir deneyim sunması hedeflenmiştir. Kullanılan temel teknolojiler aşağıda listelenmiştir:

Frontend (İstemci Tarafı)

React.js

Kullanıcı arayüzlerinin oluşturulmasında modern ve bileşen tabanlı yapı sunan JavaScript kütüphanesi.

React Router

Sayfalar arasında yönlendirme sağlamak için kullanılmıştır.

Recharts

Haftalık giriş, çıkış ve kazanç grafiklerinin görselleştirilmesinde kullanılmıştır.

CSS Modules / Özelleştirilmiş CSS

Her sayfa için ayrı yazılmış stil dosyaları ile düzenli ve karışıklıktan uzak tasarım sağlanmıştır.

Backend (Sunucu Tarafı)

Node.js & Express.js

API geliştirme, veri işlemleri ve sunucu yönetimi için kullanılmıştır.

RESTful API Mimarisi

Her veri işlemi için özelleştirilmiş GET, POST, PUT, DELETE endpoint'leri tasarlanmıştır.

Veritabanı

MySQL

Tüm kullanıcılar, araçlar, giriş-çıkış kayıtları, otopark alanları ve ödeme verilerinin saklandığı ilişkisel veritabanıdır.

MySQL Workbench

Veritabanı yönetimi, tablo yapılarının düzenlenmesi ve sorgu işlemleri için kullanılmıştır.

Kimlik Doğrulama ve Yetkilendirme

Yerel Giriş Sistemi

Kullanıcılar (personel ve admin) sisteme email ve şifre ile giriş yapar. Giriş sonrası bilgiler localStorage aracılığıyla tutulur ve ilgili role yönlendirme yapılır.

Sürüm Kontrolü

Git & GitHub

Proje geliştirme süreci boyunca versiyon kontrolü yapılmış ve GitHub üzerinde barındırılmıştır.

4. Yöntem ve Uygulama

Bu projenin geliştirme süreci, yazılım mühendisliğinde yaygın olarak benimsenen modüler ve bileşen tabanlı yaklaşım esas alınarak planlanmış ve uygulanmıştır. Proje; kullanıcı arayüzü, sunucu tarafı işlemleri ve veritabanı yönetimi olmak üzere üç temel katmanda ele alınmıştır.

4.1. Analiz ve Planlama

İlk aşamada, otopark sisteminin ihtiyaçları analiz edilmiştir. Yönetici (admin) ve personel olmak üzere iki ayrı kullanıcı rolü belirlenmiş, her bir rolün sahip olacağı yetkiler tanımlanmıştır. Temel işlevler şunlardır:

- Otopark alanlarını görüntüleme ve yönetme
- Araç ekleme, giriş-çıkış kaydı oluşturma
- Ödeme alma ve raporlar görüntüleme

4.2. Arayüz Geliştirme (Frontend)

Kullanıcı dostu ve sade bir arayüz oluşturmak için React.js teknolojisi kullanılmıştır. Her kullanıcı rolü için özelleştirilmiş ayrı paneller tasarlanmıştır:

Yönetici Paneli: Tüm veriler üzerinde tam yetkiye sahiptir. Ekleme, güncelleme, silme ve raporlama işlemleri yapabilir.

Personel Paneli: Sadece izin verilen işlemleri gerçekleştirebilir (ör. listeleme ve yeni kayıt ekleme).

Her sayfa için ayrı CSS dosyaları oluşturularak stil yönetimi modüler hâle getirilmiştir. Ayrıca, grafik gösterimleri için recharts kütüphanesi entegre edilmiştir.

4.3. Sunucu Tarafı Geliştirme (Backend)

Node.js ve Express.js ile geliştirilen RESTful API sayesinde istemci tarafı ile veritabanı arasında veri alışverişi gerçekleştirilmiştir. Her işlem için özel endpoint'ler oluşturulmuş, hata yönetimi yapılmıştır. Kullanıcı kimlik doğrulama ve yönlendirme işlemleri yine backend üzerinden kontrol edilmiştir.

4.4. Veritabanı Tasarımı ve Uygulaması

MySQL kullanılarak;

Calisanlar, Araclar, Giris_Kayitlari, Cikis_Kayitlari, Otopark_Yerleri, Odemeler tabloları oluşturulmuştur. Bu yapılar arasında ilişkiler tanımlanmış, birincil ve yabancı anahtarlar (PK/FK) ile veri bütünlüğü sağlanmıştır.

4.5. Test ve Son Kontroller

Her modül ayrı ayrı test edilmiştir. Giriş-çıkış kayıtları, ödeme akışı ve grafik verileri doğru çalışacak şekilde API üzerinden test edilmiştir. Tarayıcı konsolu ve ağ (network) analizleriyle hata takibi yapılmıştır.

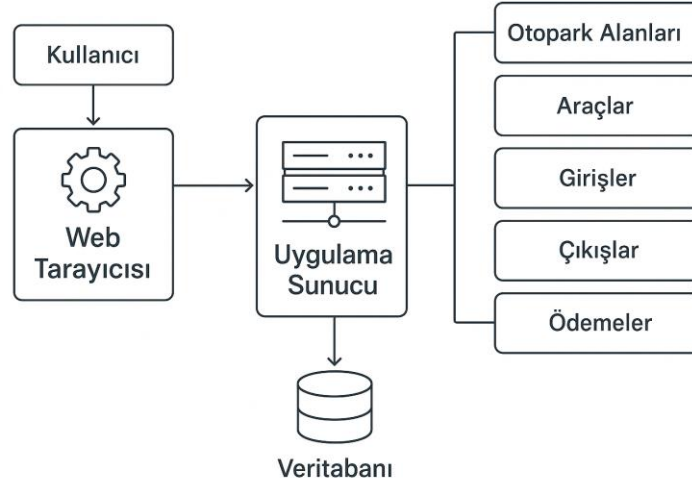
4.6. Sürüm Kontrolü ve Yayınlama

Tüm geliştirme süreci boyunca Git kullanılmış, her önemli değişiklik ayrı commit'lerle takip edilmiştir. Kodlar GitHub üzerinde barındırılmıştır.

5. Sistem Mimarisi

Bu projede geliştirilen **Web Tabanlı Otopark Otomasyon Sistemi**, kullanıcı ve yönetici etkileşimlerinin merkezi olarak yürütüldüğü bir yapıya sahiptir. Sistem, kullanıcıların araç giriş-çıkış işlemlerini yönetmesine, otopark doluluk durumunu görüntülemesine ve ödeme işlemlerini gerçekleştirmesine olanak tanır. Aynı zamanda yöneticilere (admin) çalışan, araç ve otopark alanları üzerinde tam kontrol sağlar.

Aşağıdaki mimari diyagram, sistemin temel bileşenlerini ve bunlar arasındaki veri akışını özetlemektedir:



Şekil 1 Sistem mimari yapısı

Kullanıcı (Personel/Admin) Arayüzü (React.js):

Web tarayıcısı üzerinden erişilen kullanıcı arayüzü React teknolojisi ile geliştirilmiştir.

Kullanıcı, giriş yaptıktan sonra rolüne uygun bir panelle karşılaşır (admin veya personel).

Giriş, çıkış, otopark durumu ve ödeme işlemleri burada gerçekleştirilir.

Sunucu (Node.js + Express.js API):

Arayüz ile veritabanı arasında köprü görevi görür.

RESTful API yapısıyla verileri alır, işler ve geri döner.

Kullanıcı doğrulama, kayıt, güncelleme ve raporlama işlemleri bu katmanda gerçekleşir.

Veritabanı (MySQL):

Araçlar, çalışanlar, giriş-çıkış kayıtları, otopark alanları ve ödemelere ait tüm bilgiler burada tutulur.

Veritabanı ilişkisel olarak yapılandırılmıştır ve API üzerinden sorgularla erişilir.

İşleyiş Akışı:

Kullanıcı sisteme giriş yapar.

Sistemde kullanıcı rolüne göre erişim sağlanır.

Giriş/çıkış yapan araç bilgileri kayıt altına alınır.

Ödeme yapıldığında bu işlem yine API üzerinden veritabanına kaydedilir.

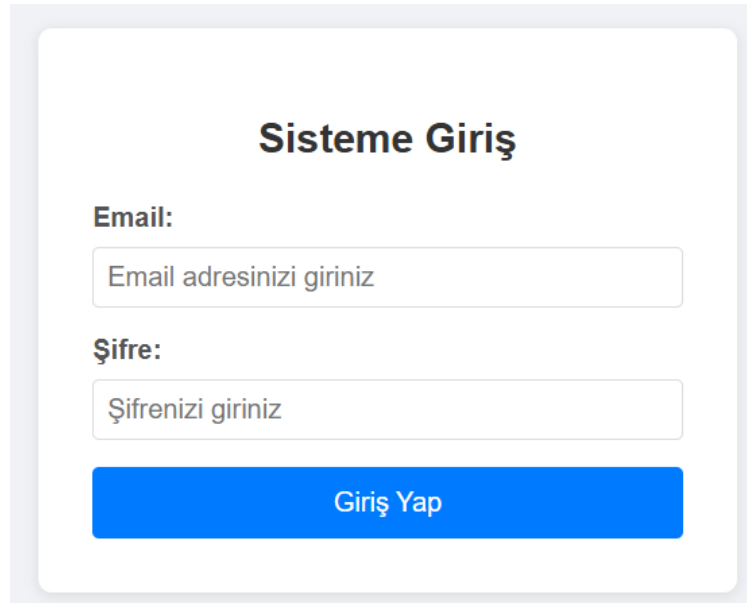
Admin panelinde grafik ve tablolar aracılığıyla bu verilerin analizi yapılabilir.

Bu yapı sayesinde sistem; kullanıcı dostu, ölçeklenebilir ve yönetimi kolay bir hale gelmiştir. Modüler mimarisi sayesinde yeni özelliklerin entegrasyonu da oldukça esnek biçimde gerçekleştirilebilir.

6. Kullanıcı Arayüzü ve İş Süreçleri

6.1. Giriş Sayfası

- Kullanıcı (admin/personel) giriş bilgilerini girer.
- Doğrulama sonrası kullanıcı rolüne göre yönlendirme yapılır.
- Başarılı girişlerde kullanıcı bilgisi localStorage'a kaydedilir.



Sisteme Giriş

Email:

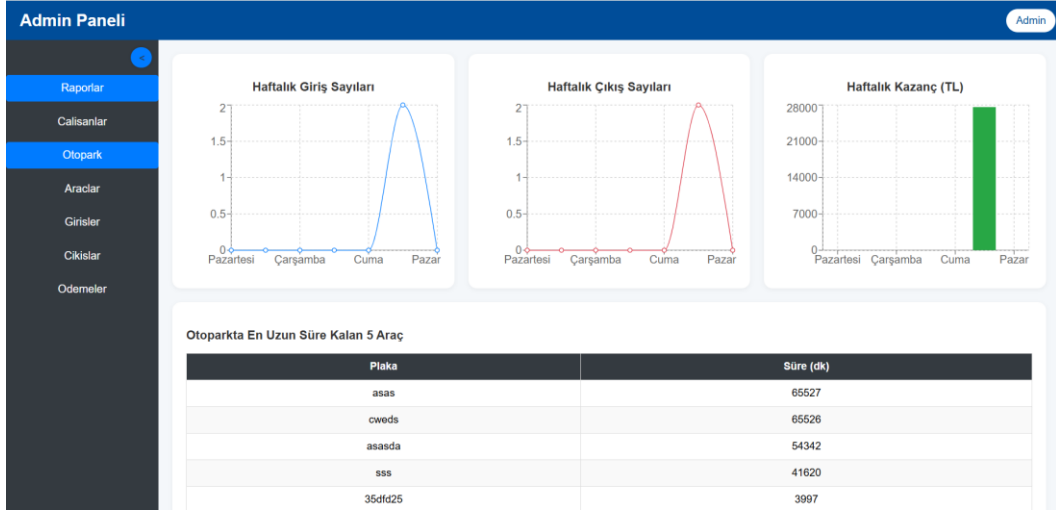
Şifre:

Giriş Yap

Şekil 2 Giriş sayfa yapısı

6.2. Admin Paneli Anasayfası

Yöneticiye özel tüm sayfalara yönlendirme yapılabilir. Üst kısımda kullanıcı adı gösterilir, çıkış butonu mevcuttur. Sol kısımda açılır-kapanır bir sidebar menü bulunur.



Şekil 3 Admin paneli ana sayfası

6.3. Otopark Alanları Sayfası

Admin Rolü:

- Alan ekleme, düzenleme ve silme yetkisi vardır.

Personel Rolü:

- Sadece görüntüleme yapılabilir. Butonlar pasiftir.

Otopark Alanları Yönetimi

Alan Adı

Boş

Ekle

ID	Alan Adı	Durum	İşlemler
1	A1	bos	Güncelle Sil
3	A2	dolu	Güncelle Sil
4	A3	dolu	Güncelle Sil

Şekil 4 admin paneli otopark alan sayfası

Otopark Alanları

ID	Alan Adı	Durum
1	A1	bos
3	A2	dolu
4	A3	dolu
5	A4	dolu

Şekil 5 personel paneli otopark alanları

6.4. Araçlar Sayfası

Yeni araç ekleme ve mevcut araçlar üzerinde düzenleme/silme yapılabilir.

Giriş işlemleri bu sayfada kullanılan araçlar üzerinden yapılır.

Araç Yönetimi

Ekle

ID	Plaka	İşlemler
11	25GH25	<button>Güncelle</button> <button>Sil</button>
4	25jnk25	<button>Güncelle</button> <button>Sil</button>
38	25klm85	<button>Güncelle</button> <button>Sil</button>
6	25sks58	<button>Güncelle</button> <button>Sil</button>
13	25SSA78	<button>Güncelle</button> <button>Sil</button>

Şekil 6 araç yönetim sayfası

6.5. Girişler Sayfası

Otoparka giren araçlar bu sayfada kaydedilir.

Girilen araçlar içeride araçlar listesine otomatik olarak eklenir.

Giriş Yönetimi

Giriş Ekle

Giriş ID	Araç Plaka	Yer Adı	Giriş Saati	Çıkış Durumu
31	65lm45	A3	18.05.2025 11:33:43	çıkmadı
30	34kom65	A4	18.05.2025 11:33:17	çıkmadı
29	25klm85	A5	10.05.2025 22:17:21	çıkta

Şekil 7 araç giriş sayfası

6.6. Çıkışlar Sayfası

İçeride bulunan araçlar listelenir.

Çıkış işlemi için ödeme yöntemi seçimi yapılır.

Ödeme tamamlandığında çıkış bilgileri ve ücret kaydedilir.

Otoparktaki Araçlar				Çıkış Yapan Araçlar				
Plaka ara...				Plaka	Giriş Saati	Çıkış Saati	Ödeme Yöntemi	Ücret
65lm45	18.05.2025 11:33:43	A3	Çıkış Yap	asas	02.04.2025 23:26:28	18.05.2025 11:33:51	nakit	13800.00 TL
34kom65	18.05.2025 11:33:17	A4	Çıkış Yap	cweds	02.04.2025 23:26:41	18.05.2025 11:33:23	kart	13800.00 TL
asasasa	02.04.2025 23:26:39	A2	Çıkış Yap	asasda	03.04.2025 04:35:34	10.05.2025 22:17:42	nakit	11400.00 TL

Şekil 8 çıkış yönetim sayfası

6.7. Ödemeler Sayfası

Tüm ödeme işlemleri listelenir. Her satırda plaka, giriş-çıkış saatleri ve ödeme bilgileri yer alır.

Ödeme Kayıtları		Ödeme Detayları	
Araç plakasına göre ara...		Ödeme ID:	23
		Araç ID:	20
		Plaka:	asas
		Giriş ID:	16
		Giriş Saati:	02.04.2025 23:26:28
		Çıkış ID:	26
		Çıkış Saati:	18.05.2025 11:33:51
		Toplam Süre:	65527 dakika
		Ücret:	13800.00 TL
		Kapat	

Ödeme ID	Ücret (TL)	Detaylar
23	13800.00	Görüntüle
22	13800.00	Görüntüle
21	11400.00	Görüntüle
20	0.00	Görüntüle
19	8700.00	Görüntüle

Şekil 9 ödemeler tablosu

7. Sonuç ve Genel Değerlendirme

Bu proje kapsamında, web tabanlı bir otopark otomasyon sistemi geliştirilmiştir. Sistem, hem yönetici (admin) hem de personel kullanıcı rollerine göre özelleştirilmiş ekranlara ve işlevselliğe sahiptir. Kullanıcı doğrulaması, rol tabanlı erişim kontrolü, araç giriş/çıkış yönetimi, ödeme takibi ve otopark raporları gibi temel işlevler başarıyla entegre edilmiştir.

Proje geliştirme süreci boyunca React.js ile kullanıcı arayüzü oluşturulmuş, Express.js ile RESTful API mimarisi kurulmuş ve veriler MySQL veritabanında güvenli şekilde yönetilmiştir. Geliştirilen grafiksel raporlama arayüzü sayesinde sistem yöneticisi haftalık giriş, çıkış ve kazanç istatistiklerini görsel olarak takip edebilmektedir.

Bu proje ile birlikte:

- Gerçek bir CRUD tabanlı sistemin uçtan uca nasıl geliştirileceği öğrenilmiş,
- Frontend ve backend arasında veri akışı kurulmuş,
- Rol bazlı kullanıcı yönetimi ve güvenlik temelleri uygulanmıştır.

Proje, ölçeklenebilir yapısıyla daha büyük sistemlere entegre edilebilir ve ek modüllerle geliştirilmeye açıktır

8. Kaynakça

- <https://react.dev/>
- <https://expressjs.com/>
- <https://dev.mysql.com/doc/>
- <https://recharts.org/en-US>
- <https://developer.mozilla.org/en-US/>
- <https://www.w3schools.com/>