**EGE ÜNİVERSİTESİ**

**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

****

**WINDOWS PROGRAMLAMA**

**2020-2021 GÜZ YARIYILI**

**PROJE RAPORU**

**PROJE KONUSU: Apartman/Site Yönetim Takip Sistemi**

**HAZIRLAYANLAR**

**05170000023 – Müge Selen ELMAS**

**TESLİM TARİHİ**

**09/02/2021**

**İÇİNDEKİLER**

[1. GİRİŞ 2](#_Toc502221584)

[1.1 Projenin Konusu ve Önemi 2](#_Toc502221585)

[1.2 Projenin Kapsamı 2](#_Toc502221586)

[1.3 Projenin Hedefleri 2](#_Toc502221587)

[1.4 Projenin Alan (Domain) Bilgisi 2](#_Toc502221587)

[2. BENZER ÇALIŞMALAR 3](#_Toc502221588)

[3. YÖNTEM VE TEKNOLOJİLER 5](#_Toc502221589)

[4. VERİ TABANI YÖNETİM SİSTEMİNİN OLUŞTURULMASI 5](#_Toc502221590)

[5. TAMAMLANAN ÇALIŞMALAR 9](#_Toc502221590)

[5.1 Analiz 9](#_Toc502221591)

[5.2 Tasarım 10](#_Toc502221592)

[5.3 Geliştirim 15](#_Toc502221593)

[5.4 Sınama 15](#_Toc502221594)

[6. GELİŞTİRİLEN PROGRAMIN TANITIMI 16](#_Toc502221595)

[7. PLANLANAN ÇALIŞMALAR 23](#_Toc502221596)

[8. SONUÇ VE DEĞERLENDİRME 23](#_Toc502221597)

[KAYNAKLAR 23](#_Toc502221598)

# GİRİŞ

## Projenin Konusu ve Önemi

Bu projede esas alınan ana konu, site/apartman sakinlerinin ve yöneticilerinin rahat bir şekilde iletişim kurabilip duyuru ekleme, aidat listesini görebilme, aidat ödeyebilme, site giderlerini takip edebilme gibi birtakım özellikleri bu yazılım sayesinde kolayca halledebilmelerini sağlayabilmektir. Yazılımı alıp kullanan sakinler, çoğu işini oturduğu yerden halledip, apartman durumunu takip edebilecektir. Bu sayede hem bilgilerin ulaşılıp paylaşılması rahat ve kolay hale gelecektir, hem de programın sunduğu fonksiyonel özellikler kullanılarak sakinler arasında daha iyi bir iletişim sağlanabilecektir.

## 1.2 Projenin Kapsamı

Proje kapsamında; kullanıcıların sisteme kullanıcı adı ve şifreleriyle giriş yapabilmeleri, kendileriyle birlikte diğer sakinlerin ad soyad, blok ve daire bilgilerini görebilmeleri, aidat bilgilerini görüp aidat ödemeleri yapabilmeleri, duyuruları takip edip duyuru ekleyebilmeleri, sitenin giderlerinden haberdar olup dinamik bir şekilde eklemeler yapabilmeleri, giderleri artı olarak grafiksel bir şekilde görebilmeleri ve kendilerinin sisteme kayıtlı bilgilerini de rahatça görebilmelerini içermektedir.

## 1.3 Projenin Hedefleri

Kolay, ergonomik, göze hoş gelen ve sade bir tasarıma sahip olması, şu an piyasada bulunan projelere yakın bir kaliteye ve fonksiyonaliteye sahip olması, gerçekleştirimi beklenen tüm gereksinimlerin doğru çalışır bir şekilde ve hata kontrolleriyle birlikte yapılması, ADO.NET ile oluşturulan veritabanının etkin bir şekilde yönetilebilmesi, dönem boyunca anlatılan teknoloji ve GUI parçalarının doğru bir şekilde kullanımı, geliştirim boyunca keşfedilen farklı yöntemlerin projeye dahil edilmesi, halihazırda bulunmakta olan sistemlere oranla artılarının olması, gerekiyorsa gerçek hayata uyarlanıp kullanılması projenin hedefleri olarak sıralanabilir.

## 1.4 Projenin Alan (Domain) Bilgisi

Bu proje ve benzeri hizmetleri sunan diğer projeler, genellikle yönetim ve takip hizmetlerini sunmaktadırlar. Kullanıcılar için kolay finansal yönetim, kolay ödeme imkanları, kanunlara uygun sistemlerle kullanıcıyı ekstra zahmete sokmayan, anlaşılır bir arayüze sahip hizmetlerdir. Projemde de apartman/site vb. birbirleriyle ortak bir alanda ortak bir şeyleri gerçekleştiren kullanıcı kitlesinin rahatça kullanacağı bir yazılımı hedefledim. Site yönetimlerinde,apartman yönetimlerinde ve sakinlerin de rahatça kullanıp bilgilerden haberdar oalbileceği, ücretlerin düzenli toplandığı, duyuruların paylaşıldığı alanlarda kullanılabilecek bir sisteme sahip.

# 2. BENZER ÇALIŞMALAR

**2.1 Biyos[1]**

Bir apartman/site/rezidans yönetim programı olan Biyos, site yöneticileri ve profesyonel yönetim şirketleri için geliştirilmiş kolay yönetim, gelir ve gider takibi programı. Kullanıcılarına sunduğu hizmetleri ise şu şekilde sıralamış: Kolay finansal hizmet, gider paylaşımı, kredi kartıyla aidat tahsilatı, anketler, sayaç yönetimi, anlaşılır raporlar ve kanuna uygun hizmet. Sistemde ana ekranda aidatlar, giderler, hesaplar, kasa/banka, yönetim, raporlar ve ayarlar bölümleri bulunmakta. Masaüstü ve mobil ortamlarda hizmet verebilmesinin yanında, veriler bulut sunucularda çalışmakta ve veriler de uçtan uca şifreli iletilmekte. Başlangıç planları ücretsiz fakat ihtiyaçların artması durumunda standart veya premium planlardan birini kullanıcıya seçmesi için sunmakta. Premium planda aylık daire başına 7,99 TL gibi bir ücret belirlenmiş. Bu fiyat yıllık 79,90 TL olmakta. Arayüzü gayet kolay kullanımlı ve hoş bir tasarıma sahip. Ücretlendirme olarak da çoğu rakibinden daha uygun bir teklif sunmakta. Oldukça iyi bir izlenim bıraktığını söylemek mümkün.

**2.2 Nst Yönetim[2]**

Karşıma çıkan bir diğer hizmet olan NST Yönetim ise İzmir özelinde hizmet sunan bir apartman/site yönetim sistemi. Apartman ve site yönetim hizmetleri dışında temizlik hizmetleri, tesis ve AVM yönetimi, güvenlik hizmeti, gayrimenkul danışmanlığı ve 7/24 call center hizmeti de sunmakta. Ayrıca 2007 yılından beri bu sektördelermiş ve şu anda hizmetini yürüttükleri yapıları sitelerinde göstermekteler. Oldukça fazla tercih edilen bir firma. Kullanıcı yorumlarını da inceleyebilmek mümkün. Örneğin bir kullanıcı site sakinlerinin aidatları ödememesinden rahatsızmış ve bu hizmeti kullanmaya başlamış. Sonrasında hizmetin sağladığı özelliklerden biri olan hukuk departmanları ile hukuki süreçleri yürütmüşler ve kullanıcı bu durumdan oldukça memnun kalmış. Başka bir kullanıcı ise sistemin oldukça şeffaf olduğundan bahsederek memnuniyetini belirtmiş.

Sunduğu hizmetleri okurken bile yüksek bir ücreti olduğunu tahmin ettim. Fakat sitelerinde belirtilen bir fiyat bulunmamakta. İletişime geçerek teklif alınabilmekte. akat sağladığı hizmetlerin ne kadar kapsamlı olduğunu düşünürsek cüzi bir ücret beklemek mantıksız olacaktır.

**2.3 Hürsoft Site Aidat Yönetim Programı[3]**

Başka biryazılım olan Hürsoft firmasının site yönetim sistemi ise kullanıcılara ücretsiz bir kullanım sunmakta. Dairelerin borç bilgilerini SMS yoluyla bilgilendirmenin yanında, Kat Mülkiyetleri Kanun ve Tüzüklerine uygun bir yazılım. Kullanıcılara sunduğu imkanlar ise şu şekilde: Kiracı ve ev sahibi bilgilerini kaydedebilme, dairelerin borç ödeme takibini yapabilme, dairelerin gecikme cezası takibini yapabilme, istenen ayların faiz sonuçlarını alabilme, daireleri aylık/yıllık olarak borçlandırabilme, dairelerin muhasebe genel durum takibini yapabilme, işletme ve kasa defterleri tutabilme, dairelere makbuz basımı yapabilme, aidat, elektrik, sıcak su, soğuk su vb. takiplerini yapabilme. Sayfada tanıtım ve eğitim videolarının yanı sıra demo indirme seçeneği de var fakat seçenek aktif olmadığı için kullanamadım. Ayrıca site yapısı bir miktar özensiz ve karmaşık olduğu için bir şeyleri ararken zorlandım. Sayfada ücretsiz demesine rağmen Menüden Site Yönetim Programını seçtiğimde 2360 TL olduğunu gördüm. Uygun bir ücret fakat sayfalarında açık hiçbir bilgi vs. vermedikleri için çok tercih edeceğim bir yazılım olmazdı.

**2.4 Apsiyon[4]**

Apsiyon, kullanıcılarına belirli şu hizmetleri sunmakta: aidat takibi, doküman yönetimi, muhasebe, finans, raporlama, online banka entegrasyonları, kişi-daire kayıtları ve kart ile online tahsilat, e-posta ve SMS gönderimi, kartlı geçiş ve plaka tanıma, sanal santral hizmetleri, web sitesi ile duyurular yapabilme, iş takibi ve icra takibi. Geniş hizmet yelpazesi sunması (özellikle plaka tanıma, banka entegrasyonları ve web sitesi yaratılması) sebebiyle de ekstra ücretlerden kaçınılması konusunda tercih edilebilir bir sistem. Fiyatlandırma olarak başlangıç fiyatı veya ortalama bir fiyat belirlenmemiş, bilgilerinizi doldurarak teklif almayı bekliyorsunuz. Halihazırda da 1 milyonu geçen kullanıcı sayısına sahip popüler bir yazılım olduğu düşünülürse çok yüksek bir fiyatlandırma olmadığını düşünüyorum.

**2.5 Yönetimcell[5]**

Web sitelerinde herkes tarafından kolayca kullanabilecek, keyifli bir kullanım sunan, sade ve anlaşılır bir yazılım olduğu ayrıca yöneticilerin tüm ihtiyaçlarına cevap verebileceği ifade edilmiş. Kredi kartıyla online tahsilatı, tüm bankalarla entegrasyonu, banka hareketlerini programa otomatik aktarabilmeyi, site sakinlerine şifre vererek hesap özetlerini görmeyi sağlayabilmeyi ve diğer birçok yeniliği vaadetmekteler. 5000’den fazla apartman/site yönetiminin bu yazılımı kullandığını da ayrıca belirtmekteler. Kullanıcılarına 3 ay, 6 ay ve 1 yıl olmak üzere 3 seçenekte lisans sunuyorlar, bağımsız bölüm sayısına ve web sitesi özelliğinin seçimiyle birlikte tutarda da değişiklikler meydana geliyor. En uygun fiyatlandırmada 8-30 bağımsız bölüm, 3 aylık lisans, web sitesiz ve 25 hediye sms hakkına sahip ve 109 TL üzerinden değerlendirilmiş. Fiyat yelpazesi ise 109.00 – 7348.00 TL aralığında değişkenlik göstermekte. Küçük bir site için ideal yazılımlardan biri olduğu söylenebilir çünkü oldukça uygun bir fiyata gayet cezbedici hizmetler sunmakta.

**2.6 SiteCloud[6]**

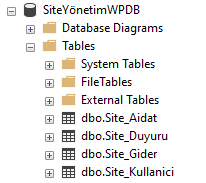
Son olarak değineceğim oldukça popüler olan bir diğer yazılım ise Akınsoft’a ait SiteCloud. Özelliklerini şu şekilde özetlemişler: internetin olduğu her yerden stok takibi yapabilme, bariyer-plaka tanıma entegrasyon modülü, tüm cari hesaplara ait işlemleri tek bir pencereden izleyip yönetebilme, havale eft ve hesaplar arasındaki transferler için banka entegrasyonu, apartman/site sakinlerinin raporları, bilgilerini kayıt etme ve kayıt altında tutabilme, apartman/site hakıında raporlamalar, yönetim ve demirbaş işlemleri, mail ve sms yoluyla faturaların paylaşılabilmesi. 30 günlük ücretsiz deneme imkanı sunmasıyla birlikte, fiyat yelpazesi çoğu sistem gibi daire sayısı, paket içeriği ve ek modüllerin ve hizmetlerin seçimiyle değişmekte. 20 daire için, tüm paket içerikleriyle ve ek modül seçilmeden verilen yıllık lisans ücreti 235.00 TL+KDV. Aynı daire sayısı için tüm ek modüllerin seçilmesiyle birlikte bu fiyat 571.00 TL+KDV’ye kadar artmakta. Küçük boyutlardaki apartman/site/komplekslerden ziyade büyük yönetimsel alanlar için hem ihtiyaçlar hem de ek modüller ve hizmetler açısından daha iyi bir seçenek olabilir.

# 3. YÖNTEM VE TEKNOLOJİLER

|  |  |
| --- | --- |
| IDE | Microsoft Visual Studio 2019 |
| Framework | .NET Framework v4.6.1 |
| Database Management System | SQL Server Management Studio 18 |
| SQL | SQL Server 2016 |
| SQL Connections | DataSet, TableAdapters ve SqlConnection |
| Others | ADO.NET Entity Framework 6.0, System.Data.SqlClient, System.Windows.Forms, WPF UserControl, XML |

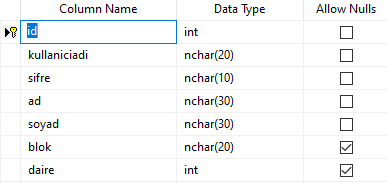
# 4. VERİTABANININ YÖNETİM SİSTEMİNİN OLUŞTURULMASI

Projede kullanılacak veritabanı için tabloları oluştururken bir sitede ne tür bilgilerin olduğu ve bunlardan hangilerinin saklanmak istediğini değerlendirdim. SiteYönetimWPDB veritabanı altında toplamda 4 tablo oluşturma kararı aldım. Bu tablolar: Site\_Kullanici, Site\_Aidat, Site\_Duyuru ve Site\_Gider.



Şekil 1. Veritabanı Tablolar Görünümü.

Öncelikle Site\_Kullanici tablosuna göz atalım. Tabloda Primary Key olarak Auto incremental bir id tutulmakta. Bunun dışında kullanıcı adı, şifre, ad, soyad, blok ve daire bilgileri de tutulmakta. Sadece blok ve daire değerleri null değer kabul etmekte.



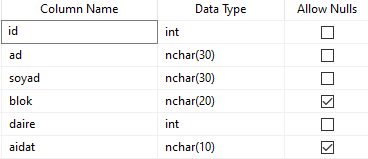
Şekil 2. Site\_Kullanici Tablo Tasarımı

Site\_Kullanici tablomuzda şu anda kayıtlı 30 kullanıcı bulunmakta. Kullanıcılar aşağıdaki gibi listelenmiş durumda:

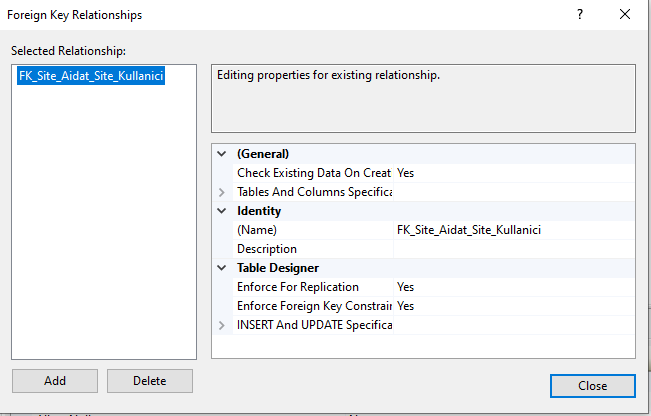


Şekil 3. Site\_Kullanici verileri.

Diğer bir tablomuz olan Site\_Aidat tablosunun tasarımı ise aşağıda gösterildiği gibidir. Buradaki id değeri bir sonraki şekilde de relationship ilişkisinden görüleceği üzere Site\_Kullanici tablosunun id column’ının Foreign Key’idir.

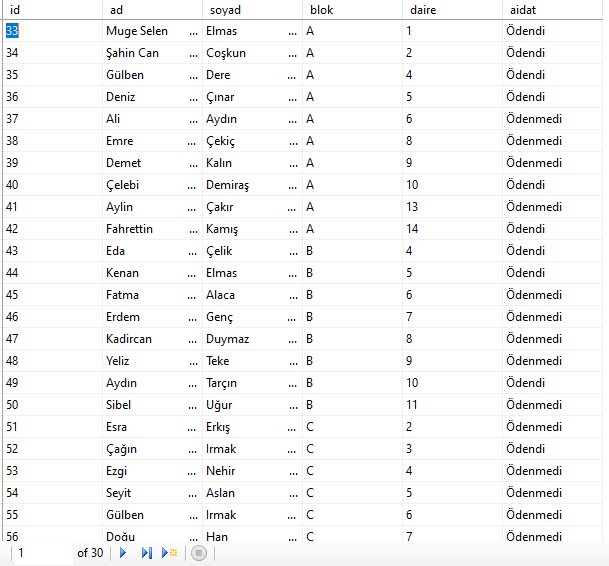


Şekil 4. Site\_Aidat Tablo Tasarımı.



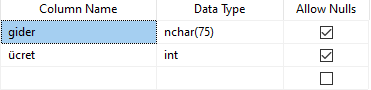
Şekil 5. Tablolar arasındaki Foreign Key ilişkisi.

Site\_Aidat tablosunda da 30 kullanıcının her biri için aidat bilgileri mevcut. Şekil 6’da tablodaki kayıtlı veriler gösterilmektedir:



Şekil 6. Site\_Aidat verileri.

Bir diğer tablomuz olan Site\_Gider tablosunun tasarımı ise gider ve ücret column’larından oluşmaktadır. İki column da null değere sahip olabilir. Bu tablonun diğer tablolarla bir ilişkisi bulunmamaktadır. İçinde bulunan veriler ise Şekil 8’deki gibidir.



Şekil 7. Site\_Gider Tablo Tasarımı

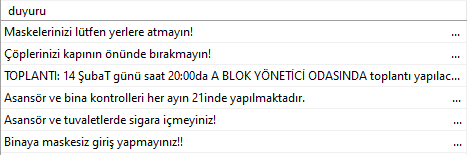


Şekil 8. Site\_Gider verileri.

Son tablo olan Site\_Duyuru tablosu ise sadece duyuru column’ından oluşmaktadır ve diğer tablolarla bir ilişkiye sahip değildir. İçindeki verilerle birlikte tablo aşağıdaki şekillerde gösterildiği gibidir:



Şekil 9. Site\_Duyuru Tablo Tasarımı



Şekil 10. Site\_Duyuru verileri.

# 5. TAMAMLANAN ÇALIŞMALAR

## 5.1 Analiz

Uygulamayı yöneticiler dahil tüm site sakinleri kullanabilmektedir. Kullanıcıların gereksinimleri ise şu şekildedir:

Her kullanıcının adı soyadı,daire numarası, varsa oturduğu blok bir veritabanında saklanmalıdır.

Her kullanıcı için ana bir sisteme giriş penceresi olmalıdır. Kullanıcı, buraya doğru kullanıcı adı ve şifreyi girerek giriş yapabilmelidir.

Yanlış kullanıcı adı veya şifre girildiğinde sistem kullanıcıyı bilgilendirmelidir.

Giriş yaptıktan sonra, sistemin ana penceresinde aidat ödeme, duyuruları görüntüleme, duyuru ekleme, ödemelerini takip edebilme, ödeme yapabilme, site giderlerini görebilme gibi seçenekler olacaktır. Kullanıcı, istediği hizmeti seçtiğinde istediği sayfa panelde görüntülenecektir.

Kullanıcı aidat ödemek istiyorsa ilgili kısma girip geçerli kredi kartı bilgilerini girip işlemi onaylamalıdır. Yanlış/eksik girilen bilgiler için kullanıcıya hata mesajı verilmelidir.

İşlem onaylandığı takdirde, kişiye ödeme işleminin tamamlandığına dair bilgilendirme yapılmalıdır.

Kullanıcı, duyuru ekleyebilmek için duyurular kısmından eklemek istediği duyuruyu text şeklinde yazmalı, yazdıktan sonra duyuru ekle butonuna tıklayarak veritabanına ve sisteme duyuruyu ekleyebilmelidir.

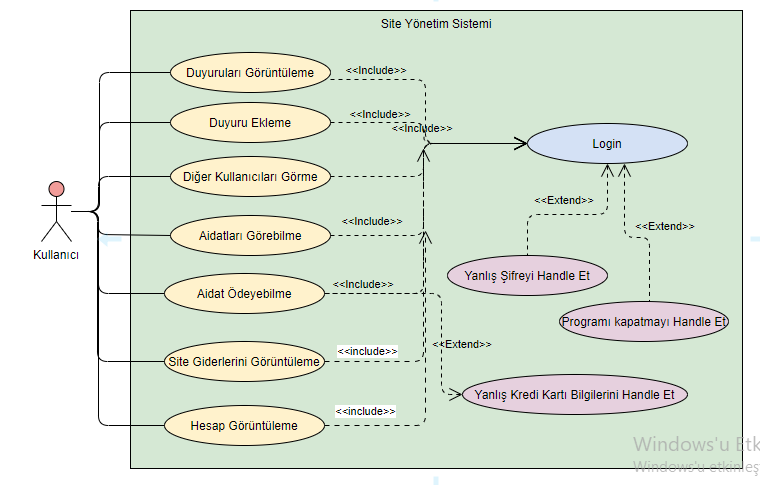
Tüm kullanıcılar; gelir giderlere, aidat ödemelerine ilişkin oluşturulan grafikleri inceleyebilmelidir.

Tüm giderlerin bilgisi yönetici tarafından eksiksiz bir şekilde sisteme eklenmelidir.

Duyurular, herkes tarafından görülebilecek bir alanda paylaşılmalıdır.

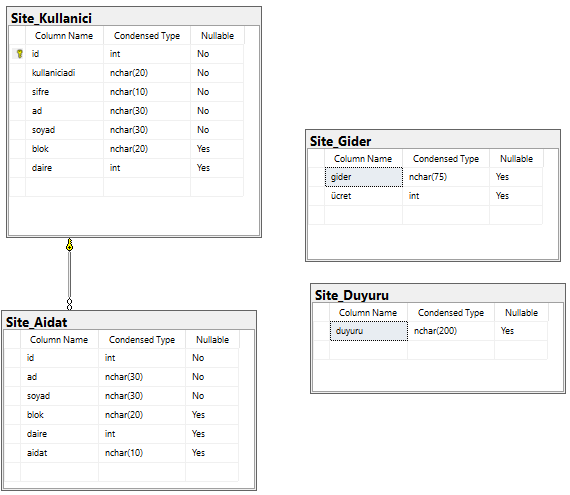
Ödenen aidat bilgileri sistemde gösterilmelidir.

**5.1.a Use Case Diagram**



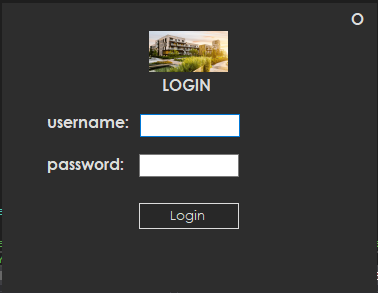
## 5.2 Tasarım

Veritabanı tabloları 4. kısımda açık bir şekilde dile getirilmiştir. Tabloların birbirleriyle ilişkilerini gösteren Database Diagram ise aşağıdaki gibidir:



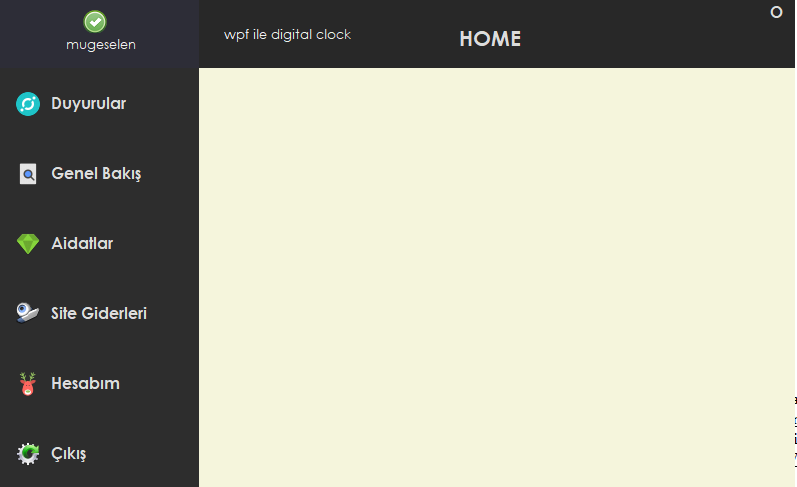
Şekil 11. SiteYönetimWPDB’nin Veritabanı diyagramı.

Programın genel şekli ise ekran görüntüleriyle birlikte şöyle özetlenebilir. Program ilk başlatıldığında öncelikle açılan Login formuna kullanıcımız sistemde kayıtlı olan kullanıcı adı ve şifre bilgilerini girer. Sağ üstte bulunan ‘O’ butonu uygulamayı kapatabilirr.



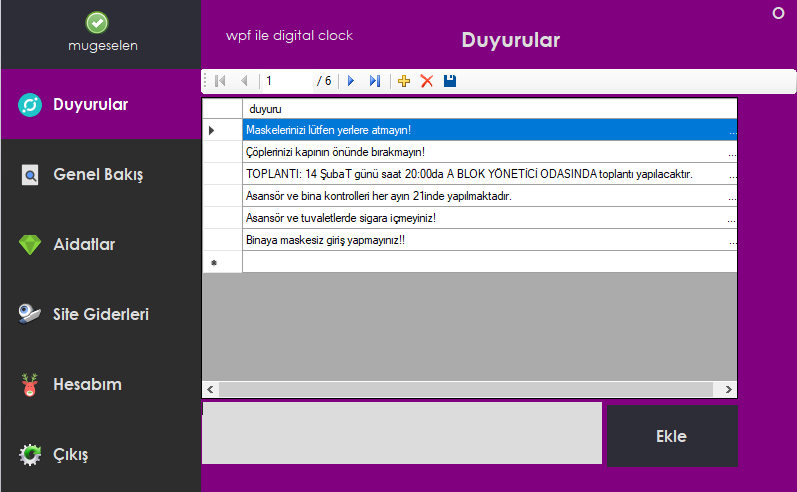
Şekil 12. Giriş Ekranı.

Başarılı bir girişin ardından aşağıdaki gibi bir ekran kullanıcıyı karşılar. Kullanıcı dilediği eylemi soldaki panelde bulunan butonlar üzerinden ilgili butonu seçerek gerçekleştirebilir. Site giderleri dışında tıklanan her bir buton home altındaki panelde görüntülenecektir. Site giderleri butonu ise MDI kullanılan yeni bir form açacaktır.



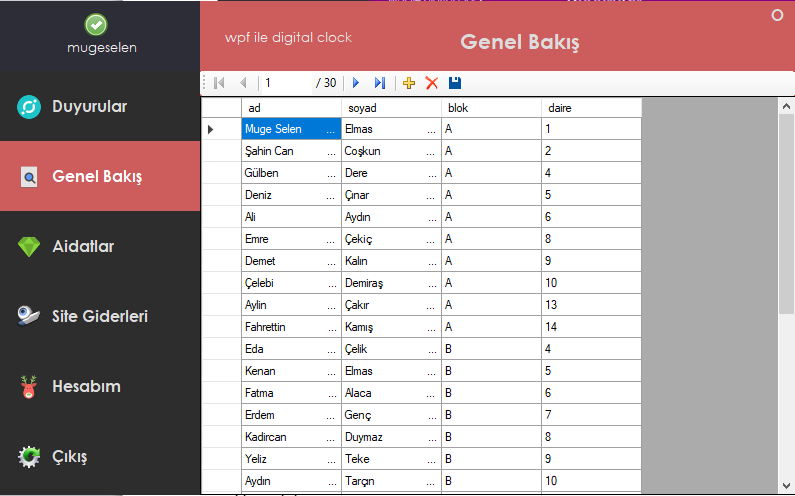
Şekil 13. Ana Ekran

Duyurular kısmında ilgili panel görünecektir. Kullanıcı yeni bir duyuru eklemek istediğinde TextBox’a istediği duyuruyu girdikten sonra ‘Ekle’ butonuna basarak duyurusunu sisteme ekleyebilmektedir.



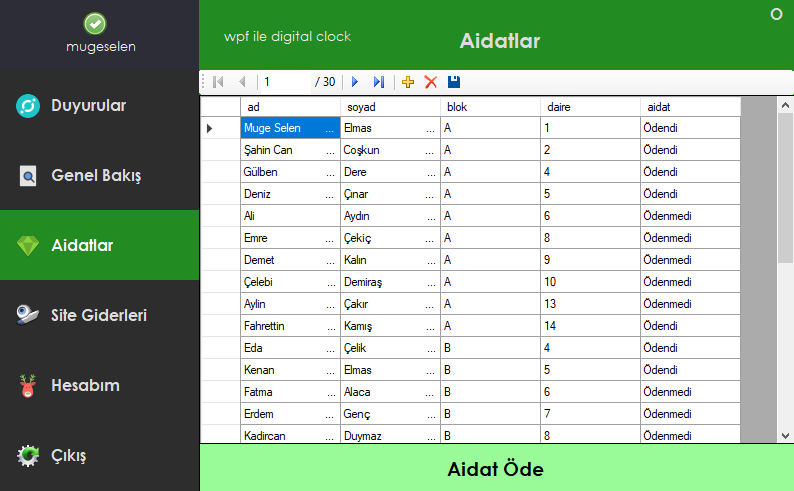
Şekil 14. Duyuru Ekranı

Genel Bakış isimli panelde ise Site\_Kullanici bilgileri görünmektedir. Yalnızca şifre column’ı DataGridView’dan çıkarılmıştır. Kullanıcı buradan diğer kişileri görebilmektedir.



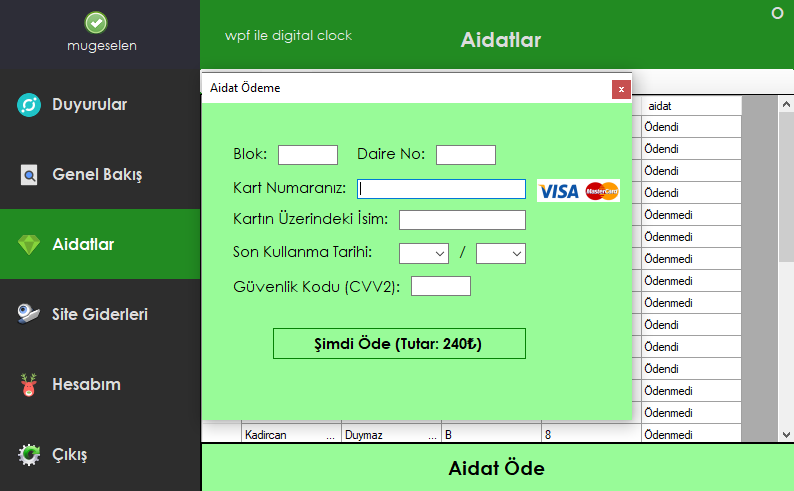
Şekil 15. Genel Bakış Ekranı

Aidatlar sekmesinde ise kullanıcılar ve aidat bilgilerinin bulunduğu Site\_Aidat tablosunun verilerini içeren bir DataGridView bulunmakta. Kullanıcı aidat ödemek istediği takdirde ‘Aidat Öde’ butonuna tıklayabilir.



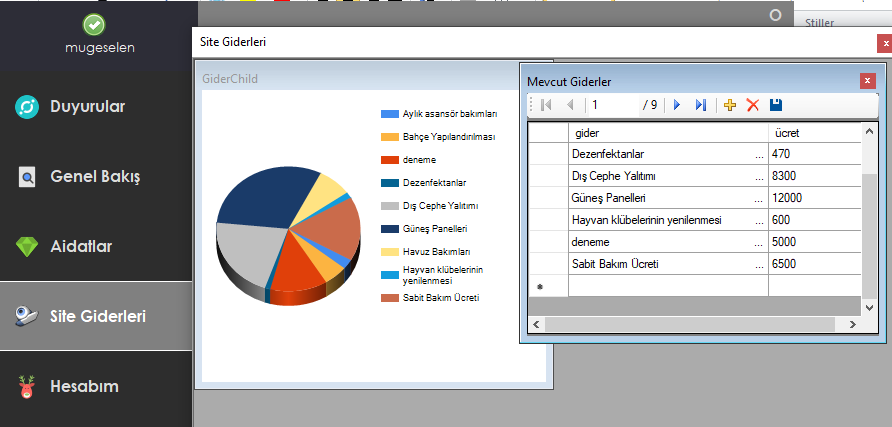
Şekil 16. Aidatlar Ekranı

Aidat öde butonuna tıklandığında ‘Aidat Ödeme’ isimli bir form açılmaktadır. Bu formda blok,daire, kart bilgileri kullanıcıdan istenmektedir. Kullanıcı eksik bilgiler girdiği takdirde sistem hata mesajı vermekte ve işlemi gerçekleştirmemektedir. Geçerli bilgiler girildiğinde ise veri tabanında update yapılmakta ve girilen bilgilere ait kullanıcının aidat durumu ‘Ödendi’ olarak değiştirilmektedir.



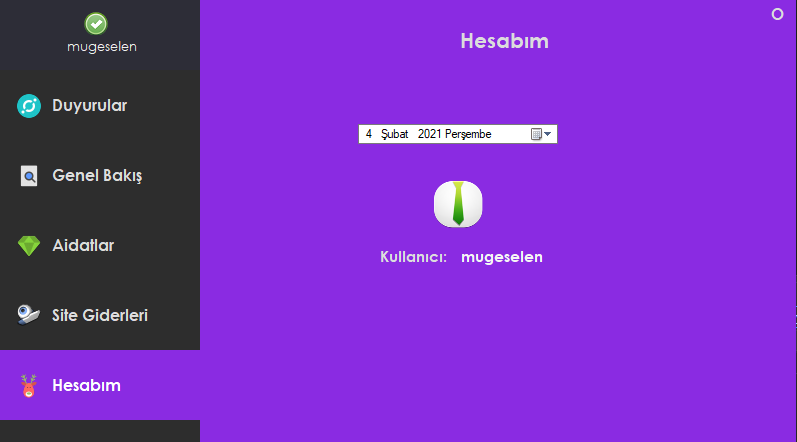
Şekil 17. Aidat Ödeme Formu

Giderler sekmesinde ise yeni bir MDI penceresi kullanıcıyı karşılamaktadır. Burada giderleri içeren bir DataGridView ve giderlerin grafikselleştirilmiş hali 2 ayrı child penceresinde bulunmaktadır.



Şekil 18. Gider Ekran Tasarımı.

Son olarak hesabım sayfasında da kullanıcı adı ve tarih görüntülenmektedir.



Şekil 19. Hesabım.

## 5.3 Geliştirim

private void openChildForm(Form childform, object btnsender){}: Bu fonksiyon sol tarafta bulunan paneldeki herhangi bir butona tıklandığında (site giderler MDI formunda ve Çıkış uygulamadan çıkmakta olduğu için onlar hariç) ilgili butonun onClick event’inde çağırılmaktadır. Aktif olan bir form varsa bu form kapatılır, mevcut buton bilgisi alınır, daha sonrasında butonun ve ana panelin rengi aynı renge ayarlanır. Aktif form (butona tıklandığında açılacak olan form) childform’a atanır ve ana sayfamızdaki homeDesktopPane isimli panel’e childfrom’umuz eklenir. Childform’umuz öne getirilir ve görüntülenir böylece tıkladığımız butona ait form ana sayfadaki panelde gösterilir.

private void loginButton\_Click(object sender, EventArgs e){}: Uygulama ilk çalıştığında kullanıcının username ve password alanına girdiği verilerin database’deki herhangi bir veriyle eşleşip eşleşmediği COUNT(\*) ile döndürülen verilerin dataTable’a aktarılmasıyla kontrol edilir. Eğer eşleşiyorsa kullanıcı hesabına başarıyla giriş yapabilir, eğer eşleşmemişse kullanıcımız bir hata mesajıyla karşılaşır ve login formunu tekrar görür.

public GiderlerForm(): Giderler formunun initialize edildiği metot yapısıdır. Burada MDI kullanımı hedeflendiği için GiderChild ve GiderChild2 adında 2 alt formu bulunmaktadır. Bu metotta her 2 child formun nesnesi yaratılmakta ve parent’larını mevcut form olarak ayarlamaktayız.

private void kartlaodeButton\_Click(object sender, EventArgs e){}: Bu metotta ise kullanıcı aidat ödemeyi seçtikten sonra açılan pencerede girdiği bilgilerin kontrolünü, bilgiler uygunsa da ödeme işlemini gerçekleştirmesi sağlanmaktadır. Kart hanesi, CVV hanesi, ay, yıl seçimleri kontrol edilir, eğer bu değerler geçerliyse database ile SqlConnection yardımıyla bir bağlantı kurulur ve kişinin aidat bilgisi ‘Ödendi’ şeklinde güncellenir.

private void duyuruekleButton\_Click\_1(object sender, EventArgs e){}: Duyuru eklemek için kullanılan bu metotta ise kullanıcıların text box’a yazdıkları duyuru database’e SqlConnection yardımıyla ve INSERT komutuyla yerleştirilir.

private void xmlbutton\_Click(object sender, EventArgs e){}: Kaydedilen aidat bilgilerini içeren XML dosyasını okur ve ilgili textBox’a iletir.

## 5.4 Sınama

Login için gerçekleştirdiğim test; öncelikle database’e bağlanıp bağlanamamaya yönelik bir testti. SqlConncetion ile başarılı bir şekilde bağlanıldığını gördüm. Bir diğeri ise textBoxlara girilen değerlerin database’teki veriyle eşleşip eşleşmediğiyle ilgiliydi. Giriş işlemi de başarılı bir şekilde yapılabilmekte.

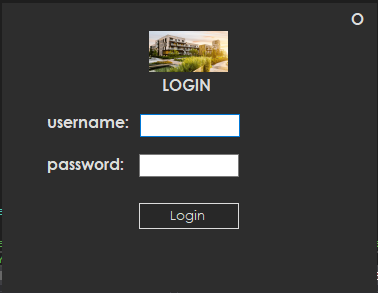
Duyurular kısmında INSERT statement’ının çalışıp çalışmadığını kontrol ettim. Yine tektbox verisiyle alınmakta ve butonla aktifleştirilmekte olan bir statement idi. Sorunsuz bir şekilde çalışmakta.

Aynı şekilde Aidat ödeme sekmesinde UPDATE işleminin çalışıp çalışmadığını denetledim. SqlConnection ile gayet uygun bir şekilde çalışmakta ve veriler güncellenmekte.

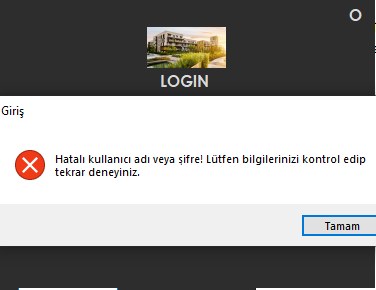
Oluşturduğum MDI child’ların görünümünü ve fonksiyonalitesini,aynı zamanda insert işleminde grafikte veritabanındaki güncellemeye göre güncelleme olup olmadığını kontrol ettim. Yenilendiği zaman gayet sorunsuz bir şekilde çalışmakta.

# 6. GELİŞTİRİLEN PROGRAMIN TANITIMI

Kullanıcı, uygulamayı ilk açtığında aşağıdaki gibi bir Login ekranıyla karşılaşır. Buraya kayıtlı olduğu kullanıcı adı ve şifresiyle giriş yapar. Eğer kullanıcı girişleri veritabanındaki herhangi bir veriyle eşleşmezse kullanıcımız hata mesajı alır. Tamam’ı seçtikten sonra ise Login formuna tekrar giriş yapabilir.

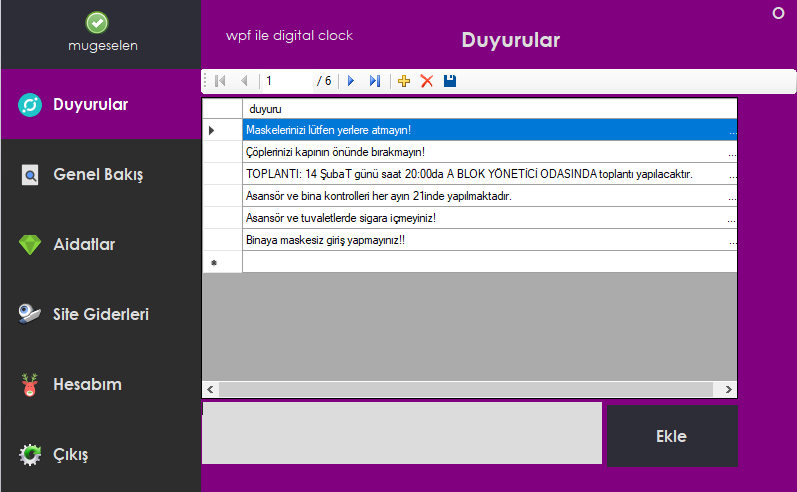


Şekil 20. Giriş Ekranı.



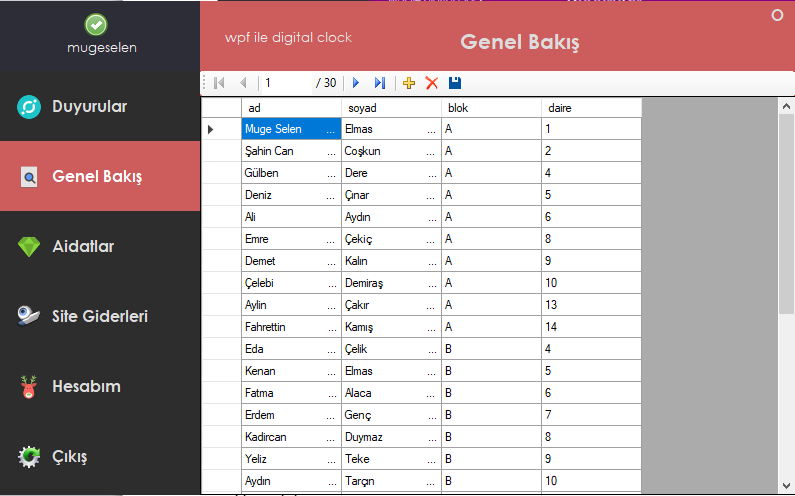
Şekil 21. Hatalı Giriş Bilgisi

Duyurular sekmesinde kullanıcı halihazırda paylaşılan duyuruları görebilir, kendisi de altındaki textBox’a istediği duyuruyu yazdıktan sonra ekle butonuna basarak duyurusunu sisteme ekleyip veritabanına kaydedebilir.



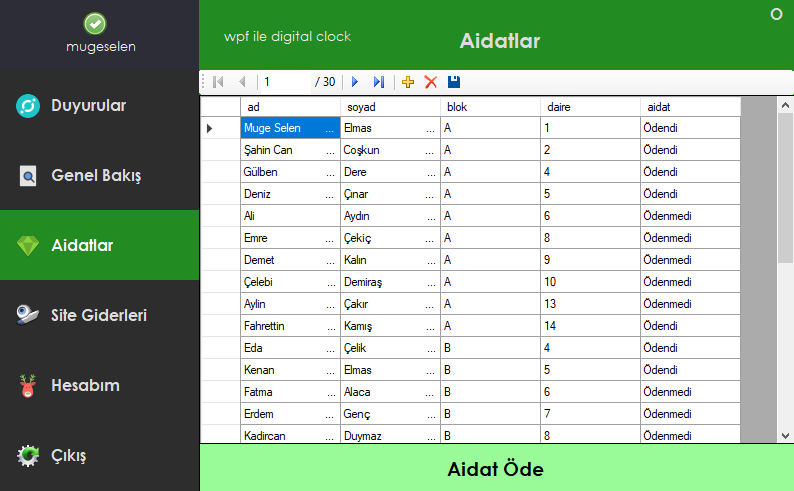
Şekil 22. Duyuru Ekranı

Genel Bakış ekranında ise kullanıcılar sistemdeki diğer kişilerin ad soyad daire ve blok bilgilerini de görebilmektedir.



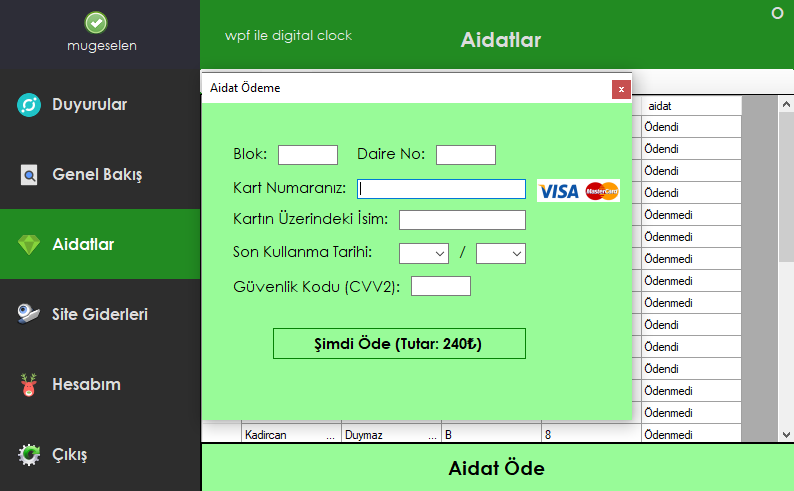
Şekil 23. Genel Bakış Ekranı

Aidatlar kısmında kullanıcı, sisteme kayıtlı kişilerin aidat durumlarını görüp takibini yapabilir. Aidatını ödemek istiyorsa aşağıda bulunan aidat öde butonuna tıklamalıdır.

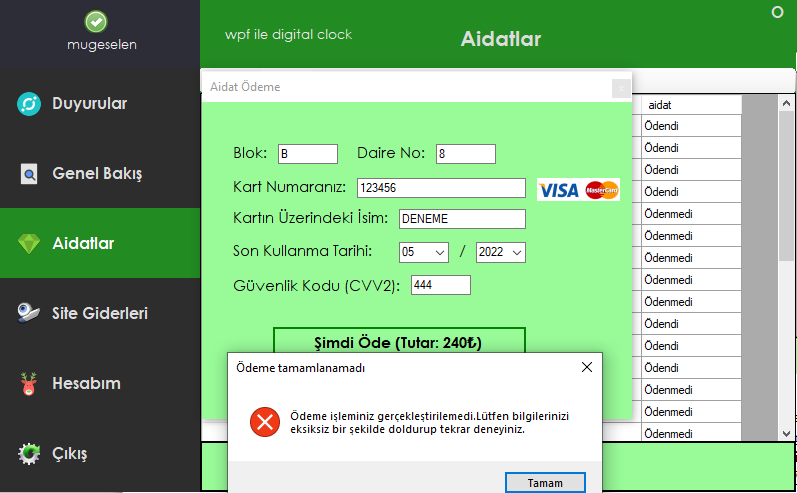


Şekil 24. Aidatlar Ekranı

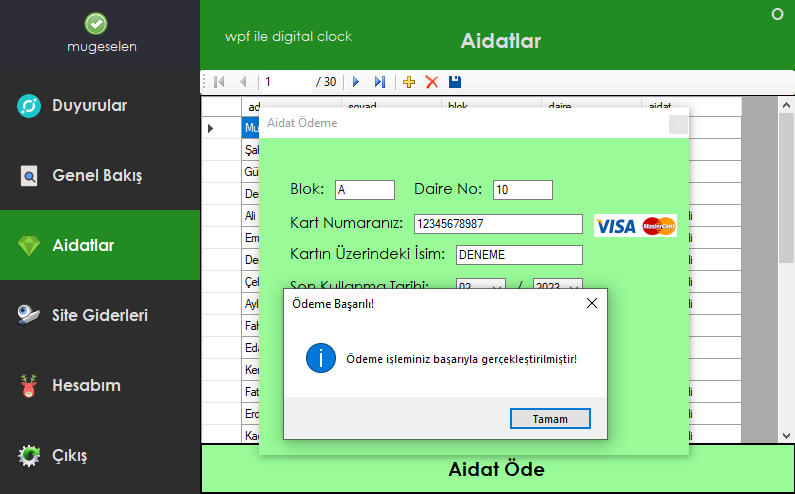
Aidat öde butonuna tıklandığında ‘Aidat Ödeme’ isimli bir form açılmaktadır. Bu formda blok,daire, kart bilgileri kullanıcıdan istenmektedir. Kullanıcı eksik bilgiler girdiği takdirde sistem hata mesajı vermekte ve işlemi gerçekleştirmemektedir. Geçerli bilgiler girildiğinde ise veri tabanında update yapılmakta ve girilen bilgilere ait kullanıcının aidat durumu ‘Ödendi’ olarak değiştirilmektedir. Aşağıdaki şekillerde başarısız ve başarılı iki aidat ödeme işlemini görmektesiniz:



Şekil 25. Aidat Ödeme Formu

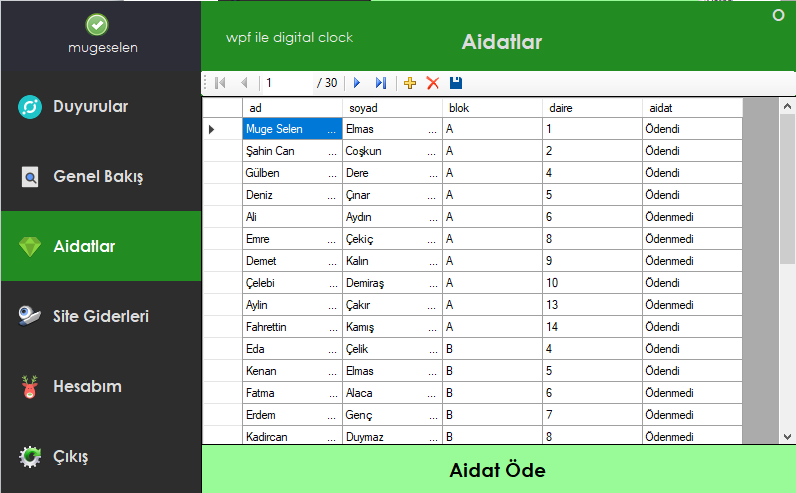


Şekil 26. Geçersiz bilgilerin girilmesiyle alınan hata mesajı.



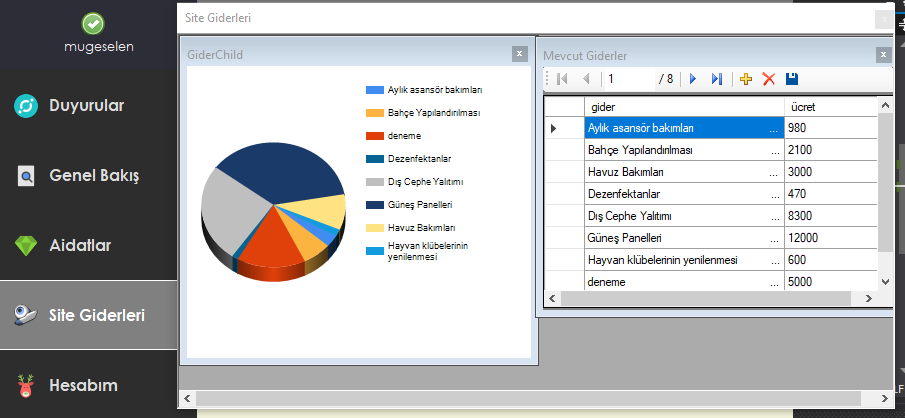
Şekil 27. Geçerli bilgilerle alınan işlem başarılı mesajı.

Ödeme başarılı bir şekilde gerçekleştirildikten ve işlemin başarılı olduğuna dair mesajı aldıktan sonra kullanıcı, bu sayfayı yeniledikten sonra girdiği bilgilere ait olan sakinin aidat durumunun Ödendi olarak güncellenmiş halini görebilir.



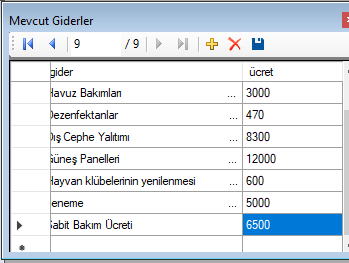
Şekil 28. Ödeme gerçekleştirildikten sonra girilen bilgilere ait kullanıcının aidatının ‘ödendi’ olarak güncellenmesi.

Giderlerin bulunduğu sekmede ise panelde görüntülenmek yerine yeni bir MDI formunda Mevcut Giderler ve Giderlerin grafiksel gösterimi bulunmakta. Yeni bir gider eklenmek isteniyorsa kullanıcı Mevcut Giderler tablosuna istediği gideri ücretiyle beraber girdikten sonra sağ üstteki kaydet butonuna tıklayarak bu gideri veritabanına ekleyebiliyor. Sayfayı yenilediğinde ise bu gider grafiksel görünüme de eklenmiş oluyor. Ekran görüntüsünde görüldüğü üzere deneme isimli bir gider belirtip kaydettikten sonra grafikte ilgili yeri edindiğini görüyoruz.

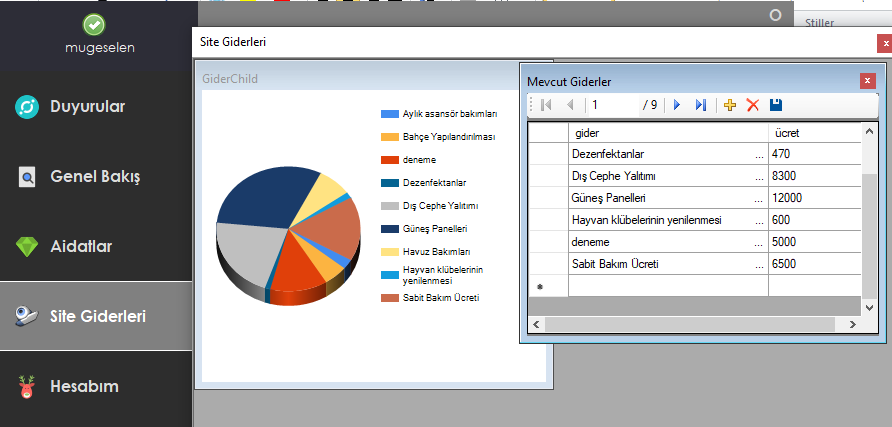


Şekil 29. Gider Ekleme ve Giderlerin Görüntülenmesi.

Tekrar denemek adına Sabit Bakım Ücreti isimli 6500 TL ücretli bir gideri tablomuza ekliyoruz. Daha sonrasında bu veriyi kaydedip gider sekmemizi tekrar açıyoruz. Açtığımızda Sabit Bakım ücretinin grafiğe ve veritabanına kalıcı olarak dahil olduğunu aşağıdaki ekran görüntülerinden görmekteyiz.

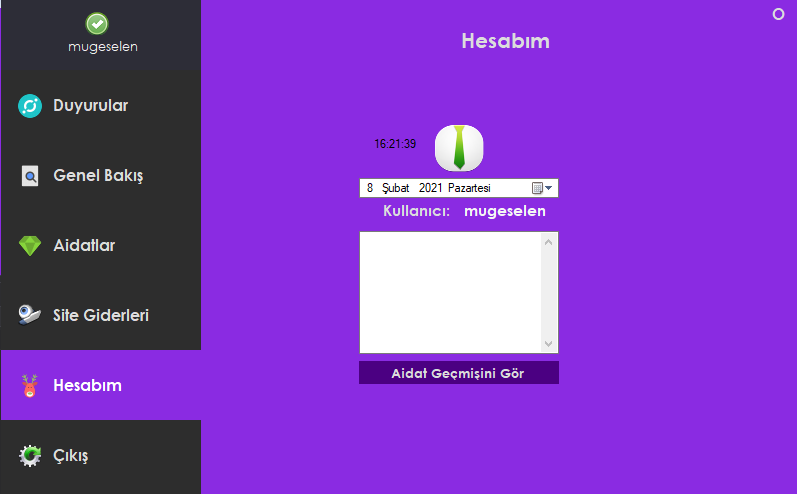


Şekil 30. Gider eklenmesi.

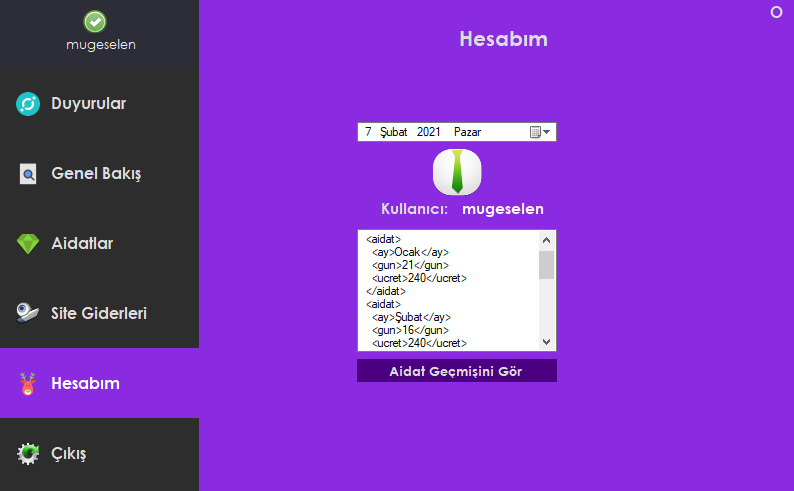


Şekil 31. Giderin Pie Chart’a eklenmesi.

Çıkıştan önceki son buton olan hesabım butonunda ise kullanıcı ve tarih görülmekte. WPF kullanarak bir Clock oluşturup bu clock yapısını panele dahil etmek istedim fakat hatayla karşılaştığım içim maalesef ekleyemedim. Bu sebeple aidat bilgilerinin bulunduğu bir XML dosyası oluşturdum ve Hesabım formuna 1 textbox ve 1 buton ekledim. Böylece butona basıldığında aidat bilgileri kullanıcıya gösterilmekte. Son olarak çıkış butonuna tıklandığında da uygulama kapanmaktadır. İlgili ekran görüntüleri de aşağıdaki gibidir:



Şekil 32. Hesabım Formu.



Şekil 33. Aidat Geçmişlerinin Görüntülenmesi.

# 7. PLANLANAN ÇALIŞMALAR

İlerleyen aşamalarda, hem fonksiyonellik hem de görünüş açısından değiştirebileceğim birçok şey bulunmakta. Örneğin kullanıcımız mail adresiyle giriş yapabilmeli veya sisteme kayıt olurken girdiği mail adresine kaydını onaylamak adına bir verify mail’i gönderilebilir. Veya kullanıcılara bir yetki kısıtlandırılması getirilebilir ve yetkilere göre yapabildikleri değişebilir. Hesabım ve Ayarlar konusunda da çok fazla gelişme sağlanabilir. Görünüş açısından bakacak olursak da DataGridView’ların göze daha hoş görünmesi için bir tasarım yapılabilir.

# 8. SONUÇ VE DEĞERLENDİRME

Ders kapsamındaki GUI kontrollerini, User Defined Controlleri ve MDI gibi fonksiyonaliteleri projemde olabildiğince kullandım. Bunlar dışında ekstra kattığım tek panelde birden çok formu görüntüleyebilmek, database’e SqlConnection ile erişip Update vb. işlemleri ve grafiksel görüntülemeleri gerçek zamanlı gerçekleştirebilmek, ilişkileri ve saklayacağım verileri düşünerek yeni bir database oluşturmak ve Database’e erişimi DataSet üzerinden gerçekleştirmek oldu. Bunlar dışında ise yaptığım WPF saat uygulamasını da elementHost aracılığıyla hesabım paneline eklemeye çalıştım fakat maalesef elementHost UserControl’ümü görmediği için gerçekleştiremedim. Bu sebeple saati gösteren bir User Defined Control ekledim.

# KAYNAKLAR

[1] Biyos, <https://www.biyos.net/tr/fiyatlar> (Son Erişim: 3 Şubat 2021)

[2] Nst Yönetim, <https://nstyonetim.com/> (Son Erişim: 3 Şubat 2021)

[3] Hürsoft, <https://www.hursoft.com.tr/sayfa/ucretsiz-site-yonetim-programi> (Son Erişim: 3 Şubat 2021)

[4] Apsiyon, <https://www.apsiyon.com/> (Son Erişim: 3 Şubat 2021)

[5] Yönetimcell, <https://www.yonetimcell.com/> (Son Erişim: 3 Şubat 2021)

[6] SiteCloud, Akınsoft <https://sitecloud.akinsoft.com.tr/> (Son Erişim: 3 Şubat 2021)