



**Beykent Üniversitesi  
Mühendislik ve Mimarlık Fakültesi  
Yazılım Mühendisliği Bölümü**

**SRS Dokümantasyonu**

## **AKILLI OTOPARK SİSTEMİ**

<b>Burak</b>	<b>FIRÇASIGÜZEL</b>	<b>130301022</b>
<b>Emre</b>	<b>SERBES</b>	<b>130301020</b>
<b>Eray</b>	<b>AKARTUNA</b>	<b>130301026</b>

## **İçindekiler :**

### **1. Giriş**

- 1.1 Amaç**
- 1.2 Kapsam**
- 1.3 Tanımlamalar ve Kısaltmalar**
- 1.4 Genel Açıklamalar**

### **2. Tanımlamalar**

- 2.1 Ürün Perspektifi**
  - 2.1.1 Sistem Arayüzleri**
  - 2.1.2 Kullanıcı Arayüzleri**
  - 2.1.3 Donanım Arayüzleri**
  - 2.1.4 Yazılım Arayüzleri**
  - 2.1.5 İletişim Arayüzleri**
  - 2.1.6 Operasyonlar**
- 2.2 Ürün Fonksiyonları**
  - 2.2.1 Yönetici Ekleme**
  - 2.2.2 Yönetici Girişi**
  - 2.2.3 Yönetici Çıkışı**
  - 2.2.4 Harita Oluşturma**
  - 2.2.5 Araç Girişi**
  - 2.2.6 Uygun Park Yeri Bulma**
  - 2.2.7 Araç Çıkış ve Ücretlendirme**
  - 2.2.8 Araçları Listeleme**
  - 2.2.9 Yönetici Güncelleme**
  - 2.2.10 Yönetici Şifre Sıfırlama**

#### **2.3 Kısıtlamalar**

### **3. Spesifik Gereksinimler**

- 3.1 Fonksiyonel Gereksinimler**
  - 3.1.1 Yönetici Ekleme**
  - 3.1.2 Yönetici Girişi**
  - 3.1.3 Yönetici Çıkışı**
  - 3.1.4 Harita Oluşturma**
  - 3.1.5 Araç Girişi**
  - 3.1.6 Uygun Park Yeri Bulma**
  - 3.1.7 Araç Çıkış ve Ücretlendirme**
  - 3.1.8 Araçları Listeleme**
  - 3.1.9 Yönetici Güncelleme**
  - 3.1.10 Yönetici Şifre Sıfırlama**
- 3.2 Fonksiyonel Olmayan Gereksinimler**

## 1. Giriş

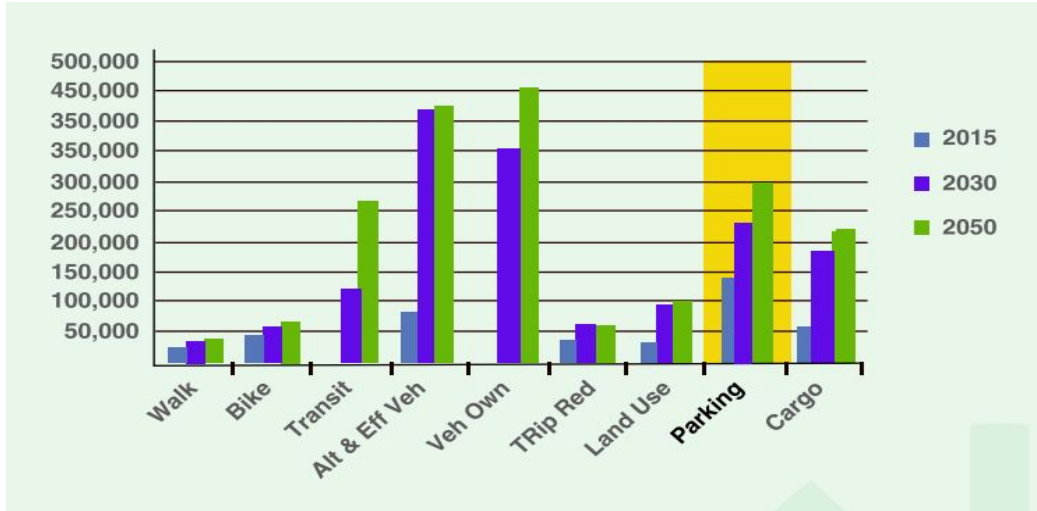
### 1.1. Amaç

Akıllı otopark sistemi, otoparklarda ki yer arama problemini çözmek amacıyla kullanılacaktır. Bu uygulama sayesinde kullanıcı otopark alanının girişinden park etme sürecine kadar çeşitli bileşenler yardımıyla kolayca aracını en uygun park alanına park edebilecektir. Projenin bir diğer amacı ise park yeri ararken ki geçen süre ve buna bağlı olarak yakılan yakıtın minimum düzeyde tutulup, CO emisyon değerini alt basamaklarda tutarak çevre ve doğaya en az tahribatı vermektir.



## 1.2. Kapsam

Akıllı otopark yazılımı, bir otoparka giriş yapan tüm araçların kaydedilebileceği, otomatik park yerleşiminin yaptırılabilmesi bir yazılım olarak geliştirilecektir. Yazılımın veri tabanı web üzerinde saklanılacaktır. Yazılım ise masaüstü ve web yazılımları sayesinde gerçekleştirilecektir.



(Grafik kaynak: www.yec.co)

Akıllı Park, 2030 yılına kadar yaklaşık 220.000 galon yakıt tasarrufu sağlayabilir. 2050 yılına kadar 3.00.000 galon yakıt tasarrufu sağlanırsa başarıyla uygulanır.

## 1.3. Tanımlamalar ve Kısaltmalar

<b>API</b>	Application Programming Interface, bir yazılımın başka bir yazılımda tanımlanmış işlevlerini kullanabilmesi için oluşturulmuş bir tanım bütünüdür.
<b>Web Service</b>	Bir web servisi, elektronik cihaz tarafından sunulan ve World Wide Web üzerinden birbiriyle iletişim kuran bir servistir.
<b>Server</b>	Bilgisayar ağlarında, diğer ağ bileşenlerinin (kullanıcıların) erişebileceği, kullanımına ve/veya paylaşımına açık kaynakları barındıran bilgisayar birimi.
<b>HTTP</b>	HTTP ağ üzerinden web sayfalarının görüntülenmesini sağlayan protokoldür. Açılımı "Hyper Text Transfer Protocol" yani "Hiper Metin Transfer Protokolü"dür
<b>HTTPS</b>	HTTPS (İngilizce Secure Hypertext Transfer Protocol, Türkçe güvenli hiper metin aktarım iletişim kuralı) hiper metin aktarım iletişim

	kuralının (HTTP) güvenli ağ protokolü ile birleştirilmiş olanıdır. Klasik HTTP protokolüne SSL protokolünün eklenmesi ile elde edilir.
<b>RESTFUL Web Service</b>	REST, servis yönelimli mimari üzerine oluşturulan yazılımlarda kullanılan bir veri transfer yöntemidir. HTTP üzerinde çalışır ve diğer alternatiflere göre daha basittir, minimum içerikle veri alıp gönderdiği için de daha hızlıdır. İstemci ve sunucu arasında XML veya JSON verilerini taşıyarak uygulamaların haberleşmesini sağlar.
<b>OAuth 2.0 Protokolü</b>	OAuth 2.0 uygulamaların birbirleri arasında veri erişimine olanak sağlayan açık bir yetkilendirme protokolüdür.
<b>Yönetici</b>	Akıllı otopark yazılımını kullanacak otoparktan sorumlu kişi.
<b>Sürücü</b>	Aracıyla otoparka giriş yapan kişi.
<b>Database</b>	Veri tabanı (Database), verilerin biriktirildiği alandır.
<b>Server</b>	Web sunucusudur.

#### 1.4. Genel Açıklamalar

Bölümler:

Bölüm 1. Giriş: Projenin tanıtımı açıklanmıştır.

Bölüm 2. Tanımlamalar: Projenin işleyişi ve hizmeti, hangi servislerin kullanılacağı, yazılım ürününün neleri etkileyeceği amaçlanmıştır

Bölüm 3. Spesifik Gereksinimler: Bu bölümde geliştirilecek projede olması istenen özellikler bölümlere ayrılarak açıklanmıştır. Sistem içerisinde sistemin yetenekleri işlevsel olan ve olmayan gereksinimleri açıklamıştır.

## 2. TANIMLAMALAR

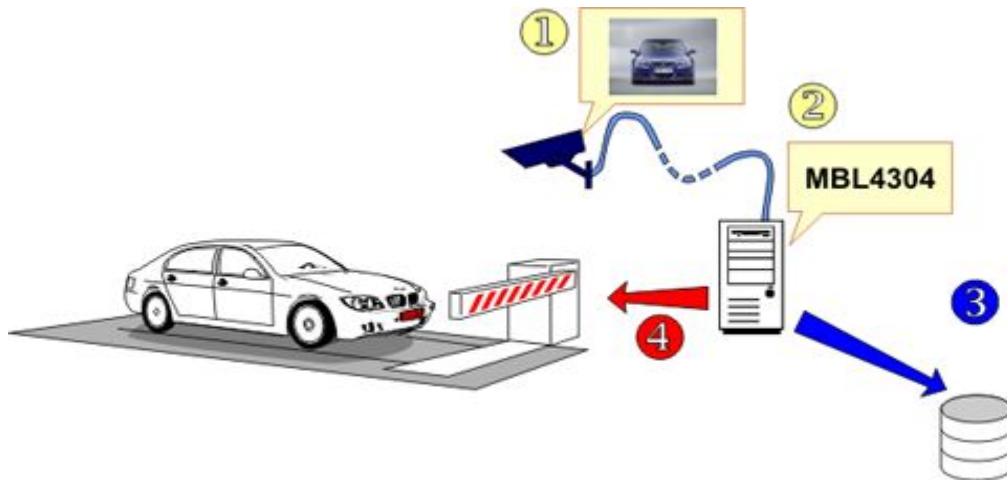
### 2.1. Ürün Perspektifi



Kullanıcının, park yeri aramadan sadece girişte plaka tanıma süreci haricinde ekstra bekleme yapmasına gerek duymaksızın otopark veri tabanında yer alan “müsait” park alanlarına geçiş evresi diyagramda gösterilmiştir. Kullanıcıyı “müsait” park alanını göstermeyi ve ulaştırmaya yardım eden modül ise navigasyondur.

#### 2.1.1. Sistem Arayüzleri

Sürücü'nün kullandığı aracın veri tabanına girdi sağlama ve tutma işlemidir. İlk etapta araç otopark bariyerinde durduğunda kamerada yer alan sistem sayesinde veri tabanında daha önce gelip gelmediği konusunda sorgulama yapar, eğer ilk defa geliyorsa gelen plaka bilgisiyle yeni bir oluşturulur. Şayet daha önce aynı plakaya dair bilgi varsa kendi alanında güncelleme yapar ve giriş-çıkış saatleri güncellenir.



## 2.1.2. Kullanıcı Arayüzleri

**2.1.2.1** Sürücü sistemi araç içinde herhangi bir ekstra elektronik alete sahip olmadan kullanabilecektir.

**2.1.2.2** Boş park alanları, bariyer açılışı sonrasında binanın tavan kısmındaki led tabelalardan sürücünün bilgisine sunulacak ve verilecek olan rotasyonu takip etmesi yeterli olacaktır.

**2.1.2.3** Yönetici giriş yapan tüm araçları filtreleyip listeleyebilecektir.

**2.1.3.4** Yönetici araçların otoparkta geçirdiği süreyi görüntüleyebilecektir.

## 2.1.3. Donanım Arayüzleri

- **CC Kamera:** Araç plaka takibi için, otopark girişinde veri tabanına veri girişi yapmak için kullanılacak olan cihazdır.
- **LED Tabela:** Sürücüye boş park alanına yönlendirme için kullanılacak olan görsel bileşenlerdir.
- **Açılır-kapanır bariyer:** Araçlardan plaka bilgisi alınması için doğru yer ve konumda olmaları için konulmuş engellerdir. Girişte plaka bilgisi alındıktan sonra açılır. Çıkışta ise ücret tahsil edildikten sonra açılır.
- **Harddisk:** Veri tabanındaki verileri tutmak için minimum 50 GB'lık disk.

(Yoğunluk ve trafiğe göre artırılması uygun görülür)

## 2.1.4. Yazılım Arayüzleri

Kullanıcı sisteme web arayüzünden ulaşabilecektir. Kameralardan gelen park verileri bir masaüstü programı aracılığıyla, web servera gönderilecektir.

## 2.1.5. İletişim Arayüzleri

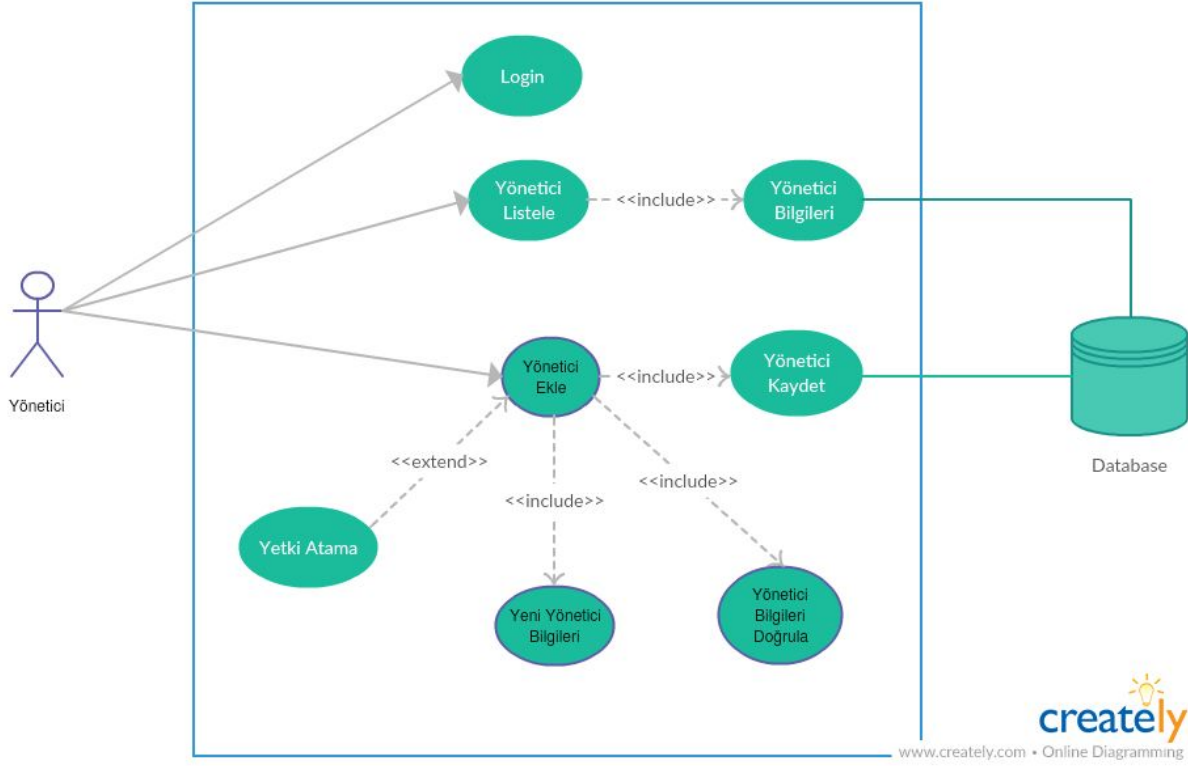
RESTful web servisler ile HTTPS güvenli iletişim protokolü üzerinden sunucu ve client yazılımları birbirleri ile haberleşir.

## 2.1.6. Operasyonlar

Sistem, veri tabanı yedeklerini günlük olarak alır ve gerekli olduğu durumlarda geri yükleme yapılır. Bu sayede sistemimizin zarar görmesini engellemiş olacağız.

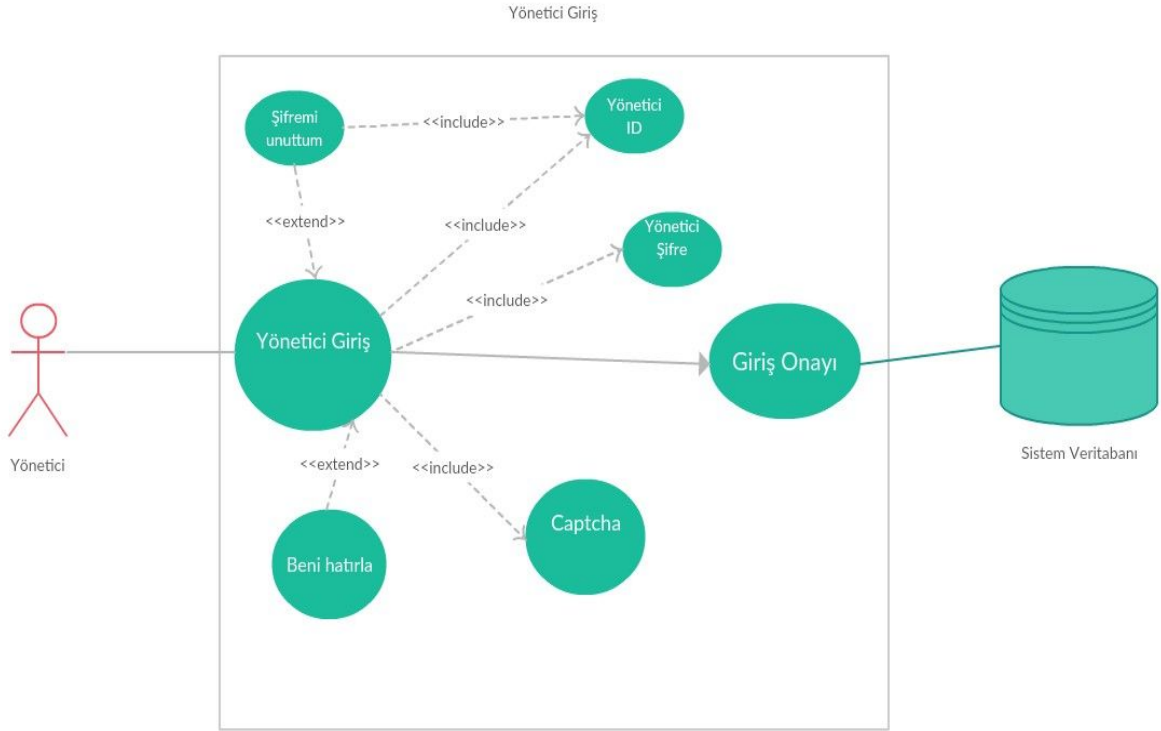
## 2.2. Ürün Fonksiyonları

### 2.2.1. Use Case 1 : Yönetici Ekleme (Emre SERBES)

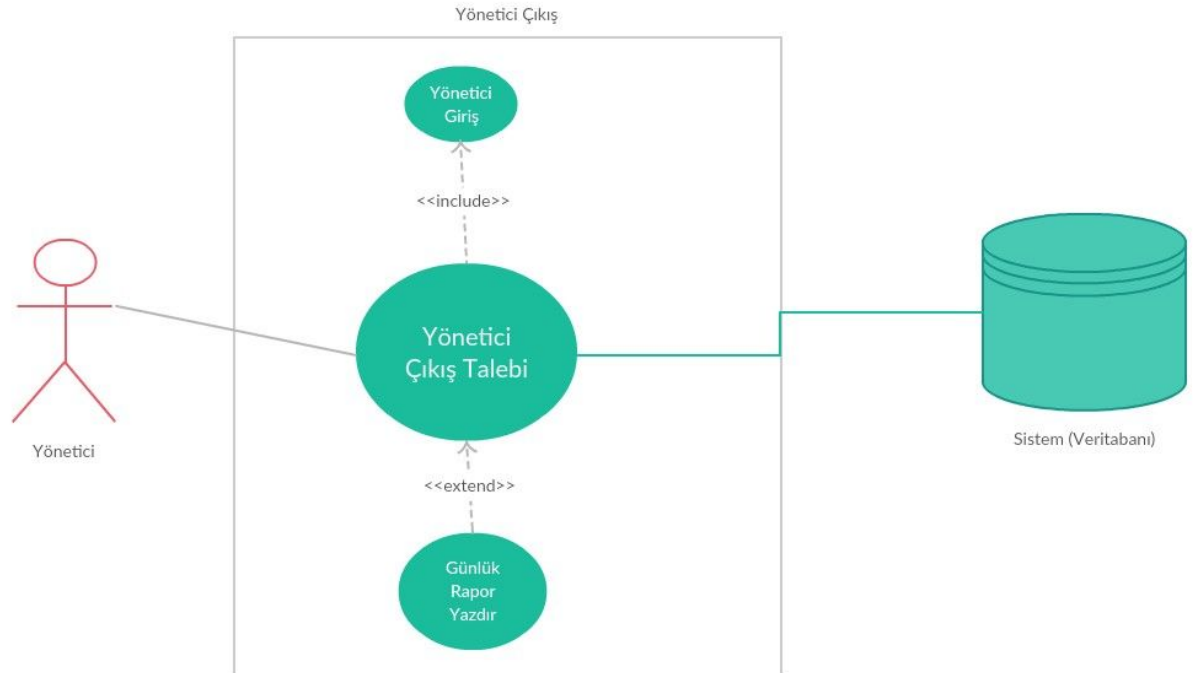




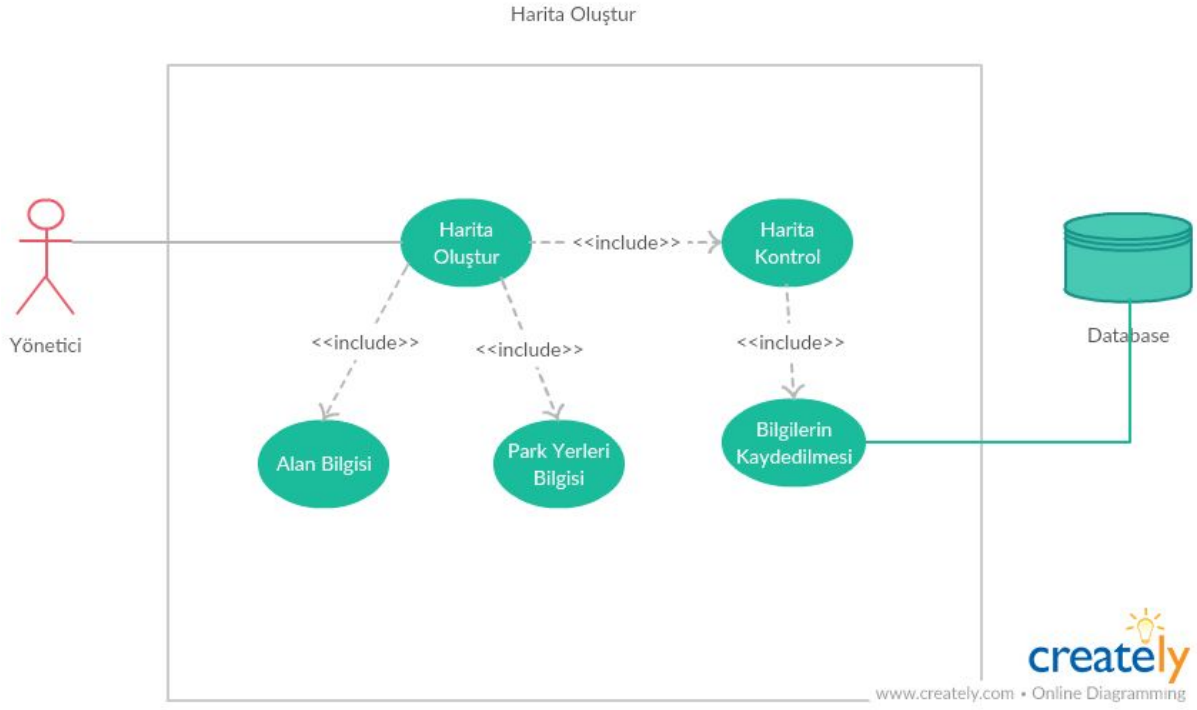
### 2.2.2. Use Case 2 : Yönetici Girişi



### 2.2.3. Use Case 3: Yönetici Çıkışı

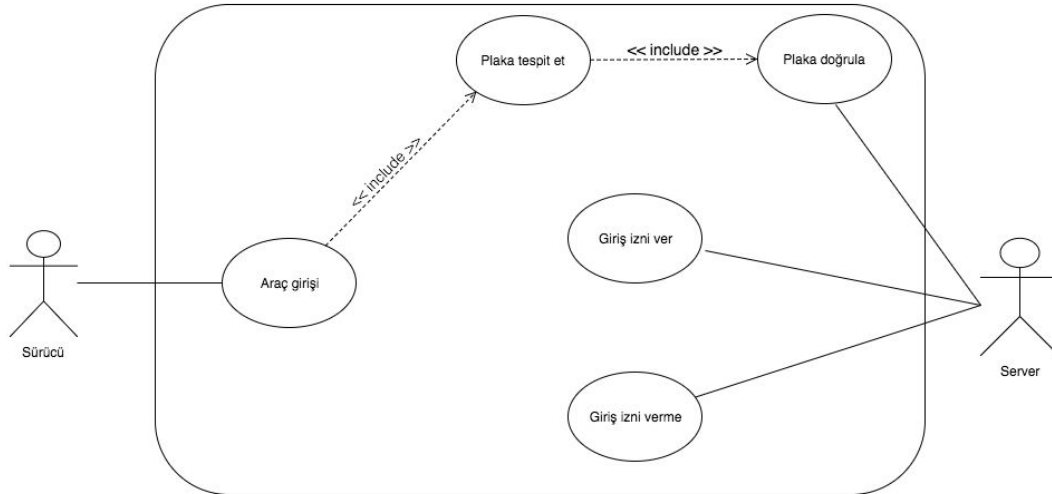


## 2.2.4. Use Case 4 : Harita Oluřturma (Emre SERBES)

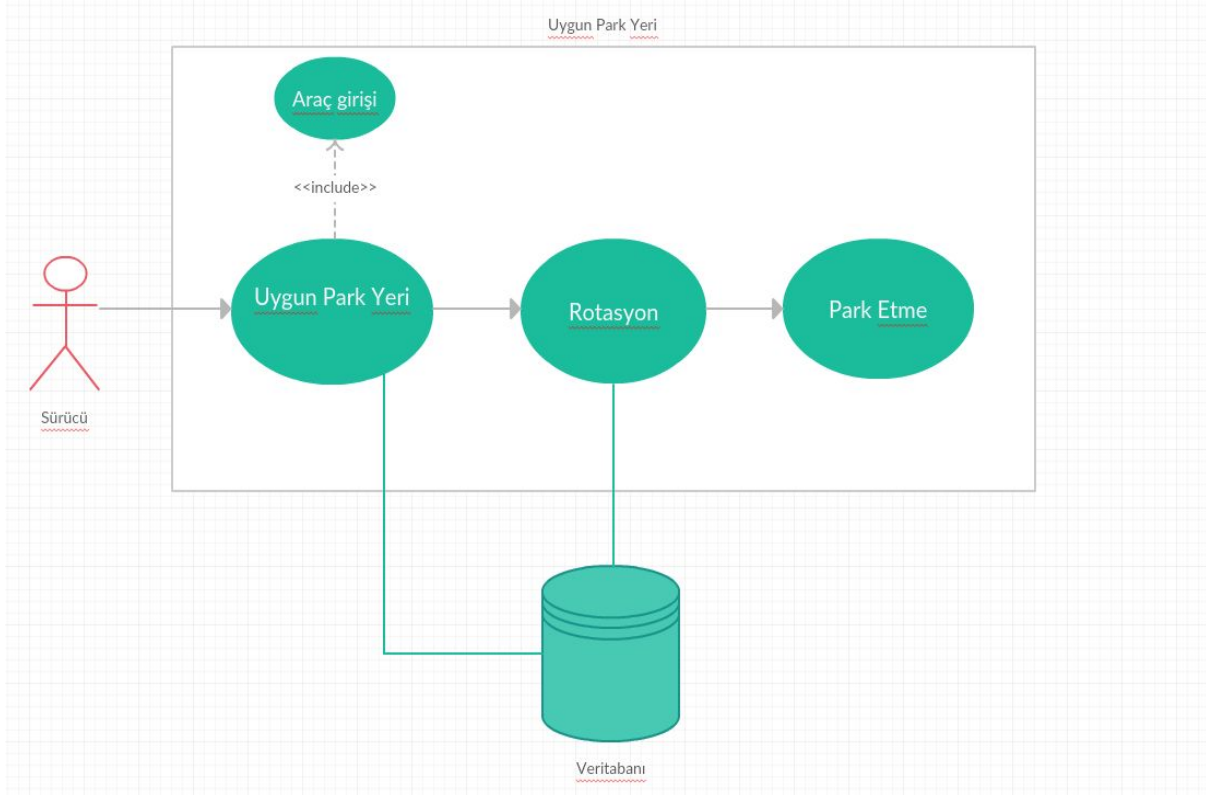


## 2.2.5. Use Case 5 : Araç Giriři

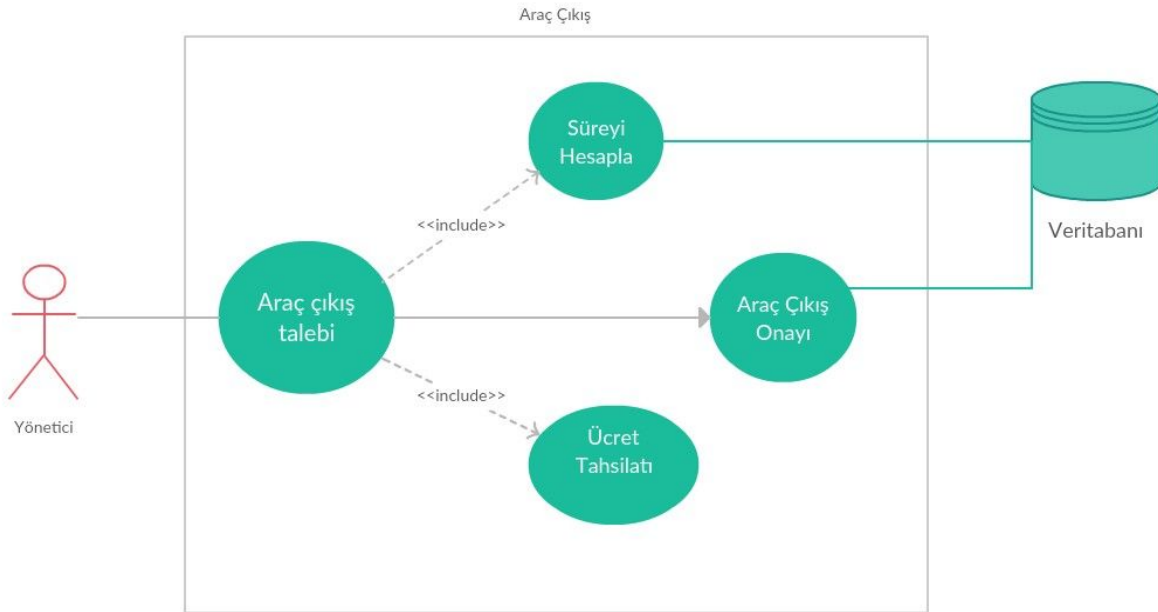
### ARAÇ GİRİŐİ



## 2.2.6. Use Case 6 : Uygun Park Yeri Bulma

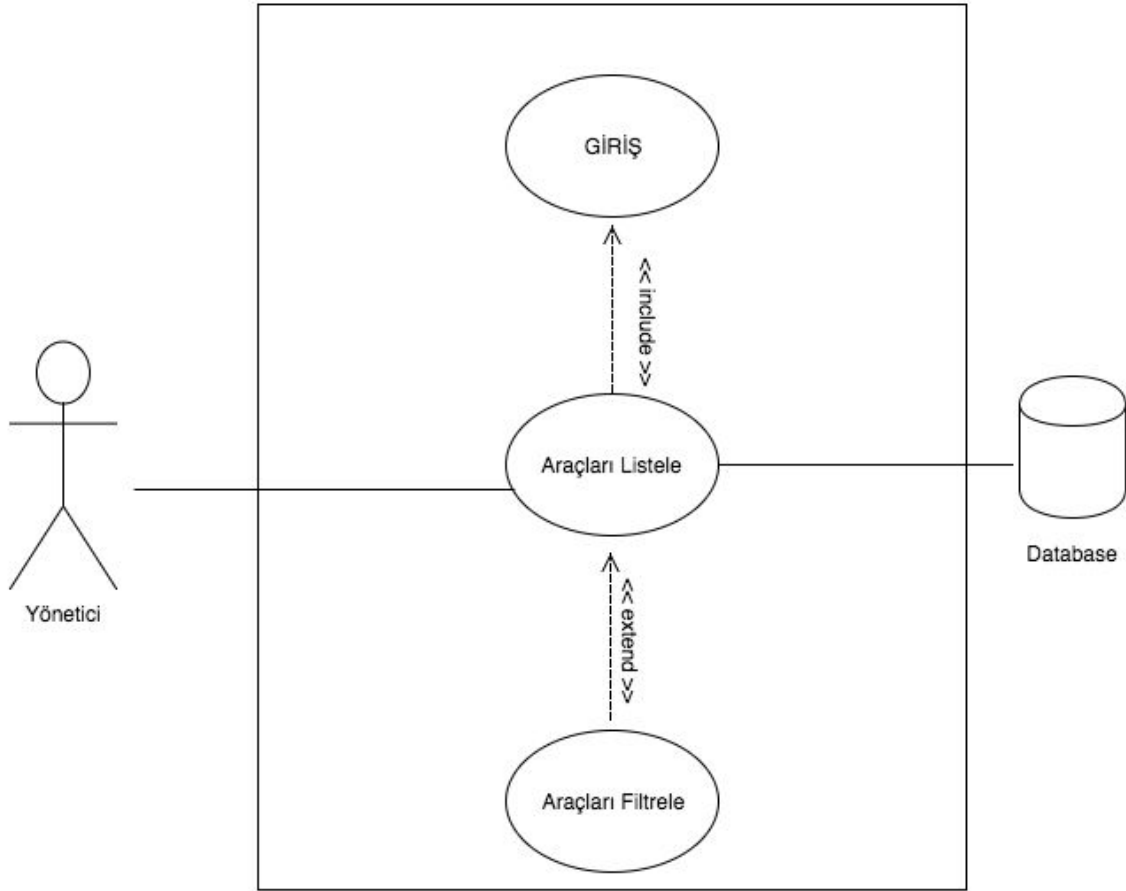


## 2.2.7. Use Case 7 : Araç çıkış ve ücretlendirme



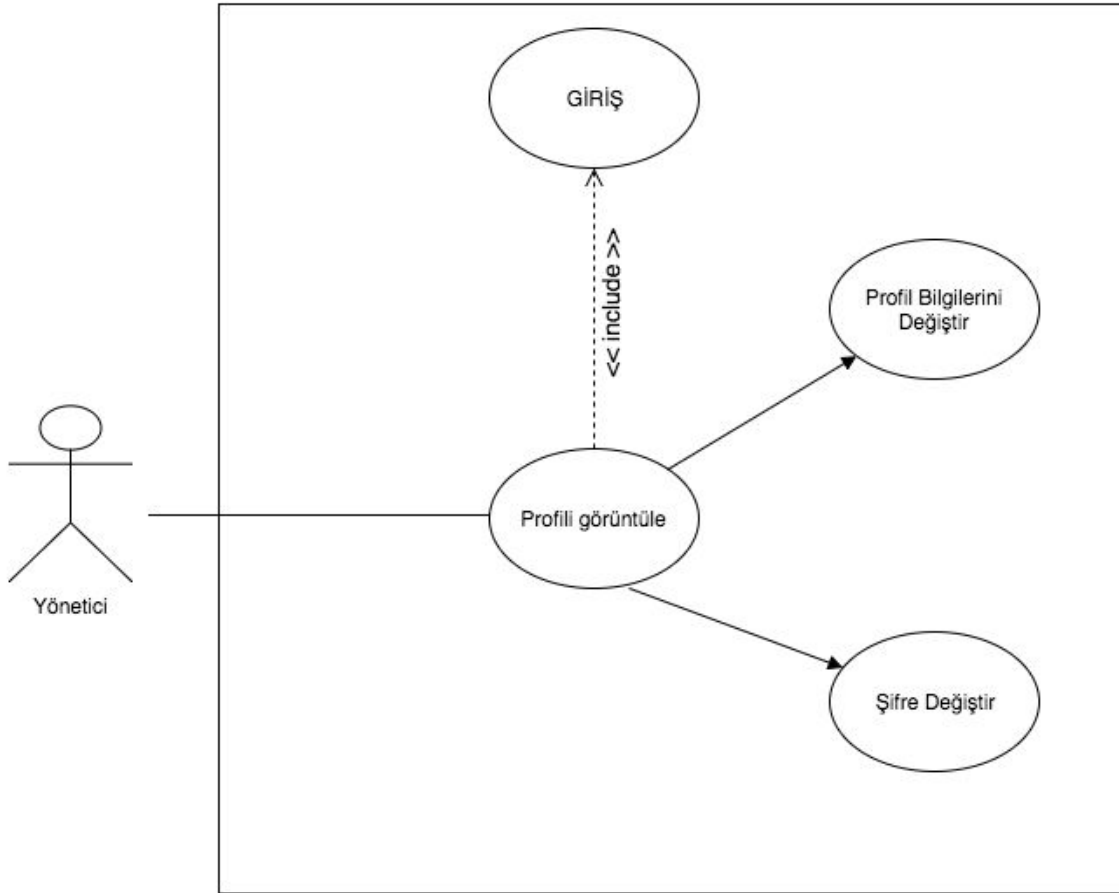
## 2.2.8. Use Case 8 : Araçları Listeleme

### Araçları Listeleme

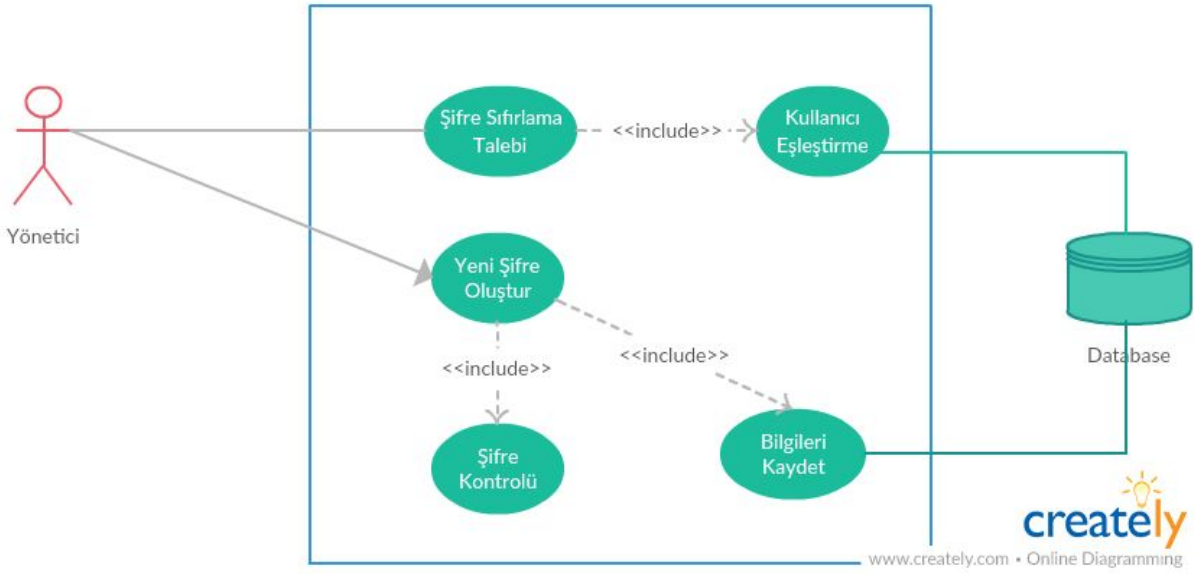


## 2.2.9. Use Case 9 : Yönetici Güncelleme

### Yönetici Güncelleme



## 2.2.10. Use Case 10 : Yönetici Şifre Sıfırlama



## 2.3. Kısıtlamalar

Otopark yöneticilerinin sistemi giriş yapabilmesi için kullanıcı adı ve şifre zorunlu tutulmuştur. Bir güvenlik kısıtlaması olarak, alt sistemlerden herhangi birinin herhangi bir donanım bileşeni başarısız olursa, kullanıcı Akıllı Otopark Sistemini etkinleştiremeyecektir. Elektrik veya internet olmaması durumunda sistem kullanılamayacaktır. Kötü amaçlı kullanımları önlemeye yardımcı olmak için OAuth2 protokolü kullanılacaktır.

## 3. Spesifik Gereksinimler

### 3.1. Fonksiyonel gereksinimler

Otoparka girecek tüm araçların plakalarını kamera ile tespit edip, uygun park yerine yönlendirmeli ve tüm verileri geriye dönük veri tabanında saklamalıdır.

### 3.1.1.

<b>Use Case Name:</b>	<b>Yönetici ekle</b>
<b>Author</b>	Emre Serbes
<b>Actor</b>	Yönetici
<b>Priority</b>	Medium
<b>Description</b>	Bu use case, yöneticinin yeni bir yönetici oluşturmalarını içermektedir.
<b>Trigger</b>	Yönetici, yönetici ekleme talebinde bulunur.
<b>Precondition</b>	Yönetici giriş yapmış olmalıdır ve gerekli izinler verilmiş olmalıdır
<b>Basic Path</b>	<ol style="list-style-type: none"><li>1. Yönetici, yeni yönetici ekleme isteğini belirtir.</li><li>2. Yönetici, sisteme yeni yönetici bilgilerini girer.</li><li>3. Sistem aynı bilgilere sahip başka bir yönetici olup olmadığını kontrol eder.</li><li>4. Yönetici, bilgilerin doğruluğunu onaylar.</li><li>5. Sistem yöneticiyi kaydeder.</li><li>6. Yönetici sistemden çıkar.</li></ol>
<b>Alternate Path</b>	<p>3a. Aynı bilgilere sahip kullanıcı varsa. Sisteme giriş yapılması istenir.</p> <p>3a1.Eğer parolayı unutmuşsa Şifre Sıfırlama devreye girer.</p>
<b>Postcondition</b>	<ol style="list-style-type: none"><li>1. Yönetici, sisteme yeni bir yönetici eklemiş olacak.</li></ol>
<b>Exception Path</b>	<p>1a. Yöneticinin gerekli izinleri yoktur</p> <p>Sistem gerekli izinlerin olmadığını yöneticiye bildirir.</p>
<b>Other</b>	Yönetici bilgileri eklenirken zorunlu alanların sonunda (*) işareti bulunmalıdır.
<b>Reference</b>	<b>SRS 3.1.1</b>

### 3.1.2.

<b>Use Case Name:</b>	<b>Yönetici giriş</b>
<b>Author</b>	Burak FIRÇASIGÜZEL
<b>Actor</b>	Yönetici
<b>Priority</b>	Medium
<b>Description</b>	Sistemi yönetmek ve düzenlemeler yapmak için, yöneticinin sisteme giriş yapması
<b>Trigger</b>	Yönetici sisteme giriş isteği yapar
<b>Precondition</b>	Sistemin anasayfası yüklenmelidir
<b>Basic Path</b>	<ol style="list-style-type: none"><li>1. Yönetici, yönetici ID ve şifre bilgisini sisteme bildirir.</li><li>2. Sistem, girilmiş olan bilgileri kontrol eder ve yöneticiyi, sistem ekranına yönlendirir.</li></ol>
<b>Alternate Path</b>	<ol style="list-style-type: none"><li>3. Şifresini unutan yönetici, şifremi unuttum bölümünden yeni şifre belirleyebilir.</li><li>4. Beni hatırla seçeneğini işaretleyerek, bir sonraki girişte yönetici bilgi ve şifreleri otomatik olarak hatırlanır</li></ol>
<b>Postcondition</b>	Yönetici giriş onayı sonrası, sisteme giriş yapmış olur.
<b>Exception Path</b>	N/A
<b>Other</b>	<ol style="list-style-type: none"><li>1. Yönetici ID Türkçe karakter içeremez</li><li>2. Şifre 5 haneden az olamaz</li></ol>
<b>Reference</b>	<b>SRS 3.1.2</b>



### 3.1.3.

<b>Use Case Name:</b>	<b>Yönetici çıkış</b>
<b>Author</b>	Burak FIRÇASIGÜZEL
<b>Actor</b>	Yönetici
<b>Priority</b>	Medium
<b>Description</b>	Yöneticinin sistem dışına çıkma olayıdır
<b>Trigger</b>	Yönetici sisteme giriş yapmış olmalıdır
<b>Precondition</b>	Sistemin anasayfası yüklenmelidir
<b>Basic Path</b>	<ol style="list-style-type: none"><li>1. Yönetici, çıkış yapma isteğini sisteme bildirir. Yönetici bu isteğini yönetici ekranından yapar.</li><li>2. Sistem, çıkış işlemi için onay ister.</li><li>3. Yönetici çıkış işlemine onay verir.</li><li>4. Sistem, girilmiş olan bilgileri kontrol eder ve yöneticinin çıkış işlemini gerçekleştirerek süreci sonlandırır.</li><li>5. Yönetici isterse günlük otopark-araç raporurunu görebilir ve yazdırabilir.</li></ol>
<b>Alternate Path</b>	N/A
<b>Postcondition</b>	<ol style="list-style-type: none"><li>1. Yönetici çıkış onayı sonrası, sistemden çıkış yapmış olur.</li></ol>
<b>Exception Path</b>	<ol style="list-style-type: none"><li>1. Yöneticinin çıkış işlemine onay vermemesi durumu: Sistem yöneticinin çıkış işlemini gerçekleştirmez</li></ol>
<b>Other</b>	
<b>Reference</b>	<b>SRS 3.1.3</b>

### 3.1.4.

<b>Use Case Name:</b>	<b>Harita oluştur</b>
<b>Author</b>	Emre Serbes
<b>Actor</b>	Yönetici
<b>Priority</b>	High
<b>Description</b>	Bu use case, Yöneticinin harita oluşturmalarını ve park yerlerini belirlemesini içermektedir.
<b>Trigger</b>	Yönetici harita oluşturma isteğini belirtir
<b>Precondition</b>	Yönetici girişi yapılmış olmalıdır ve gerekli izinler verilmiş olmalıdır
<b>Basic Path</b>	<ol style="list-style-type: none"><li>1. Yönetici, yeni yönetici oluşturma isteğini belirtir.</li><li>2. Yönetici sisteme yeni yönetici bilgilerini girer.</li><li>3. Sistem, daha önce böyle bir yönetici olup olmadığını kontrol eder.</li><li>4. Bariyer ve yolları belirtir.</li><li>5. Yönetici harita doğruluğunu onaylar.</li><li>6. Sistem haritayı kaydeder.</li><li>7. Yönetici sistemden çıkar.</li></ol>
<b>Alternate Path</b>	N/A
<b>Postcondition</b>	<ol style="list-style-type: none"><li>1. Yönetici, sistem için bir haritaya sahip olacak.</li><li>2. Park yerleri belirlenmiş olacak.</li></ol>
<b>Exception Path</b>	<ol style="list-style-type: none"><li>1a. Yöneticinin gerekli izinleri yoktur Sistem gerekli izinlerin olmadığını yöneticiye bildirir.</li><li>2a. Otopark alanı girilmemiştir Sistem, alan belirlenmeden işlem yapılamayacağını bildirir.</li><li>3a. Park yeri belirlenmemiştir Sistem, park yeri belirlenmeden işlem yapılamayacağını bildirir.</li><li>4a. Bariyer ve yollar girilmemiştir Sistem, bariyer ve yolların belirlenmeden işlem yapılamayacağını bildirir.</li><li>6a. Sistem hatası oluşma durumu Sistem, yöneticiye hata mesajını sunar.</li></ol>
<b>Other</b>	
<b>Reference</b>	<b>SRS 3.1.4</b>

### 3.1.5.

<b>Use Case Name:</b>	<b>Araç girişi</b>
<b>Author</b>	Eray Akartuna
<b>Actor</b>	Sürücü
<b>Description</b>	Bu use case, sürücünün aracıyla otoparkına girişiyle ilişkin senaryoları ve diğer bir takım ayrıntıları içermektedir.
<b>Priority</b>	High
<b>Trigger</b>	Sürücünün araç ile otopark girişine gelmesi
<b>Precondition</b>	Sürücünün aracında ön plaka olmalıdır. Araç bariyerlere yaklaşmış olmalıdır.
<b>Basic Path</b>	<ol style="list-style-type: none"><li>1. Bu use case, sürücünün otopark bariyerine yaklaşmasıyla başlar.</li><li>2. Sistem kamera aracılığıyla plakayı tespit eder.</li><li>3. Tespit edilen plaka sisteme iletilir.</li><li>4. Sistem plakayı doğrular.</li><li>5. Sistem plakayı veri tabanına kaydeder.</li><li>6. Sistem sürücüye geçiş izni verir.</li></ol>
<b>Alternate Path</b>	N/A
<b>Postcondition</b>	<ol style="list-style-type: none"><li>1. Araç otoparka giriş yapmış olur.</li><li>2. Giriş yapan araç sisteme kaydedilmiş olur.</li></ol>
<b>Exception Path</b>	4a. Sistem plakayı doğrulayamaz veya okuyamaz. Sistem hata mesajı verir ve geçiş izni tanımaz.
<b>Other</b>	Hata mesajının verileceği ekran sürücünün rahat görebileceği bir yere konulur.
<b>Reference</b>	<b>SRS 3.1.5</b>

### 3.1.6.

<b>Use Case Name:</b>	<b>Uygun park yeri</b>
<b>Author</b>	Burak FIRÇASIGÜZEL
<b>Actor</b>	Sürücü
<b>Priority</b>	High
<b>Description</b>	Otoparka gelen aracın, otopark hizmetinden yararlanması için, en uygun park alanının bulunması
<b>Trigger</b>	Araç girişi yapılması
<b>Precondition</b>	Araç girişi yapılmış olmalıdır
<b>Basic Path</b>	<ol style="list-style-type: none"><li>1. Araç girişi,sonrasında giriş yapan araç plakasına göre en uygun park yeri bulunur.</li><li>2. Premium sürücüler, boşta olan çıkışa yakın park alanlarına yönlendirilir. Bu önceliktir.</li><li>3. veri tabanındaki park alanı eşleşmesi sonucunda en uygun park yeri rotasyon bilgisine sunulur</li><li>4. Rotasyon, uygun park yeri için rotayı çizer.</li><li>5. Led tabelalardan verilen rotalar ile birlikte, park alanına ulaşılır</li><li>6. Park işlemi yapılır</li></ol>
<b>Alternate Path</b>	<ol style="list-style-type: none"><li>1. Sürücü için park yeri, çıkışa en yakın park alanlarından bir tanesiye rota orayı çizer.Premium opsiyonu aranmaz.</li><li>2. Premium park alanlarından çıkan araçların yerine, herhangi bir araç gelebilir. Premium opsiyonu aranmaz.</li></ol>
<b>Postcondition</b>	Park işlemi yapılır
<b>Exception Path</b>	<ol style="list-style-type: none"><li>3.a. veri tabanı çökmesi durumunda hizmet durması</li><li>5.a. Led tabelaların zaman aşımından piksel bozulması</li></ol>
<b>Other</b>	<ol style="list-style-type: none"><li>1. Yönetici ID Türkçe karakter içeremez</li><li>2. Şifre 5 haneden az olamaz</li></ol>
<b>Reference</b>	<b>SRS 3.1.6</b>

### 3.1.7.

<b>Use Case Name:</b>	<b>Araç çıkış ve ücretlendirme</b>
<b>Author</b>	Burak FIRÇASIGÜZEL
<b>Actor</b>	Yönetici
<b>Priority</b>	Medium
<b>Description</b>	Otoparktan ayrılmak isteyen aracın, otoparktan ayrılışı ve ücretlendirme
<b>Trigger</b>	Araç çıkış talebinde bulunur
<b>Precondition</b>	Yönetici sistem içinde olması gerekir.
<b>Basic Path</b>	<ol style="list-style-type: none"><li>1. Araç, otoparktan çıkma talebini sisteme bildirir.</li><li>2. Yönetici araç plaka bilgileri ile veri eşleşmesi yapar.</li><li>3. veri tabanından elde edilen bilgilerle, ne kadar süre otoparkta olduğu hesaplanır.</li><li>4. Veriler, veri tabanında yedeklenir.</li><li>5. Hesaplanan süre üzerinden ücretlendirme uygulanır.</li><li>6. Yönetici ücreti tahsil eder</li><li>7. Araç çıkışına onay verilir.</li><li>8. Araç otoparktan çıkar.</li></ol>
<b>Alternate Path</b>	<ol style="list-style-type: none"><li>2.a. Araç “premium” opsiyonuna sahiptir.</li><li>2.b. Ücretlendirme “premium” opsiyonuna göre yapılır.</li><li>2.c.a. Ücret “premium” opsiyonuna göre alınır.</li><li>2.c.b. Premium olarak girilen aracın opsiyonu, otoparkta olduğu süresince bitmiştir.</li></ol>
<b>Postcondition</b>	<ol style="list-style-type: none"><li>1. Araç, otopark hizmeti sonrası otoparktan ayrılacak.</li></ol>
<b>Exception Path</b>	<ol style="list-style-type: none"><li>6.a Yönetici ücreti tahsil edemez, araç otoparktan çıkamaz</li><li>7.a Araç çıkışına onay verilmez</li></ol>
<b>Other</b>	
<b>Reference</b>	<b>SRS 3.1.7</b>

### 3.1.8.

<b>Use Case Name:</b>	<b>Araçları görüntüleme</b>
<b>Author</b>	Eray Akartuna
<b>Actor</b>	Yönetici
<b>Description</b>	Bu use case, yöneticinin araçları listelemesi ve görüntülemesiyle ilgili senaryoları ve diğer bir takım ayrıntıları içermektedir.
<b>Priority</b>	Medium
<b>Trigger</b>	Yöneticinin araçları listele butonuna basması
<b>Precondition</b>	Yönetici giriş yapmış olmalıdır.
<b>Basic Path</b>	<ol style="list-style-type: none"><li>1. Bu use case, yöneticinin araçları listele butonuna basmasıyla başlar.</li><li>2. Sistem araçların listelenebilmesi için tarih aralığı filtresi getirir.</li><li>3. Yönetici tarih aralıklarını girer.</li><li>4. Sistem belirlenmiş tarih aralığındaki araç bilgilerini getirir.</li><li>5. Yönetici görüntülemek istediği aracı seçer.</li></ol>
<b>Alternate Path</b>	3a Yönetici tarih aralıklarını boş geçer. Sistem otoparkta bulunan mevcut araçları getirir.
<b>Postcondition</b>	<ol style="list-style-type: none"><li>3. Araç otoparka giriş yapmış olur.</li><li>4. Giriş yapan araç sisteme kaydedilmiş olur.</li></ol>
<b>Exception Path</b>	4a. Sistem herhangi bir araç bilgisi getiremez. Sistem ekrana “Belirlediğiniz tarih aralığında herhangi bir araç bulunmamaktadır.” mesajını getirir.
<b>Other</b>	Girilen tarih aralıkları liste sayfasının üzerinde yazılı olmalıdır. Liste sayfasında araç plakaları ve hangi tarihlerde girdiği yazılı olmalıdır.
<b>Reference</b>	<b>SRS 3.1.8</b>

### 3.1.9.

<b>Use Case Name:</b>	<b>Profil düzenleme</b>
<b>Author</b>	Eray Akartuna
<b>Actor</b>	Yönetici
<b>Description</b>	Bu use case, yöneticinin profil bilgilerini düzenlemesiyle ilgili senaryoları ve diğer bir takım ayrıntıları içermektedir.
<b>Priority</b>	Low
<b>Trigger</b>	Yöneticinin profilimi görüntüle butonuna basması.
<b>Precondition</b>	Yönetici giriş yapmış olmalıdır.
<b>Basic Path</b>	<ol style="list-style-type: none"><li>1. Bu use case, yöneticinin profilimi görüntüle butonuna basmasıyla başlar.</li><li>2. Yönetici profil bilgilerini görür.</li><li>3. Yönetici profil bilgilerimi düzenle butonuna basar.</li><li>4. Yönetici değiştireceği profil bilgilerini girer.</li><li>5. Yönetici profil bilgilerimi güncelle butonuna basar.</li><li>6. Sistem yeni profil bilgilerini doğrular.</li><li>7. Yeni bilgiler veri tabanına kaydolur.</li><li>8. Sistem kullanıcıyı profil sayfasına yönlendirir.</li><li>9. "Profil bilgileriniz başarıyla güncellendi." mesajı ekrana yazılır.</li><li>10. Kullanıcı şifremi değiştir butonuna basar.</li><li>11. Kullanıcı mevcut şifresini girer.</li><li>12. Kullanıcının mevcut şifresi doğrulanır</li><li>13. Kullanıcı yeni şifresini girer.</li><li>14. Yeni şifre doğrulanır.</li><li>15. Sistem kullanıcıyı profil sayfasına yönlendirir.</li><li>16. "Şifreniz başarıyla güncellendi." mesajı ekrana yazılır</li></ol>
<b>Alternate Path</b>	N/A
<b>Postcondition</b>	
<b>Exception Path</b>	<p>6a Sistem yeni profil bilgilerini doğrulayamaz. Sistem düzenleme sayfasına döner Sistem ekrana "Profil bilgileriniz doğrulanamadı." mesajı verir</p> <p>12a Kullanıcının mevcut şifresi doğrulanamaz Sistem ekrana "Mevcut şifreniz doğrulanamadı" mesajı verir.</p> <p>14a Yeni şifre doğrulanamaz. Sistem ekrana "Yeni şifreniz doğrulanamadı" mesajı verir.</p>

<b>Other</b>	Profil bilgileri düzenlenirken zorunlu alanların sonunda (*) işareti bulunmalıdır.
<b>Reference</b>	<b>SRS 3.1.9</b>

### 3.1.10.

<b>Use Case Name:</b>	<b>Yönetici şifre sıfırlama</b>
<b>Author</b>	Emre Serbes
<b>Actor</b>	Yönetici
<b>Priority</b>	Medium
<b>Description</b>	Bu use case, yöneticinin unuttuğu şifresini değiştirmeyi içermektedir.
<b>Trigger</b>	Yönetici şifre sıfırlama isteğini belirtir
<b>Precondition</b>	Yönetici girişi denenmiş olmalıdır ve şifre kabul edilmemiş olmalıdır
<b>Basic Path</b>	<ol style="list-style-type: none"> <li>1. Yönetici, şifre sıfırlama isteğini belirtir.</li> <li>2. Yönetici, e-posta adresini girer.</li> <li>3. Sistem, eşleşen kullanıcıya şifre sıfırlama talimatı gönderir.</li> <li>4. Şifre belirtilen kriterlere göre yeniden oluşturulur.</li> <li>5. Yönetici şifrenin doğruluğunu onaylar.</li> <li>6. Sistem yeni şifreyi kaydeder.</li> <li>7. Yönetici sistemden çıkar.</li> </ol>
<b>Alternate Path</b>	2a. Yönetici, ID girer.
<b>Postcondition</b>	1. Yönetici şifresi sıfırlanmış olur.
<b>Exception Path</b>	<p>2a. Sistem yöneticiyi bulamaz Sistem, yöneticiye hata mesajını sunar.</p> <p>4a. Şifre oluştururken istenilen kriterler yerine getirilmez. Sistem, yöneticiye hata mesajını sunar.</p>
<b>Other</b>	
<b>Reference</b>	<b>SRS 3.1.10</b>



### 3.2. Fonksiyonel Olmayan Gereksinimler

**Kullanılabilirlik :** Yönetici ekranında araçlar listelenirken ekranın üst tarafında filtre alanları bulunmalıdır.Yapılacak filtreler ile hızlıca araçları listeleyebilmelidir.Araç sürücüsü ise giriş yaptıktan sonra LED tabelaları direkt fark edebilmelidir.

**Güvenilirlik :** Veri tabanının aylık yedeği alınmalıdır.

**Performans :** Sistem aynı anda binlerce kişiye hizmet verebilecek şekilde tasarlanmıştır.

**Desteklenebilirlik :** Kullanıcılar herhangi bir web browser üzerinden girecekleri kullanıcı adı ve şifre ile gerekli bilgilere ulaşabilecektir.

**Arayüz :** Sistem yazılımlarının ekran çözünürlükleri, çalışacağı cihazın desteklediği maksimum çözünürlükler olacaktır. Sistem arayüzleri buna göre relatif olarak tasarlanacaktır.

**Taşınabilirlik :** Sistem web tarayıcısı bulunan tüm işletim sistemlerinden erişilebilecektir.



**Beykent Üniversitesi**  
**Mühendislik ve Mimarlık Fakültesi**  
**Yazılım Mühendisliği Bölümü**

**SDD Dokümantasyonu**

## **AKILLI OTOPARK SİSTEMİ**

<b>Burak</b>	<b>FIRÇASIGÜZEL</b>	<b>130301022</b>
<b>Emre</b>	<b>SERBES</b>	<b>130301020</b>
<b>Eray</b>	<b>AKARTUNA</b>	<b>130301026</b>

## **İçindekiler :**

### **1. Giriş**

- 1.1 Amaç**
- 1.2 Kapsam**
- 1.3 Genel Bakış**
- 1.4 Referanslar**
- 1.5 Tanımlama ve Kısaltmalar**

### **2. Sistem Genel Tanımlamaları**

### **3. Sistem Mimarisi**

- 3.1 Mimari Tasarım**
- 3.2 Ayrışma Açıklaması**
- 3.3 Tasarım Gerekçesi**

### **4. Veri Tasarımı**

- 4.1 Veri Tanımlama**
- 4.2 Veri Sözlüğü**
- 4.3 Ayrıntılı Veri Tabanı Programı**

### **5. Bileşenlerin Tasarımı**

- 5.1 Giriş**
- 5.2 Context Viewpoint**
- 5.4 Logical Viewpoint**
  - 5.4.1 Ana Sınıf Diyagramı**
  - 5.4.2 Araç Listeleme**
  - 5.4.3 Ücretlendirme Sistemi**
- 5.5 Modül: Camera**
- 5.6 Modül: ParkingSpace**
- 5.7 Modül: Park Controller**
- 5.8 Modül: Car**
- 5.9 Gate Modülü**
  - 5.9.1 Gate**
  - 5.9.2 GateController**
- 5.10 Interaction Viewpoint**
  - 5.10.1 Araç Girişi**
  - 5.10.2 Araç Listeleme**
  - 5.10.3 Ücretlendirme Sistemi**
- 5.11 State Dynamics Viewpoint**

### **6. İnsan Arayüz Tasarımı**

- 6.1 Kullanıcı Arayüzüne Genel Bakış**
- 6.2 Ekran Resimleri**
  - 6.2.1 Anasayfa**

**6.2.2 Araç Giriş i Yap**

**6.2.3 Kroki Oluştur**

**6.2.4 İçerdeki Araçları Listele**

**6.2.5 Geçmiş Kayıtları Listele**

### **6.3 Ekran Objeleri ve Olayları**

**6.3.1 Ana sayfa Ekran Objeleri**

**6.3.2 Araç Giriş i Yap Ekran Objeleri**

**6.3.3 Kroki Oluştur Ekran Objeleri**

**6.3.4 İçerideki Araçları Listele Ekran Objeleri**

**6.3.5 Geçmiş Kayıtları İncele Ekran Objeleri**

## 1.Giriş

### 1.1. Amaç

Bu SDD dokümanı SRS dokümanımızda belirlediğimiz gereksinimlerin entegrasyonunun başarılı bir şekilde sonuçlanması için gerekli açıklamaları bulundurmakta ve yönetim sisteminin yazılım mimarisini ve bunların bileşenlerinin etkileşimini içermektedir. Aynı zamanda yazılım geliştirici ekibe yol göstermesi amaçlanmıştır.

### 1.2. Kapsam

Akıllı otopark yazılımı, bir otoparka giriş yapan tüm araçların kaydedilebileceği, otomatik park yerleşiminin yaptırılabilceği bir yazılım olarak geliştirilecektir. Yazılımın veri tabanı web üzerinde saklanılacaktır. Yazılım ise masaüstü ve web yazılımları sayesinde gerçekleştirilecektir.

### 1.3. Genel Bakış

Akıllı otopark sistemi, otoparklarda ki yer arama problemini çözmek amacıyla kullanılacaktır. Uygulama sayesinde kullanıcı otopark alanının girişinden park etme sürecine kadar çeşitli bileşenler yardımıyla kolayca, aracını en uygun park alanına park edebilecektir. Bu sayede park arama süresini minimuma düşürerek zamandan tasarrufta sağlanmaktadır.

Projenin bir diğer amacı ise park yeri ararken ki geçen süre ve buna bağlı olarak yakılan yakıtın minimum düzeyde tutulup, CO2 emisyon değerini alt basamaklarda tutarak çevre ve doğaya en az tahribatı vermektir.

Bu Yazılım Dizayn Dökümantasyonunda:

- 1.Bölümde Akıllı Otopark Sistemi'nin amaçları ve döküman hakkında genel bilgi verilmiştir.
- 2.Bölümde ilgili sistemin genel tanımlamaları yapılarak çalışma şekli hakkında bilgi verilmektedir.
- 3.Bölümde sistem mimarisinin nasıl dizayn edileceği ve kullanılacak mimari tasarım hakkında detaylı bilgi verilmesi planlanmaktadır.
- 4.bölümde veri dizaynı tanımlamaları , veri tabanı ile ilgili tüm gerekli bilgiler , veri tabanı parametreleri, veri tipleri ile birlikte tanımlanacaktır.
- 5.bölümde sistem tasarımının belirlenmesinde kullanılacak diyagramlar çizilir.
- 6.bölümde insan ara yüz tasarımları yapılacak ekran tasarım dizaynlarının nasıl olacağı belirlenecektir.

### 1.4. Referanslar

Bu belgede, IEEE Software Design Descriptions (IEEE SDD 1016) dokümanından, UML Tutorial Part 1 (Sparks System) ve UML Tutorial Part 2(SparksSystems) belgelerinden referans alınmıştır.

## 1.5 Tanımlama ve Kısaltmalar

JAVA	Açık kodlu, nesneye yönelik, zeminden bağımsız, yüksek verimli, çok işlevli, yüksek seviye, adım adım işletilen (yorumlanan-interpreted) bir dildir.
PHP	Geniş bir kitle tarafından kullanılan, özellikle sanal yöreler üzerinde geliştirme için tasarlanmış HTML içine gömülebilen bir betik dilidir.
API Client	JSON, SOAP , XML gibi iletişim katmanı üreten program ara yüzü
OAuth2	Uygulamaların birbirleri arasında veri erişimine olanak sağlayan açık bir yetkilendirme protokolüdür
MySQL DB	Çoklu iş parçacıklı, çok kullanıcı, hızlı ve sağlam bir veri tabanı yönetim sistemidir
Park Control Machine	Otoparktaki kamera,kapı ve led tabelaları kontrol edecek java tabanlı uygulama.
Gate	Otoparkta bulunan giriş ve çıkış kapıları.

## 2. Sistem Genel Tanımlamaları

Sistem otoparklarda boş park alanı bulma işleminin hızlı olması için üretilmiş, kullanıcıya led tabela sistemi ile direktifler vererek en uygun park yerine götüren sistemdir.

1. Araç otopark bariyerinde durur pozisyonda iken "Akıllı Otopark Sistemi"nin kameraları tarafından plaka algılanır.
1. Web Sunucusu tarafından Otopark Veri tabanı ile bağlantı kurulur.
1. En uygun park yeri seçilir ve plaka o yer bilgisine işlenir.
1. En uygun park yeri için verilen direktifler, led tabeladan kullanıcıya sunulur.
1. Kullanıcı direktifleri takip ederek en uygun park yerine ulaşır.

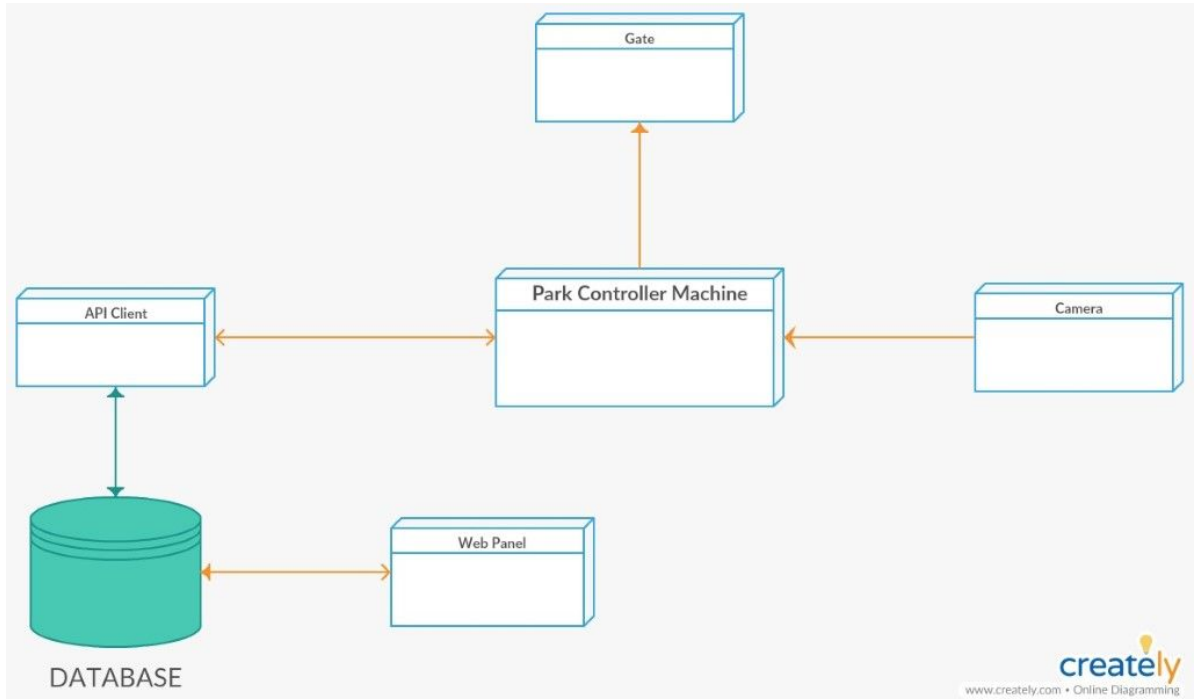
## 3. Sistem Mimarisi

### 3.1. Mimari Tasarım

Projemiz Web Sunucusu, Otopark Veri tabanı ve Masaüstü Ekranı'ndan oluşan 3 ana modülden oluşmaktadır. Otoparkta bulunan masaüstü uygulaması JAVA ile yazılacaktır. Web paneli ve web servisler PHP ile kodlanacak ve mysql veri tabanı kullanılacaktır.

1. Kamerada algılanan plaka otoparkta bulunan masaüstü uygulaması üzerinden web servise gönderilecektir.
1. API Client'a gönderilen plaka veri tabanına kaydedilecektir.
1. API Client araç için uygun park yerini belirleyecektir.
1. Database daima çalışır ve erişebilir halde olması gerekir.
1. Web servise erişim güvenlik gerekçesiyle OAuth2 protokollerine uygun olmalıdır.

### 3.2. Ayrışma Açıklaması



Birimler arası geçişler diyagramda gösterildiği gibi açık ve nettir.

### 3.3 Tasarım Gerekçesi

Tasarımda Nesneye Dayalı yaklaşım seçildi çünkü bu sayede donanım bölümü ve yazılım bölümü kolaylıkla birleştirilecek. Yazılım bölümü, ayrıştırma ve not okuma ve veri iletme içerir. Bu çok sayıda paket tasarlamak için kullanıldı. Bu paketler birbirlerine bağlıdır ve ayrı ayrı kontrol edilebilirler. Ayrıca, donanım kısmı için bir paket kullanılmış ve yazılım ve donanım parçalarını bir araya getiren bir başka paket var.

## 4. Veri Tasarımı

### 4.1. Veri Tanımlama

Projemizde veri tabanı olarak MySQL DB kullanılacaktır. Tüm veriler ilişkisel veri tabanı mantığına uygun bir şekilde kayıt edilecektir. Sistemimizi geliştirmeye uygun tasarladık.

### 4.2. Veri Sözlüğü

#### ADMINS

Admin_id	Int   NN
Admin_firstname	String(55)    NN
Admin_lastname	String(55)    NN
Admin_email	String(55)    NN
Admin_password	String(55)    NN

#### CARS

Car_id	Int   NN
Plate_number	String(25)    NN



## CAR\_PARKINGS

<b>Parking_id</b>	<b>Int   NN</b>
<b>Car_id</b>	<b>Int   NN</b>
<b>Car_entry_date</b>	<b>Datetime   NN</b>
<b>Car_release_date</b>	<b>Datetime</b>
<b>Parking_space_id</b>	<b>Int   NN</b>

## PLANS

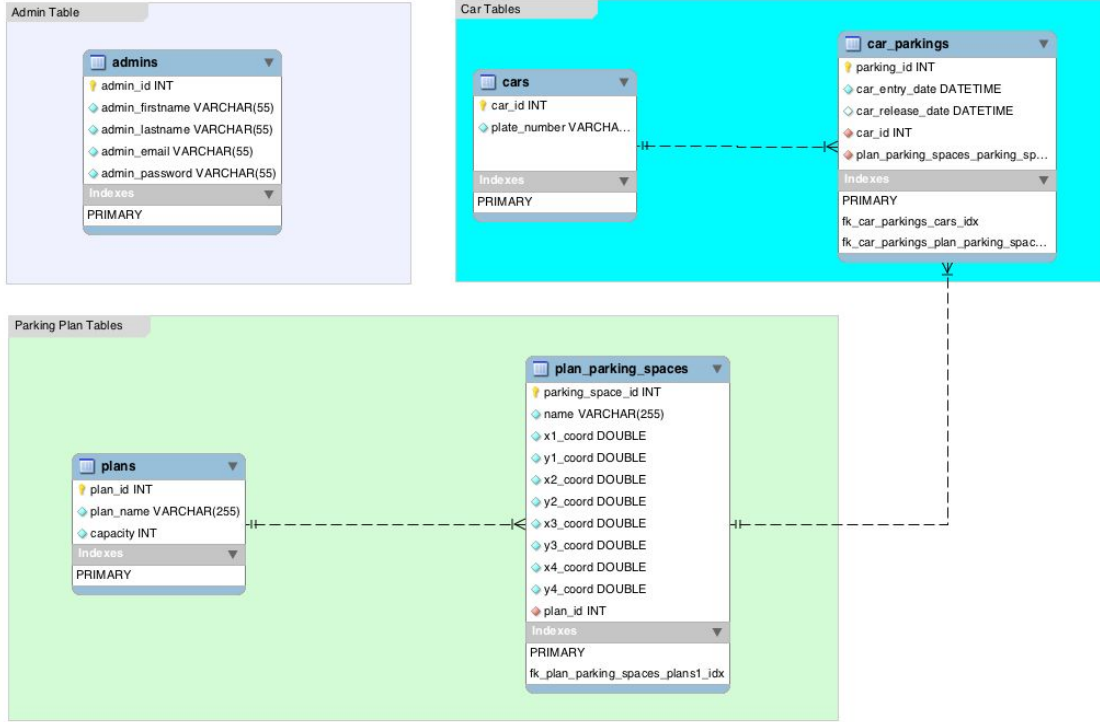
<b>Plan_id</b>	<b>Int   NN</b>
<b>Plan_name</b>	<b>String(255)   NN</b>
<b>Capacity</b>	<b>Int   NN</b>

## PLAN\_PARKING\_SPACES

Parking_space_id	Int   NN
name	String(255)   NN
Plan_id	Int   NN
X1_coord	Double   NN
Y1_coord	Double   NN
X2_coord	Double   NN
Y2_coord	Double   NN
X3_coord	Double   NN
Y3_coord	Double   NN
X4_coord	Double   NN
Y4_coord	Double   NN

\*NN = Not null boş geçilemez.

### 4.3. Ayrıntılı Veri Tabanı Programı

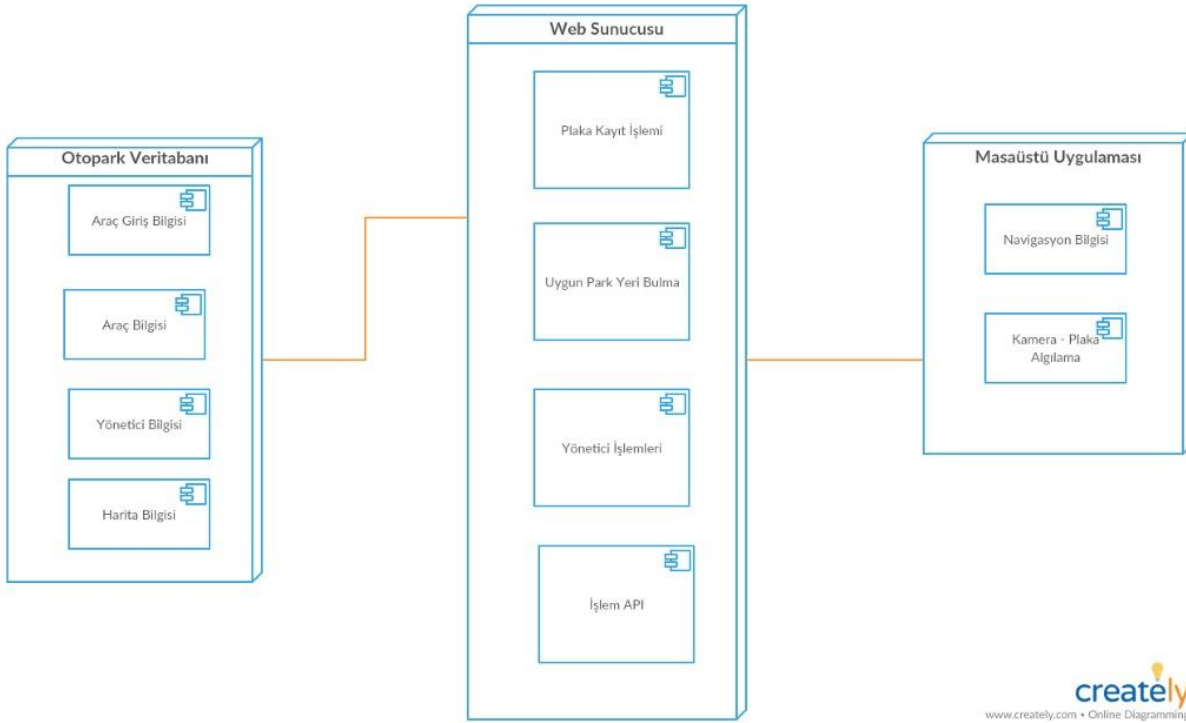


## 5. Bileşenlerin Tasarımı

### 5.1 . Giriş

Bileşenlerin tasarım kısmındaki tanım ve açıklamaları sade ve açık bir dille yazımı amaçlanmıştır. Diyagramlar altında yer alan açıklamalar, diyagramın anlaşılamadığı noktada yol göstericidir.

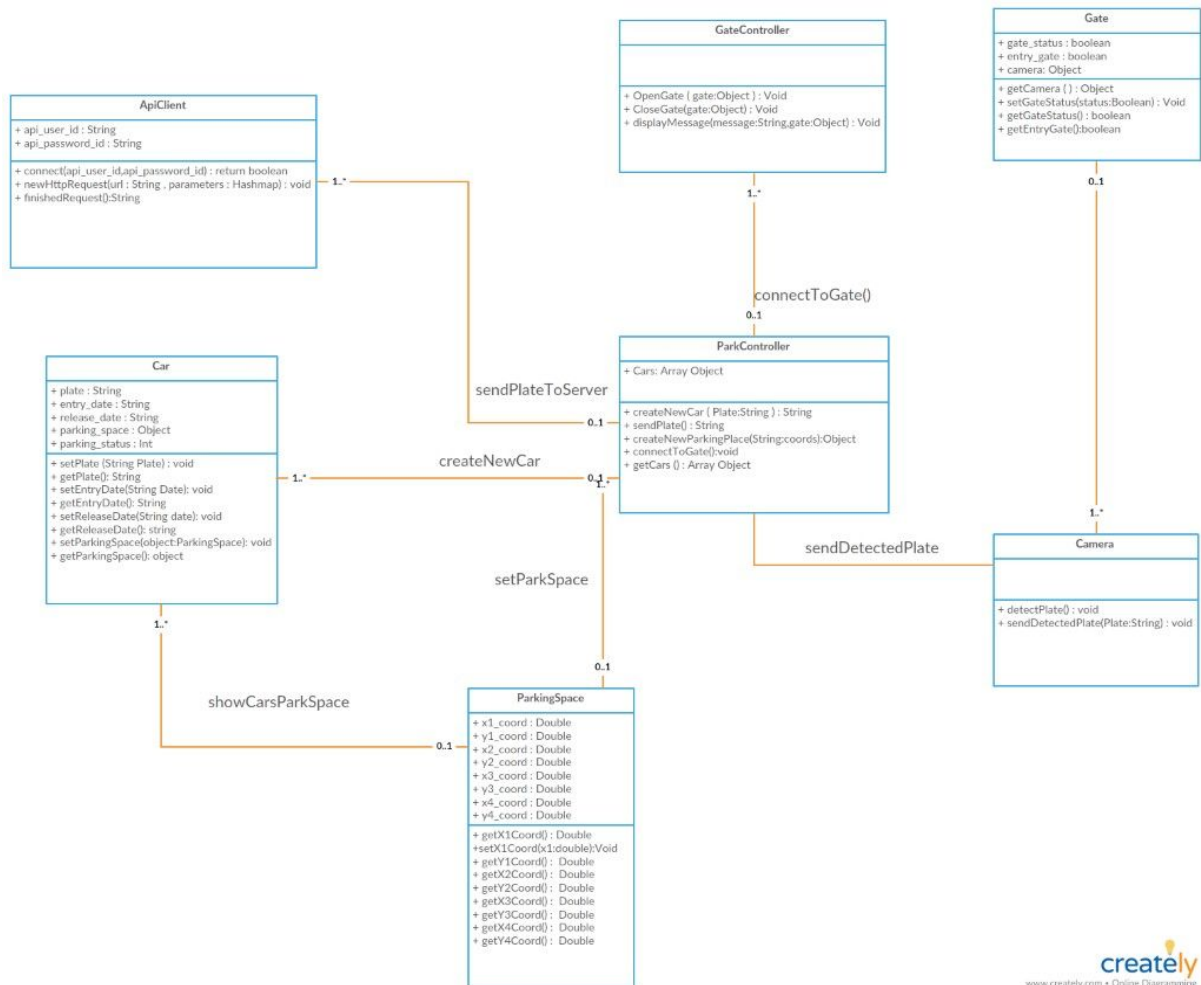
## 5.2. Context View Point



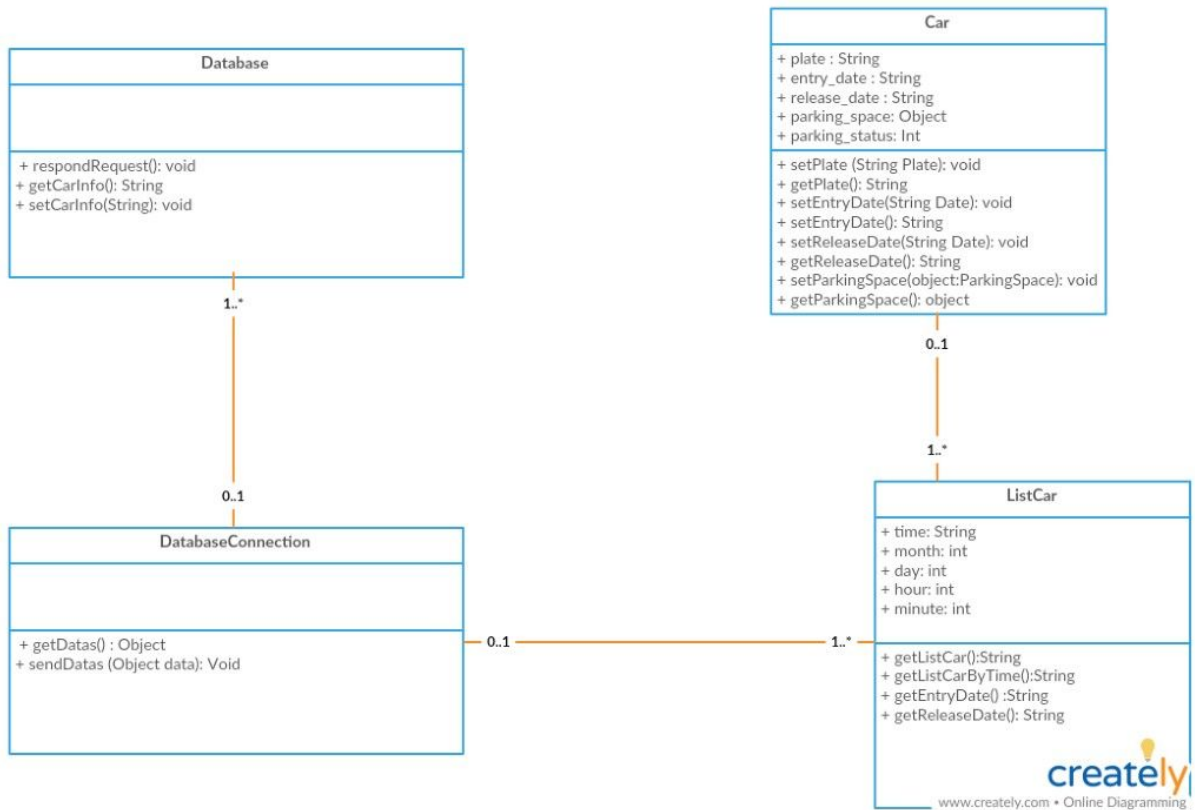
Her yeni araç girişinde veri tabanına giden bilgiler güncellenecek ve yönetici masaüstü uygulaması sayesinde bu bilgilere erişilecek.

## 5.4 Logical View Point

### 5.4.1 Ana Sınıf Diagramı

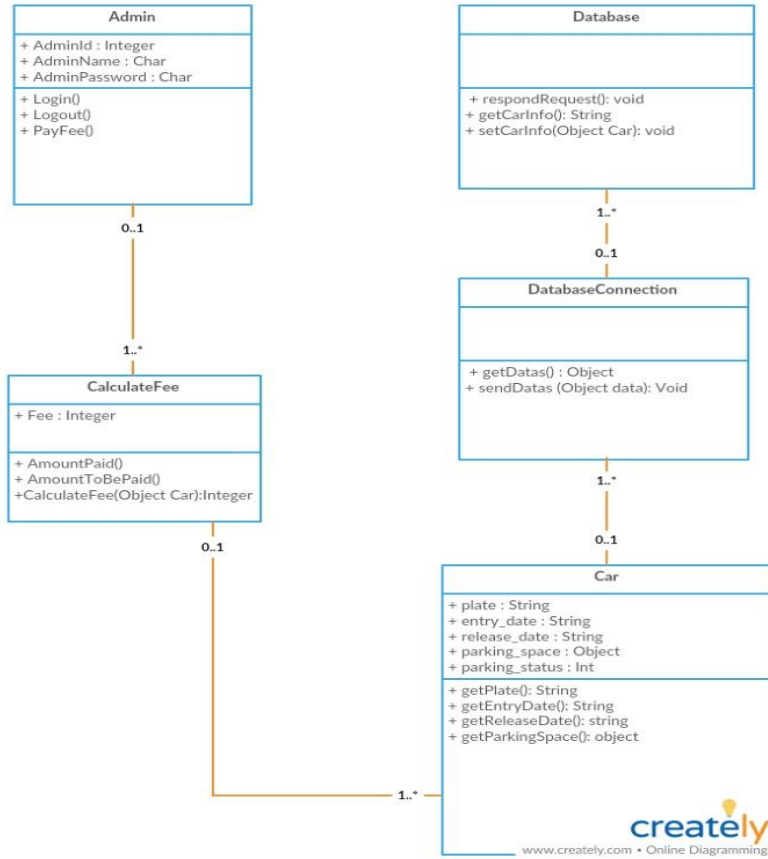


### 5.4.2 Araç Listeleme



### 5.4.3 Ücretlendirme Sistemi

#### Fiyatlandırma Sistemi



### 5.5 Modül : Camera

ParkController ile ilişkilidir.

detectedPlate(): void tipinde değer döndürür. Plakanın algılanıp, algılanmadığı işlemi hakkında değer döndürür.

SendDetectedPlate(Plate:String): void tipinde değer döndürür. Algılanan plaka bilgisini, plaka ismi ile birlikte göndermeye yarayan metottur.

### 5.6. Modül: ParkingSpace

Car ve ParkController sınıfları ile ilişkilidir

(x1\_coord:Double),(y1\_coord:Double)(x2\_coord:Double),(y2\_coord:Double),(x3\_coord:Double),(y3\_coord:Double),(x4\_coord:Double)(y4\_coord:Double) tipinde değişkenlere sahiptir.

Park alanı oluşturmada x ve y leri tanımlamada kullanılır.

(getX1Coord():Double),(setX1Coord(x1:double):Void),(getY1Coord():Double),(getX2Coord():Double),(getY2Coord():Double),(getX3Coord():Double),(getY3Coord():Double),(getX4Coord():Double),(getY4Coord():Double) metotlarına sahiptir.

Park alanı oluşturmada x ve y leri tanımlandıktan sonra belirlenen değerler için park alanı oluşturur.

## 5.7 Modül: Park Controller

Diyagramdan anlaşılacağı üzere en çok işlem gören sınıftır. Park işleminin ana hatları bu sınıf diyagramı altında işlenir.

Api Client, Gate, ParkingSpace ve Car Plate sınıfları ile ilişkilidir.

(Cars: Array Object) dizi değişkenine sahiptir.

Araç objeleri bu dizide tutulur.

(createNewCar(Plate:String): String) metodu yeni aracı, plaka bilgisiyle döndürür.

(sendPlate:String) metodu yeni plaka girişini servera gönderir.

(createNewParkingPlace(String:coords):Object) metodu girilen koordinatlara göre yeni park alanı belirler.

(connectToGate():void) metodu kapı ile bağlantının kontrol edildiği metottur.

(getCars(): Array Object) metodu arabaların bilgilerinin dizi tipinde döndürüldüğü metottur.

## 5.8 Modül : Car

ParkingSpace ve ParkController sınıfı ile ilişki halindedir.

(plate:String) aracın plaka bilgisini String tipinde tutar.

(entry\_date:String) araç giriş tarihini String tipinde tutar.

(release\_date:String) araç çıkış tarihini String tipinde tutar.

(parking\_space:Object) aracın hangi alanda park halinde olduğunu ParkingSpace nesnesi olarak saklayabilir.

(parking\_status: Int) aracın park durumunu belirtir.

(setPlate(String Plate):void) metodu aracın plakasına değer atar.

(getPlate():String) metodu aracın plaka bilgisini döndüren metottur.

(setEntryDate(String Date): void) ----

(setEntryDate(): void) entry\_date özelliğine aracın giriş yaptığı tarihi atar.

(setReleaseDate(String Date): void) release\_date özelliğine aracın çıkış yaptığı tarihi atar.

(getReleaseDate(): string) release\_date özelliğini döndürür.

(setParkingSpace (object:ParkingSpace): void) ---- parking\_space özelliğine parametre olarak gönderilen ParkingSpace nesnesini gönderir.

(getParkingSpace(): object) ---- parking\_space özelliğini döndürür.

## 5.9 Gate Modülü

### 5.9.1 Gate

Camera sınıfı ile ilişkilidir.

(gate\_status: boolean) kapı bariyerinin açık veya kapalı olduğunu belirten boolean ifadedir.

(entry\_gate:boolean) kapının giriş veya kapalı olduğunu belirten boolean ifadedir.

(getCamera(): Object) Camera nesnesine değer atan metottur.

(setGateStatus(status:Boolean):Void) kapının açık veya kapalı durumda olduğunu belirten metottur.



(getGateStatus():boolean) kapının hangi durumda olduğunu sorgulayan boolean tipinde değer döndüren metottur.

(getEntryGate():boolean) kapının açılıp-kapanması olayının belirtilmesi için (entry\_gate) değişkenin çağırıldığı metottur.

### 5.9.2 GateController

ParkController sınıfı ile ilişkilidir.

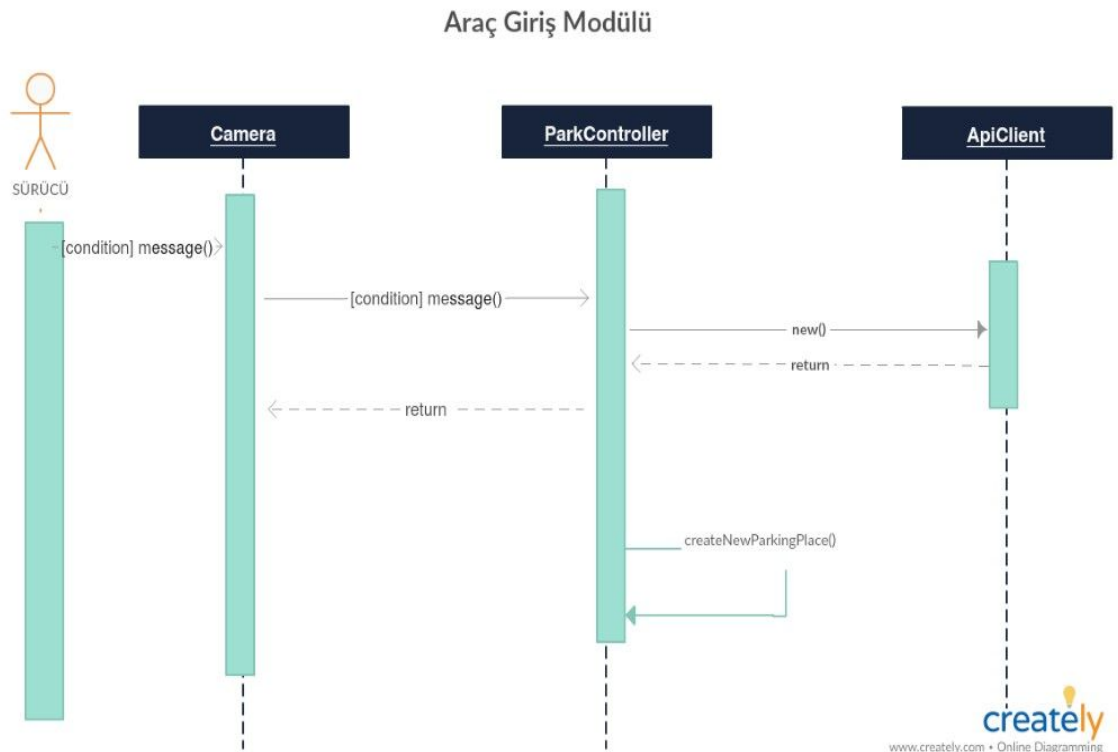
(openGate(gate:Object):Void) kapı açılması olayının gerçekleştirildiği metottur. Açılacak kapı "gate" nesnesi parametre olarak gönderilerek işlem yapılır.

(closGate(gate:Object):Void) kapı kapanması olayının gerçekleştirildiği metottur. Kapanacak kapı "gate" nesnesi parametre olarak gönderilerek işlem yapılır.

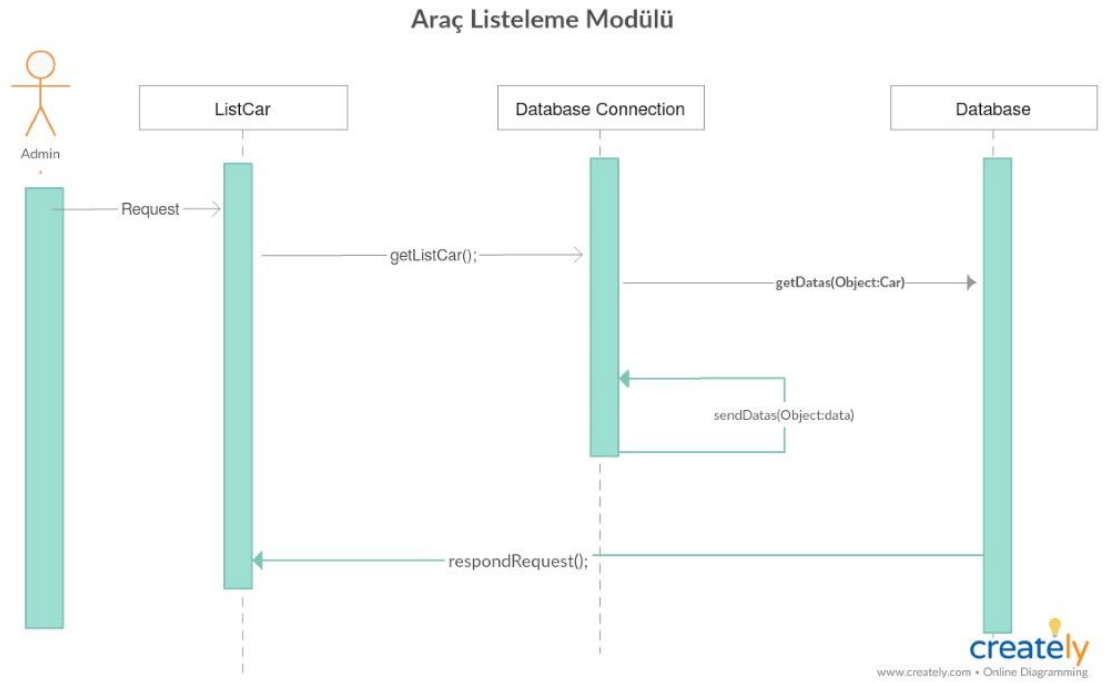
(displayMessage(message:String,gate:Object): Void) kapının durumunu String tipte değer döndürülerek ekrana mesaj vermeyi sağlayan metottur. Mesaj ve mesajın gösterileceği kapı parametre olarak gönderilerek işlem yapılır.

## 5.10 Interaction Viewpoint

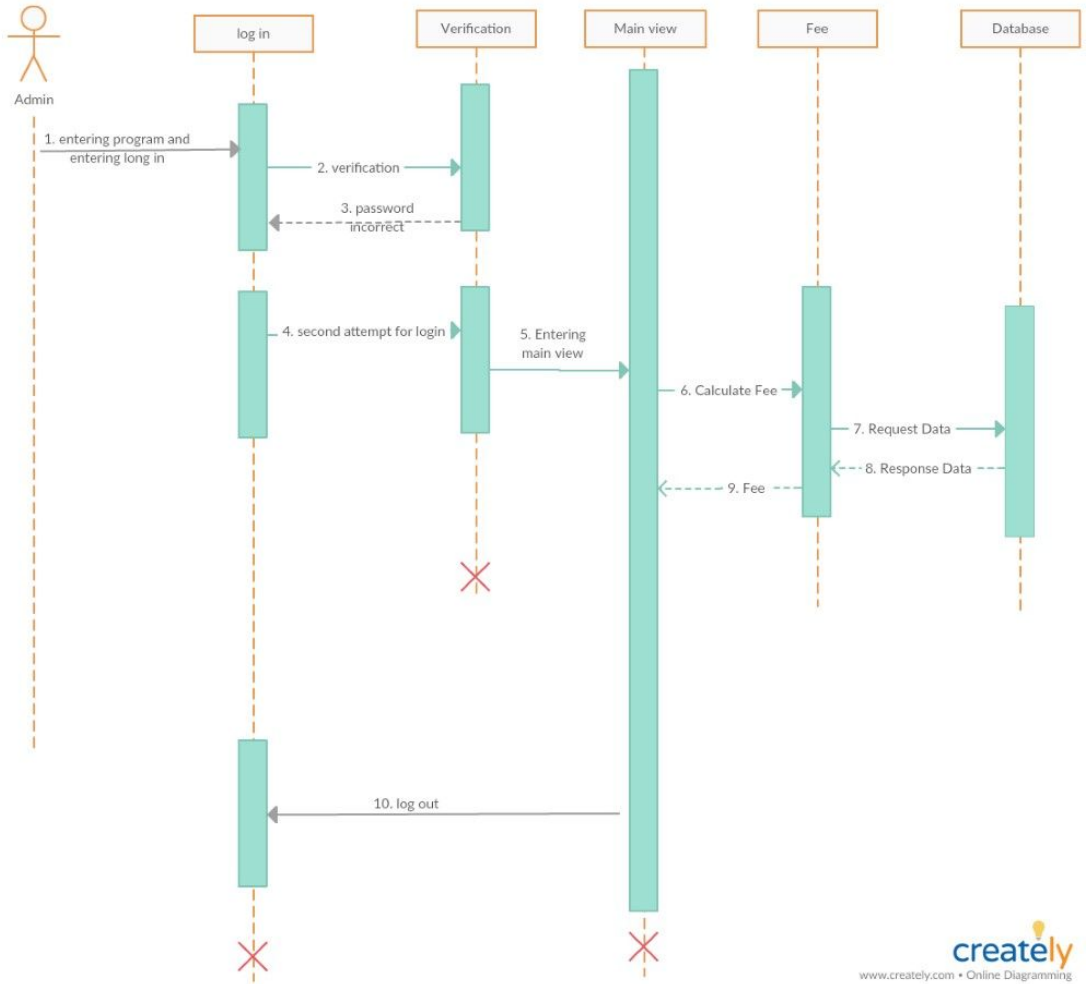
### 5.10.1 Araç Girişi



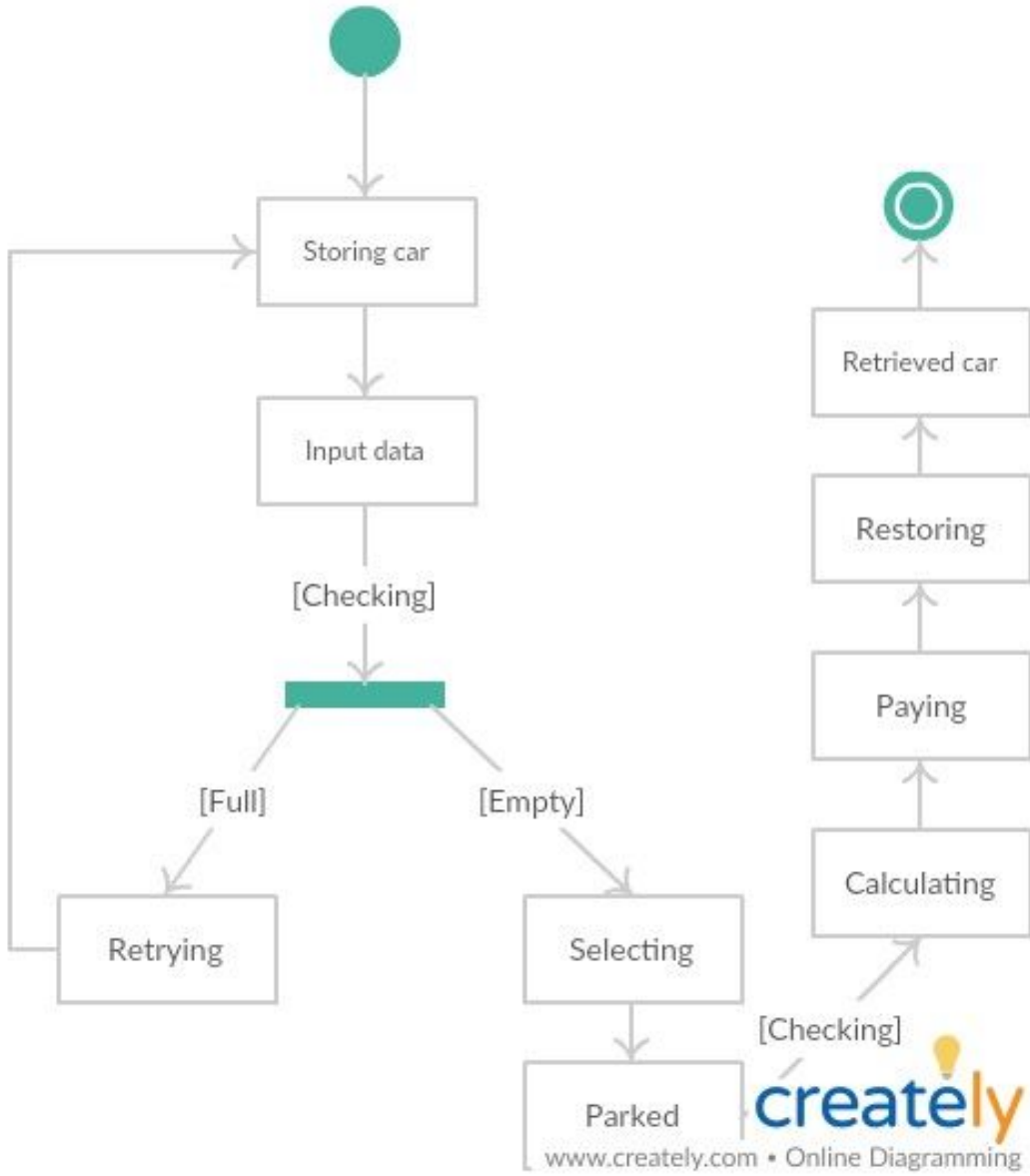
### 5.10.2 Araç Listeleme



### 5.10.3 Ücretlendirme Sistemi



## 5.11 State Dynamics Viewpoint



## 6. İnsan Arayüz Tasarımı

### 6.1 Kullanıcı Arayüzüne Genel Bakış

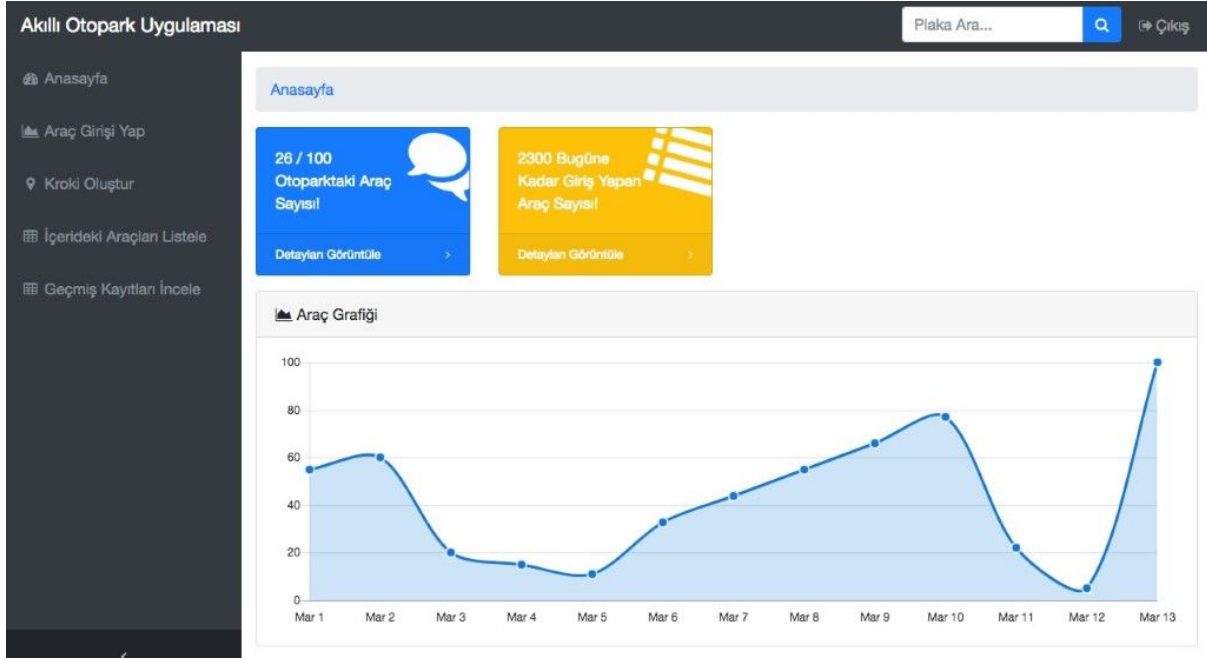
Uygulama daha çok yönetici odaklı dizayn edilmiştir. Arayüzde kullanım verimliliği ve takibin artırılması için tablosal olarak düzenlenmiş tasarımlar yer almaktadır. Kolay kullanımın yanı sıra menülerin isimlendirilmesinde açık ve sade dil kullanılması ile uygulamanın daha anlaşılabilir olması amaçlanmıştır. Sol tarafta yapışık olarak yer alan kenar çubuğu sayesinde uygulama fonksiyonlarına erişilebilir.

Ara yüzler 5 kısımda sınıflandırılmıştır.

- Anasayfa
- Araç Girişi Yap
- Kroki Oluştur
- İçerideki Araçları Listele
- Geçmiş Kayıtları İncele



## 6.2 Ekran Resimleri

### 6.2.1 Anasayfa



### 6.2.2 Araç Girişi Yap


Akıllı Otopark Uygulaması


Plaka Ara...  

Anasayfa / Araç Girişi Yap

Plaka

Park Yeri





## 6.2.3 Kroki Oluştur

Akıllı Otopark Uygulaması

Plaka Ara... Q Çıkış

Anasayfa

Araç Girişi Yap

Kroki Oluştur

İçerideki Araçları Listele

Geçmiş Kayıtları İncele

Anasayfa / Kroki Oluştur

Kroki İsmi

Lütfen Krokinizi İsmiendirmiş

Araç Kapasitesi

Lütfen Araç Kapasitesini Giriniz

En

Lütfen Alanın Enini Giriniz

Boy

Lütfen Alanın Boyunu Giriniz

Kaydet

## 6.2.4 İçerideki Araçları Listele

Akıllı Otopark Uygulaması

Plaka Ara... Q Çıkış

Anasayfa

Araç Girişi Yap

Kroki Oluştur

İçerideki Araçları Listele

Geçmiş Kayıtları İncele

İçerideki Araçları Listesi

Plaka Ara:

Plaka	İçeride Geçirilen Süre	Giriş Tarihi	Park Edilen Alan
34 AB 2525	2 Saat 15 Dakika	20.02.2017 10:30	A3
34 AB 2525	2 Saat 15 Dakika	20.02.2017 10:30	A3
34 AB 2525	2 Saat 15 Dakika	20.02.2017 10:30	A3
34 AB 2525	2 Saat 15 Dakika	20.02.2017 10:30	A3
34 CB 0953	1 Saat 10 Dakika	20.02.2017 10:30	B1
34 CB 0953	1 Saat 10 Dakika	20.02.2017 10:30	B1
34 CB 0953	1 Saat 10 Dakika	20.02.2017 10:30	B1
34 CB 0953	1 Saat 10 Dakika	20.02.2017 10:30	B1
Plaka	İçeride Geçirilen Süre	Giriş Tarihi	Park Edilen Alan

Previous 1 Next

## 6.2.5 Geçmiş Kayıtları İncele

Akıllı Otopark Uygulaması

Plaka Ara...

Çıkış

Anasayfa

Araç Giriş Yap

Kiraki Oluştur

İçerideki Araçları Listele

Geçmiş Kayıtları İncele

Anasayfa / Geçmiş Kayıtlar

Başlangıç Tarihi

20.02.2017

Bitiş Tarihi

21.02.2017

Filtrele

20.02.2017 - 21.02.2017 Tarihleri Arasındaki Kayıtlar

Plaka Ara:

Plaka	İçeride Geçirilen Süre	Giriş Tarihi	Çıkış Tarihi	Park Edilen Alan
34 AB 2525	2 Saat 15 Dakika	20.02.2017 10:30	20.02.2017 12:45	A3
34 AB 2525	2 Saat 15 Dakika	20.02.2017 10:30	20.02.2017 12:45	A3
34 AB 2525	2 Saat 15 Dakika	20.02.2017 10:30	20.02.2017 12:45	A3
34 AB 2525	2 Saat 15 Dakika	20.02.2017 10:30	20.02.2017 12:45	A3
34 CB 0953	1 Saat 10 Dakika	20.02.2017 10:30	20.02.2017 11:40	B1
34 CB 0953	1 Saat 10 Dakika	20.02.2017 10:30	20.02.2017 11:40	B1
34 CB 0953	1 Saat 10 Dakika	20.02.2017 10:30	20.02.2017 11:40	B1
34 CB 0953	1 Saat 10 Dakika	20.02.2017 10:30	20.02.2017 11:40	B1

Previous1Next

## 6.3 Ekran Objeleri ve Olayları

### 6.3.1 Anasayfa Ekran Objeleri

Ekran Arayüz No	6.3.1
Ekran Arayüz Adı	Anasayfa
Tanımlayan	
Ana Akış	<ul style="list-style-type: none"><li>• Giriş yapan yönetici bu sayfada karşılanır.</li><li>• Yönetici, otoparkta park halinde olan araç sayısına erişebilir.</li><li>• Yönetici, otoparkta giriş yaptığı güne kadar kaç arabanın otoparka giriş yaptığını öğrenebilir.</li><li>• Yönetici işlem yapmak istediği kısımlara menülerden ulaşır.</li><li>• Otoparkın günlük,haftalık,aylık istatistiklerine ulaşabilir.</li><li>• Plaka aranmak istenirse, sağ üstteki arama çubuğu kullanılır.</li><li>• Yönetici çıkış yapmak isterse, sağ köşedeki "Çıkış Yap" butonuna basar.</li></ul>
Referans Kodu	6.2.1
Not	

### 6.3.2 Araç Giriş Yap Ekran Objeleri

Ekran Arayüz No	6.3.2
Ekran Arayüz Adı	Araç Giriş Yap
Tanımlayan	
Ana Akış	<ul style="list-style-type: none"><li>• Yönetici, araç girişi için "Plaka" bilgisini tanımlanan alana girer.</li><li>• Yönetici, "Park Yeri" bilgisini tanımlanan değer aralığındaki seçeneklerden seçer.</li><li>• "Kaydet" butonuna basılır ve araç girişi bilgileri ile birlikte veri tabanına kaydedilir.</li></ul>
Referans Kodu	6.2.2
Not	



### 6.3.3 Kroki Oluştur Ekran Objeleri

Ekran Arayüz No	6.3.3
Ekran Arayüz Adı	Kroki Oluştur
Tanımlayan	
Ana Akış	<ul style="list-style-type: none"><li>• Yönetici, yeni oluşturacağı kroki için, "Kroki İsmi" kısmından isim belirler.</li><li>• Yönetici, kapasite seçimi yapmak için, "Araç Kapasitesi" kısmından kapasite sayısını belirler.</li><li>• Yönetici, yeni oluşturacağı kroki için en uzunluğunu, "En" kısmından belirler.</li><li>• Yönetici, yeni oluşturacağı kroki için boy uzunluğunu, "Boy" kısmından belirler.</li><li>• Bütün alanlar belirlendikten sonra "Kaydet" tuşuna basılır ve yeni kroki oluşturulmuş olunur</li></ul>
Referans Kodu	6.2.3
Not	

#### 6.3.4 İerideki Araları Listele Ekran Objeleri

Ekran Arayüz No	6.3.4
Ekran Arayüz Adı	İerideki Araları Listele
Tanımlayan	
Ana Akış	<ul style="list-style-type: none"><li>• Yönetici, otoparkta hali hazırda park durumunda olan araçları listeler.</li><li>• "İerde geçirilen süre", "giriş tarihi" ve "park alan" bilgisine "plaka" bilgisi ile ulaşır.</li></ul>
Referans Kodu	6.2.4
Not	<ul style="list-style-type: none"><li>• Listelenen araç tablosu, opsiyonel olarak "Plaka", "İeride Geçirdiği Süre", "Giriş Tarihi" ve "Park Edilen Alan" unsurlarına göre de üzerine tıklanarak, değ er tipine göre sıralanabilir.</li></ul>

### 6.3.5 Geçmiş Kayıtları İncele Ekran Objeleri

Ekran Arayüz No	6.3.5
Ekran Arayüz Adı	Geçmiş Kayıtları İncele
Tanımlayan	
Ana Akış	<ul style="list-style-type: none"><li>• Yönetici, görüntülemek istediği "Başlangıç Tarihi"ni belirler.</li><li>• Yönetici, görüntülemek istediği "Bitiş Tarihi"ni belirler.</li><li>• "Filtrele" butonuna basılır.</li><li>• Belirlenen değer aralıklarındaki araç listesi yöneticiye sunulur.</li></ul>
Referans Kodu	6.2.5
Not	<ul style="list-style-type: none"><li>• Listelenen araç tablosu, opsiyonel olarak "Plaka", "İçeride Geçirdiği Süre", "Giriş Tarihi", "Çıkış Tarihi" ve "Park Edilen Alan" unsurlarına göre de üzerine tıklanarak, değer tipine göre sıralanabilir.</li></ul>



**Beykent Üniversitesi**  
**Mühendislik ve Mimarlık Fakültesi**  
**Yazılım Mühendisliği Bölümü**

**STD Dokümantasyonu**

## **AKILLI OTOPARK SİSTEMİ**

<b>Burak</b>	<b>FIRÇASIGÜZEL</b>	<b>130301022</b>
<b>Emre</b>	<b>SERBES</b>	<b>130301020</b>
<b>Eray</b>	<b>AKARTUNA</b>	<b>130301026</b>

## **İçindekiler :**

- 1. Testin Planlanması**
- 2. Referanslar (References)**
- 3. Giriş (Introduction)**
- 4. Test Edilecek Birimler (Test Items)**
- 5. Yazılıma İlişkin Riskler (Software Risk Issues)**
- 6. Test edilecek özellikler ( Features to be Tested)**
  - 6.1 Yönetici Giriş Testi**
  - 6.2 Plaka Tanıma Testi**
  - 6.3 Ücretlendirme Testi**
  - 6.4 Uygun Park Yeri Bulma Testi**
  - 6.5 Harita Oluşturma Testi**
  - 6.6 Yönetici Ekleme Testi**
  - 6.7 Geçmiş Kayıtları Listeleme Testi**
  - 6.8 Araç Giriş Testi**
  - 6.9 Araç Çıkış Testi**
  - 6.10 Yöneticileri Listeleme Testi**
  - 6.11 Web Servis Bağlantı Testi**
  - 6.12 Veri Tabanı Bağlantısı Testi**
- 7. Test edilmeyecek özellikler (Features not to be Tested)**
- 8. Test Stratejisi (Approach)**
- 9. Birimin Başarı/Başarısızlık Kriterleri (Item Pass/Fail Criteria)**
- 10. Test Versiyon Bilgileri**

## **1- Testin Planlanması**

Bu test planını, seviyesini ve ilişkili olduğu yazılım düzeyini belirlemek için benzersiz bir şirket numarası ürettik. Test planı seviyesi, ilgili yazılım seviyesiyle aynı olacaktır. Numara, aynı zamanda test planının bir Master plan, bir Düzey planı, bir entegrasyon planı veya hangisi temsil seviyesini temsil ediyorsa tanımlanabilir. Bu, yapılandırma yazılım ve testware sürümlerinin koordinasyonunda yardımcı olmak içindir.

## **2. Referanslar (References)**

Bu test dokümanında IEEE System Test Documentation (IEEE Std 829) standartları referans alınmıştır.

## **3- Giriş (Introduction)**

Bu yazılım test belgeleri (STD), IEEE Std 829 standardına göre Akıllı Otopark yazılımının tüm test prosedürlerini, yazılımın hedeflenen ihtiyaçları karşıladığını doğrulamak için gerekli olan test ortamının hazırlanışı, test durumları ve yapılacak olan kullanıcı test faaliyetlerini göstermektedir.

## **4- Test Edilecek Birimler (Test Items)**

- 4.1 Yönetici Giriş Testi**
- 4.2 Plaka Tanıma Testi**
- 4.3 Ücretlendirme Testi**
- 4.4 Uygun Park Yeri Bulma testi**
- 4.5 Harita Oluşturma Testi**
- 4.6 Yönetici Ekleme Testi**
- 4.7 Geçmiş Kayıtları Listeleme Testi**
- 4.8 Araç Giriş Testi**
- 4.9 Araç Çıkış Testi**
- 4.10 Yöneticileri Listeleme Testi**
- 4.11 Web Servis Bağlantı Testi**
- 4.12 Veri Tabanı Bağlantısı Testi**

## **5.Yazılıma İlişkin Riskler (Software Risk Issues)**

### **5.1 Zaman Riskleri**

Projenin, yanlış görev ve malzeme paylaşımından dolayı beklenen süre içerisinde gerçekleşmesine veya tamamlanmasına engel olan veya sebep olan risk çeşididir.

### **5.2 Bütçe (maliyet) riskleri**

Gerçekçi olmayan bütçe tahminleri sonucu finansal sorunlara yol açan risklerdir. Bu risklerin gerçekleşmesi durumunda Tablolar değişmekte, maliyetler artmaktadır.

### **5.3 Yönetim riskleri**

Yönetim riskleri; amaçların net olmayışı, planlama eksikliği, yönetim tecrübesi ve eğitim eksikliği, iletişim sorunları, örgütsel sorunlar, otorite eksikliği ve kontrol problemlerini kapsamaktadır.

### **5.4 Teknik riskler**

Genelde fonksiyonların yanlış olmasından kaynaklanır. Müşteri taleplerinin sürekli değişmesi, gelişmiş tekniklerin kullanılmaması ve geliştirilecek olan projenin zor faaliyetler içermesi gibi sebeplerden kaynaklanmaktadır.

### **5.5 Program riskleri**

Proje kapsamının dışına çıkan, kontrol dışı durumlardan veya önceliklerin sürekli değişmesinden doğan risklerdir.

### **5.6 Sözleşme ve yasal riskler**

Sözleşme ve yasal riskler; değişen ihtiyaçları, pazar odaklı programları, sağlık ve güvenlik sorunları, hükümet düzenlemeleri ve ürün garantisi konularını içerir.

### **5.7 Personel riskleri**

Personel riskleri; personel duraklamaları, deneyim ve eğitim sorunları, etik ve ahlak

konularını, personel çatışmalarını ve verimlilik sorunlarını içermektedir.

### 5.8 Diğer kaynaklı riskler

Diğer kaynaklı riskler, mevcut olmayan veya geç teslim edilen ekipman ve sarf malzemeleri, yetersiz aracı, yetersiz tesisleri, dağıtılan bölgeleri, bilgisayar kaynaklarının olmayışı ve yavaş tepki sürelerini kapsamaktadır.

## 6. Test edilecek özellikler ( Features to be Tested)

### 6.1 Yönetici Giriş Testi

<b>Risk Seviyesi</b>	Yüksek
<b>Amaç</b>	Yöneticinin, yönetim sistemine giriş yapması
<b>Girdiler</b>	<ul style="list-style-type: none"><li>• Yönetici e-mail ve şifresi</li><li>• Veri tabanı</li><li>• Yönetim Paneli</li></ul>
<b>Beklenen Çıktılar</b>	<ul style="list-style-type: none"><li>• Yönetim paneli ana sayfası</li></ul>
<b>Geçiş Kriterleri</b>	<ul style="list-style-type: none"><li>• Veri tabanından gelen geri bildirimin olumlu olması</li></ul>
<b>Başarısız Kriterler</b>	<ul style="list-style-type: none"><li>• Veri tabanından gelen geri bildirimin olumsuz olması</li></ul>
<b>Test Prosedürü</b>	Testin başarılı veya başarısızlık durumunun belirtilerek nedenlerinin raporlanması

### 6.2 Plaka Tanıma Testi

<b>Risk Seviyesi</b>	Yüksek
<b>Amaç</b>	<ul style="list-style-type: none"><li>• Araç plakasının resimden tanınabilmesi</li></ul>
<b>Girdiler</b>	<ul style="list-style-type: none"><li>• Araç plakası Resmi</li></ul>
<b>Beklenen Çıktılar</b>	<ul style="list-style-type: none"><li>• Yazı olarak araç plakası</li></ul>
<b>Geçiş Kriterleri</b>	<ul style="list-style-type: none"><li>• Web servis'den araç plakasının yazı olarak döndürülmesi</li></ul>
<b>Başarısız Kriterler</b>	<ul style="list-style-type: none"><li>• Araç plakasının okunamaması.</li></ul>
<b>Test Prosedürü</b>	Web servis'den dönen sonuçlara göre başarılı veya başarısız sonuçların raporlanması



### 6.3 Ücretlendirme Testi

<b>Risk Seviyesi</b>	Orta
<b>Amaç</b>	Alınan otopark hizmetinin ücretlendirilmesi
<b>Girdiler</b>	<ul style="list-style-type: none"><li>• Araç plaka bilgisi</li><li>• Araç giriş-çıkış bilgileri</li><li>• Web servis'den gelen cevap</li></ul>
<b>Beklenen Çıkışlar</b>	<ul style="list-style-type: none"><li>• Araç bilgisi ve giriş-çıkış bilgileri göz önünde bulundurularak ücretin double cinsinden değer döndürmesi</li></ul>
<b>Geçiş Kriterleri</b>	<ul style="list-style-type: none"><li>• Web servisin beklenen şekilde cevap vermesi</li></ul>
<b>Başarısız Kriterler</b>	<ul style="list-style-type: none"><li>• Web servisin hatalı cevap döndürmesi</li></ul>
<b>Test Prosedürü</b>	Testin başarılı veya başarısızlık durumunun belirtilerek nedenlerinin raporlanması

### 6.4 Uygun Park Yeri Bulma Testi

<b>Risk Seviyesi</b>	Yüksek
<b>Amaç</b>	Araç için en uygun park yerini bulmak
<b>Girdiler</b>	<ul style="list-style-type: none"><li>• Araç plakası</li><li>• Giriş zamanı</li><li>• Web Server bağlantısı</li></ul>
<b>Beklenen Çıkışlar</b>	<ul style="list-style-type: none"><li>• Web servis bağlantı işlemi için geri bildirim döndürülmesi</li></ul>
<b>Geçiş Kriterleri</b>	<ul style="list-style-type: none"><li>• Web servis gelen geri bildirimin olumlu olması</li></ul>
<b>Başarısız Kriterler</b>	<ul style="list-style-type: none"><li>• Web servis'den gelen geri bildirimin olumsuz olması</li><li>• Otoparkın dolu olması</li></ul>
<b>Test Prosedürü</b>	Web servis'ten gelen bildirimin durumuna göre başarılı veya başarısızlıklara yol açan etkenlerin raporlanması

### 6.5 Harita Oluřturma Testi

<b>Risk Seviyesi</b>	Yüksek
<b>Amaç</b>	Yöneticinin otopark için harita oluřturması
<b>Girdiler</b>	<ul style="list-style-type: none"><li>• x1,x2,x3,x4,y1,y2,y3,y4 koordinat deęerleri</li></ul>
<b>Beklenen Çıkıřlar</b>	<ul style="list-style-type: none"><li>• Belirlenen düzende otopark alanının oluřturulması</li></ul>
<b>Geçiř Kriterleri</b>	<ul style="list-style-type: none"><li>• Web servis'den gelen otopark haritalandırma isteęinin olumlu cevaplanması</li></ul>
<b>Başarısız Kriterler</b>	<ul style="list-style-type: none"><li>• Koordinatların uyuřmamazlık göstermesi</li></ul>
<b>Test Prosedürü</b>	Testin başarılı veya başarısızlık durumunun belirtilerek nedenlerinin raporlanması

### 6.6 Yönetici Ekleme Testi

<b>Risk Seviyesi</b>	Orta
<b>Amaç</b>	Sisteme yönetici ekleyebilme
<b>Girdiler</b>	<ul style="list-style-type: none"><li>• Email</li><li>• Şifre</li><li>• Ad</li><li>• Soy ad</li></ul>
<b>Beklenen Çıkıřlar</b>	<ul style="list-style-type: none"><li>• Yönetim paneli yönetici liste sayfası.</li></ul>
<b>Geçiř Kriterleri</b>	<ul style="list-style-type: none"><li>• Veri tabanından gelen geri bildirimin olumlu olması</li></ul>
<b>Başarısız Kriterler</b>	<ul style="list-style-type: none"><li>• Veri tabanından gelen geri bildirimin olumsuz olması</li></ul>
<b>Test Prosedürü</b>	Veri tabanından gelen bildirimin durumuna göre başarılı veya başarısızlıklara yol açan etkenlerin raporlanması

### 6.7 Geçmiş Kayıtları Listeleme Testi

<b>Risk Seviyesi</b>	Düşük
<b>Amaç</b>	Otopark geçmiş kayıtlarının listelenmesi
<b>Girdiler</b>	<ul style="list-style-type: none"><li>• Tarih aralığı</li><li>• Plaka bilgisi</li></ul>
<b>Beklenen Çıktılar</b>	<ul style="list-style-type: none"><li>• Belirlenen aralıkta geçmiş kayıtların listelenmesi</li></ul>
<b>Geçiş Kriterleri</b>	<ul style="list-style-type: none"><li>• Web servis'ten gelen geri bildirimin olumsuz olması</li></ul>
<b>Başarısız Kriterler</b>	<ul style="list-style-type: none"><li>• Girdilerin istenilen formata uygun girilmemesi</li></ul>
<b>Test Prosedürü</b>	Testin başarılı veya başarısızlık durumunun belirtilerek nedenlerinin raporlanması

### 6.8 Araç Giriş Testi

<b>Risk Seviyesi</b>	Yüksek
<b>Amaç</b>	Otopark'a yeni araç girişini, bilgileri ile birlikte sağlamak
<b>Girdiler</b>	<ul style="list-style-type: none"><li>• Araç plakası</li><li>• Araç plaka tanıma</li><li>• Araç giriş zamanı</li></ul>
<b>Beklenen Çıktılar</b>	<ul style="list-style-type: none"><li>• Sistemin yeni araç girişine onay vermesi</li></ul>
<b>Geçiş Kriterleri</b>	<ul style="list-style-type: none"><li>• Web servisin hatasız çalışması</li></ul>
<b>Başarısız Kriterler</b>	<ul style="list-style-type: none"><li>• Sistemin yeni araç girişine onay vermemesi</li></ul>
<b>Test Prosedürü</b>	Servisten gelen değerlere göre başarı veya başarısızlığa yol açan nedenlerin raporlanması

## 6.9 Araç Çıkış Testi

<b>Risk Seviyesi</b>	Orta
<b>Amaç</b>	Otoparktan araç çıkışının sağlanması
<b>Girdiler</b>	<ul style="list-style-type: none"><li>• Araç plaka bilgisi</li><li>• İçerde geçirilen zaman</li><li>• Veri tabanı</li></ul>
<b>Beklenen Çıktılar</b>	<ul style="list-style-type: none"><li>• Araç çıkış işleminin başarılanması</li></ul>
<b>Geçiş Kriterleri</b>	<ul style="list-style-type: none"><li>• Veri tabanından gelen geri bildirimin olumlu olması</li></ul>
<b>Başarısız Kriterler</b>	<ul style="list-style-type: none"><li>• Veri tabanından olumsuz mesaj alma</li></ul>
<b>Test Prosedürü</b>	Veri tabanından gelen mesajın durumuna göre başarılı veya başarısızlıklara yol açan etkenlerin raporlanması

## 6.10 Yönetici Listeleme Testi

<b>Risk Seviyesi</b>	Düşük
<b>Amaç</b>	Sistemdeki Yöneticileri Listeleme
<b>Girdiler</b>	<ul style="list-style-type: none"><li>• Görüntülenecek yöneticilerin limiti</li></ul>
<b>Beklenen Çıktılar</b>	<ul style="list-style-type: none"><li>• Yönetici listesini görüntüleyebilme</li></ul>
<b>Geçiş Kriterleri</b>	<ul style="list-style-type: none"><li>• Veri tabanından yönetici listesinin alınabilmesi.</li></ul>
<b>Başarısız Kriterler</b>	<ul style="list-style-type: none"><li>• Veri tabanından olumsuz mesaj alma</li></ul>
<b>Test Prosedürü</b>	Veri tabanından gelen mesajın durumuna göre başarılı veya başarısızlıklara yol açan etkenlerin raporlanması

### 6.11 Web Servis Bağlantı Testi

<b>Risk Seviyesi</b>	Yüksek
<b>Amaç</b>	Web servise bağlantı isteğinin test edilmesi
<b>Girdiler</b>	<ul style="list-style-type: none"><li>• Web Servis Kullanıcı Adı</li><li>• Web Servis Kullanıcı Şifresi</li></ul>
<b>Beklenen Çıktılar</b>	<ul style="list-style-type: none"><li>• Web servis bağlantı işlemi için geri bildirim döndürülmesi</li></ul>
<b>Geçiş Kriterleri</b>	<ul style="list-style-type: none"><li>• Web serviste kullanılacak token'ın döndürülmesi</li></ul>
<b>Başarısız Kriterler</b>	<ul style="list-style-type: none"><li>• Herhangi bir token dönmemesi</li><li>• Bağlantı hatası oluşması.</li></ul>
<b>Test Prosedürü</b>	Testin başarılı veya başarısızlık durumunun belirtilerek nedenlerinin raporlanması

### 6.12 Veri tabanı Bağlantı Testi

<b>Risk Seviyesi</b>	Yüksek
<b>Amaç</b>	Veri tabanına bağlantının sağlanması
<b>Girdiler</b>	<ul style="list-style-type: none"><li>• Veri tabanı url, veri tabanı adı, kullanıcı adı, şifresi</li></ul>
<b>Beklenen Çıktılar</b>	<ul style="list-style-type: none"><li>• Veri tabanı bağlantı işlemi için geri bildirim döndürülmesi</li></ul>
<b>Geçiş Kriterleri</b>	<ul style="list-style-type: none"><li>• Veri tabanından gelen geri bildirim olumlu olması</li></ul>
<b>Başarısız Kriterler</b>	<ul style="list-style-type: none"><li>• Veri tabanından gelen geri bildirim olumsuz olması</li></ul>
<b>Test Prosedürü</b>	Veri tabanından gelen bildirim durumuna göre başarılı veya başarısızlıklara yol açan etkenlerin raporlanması

## 7. Test edilmeyecek özellikler (Features not to be Tested)

Aşağıdaki eylemler çok düşük risk seviyesinde bulunduğu için test edilmeyecektir.

- 7.1 Haritanın doğruluğu
- 7.2 Yönetici kullanıcı adı

## 8. Test Stratejisi (Approach)

Akıllı Otopark yazılım bileşenlerinin tamamı risk seviyelerine göre sınıflandırılmış, test edilecek ve edilmeyecek bileşenler bu dokümantasyonda verilmiştir. Test işlemleri 6.TEST DURUMLARI bölümünde sırası ile verilen tanım ve kısıtlamalar doğrultusunda yapılarak 9.TEST LOG FORMATI bölümünde verilen raporlama formatına uygun olarak Akıllı Otopark test raporu dokümanı hazırlanmıştır.

## 9. Birimin Başarı/Başarısızlık Kriterleri (Item Pass/Fail Criteria)

### 9.1. Yönetici Giriş Testi

Test Durumu	Gerçekleşen Sonuç	Geçti / Kaldı
Yöneticinin, otopark yönetim paneline yönetici adı ve şifresiyle istek yollaması ve veri tabanının gelen isteğe cevap vermesi beklenmektedir	Tamam	G

### 9.2. Plaka Tanıma Testi

Test Durumu	Gerçekleşen Sonuç	Geçti / Kaldı
Otoparka gelen yeni aracın, otopark kamera ve yazılımları aracılığıyla plaka tespiti yapılması	Tamam	G

### 9.3. Ücretlendirme Testi

Test Durumu	Gerçekleşen Sonuç	Geçti / Kaldı
Araç bilgilerine karşılık gelen bilgilerle web serviste işlenen giriş-çıkış zamanına göre hizmet ücreti miktarının belirlenmesi	Tamam	G

#### 9.4. Uygun Park Yeri Bulma Testi

Test Durumu	Gerçekleşen Sonuç	Geçti / Kaldı
Araç bilgisinin veri tabanı-web servis arasındaki bağlantıyı sağladıktan sonra, uygun park yeri yazılımını kullanarak, araç için uygun park yeri bilgilerini vermesi beklenmektedir	Tamam	G

#### 9.5. Harita Oluşturma Testi

Test Durumu	Gerçekleşen Sonuç	Geçti / Kaldı
Belirlenen x1,x2,x3,x4,y1,y2,y3,y4 değerler için otopark park alanı oluşması	Tamam	G

#### 9.6. Yönetici Ekleme Testi

Test Durumu	Gerçekleşen Sonuç	Geçti / Kaldı
Yönetici oluşturmak için e-mail,şifre,ad,soy ad bilgilerinin veri tabanının onay vermesi ve yeni yönetici oluşturulması	Tamam	G

#### 9.7. Geçmiş Kayıtları Listeleme Testi

Test Durumu	Gerçekleşen Sonuç	Geçti / Kaldı
Otopark veri tabanında yer alan geçmiş kayıtlarının, istek üzerine veri tabanındaki listelerin gösterilmesi	Tamam	G

#### 9.8. Araç Giriş Testi

Test Durumu	Gerçekleşen Sonuç	Geçti / Kaldı
Yeni gelen aracın gerekli bilgileri alınarak, otoparka giriş olayının sağlanması	Tamam	G

#### 9.9. Araç Çıkış Testi

Test Durumu	Gerçekleşen Sonuç	Geçti / Kaldı
Otoparkta bulunan aracın gerekli bilgileri alınarak,	Tamam	G

otoparktan çıkış yapması		
--------------------------	--	--

#### 9.10. Yöneticileri Listeleme Testi

Test Durumu	Gerçekleşen Sonuç	Geçti / Kaldı
Veri tabanında kayıtlı yöneticilerin listelenmesi	Tamam	G

#### 9.11. Web Servis Bağlantı Testi

Test Durumu	Gerçekleşen Sonuç	Geçti / Kaldı
Web servise kullanıcı adı ve şifre ile bağlantı isteği gönderilmesi ve olumlu yada olumsuz sonucun geri döndürülmesi	Tamam	G

#### 9.12. Veri Tabanı Bağlantısı Testi

Test Durumu	Gerçekleşen Sonuç	Geçti / Kaldı
Veri tabanı url'i , veri tabanı adı, kullanıcı adı ve şifresi verileri ile veri tabanı bağlantısını gerçekleştirecek class'ın olumlu yada olumsuz sonucu geri döndürmesi	Tamam	G

### 10. Test Versiyon Bilgileri

Testi Yapan	Test Tarihi	Versiyon
Eray Akartuna	15.12.2017	0.1
Burak Fırçasıgüzel	16.12.2017	0.5
Emre Serbes	18.12.2017	0.9
Emre Serbes	19.12.2017	1