

Resim Yapıda Veritabanı Uygulaması

Ebubekir Yaşar^{1*}, Turgut Özseven²

¹Tokat Gaziosmanpaşa Üniversitesi Mühendislik ve Doğa Bilimleri Fakültesi Makine Mühendisliği

²Tokat Gaziosmanpaşa Üniversitesi Mühendislik ve Doğa Bilimleri Fakültesi Bilgisayar Mühendisliği

*ebubekir.yasar@gop.edu.tr

Özet – İşletmelerin hesaplarının tutulduğu otomasyon yazılımlarının en önemli bileşenleri şüphesiz veritabanı yazılımlarıdır. Veritabanı sistemleri, verilerin paylaşılması, yetkilendirilmesi, işlenmesi, hızlı arama, sorgulamaya imkan vermesi, veri güvenliği, yedekleme gibi konularda birçok avantaja sahiptir. Bu avantajların yanında bazı problemleri de beraberinde getirmektedir. Gelişmiş veritabanı uygulamalarının pahalı olması ve ekonomik olanların ise güvenlik riskleri ve zamansız hata potansiyelleri özellikle küçük çaplı işletmelerde veritabanı kullanımını problemli veya pahalı bir seçenek haline getirmektedir. Bu nedenle gelişmiş veritabanı kullanımının küçük işletmeler için avantaj ve dezavantajının tartışılmasına neden olabilmektedir. Bu çalışmada şişme çocuk parkı ve trambolin hizmeti veren bir işletmeye, müşteri takibi ve gelir hesabı amacıyla resim yapıda veritabanına sahip bir otomasyon çalışması yapılmıştır.

Resimler ekranda her biri piksel denem noktalarından oluşan iki boyutlu matris formunda yapıya sahiptir. Bunun yanında resimler üzerinde hızlı işlem yapabilen ve programcılarının kullanımına izin verilen çeşitli metot ve özelliklere sahip sınıflar geliştirilmiştir. Bu sayede verilerin kaydı ve okunup raporların hazırlanması standart resim işleme metotlarına göre daha hızlı olabilmektedir.

Yapılan çalışmada şişme çocuk parkı hizmeti veren bir işletmeye ait günlük hesaplar hazırlanan uygulamada, resim yapıyı oluşturan pikseller üzerinde tutularak bir otomasyon sistemi gerçekleştirilmiştir. Şişme çocuk parklarında çocuk müşterilerin takibi amacıyla oluşturulan otomasyon açık havada çalışan oldukça düşük konfigürasyona sahip bir bilgisayar üzerinde yürütülmüştür. Hazırlanan programın toplam boyutu veritabanı dosyası dahil <1MB dır.

Anahtar Kelimeler – veritabanı, resim, otomasyon yazılımı

Abstract – The most important components of automation software, which hold the accounts of the enterprises, are undoubtedly database software. Database systems have many advantages in terms of sharing, authorizing, processing data, enabling quick search, querying, data security, and backup. It also brings with it some problems. The cost of advanced database applications and the security risks and untimely error potentials of the economic ones make the database use a problematic or expensive option, especially in small enterprises. Therefore, the advantages and disadvantages of advanced database usage for small enterprises can be discussed. In this study, an automation study of an inflatable playground was performed.

The images have a two-dimensional matrix in the form of dots on the screen, each consisting of dots called pixels. In addition to this, classes have been developed with various methods and features that allow the use of programmers to process images quickly. In this way, recording data and reading and preparing reports can be faster than standard image processing methods.

In this study, an automation system was realized by keeping the daily accounts of an enterprise providing inflatable playground services on the pixels that make up the image structure. The automation, which was created for the monitoring of children's customers in inflatable playgrounds, was carried out on a computer with a very low configuration working outdoors. The total size of the prepared program is <1MB, including the database file.

Keywords – database, image, automation application

I. GİRİŞ

Veritabanları, 1960'lı yılların başlarından itibaren önemli aşamalar geçirmiş ve günümüzde internetin de yaygın kullanımı ile bulut veritabanı, kendi kendini yöneten veritabanları gibi yapılar verileri toplama, depolama, yönetme ve kullanma konusunda çığır açmışlardır[1]. Zamanla gelişen donanım ve yazılım teknolojileri kullanıcı karşısına hep artan maliyetlerle çok hızlı eskijen ve ekonomik değerini yitiren yapılar çıkartmıştır. Yaklaşık on yıl sonra şimdiki güncel bilgisayar sistemleri çöp olmakta ve hurda fiyatına düşmektedir. Veritabanı yapılarının asli görevleri(ekleme, silme, arama, güncelleme...) değişmemekle birlikte yeni eklenen özellikler lineer olmayan bir şekilde hafıza ve bellek

gereksinimlerini artırmaktadır. Bu hızlı eskime sadece veritabanı yapılarında değil işletim sistemi, uygulama geliştirme ortamları, program çatıları... hızlı bir şekilde üzerinde çalışacağı platformun daha fazla işlem kapasitesine haiz hafıza ve işlemciye sahip olmasını gerektirmektedir. Yeni özelliklere her işletme ve uygulamanın ihtiyaçları olmasa da sistem üzerinde kurulum alanları eski versiyonlara göre oldukça fazla olmaktadır. Buda güncel yazılımların eski bilgisayarlar üzerinde çalışmaması ve sistemlerin yenilenmesi gerekliliği anlamına gelmektedir. Bu çalışmada eski bilgisayarlar üzerinde çalışabilen düşük boyutlara sahip işlevsel bir yazılım oluşturma yoluna gidilmiştir. Küçük işletmelerin girdilerini ve çıktılarını tutmak için gelişmiş

veritabanı yapılarının kullanımı bazen masraflı olabilmektedir. Güncel veritabanlarının bakımı, onarımı, maliyeti, güvenliği, yedekleme ve sistem gereksinimleri gibi birçok problem küçük ölçekli işletmeler için hazırlanan otomasyon yazılımının ve üzerinde çalışacağı bilgisayar sisteminin toplam maliyetine yansımaktadır. Kendine has standart dışı yöntemlerle hazırlanacak veritabanı oluşumları sisteme başka yetkisiz kişilerin sızmaları için önemli bir engel oluşturmaktadır. Küçük bir işletmenin verilerinin bilgisayar ortamında tutulabilmesi amacıyla hazırlanan yazılımın ekonomik, bakım gereksinimi olmayan, sistem gereksinimi minimum olan güvenlik ve maliyet açısından resim yapı üzerinde verileri saklayan bir otomasyon uygulaması gerçekleştirilmiştir.

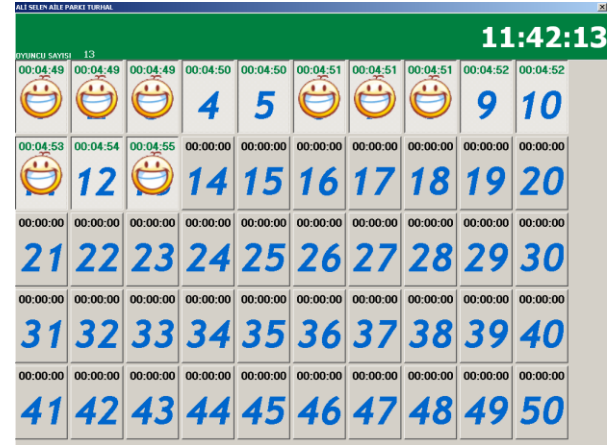


Şekil 1. Şişme oyun parkı (solda), trampolin (sağda)

II. MATERYAL VE METOT

Bu çalışmada bünyesinde çok fazla ürün/hizmet çeşidi olmayan küçük bir işletmeye (şişme oyun parkı ve trampolin) müşteri ve kazanç takibi amaçlı bir otomasyon programı hazırlanmıştır. Bu tür parklarda bazı müşteriler fazla, bazı müşteriler ise süresi dolmadan çıkarılmakta ve bu durum zaman zaman özellikle kalabalık dönemlerde probleme dönüşmektedir. Hem girdileri tutmak, hem de parktaki her müşterinin süresi kadar ne fazla ne eksik oynayabilmelerini sağlamak için bir otomasyon yazılımı gerçekleştirilmiştir. Hazırlanan program üzerinde parkta aynı anda en fazla oynayabilecek sayıda üzerlerinde sıralı sayılar olan butonlar vardır. Bu arayüz Şekil 2’de gösterilmiştir. Parkta oynamak isteyen çocuk için boş bir butona tıklanır ve buton üzerinde, kalan süre yanıp sönmeye başlar. Süresi dolmayan çocuk için basılan buton üzerinde, gülen yüz ikonu çıkmaktadır. Butona basıldıktan sonra yazıcıdan yapışkanlı kağıda (etikete) butonun numarası, giriş ve çıkış saatleri basılır. Örnek bir etiket Şekil 2’de gösterilmiştir. Veritabanında ilgili saatteki müşteri sayısı bir artırılarak veritabanına işlenir. Yazıcı çıktısı etiket, yazıcıdan sökülerek çocuğun giysisine yapıştırılır. Süresi dolan müşteriye ait butonun rengi değişir, üzerindeki gülen yüz kızgın kırmızı yüze döner ve park görevlisi süresi dolan çocuğu üzerindeki etiketin numarası yardımıyla uyararak parktan çıkmasını sağlar. Parktan çıkarılan çocuğun üzerindeki etiket sökülüp atılır ve ilgili butona tıklanarak yeni bir müşteri için hazır hale getirilir.

Otomasyonda kullanılan termal yazıcı etiket basmak için özel olarak tasarlanmış kompakt yapıya sahiptir. Yazıcıya ait etiket rulusunda her bir etiketin boyutu 10x7.5cm boyutlarına sahiptir.



Şekil 2. Park otomasyonu program arayüzü ve basılan etiket görüntüsü (altta)

Bu programda veritabanı olarak resim dosyası kullanılmıştır. Müşterilere ait veriler, resim dosyası içerisinde, resmin piksel (RGB) yapıları korunarak saklanmıştır. 24 bit BMP formatında renkli yapıda yaklaşık 275 KB büyüklüğünde disk üzerinde yer kaplayan bir resim dosyasında haftanın 7 günü 24 saat kayıt alabilecek şekilde 10 yıllık verinin saklanabileceği bir veritabanı otomasyon sistemi yapılmıştır. Veritabanı olarak kullanılan resmin boyutu hep sabit kalmaktadır.

Resim Verilerinin Disk Üzerinde Tutulması

Resimler piksel denem adreslenebilen küçük noktalardan oluşmaktadır [2]. Resimlerin kayıt yapısı 2 boyutlu matris formuna yakın bir yapıya sahiptir. Her bir piksel resim üzerinde 2 indisle adreslenir ve bu indisler satır ve sütun numaralarıdır. Bu numaralar resmin sol-üst köşesinden sıfır değerinden başlayarak sağa ve aşağı doğru artarak devam eder. Sağa doğru sütun indisi artarken aşağı doğru satır değeri artmaktadır. Diğer bir ifadeyle sağa doğru x bileşeni artarken aşağı doğru y bileşeni artmaktadır. Örnek olarak resim içerisinde bir X harfinin bileşenleri Şekil 3 de gösterilmiştir. Bu resimde renkler ikili değerlerden oluşmaktadır. Bu tür resimler siyah beyaz resimler olarak adlandırılır.

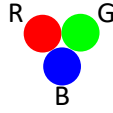
| 0x0 | 0x1 | 0x2 | 0x3 | 0x4 |
|-----|-----|-----|-----|-----|
| 1 | 0 | 0 | 0 | 1 |
| 0 | 1 | 0 | 1 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 1 | 0 | 0 | 0 | 1 |

4x4

X

Şekil 3: İki renkten oluşan X harfine ait bir resmin bileşenleri ve piksel adresleri.

Resimler renkli, gri ve siyah-beyaz olmak üzere üç sınıfa ayrılmaktadır[2]. Renkli resimlere ait her bir piksel üç alt pikselden oluşmaktadır. Bu pikseller R(Red=Kırmızı), G(Green=Yeşil), B(Blue=Mavi) olarak isimlendirilir. Ekranlar renkleri ışık yardımıyla gösterirler. Işıklı oluşturulan bir renk üç ana renkten(RGB) oluşmaktadır. 24 bit renk derinliğine sahip bir resmin bir pikseli $24(8+8+8)$ bitten oluşmaktadır. Bu pikselin kırmızı, yeşil ve mavi bileşenleri 8 bit değer taşımaktadır. Örneğin her bir pikselin kırmızı değeri 0 – 255 arasında toplam 256 farklı değer almaktadır. Dolayısı ile bir piksel $256 \times 256 \times 256 = 16.777.216$ adet renkten oluşmaktadır. Yukarıda Şekil 3'deki 5×5 resim iki renkten oluşsaydı hafızada toplam $5 \times 5 = 25$ bit yer kaplayacaktı. Renkli resim olsaydı $5 \times 5 \times 24 = 600$ bit değere sahip olacaktı. Bir pikselin R,G,B değerleri birbirine eşitse o piksel gri tonu olmaktadır. Örneğin orta koyulukta gri değeri için RGB değerleri(127,127,127) değerlerini almalıdır. Bir pikselin tüm RGB değerleri 255(255,255,255) ise beyaz renk, 0(0,0,0) ise siyah rengini almaktadır. Dolayısı ile Şekil 3'deki resim gri bir resim olsaydı hafızada $5 \times 5 \times 8 = 200$ bit yer kaplayacaktı. Gri piksellere sahip bir resim noktasını hafızada tutmak için 8 bit yeterlidir. Çünkü tüm RGB değerleri birbirinin aynıdır.



Şekil 4: Renkli bir resme ait piksel yapısı ve bileşenleri

Resmin bir pikseline ait RGB değerleri onaltılı sayı sistemine çevrilerek yan yana yazılıp topluca tamsayıya çevrilir[3]. Örneğin beyaz rengin tamsayı karşılığı bulmak isteyelim. Beyaz renge sahip bir pikselin alt pikselleri $R=G=B=255$ değerine sahiptir. Şimdi 255 in onaltılı sayı sistemindeki karşılığı FF değerlerini yan yana yazdıktan sonra ve beyaz rengin onaltılı sayı sistemindeki kod karşılığı FFFFFFFF olacaktır. Bu sayının onluk sayı sistemindeki karşılığı 16.777.215 değeridir. Örnek olarak FF0000 değeri saf kırmızı rengini, 00FF00 ise saf yeşil rengini göstermektedir. Diğer tüm renkler bu üç rengin bileşiminden oluşan ara renklerdir.

Resim formatları, sıkıştırılmış(jpg, png, gif...) ve sıkıştırılmamış(bmp) olmak üzere ikiye ayrılırlar[4]. Sıkıştırılmış olanlarda kendi aralarında kayıplı ve kayıpsız olmak üzere ikiye ayrılmaktadır[5]. Sıkıştırılmamış resim formatlarında hem resim üzerindeki orijinal piksel değerleri hem korunur, hem de piksel verilerinin elde edilmesi hızlıdır[6]. Buna karşın sıkıştırılmamış formatlar disk üzerinde daha fazla yer kaplarlar. Özellikle kayıplı sıkıştırılmış formatlarda orijinal piksel değerleri korunmazlar[7].

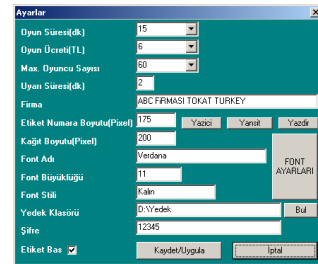
Veritabanları birbiriyle ilişkili verileri tutan depolama yapısıdır. Veriler tablo denilen yapılarda kendileri için oluşturulan alanlarda saklanmaktadır[8][9]. Yapılan çalışmada bu alanlar yerine resim yapılara ait piksel değerleri kullanılmıştır. Bu amaçla uygulamada her bir pikseli 24 bit renk derinliğine sahip bmp formatında 775×121 boyutlarında bir resim dosyası kullanılmıştır. Dolayısı ile bir piksel içerisine üç farklı karakter sığabilmektedir. Sayı kaydı olarak

düşünüldüğünde bir piksel içerisine en fazla 255 sayısı yazılabilir. Bu resmin hafızada kapladığı alan tam olarak hesaplandığında $275KB((775 \times 121 \times 24 / 8) / 1024)$ büyüklüğüne sahip olduğu görülecektir. Yapılan uygulamada başlangıçta resmin tüm pikselleri beyaz(FFFFFF) değerine sahiptir. Tamamen beyaz pikseller boş kayıt alanlarını göstermektedir.

Resim Üzerinde Kayıt Formatının Belirlenmesi ve Verilerin Tutulması

Resim üzerinde her bir satır bir aylık verileri alması için tasarlanmıştır. Her gün 24 saat ve bir ay en fazla 31 gün çektiği için resmin genişliği $31 \times 25 = 775$ seçilmiştir. Her gün için belirlenen 25 pikselin 24 tanesi çalışma saatlerini tutarken diğer piksel park görevlisine ait kodu tutmaktadır. Toplam 3 adet park görevlisi kodu ve çalıştığı saat sayısı tek bir piksele yazılabilmektedir. Fiyat ve süre değişimleri saat başında yapılabilmektedir. Her saat için kullanılan bir piksel müşteri sayısını, birim ücreti ve birim süreyi tutmaktadır. Bu değerler gün başlangıcında değiştirilmezse bir önceki günün değerleri geçerli olmaktadır. Bir satırda 31 gün ve her gün 24 saat için ayrılan piksel değerleri sayesinde her bir müşteri girişinde kayıt yaparken saat ve tarih bilgilerini tekrar yazmaya gerek kalmamaktadır. Bu nedenle gece saatlerinde özellikle turizmin yoğun olduğu kesimlerde müşteri olabilir ihtimaline karşı 24 saat üzerinden kayıt alanı bırakılmıştır. Sadece jeton adetleri(müşteri sayısı) tutulmak istenseydi resmin boyutları çok daha küçük olabilirdi. Bir saatte en fazla 1000 müşteri öngörülmüş ve bu amaçla R bandından 8 bit G den 2 bit olmak üzere toplam 10 bit alan kullanılmıştır. G deki diğer bitler ve B bandı bitleri(toplam 14 bit) oyun süresi ve birim ücreti yazmak amaçlı kullanılmıştır.

Resimde toplam 121 satır bulunmaktadır. Son satır programı çalıştıran ve kullanan işletmenin adı, yazıcı ayarları, kullanıcı şifresi, font ayarları, yedekleme klasörü, maksimum oyuncu sayısı, uyarı süresi gibi saklanması amacıyla eklenmiştir. Bu alana veri eklemek için ayarlar sayfası kullanılmaktadır. Bu sayfa Şekil 5'te gösterilmiştir. Burada her bir alana ait uygun uzunlukta alan tanımlanarak kayıtlar piksellerin RGB değerlerine sırayla yazılmaktadır. Her bir satırda bir aylık veriler tutulduğuna göre toplam $120/12 = 10$ yıllık veriler diğer 120 satırın tamamında tutulabilmektedir. Resmin uzunluğu artırılarak bu değer daha fazla yıla çıkartılabilir.



Şekil 5: Park otomasyonuna ait ayarlar sayfası

Arama / Sorgulama

İstenen güne ve günlere ait arama ve sorgulama yapılabilmektedir. Günlere ait veriler adresi sabit olan satır ve

sütunlardan hızlı bir şekilde okunabilmektedir. 275KB toplam boyuta sahip bir resim üzerinde istenen piksellere ulaşmak ve çözümlmek 1sn(Pentium 4, 3 GHZ, 512 MB RAM ve Windows XP üzerinde) altında süre almaktadır. Elde edilen veriler txt formatında istenirse kaydedilebilecek şekilde ayarlanmıştır. “Özet yazdır” butonu ise özet bir hesap çıkartıp mevcut yazıcıya basabilmektedir. Şekil 6’da konu ile ilgili ekran görüntüleri verilmiştir.

| GÜN | Jeton | Gün Sonu |
|-------------------------|-------|----------|
| 01.07.2010 | 52 | 156 TL |
| 02.07.2010 | 123 | 369 TL |
| 03.07.2010 | 150 | 450 TL |
| 04.07.2010 | 130 | 390 TL |
| 05.07.2010 | 76 | 228 TL |
| 06.07.2010 | 72 | 200 TL |
| Toplam Jeton : 603 Adet | | |
| Toplam Tutar : 1793 TL | | |

Şekil 6. Arama sorgulama sayfası ve özet çıktı örneği(altta)

Rapor sayfasında hangi saatte kaç adet müşteri geldiği ve o saat içerisinde aktif tarife ücreti ve tarife süresi tek tek saat sütunu içerisinde görüntülenmektedir. Ayrıca toplam kişi sayısı, toplam elde edilen kazanç değerleri gibi bilgiler rapor sayfasının durum çubuğunda yer almaktadır. Bu sayfada istenirse veritabanındaki tüm veriler silinerek yeni dönem için başlangıç yapılır. Silme işlemi gerçekleştirildiğinde güncel tarihten itibaren veritabanı tekrar formatlanır.

Veritabanı Güvenliği

Verilerin her ne kadar resim yapıda saklandığı bilinse dahi format yapısı bilinmediğinden içerisine girilip veriler üzerinde işlem yapmak kolay değildir. Veritabanı olarak kullanılan resim dosyasının uzantısı (.eby) dir. Dolayısı ile resim dosyası olarak disk üzerinde gözükmektedir. İşletim sistemi dosyayı tanımadığı dosya simgesi tipinde göstermektedir. Dosya böylece kötü niyetli kişilerden saklanmıştır. Ayrıca veritabanı dosyasının belirlenen süreler içerisinde istenen bir klasöre yedeklenmesi sağlanmıştır. Harici belleklere alınıp taşınması ve yedeklenmesi de kolaydır. Program veritabanını RAM hafızada tutarak her 10 dakikada bir disk üzerinde kendi klasörüne ve belirlenen yedek klasörüne kaydetmektedir. Yedek klasörün yolu ayarlar sayfasında ilgili alandan belirlenebilmektedir. Ayrıca silinen dosyaları yedek üzerinden geri getirebilme yetisi kazandırılmıştır. Tüm bu işlemler konfigürasyonu düşük bir bilgisayarda gerçekleştirilmiştir. Buda şüphesiz veritabanının küçük boyutta olmasından kaynaklanmaktadır.

Sistem Gereksinimleri

Program boyut olarak 1MB civarındadır. Bu büyüklüğe exe uygulama dosyası, veritabanı ve diğer yardımcı dosyalar her şey dahildir. Hazırlanan yardım dosyası yaklaşık 300KB yer kaplamakta ve toplam kurulum programı 1.5 MB değerine ulaşmaktadır. Program 500Mhz işlemci ve 256 MB RAM üzerinde çalışabilmektedir. Hazırlanan yazılımın çalıştırıldığı en küçük konfigürasyon değerleri Pentium4 3Ghz işlemci, 60GB disk ve 512 MB RAM olan sistemdir.

Ekonomik Değeri

Hazırlanan yazılım için veritabanı gideri bulunmamaktadır. Ayrıca çok küçük konfigürasyonda da çalışabildiğinden ihtiyaç duyulan bilgisayar sistem maliyeti olabildiğince düşük olmaktadır. Günümüzde bu yazılımı çalıştırabilecek konfigürasyona sahip bilgisayarlar hurda fiyatına alınıp satılmaktadır. Programı oluşturan mantığın en özgün tarafı eski bilgisayarların da işe yaramasıdır. Yapılan çalışma diğer veritabanı ile oluşturulmuş yazılımların sahip olduğu avantajların birçoğunu eski sistemler üzerinde karşılayabilmektedir.

Boyut

Veritabanı olarak kullanılan resmin boyutu 275KB büyüklüğündedir. Bu büyüklük veritabanındaki tüm pikseller veri taşırken dahi değişmemektedir. Dolayısı ile artan veri miktarının veritabanı performansını modern veritabanı yapılarındaki gibi fazla kasmadığı görülmüştür. Bu durum programa ait veritabanının yedeklenmesi, program tarafından kullanılmasının zaman içerisinde performans kaybı oluşturacağı gibi problemlerle karşılaşılacağı anlamına gelmektedir.

Ayrıca programın uygulama dosyasının(1 adet dll siz) boyutunun küçük olması(600 KB) ve basit yapıya sahip olması, sisteme kalıcı performans sağlamaktadır.

III. SONUÇ

Verilerin resim yapısı içerisinde tutulmasında kısıtlar mutlaka oluşturulmalıdır. Her bir kaydın sınırlı alanı olması gerekmektedir. Master detay tablo yapıları bu yöntem içerisinde kullanılamaz. Bunun yanında aynı resim içerisinde sınırlı lookup tablo oluşturulabilir. Bu çözüm hizmet ve ürün çeşidi fazla olmayan müşteri borç alacak bilgilerinin tutulmasının zorunlu olmadığı işletmeler için uygun olabilir. Bu işletmelere örnek restoran, kafe, çay ocağı, ücretli oyun parkları gibi alanlar olabilir.

Hazırlanan otomasyon yazılımının program ve veritabanı büyüklüğünün küçük olması düşük konfigürasyonlara sahip bilgisayarlar üzerinde bile çalışabileceğini göstermiştir. Hazırlanan yazılımın her gün 7/24 işletme verilerini tutabilecek olması ve sadece 275 KB büyüklüğünde yer kaplaması çok büyük bir avantajdır.

Oluşturulan yapının otomasyona ve programcıya özgü tasarım gerektirmesi standart yönünün olmaması programı güvenli kılmaktadır. Hızlı resim işleme metodlarına sahip sınıfların kullanılması yapının performansını üst seviyede tutmaktadır.

KAYNAKLAR

- [1] (2019) Veritabanı. Oracle Website [Online]. Available: <https://www.oracle.com/tr/database/what-is-database.html>.
- [2] Darko Stipaničev. Introduction to Digital Image Processing and Analysis. Chronolab A.G. Zug Switzerland .1994.
- [3] BURKE M., Image Acquisition-Handbook Of Machine Vision Engineering. Chapman & Hall. 1996.
- [4] Vipin Tyagi . Understanding Digital Image Processing. CRC Press. 2018.
- [5] EFFORD N., Digital Image Processing. Addison-Wesley, USA. 2000.
- [6] RITTER G.X., WILSON J. N., Hand Book Of Computer Vision Algorithms In Image Algebra. Crc Press. USA. 1996.
- [7] RUSS J.C., The Image Processing Handbook. CRC Pres and IEEE Pres. USA. 1999.
- [8] R Ramakrishnan, J. Gehrke. Database Management Systems, McGraw-Hill Education. 2002.
- [9] H.Garcia-Molina, J.D. Ullman, J. Widom, Database Systems: The Complate Book. Prentice Hall. 2009.