

OTOPARK OTOMASYONU

PARKING AUTOMATION

Can Ata Tekirdađlı, Halit Çelenk, Muhammed Emin Babayiđit

Biliřim Sistemleri Mühendisliđi

Kocaeli Üniversitesi

Kocaeli, Türkiye

201307032@kocaeli.edu.tr/201307056@kocaeli.edu.tr/201307005@kocaeli.edu.tr

Öz

Teknoloji günümüzde hayatımıza inkâr edilemeyecek kadar büyük katkılar ve kolaylıklar sağlamıştır. Her alanda kullanılan teknolojik aletler ve yazılımlar insanoğlunun yaşam kalitesini önemli ölçüde arttırmış ve aynı zamanda yapılması zor olan işleri de çok kolay hale getirmiştir. Kalem kâğıda gerek kalmadan bilgileri tutma, işlem yapma, bilgiye kolayca ulaşabilme gibi birçok faydası bulunan teknoloji günümüzde birçok alanda tercih edilmeye başlanmıştır ve hızla kullanıldığı alan sayısı da artmaktadır.

İnsanoğlunun ihtiyaçlarının her geçen gün artmasıyla birlikte her geçen gün büyümeye devam eden, hayatımızda artık çok önemli yere sahip teknoloji artık en basit yerlerde de kullanılmaya başlanmıştır. Biz de yaptığımız otomasyon sisteminde; otoparklarda kullanılabilecek, sade ve kolay tasarıma sahip, giriş çıkışların kolayca yapılabilmesi, hesap makinasına ihtiyaç duymadan tarifeye göre ücretin hesaplandığı, daha önce giriş yapmış kullanıcıların bilgilerini kaydeden bir otomasyon sistemi tasarladık. Kâğıt kalem kullanmadan kolaylıkla tüm işlemleri yapabildiğiniz bu otomasyon otoparkların ihtiyaçlarına göre tasarlanmıştır.

Anahtar Kelimeler: Otomasyon, teknoloji, insan, otopark, inavasyon

Abstract

Today, technology has provided undeniably great contributions and conveniences to our lives. Technological tools and software used in every field have significantly increased the quality of life of human beings and at the same time have made difficult jobs very easy. Technology, which has many benefits such as keeping information, making transactions, and accessing information easily without the need for pen and paper, has started to be preferred in many areas today and the number of areas where it is used is increasing rapidly.

Technology, which continues to grow day by day with the increasing needs of human beings and has a very important place in our lives, has now started to be used in the simplest places. In the automation system we make; We have designed an automation system that can be used in car parks, has a simple and easy design, can be easily entered and exited, the fee is calculated according to the tariff without the need for a calculator, and records the information of the users who have logged in before. This automation, where you can easily do all the operations without using paper and pen, has been designed according to the needs of parking lots.

Key Words: Automation, technology, human, parking lot, innovation

1. Problem Tanımı

Elektronik ortamda otoparkta kullanılması için bir otomasyon yapılması istenmektedir. Personelin giriş yaparak, müşterilerin bilgilerini kayıt edebildiği, kayıtlı kullanıcıyı gerekli durumlarda silebildiği, müşterinin otoparka giriş ve çıkışlarını kayıt altına alabildiği, otopark durumundan otoparkta hangi kısımların boş ya da dolu olduğunu görebileceği, toplam ne kadar kazanıldığını görebildiği aynı zamanda ayarlar kısmından şifresini değiştirebileceği, yeni kullanıcı eklemesini ya da kayıtlı olan kullanıcıyı silebilmesini sağlayan aynı zamanda yapılması istenen otomasyon sisteminde gerçekleşen tüm işlemler arka planda çalışan veri tabanı ile senkronize şekilde çalışmakta olan, bir otopark otomasyon sistemi geliştirdik.

2. Yapılan Araştırmalar

- Projemizi yaparken veri tabanı olarak Microsoft SQL Server'ı kullandık. Bu veri tabanı ile C# form tasarımıımızı bağlamamız gerekiyordu. Bu problemle ilgili videolar izledik ve araştırmalar yaptık. Veri tabanı bağlantılarını sağlamak için gerekli nesneleri türettik ve bağlantıyı gerçekleştirdik.[1][3]

- Kullanıcı formu büyütüp küçültmek istediği zaman form ekranında kayma vs. gibi problemler olma ihtimaline karşın formu büyütüp küçültmemeye kararı aldık. Bununla ilgili araştırma yapıp kaynak kod örneklerine baktık.

- Kullanıcıdan bilgilerini veri tabanına aktarırken verileri textboxlardan aktaracağımız için kullanıcı textboxlardan birini veya birden fazlasını boş bıraktığında kullanıcıya hata mesajı çıkarmak için textboxlarını if koşuluyla kontrol ettik.

- Formda kullanıcı formu control boxtan kapattığı zaman hata aldığımız için kendimiz label ile kapatma butonu oluşturduk. Kullanıcı oluşturduğumuz kapatma butonuna basınca form kapatma eventini kullandık ve Application.Exit() ile uygulamayı kapatarak sorunu çözdük.

- Bazı kısımlarda textboxlara girilen verilerin sayı veya harf içermemesi gerekiyordu biz de araştırıp gerekli yerlerde harf veya sayı girişini engelledik.[4]

- Giriş sayfasında şifreyi dışarıdan okunmasını önlemek için gizledik ve istenildiği zaman açabilecekleri parolayı göster seçeneğini eklemek için gerekli araştırmaları yaptık.[2]

- Giriş sayfasında admin girişi yapılırken kullanıcı adını ve şifreyi kontrol ettirmemiz gerekiyordu bunun öncesinde bilgilerin girildiği textboxların boş olmadığına emin olmak için if koşuluyla textboxları kontrol ettirdik. Textboxlardan en az 1 tanesi boşsa ekrana hata mesajı yazdırdık.

- Uygulamayı tam anlamıyla kullanmak ve hata almamak için arka planda veri tabanın açık olması gerekiyor. Kullanıcının daha sonradan hata almasını önlemek için bağlantıyı kontrol etmemiz gerekiyordu. Bağlantıyı kontrol eden metod oluşturduk. Bu metod sayesinde kullanıcı uygulamayı çalıştırdığında eğer bağlantı başarılı bir şekilde sağlanamazsa uyarı mesajı verdirdik ve otomatik uygulamanın kapanmasını sağladık.

- Veri tabanındaki bilgileri comboboxa eklerken sorun yaşadık. Gerekli araştırmaları yapıp başarılı bir şekilde veri tabanından comboboxa verileri aktarabildik.

- Kayıtlı araç girişi yapılırken seçilen plaka sahibinin bilgileri dışarıdan değiştirilebiliyordu biz de bu değişikliklerden dolayı hata almamak için araştırma yapıp textboxların enabled özelliğini "false" olarak ayarladık.

- Müşteri kaydı yapılırken bütün araba markaları aklımıza gelmediği için hepsini ekleyemedik daha sonra veri tabanında olmayan arabanın otoparka gelmesi durumunda sorunla karşılaşmamak için comboboxa yazılan markayı veri tabanına eklemesi için bir buton oluşturduk.

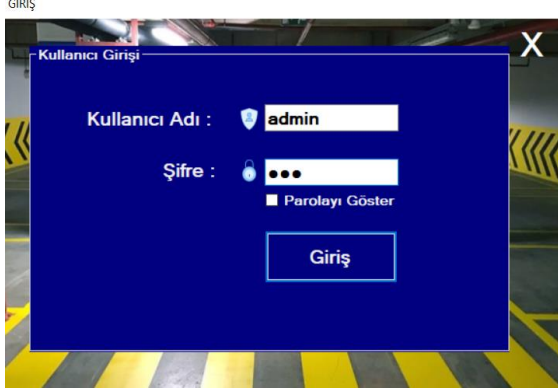
- Müşterinin talebi doğrultusunda kişisel bilgilerini silmemiz gerekebileceğini fark ettik. Bu durumda tablolardan birinden silince diğerinde silinmediği gördük. Diğer tablodan da silebilmek için plaka bilgisini de aldık. If döngüsüyle de kullanıcı bilgilerinin doğrulunu kontrol ettirdik. Kullanıcı bilgileri hatalı ya da eksik girildiğinde kullanıcıyı bilgilendiren hata mesajları verdirdik.

- Admin giriş yaptığı zaman kullanıcı değiştirmek için uygulamayı kapatıp açması gerekiyordu. Oturum kapatma kısmı ekleyerek sorunu çözdük.

- Araç çıkışı kısmında kullanıcı kaydı yapılırken giriş yapan araçların plakası gözüküyordu. Kodlardaki hataları araştırıp hatayı düzelttik.

- Araçlar giriş-çıkış yaptıkları zamanları veri tabanında tutmamız gerekiyordu biz de bu bilgileri elle doldurmak yerine daha doğru olması için bilgisayar saatinden çekmek istedik bunu yaparken farklı bilgisayarlarda hatayla karşılaştık.

3. Tasarım



Şekil 1

Şekil 1: Uygulamayı kullanmak için yetkisi bulunan kişinin giriş yapabileceği giriş ekran.



Şekil 2

Şekil 2: Ana sayfa tasarımı.



Şekil 3

Şekil 3: Müşteri kaydının yapılıp, aracın otoparka girişinin sağlandığı ekran.



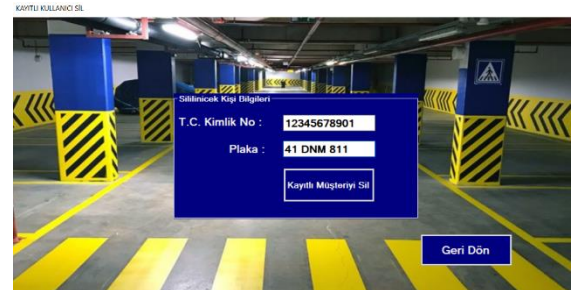
Şekil 4

Şekil 4: Kayıtlı kullanıcının plakasını seçip otoparka giriş yapılmasının sağlandığı ekran.



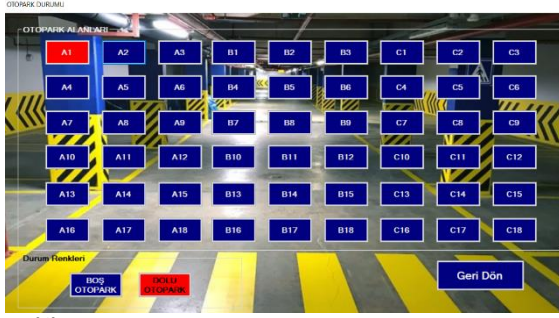
Şekil 5

Şekil 5: Otoparkta bulunan aracın çıkışının sağlandığı aynı zamanda giriş tarihinin, otoparkta kalma süresinin ve ödeyeceği tutarı gösteren araç çıkış ekranı.



Şekil 6

Şekil 6: Gerekli durumlarda kayıtlı olan müşterinin TC kimlik numarasının ve plakasını girerek müşterinin bütün bilgilerini sildiren kayıtlı kullanıcı silme ekranı.



Şekil 7

Şekil 7: Otoparkta boş ve dolu alanların görülebildiği otopark durumu ekranı.



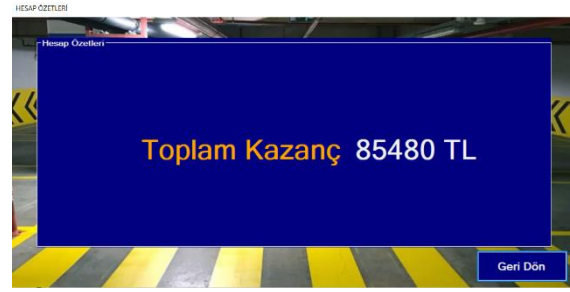
Şekil 11

Şekil 11: Uygulamaya giriş yapma yetkisine sahip olan kullanıcının şifresini değiştirebildiği ekran.



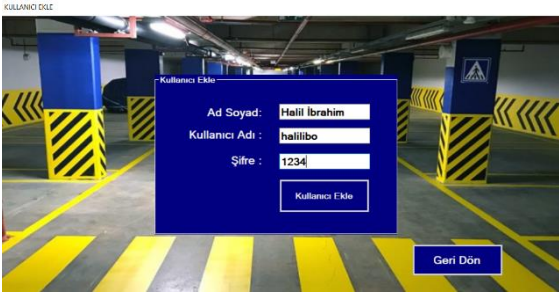
Şekil 8

Şekil 8: Ayarların bulunduğu ekran.



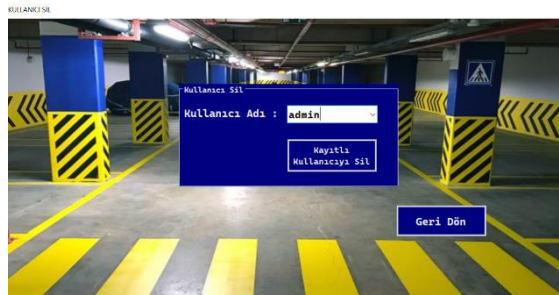
Şekil 12

Şekil 12: Bu zamana kadar ne kadar kazanıldığını gösteren hesap özetleri sayfası.



Şekil 9

Şekil 9: Uygulamayı açma yetkisine sahip başka bir kullanıcı eklemek için tasarlanan kullanıcı ekle sayfası.



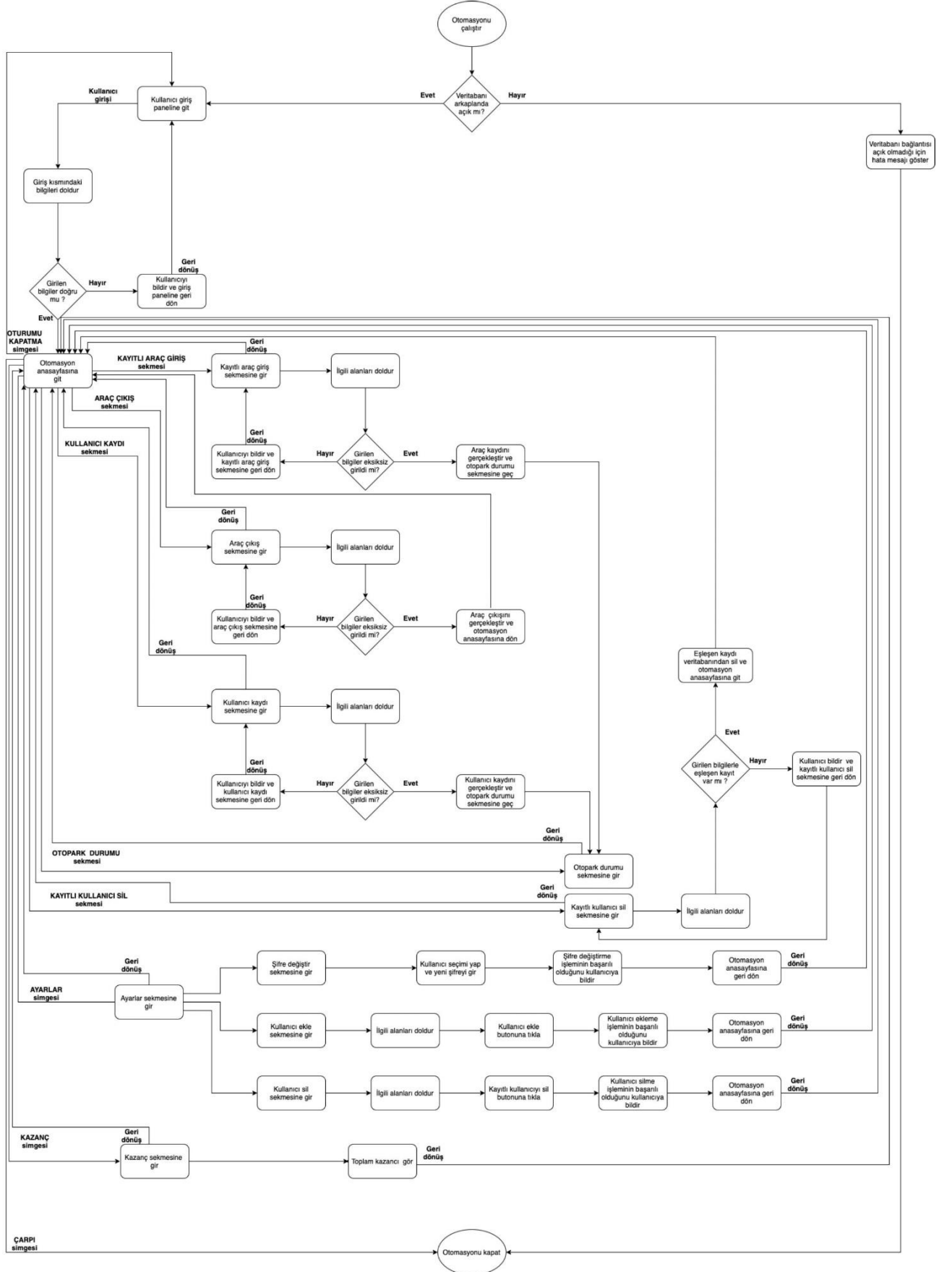
Şekil 10

Şekil 10: Uygulamaya giriş yapma yetkisine sahip olan kullanıcı silme ekranı.

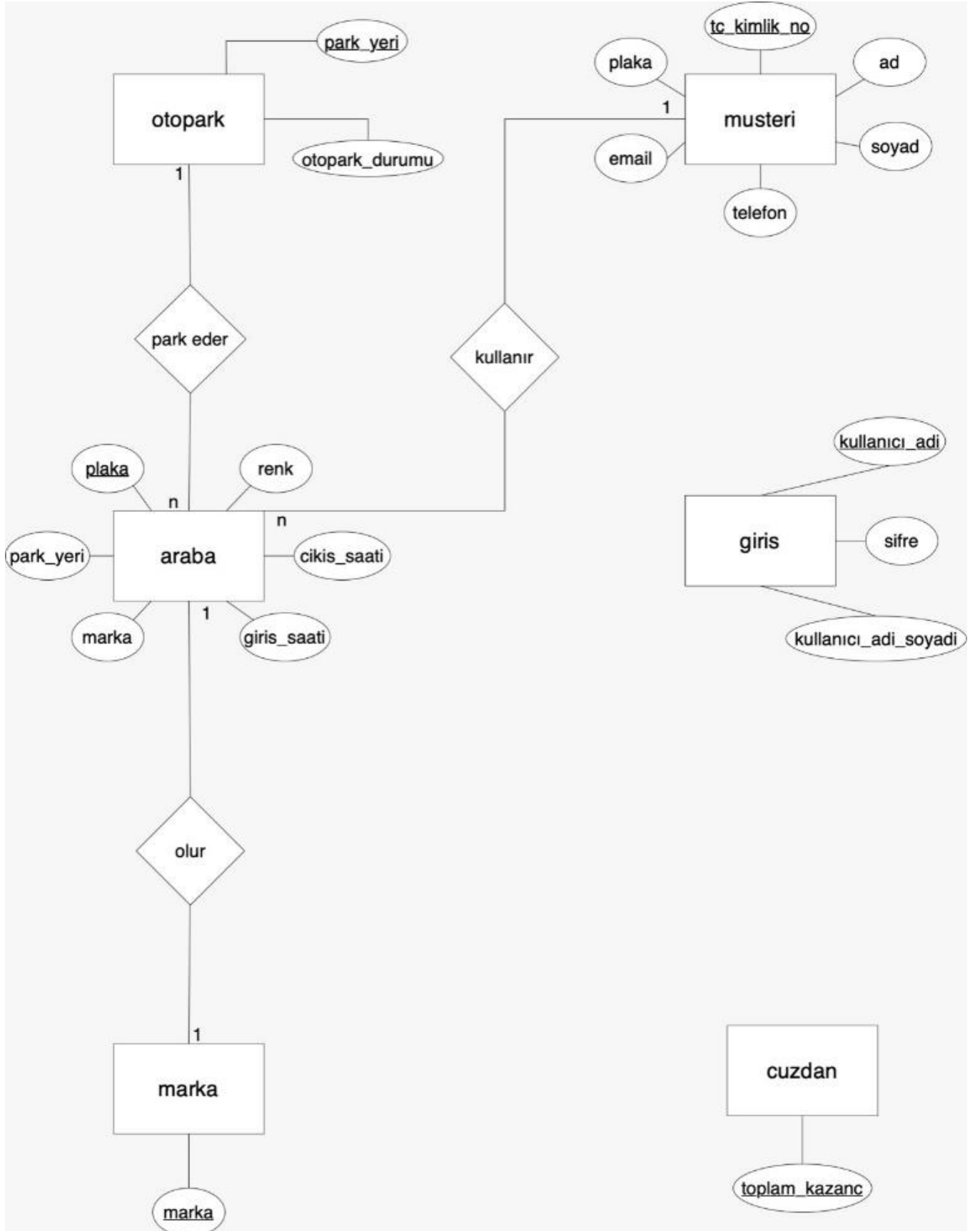
4. Yazılım Mimarisi

Otomasyonu geliştirirken kullandığımız veri tabanı yönetim sistemi Microsoft SQL Server'dır. Son kullanıcının kullanacağı program ara yüzünü C# ile geliştirdik. Otomasyon ara yüzünün veri tabanı ile olan bağlantıları ve veriler üzerindeki işlemlerinde tekrar tekrar aynı kodları yazmamak için genel bir sınıf oluşturduk. Bu sınıfta veri tabanı bağlantısını açma ve kapatma, yollanan sorguların şemini yapma, veri okuma gibi işlemlerin metotlarını yazdık. Böylelikle form ekranlarının backend kodlarında bu sınıfın nesnesini oluşturarak ve sadece metotları çağırarak kodun daha anlaşılır ve tekrarlayan alanlardan uzak olmasını sağladık.

5. Akış Şeması



6. Veri Diyagramı



7. Genel Yapı

Projemize başlarken öncelikle veri tabanının tablolarını belirledik. Otomasyonumuzun nasıl işleyeceğini yani sistemimizi konuştuktan sonra belirlediğimiz tabloların niteliklerini belirledik. Veri tabanımızı tasarlarken normalizasyon kurallarına göre hareket ettik. Tablolarımızın niteliklerini belirledikten sonra draw.io'da projemizin varlık-iliski (ER) diyagramını oluşturduk. Varlık-iliski diyagramını bitirdikten sonra veri tabanımızın tablolarını ve niteliklerini Microsoft SQL Server'da oluşturduk. Veri tabanı tasarımı oluştururken aşağıdaki adımlarımıza göre hareket ettik;

- Uygulamayı kullanacak olan görevlilerin adı, soyadı, kullanıcı adı ve şifresi vardır.
- Her müşterinin TC Kimlik numarası, adı, soyadı, cep telefonu, mail adresi tutulur.
- Her müşterinin arabası vardır. Bu arabanın plakası, markası, rengi vardır.
- Her aracın bir modeli vardır.
- Bir model birden fazla araçta olabilir.
- Otoparkta her bölgenin kendine özel adı vardır.
- Müşteri giriş yaptığı zaman bir park yeri kullanır.
- Her müşterinin otoparka giriş-çıkış saati vardır.
- Her müşteri birden fazla kez giriş çıkış yapabilir.
- Bir araç aynı zamanda farklı otopark yerlerinde bulunamaz.
- Aynı kullanıcı adına sahip birden fazla kişi olamaz.
- Bir plaka yalnızca bir kişiye ait olabilir.
- Farklı zamanlarda aynı otopark yerine farklı araçlar giriş yapabilir.

Veri tabanı işlemlerimizi bitirdikten sonra projenin masaüstü uygulamasını geliştirmek için C# form kısmına geçtik. Projenin ara yüzünü tasarlanması evresine geldik. Ara yüzü tasarlarken programı kullanacak kişilerin rahat ve anlaşılır şekilde programı kullanmasına yönelik tasarımı gerçekleştirdik. Otomasyonu açtığımızda bizi giriş ekranı karşılamaktadır ayrıca sağ üstteki "X" işaretinden uygulama kapatılabilmektedir. Giriş yapıldıktan sonra;

- "KULLANICI KAYDI" sayfasından daha önce otoparkı kullanmamış ya da kaydı silinen kullanıcıyı kayıt edilmesi ve aynı zamanda otoparka giriş yapması sağlanır. Kayıt yapılan aracın modeli daha önce eklenmediyse yazıldıktan sonra "+" butonuna basılarak eklenebilmektedir.
- "KAYITLI KULLANICI GİRİŞ" sayfasında daha önce kayıt edilmiş kullanıcın gireceği otopark yeri seçilerek kolayca otoparka girişi sağlanır.

- Gerekli durumlarda müşteri isterse bilgilerini silmek isterse "KAYITLI KULLANICI SİL" sayfasından müşterinin bütün bilgileri silinebilmektedir.
- "OTOPARK DURUMU" sayfasından da rahatça anlaşılabilir, renklendirilmiş ve sade otopark krokisinden hangi otopark yerinin dolu hangisinin boş olduğunu görebilirsiniz.
- "ARAÇ ÇIKIŞI" sayfasında otoparka giriş yapmış aracın plakasını seçerek ne kadar süre otoparkta kaldığını ve ne kadar ödeyeceğini görebilir ve otoparktan çıkışını sağlayabiliriz.
- "HESAP ÖZETLERİ" sayfasında da toplam ne kadar kazanıldığını görebiliriz.
- "AYARLAR" kısmında giriş yapabilen görevliler ekleyebilir, ekli olan kullanıcının şifresini güncelleyebilir isterseniz de kayıtlı kullanıcıyı silebilirsiniz.
- Giriş yapan kullanıcıyı değiştirmek istediğinizde de ana sayfada sol alt kısımda bulunan "OTURUM KAPATMA" simgesinden oturum kapatabilir ya da sağ üstte bulunan "X" simgesine tıklayarak uygulamayı kapatabilirsiniz.

Referanslar

- [1] H. Maqbool, «Connection Class in C#», 22 Temmuz 2015. [Çevrimiçi]. Available: c-sharpcorner.com/UploadFile/f26fb0/connection-class-in-C-Sharp/.
- [2] A. Yazılım, «C# Kullanıcı Adı ve Şifre ile Giriş Yapma(Kullanıcı Ekleme ve Şifre Değiştirme)», 2018.
- [3] M. Yücedağ, «C# Ders 31 DataGridView Ekleme Silme Güncelleme ve Arama», Youtube, 2014.
- [4] M. Bükülmez, «mustafabukulmez», 28 Kasım 2018. [Çevrimiçi]. Available: https://mustafabukulmez.com/2018/11/28/c-textbox-sadece-sayi_harf-girisi-yaptirmak/.