**İÇİNDEKİLER**

TABLO LİSTESİ ..............................................................................................iv

ŞEKİL LİSTESİ ……..........................................................................................v

1. Giriş 1

2. Veritabanı Tasarımı 2

2.1. Gereklilik Analizi 2

2.2. Kavramsal Tasarım 2-3

2.2.1. Öğrenci Varlık Kümesi 3-4

2.2.2.Ayarlar Varlık Kümesi: 5

2.2.3. Öğretmen Varlık Kümesi 5-6

2.2.4. Veli Varlık Kümesi 6-7

2.2.5. Branş Varlık Kümesi 7-8

2.2.6. İl ve İlçe Varlık Kümesi 8

2.3. Mantıksal Tasarım 9

2.3.1. Veli-Öğrenci İlişkisi 9-10

2.3.2 Öğretmen-Öğrenci İlişkisi 10-11

2.3.3 Yönetici-Öğretmen İlişkisi 11-12

2.3.4. Öğretmen-Ayarlar İlişkisi 12-13

2.3.5. Öğrenci-Ayarlar İlişkisi 13

2.3.6. Genel Gösterim 14

2.4. Fiziksel Tasarım 14-15

2.4.1. TBL\_AYARLAR Tablosu 15-16

2.4.2. TBL\_BRANSLAR Tablosu 17

2.4.3. TBL\_ILCELR Tablosu 16-17

2.4.4. TBL\_ILLER Tablosu 18

2.4.5. TBL\_OGRAYARLAR Tablosu 18-19

2.4.6. TBL\_OGRENCILER Tablosu 19-20

2.4.7. TBL\_OGRETMENLER Tablosu 20-21

2.4.8. TBL\_VELILERTablosu 21-22

2.4.9. Tabloların Sql Kodları 22

2.4.9.1. TBL\_AYARLAR 22

2.4.9.2. TBL\_BRANSLAR 23

2.4.9.3. TBL\_ILCELER 23

2.4.9.4. TBL\_ILLER 23-24

2.4.9.5. TBL\_OGRAYARLAR 24

2.4.9.6. TBL\_OGRENCILER 24-25

2.4.9.7. TBL\_OGRETMENLER 25

2.4.9.8. TBL\_VELILER 26

2.4.9.9. AyarlarOgrencıler(Procedure) 26

2.4.9.10. AyarlarOgretmenler(Procedure) 26-27

2.4.9.11. MınıkDahıler(Procedure) 27

2.4.9.12. MınıkEller(Procedure) 27

2.4.9.13. MınıkKelebekler(Procedure) 27-28

2.4.9.14. MınıkMucıtler(Procedure) 28

2.4.10. Veritabanı Diyagramı 29

2.5. Projenin Akış Şeması 30

2.5.1. Giriş Ekranı Akış Şeması 30

2.5.2. Yönetici Ekranı Öğrenci Ekleme Modülü Akış Şeması 31

2.5.3. Yönetici Ekranı Öğrenci Silme Modülü Akış Şeması 32

2.5.4. Yönetici Ekranı Öğrenci Güncelleme Modülü Akış Şeması 33

2.5.5. Yönetici Ekranı Öğrenci Görüntüleme Modülü Akış Şeması 34

2.5.6. Yönetici Ekranı Öğretmen Ekleme Modülü Akış Şeması 35

2.5.7. Yönetici Ekranı Öğretmen Silme Modülü Akış Şeması 36

2.5.8. Yönetici Ekranı Öğretmen Güncelleme Modülü Akış Şeması 37

2.5.9. Yönetici Ekranı Veli Ekleme Modülü Akış Şeması 38

2.5.10. Yönetici Ekranı Veli Silme Modülü Akış Şeması 39

2.5.11. Yönetici Ekranı Veli Güncelleme Modülü Akış Şeması 40

2.5.12. Yönetici Ekranı Fatura Ekleme Modülü Akış Şeması 41

2.5.13. Yönetici Ekranı Giderler Ekleme Modülü Akış Şeması 42

2.5.14. Yönetici Ekranı Kasa Ekleme Modülü Akış Şeması 43

2.5.15. Yönetici Ekranı Raporlar Ekleme Modülü Akış Şeması 44

2.5.16. Yönetici Ekranı Mesaj Ekleme Modülü Akış Şeması 45

2.5.17. Yönetici Ekranı Ayarlar Ekleme Modülü Akış Şeması 46

2.5.18. Yönetici Ekranı Ayarlar Güncelleme Modülü Akış Şeması 47

2.5.19. Öğretmen Ekranı Davranış Ekleme Modülü Akış Şeması 48

2.5.20. Öğretmen Ekranı İstatistik Ekleme Modülü Akış Şeması 49

2.5.21. Öğretmen Ekranı Not Ekleme Modülü Akış Şeması 50

2.5.22. Öğrenci Ekranı Davranış ve İstatistik Görüntüleme Modülü 51

2.6. Proje Uygulama Kısmı 52

2.6.1.Anaokulu Yönetim Sistemi Giriş Ekranı 52

2.6.2.Anaokulu Yönetim Sistemi Yönetici Ana Ekranı 52-53

2.6.2.1.Öğretmen Bilgi İşlemleri Sekmesi 53-56

2.6.2.2.Öğrenci Bilgi İşlemeleri Sekmesi 56-60

2.6.2.3.Veli Bilgi İşlemeleri Sekmesi 60-62

2.6.2.4.Davranış Bilgi Sekmesi 63

2.6.2.5.İstatistik Bilgi Sekmesi 63-64

2.6.2.6.Faturalar Bilgi İşlemeleri Sekmesi 64-65

2.6.2.7.Giderler Bilgi İşlemeleri Sekmesi 66-67

2.6.2.8.Kasa Bilgi İşlemeleri Sekmesi 67-69

2.6.2.9.Raporlar Bilgi İşlemeleri Sekmesi 69-71

2.6.2.10.Mesaj Bilgi İşlemeleri Sekmesi 71-72

2.6.2.11.Ayarlar Bilgi İşlemeleri Sekmesi 72-76

2.6.2.12.Web Sayfası Sekmesi 77

2.6.3.Anaokulu Yönetim Sistemi Öğretmen Ana Ekranı 78

2.6.3.1.Öğretmen Bilgi Sekmesi 78-79

2.6.3.2.Öğrenci Bilgi Sekmesi 79-80

2.6.3.3.Veli Bilgi Sekmesi 80

2.6.3.4.İstatistik Bilgi İşlem Sekmesi 81-82

2.6.3.5.Davranış Bilgi İşlem Sekmesi 83-84

2.6.3.6.Notlar Bilgi İşlem Sekmesi 85-86

2.6.4.Anaokulu Yönetim Sistemi Öğrenci Ana Ekranı 86-87

2.7. Projenin Kodları 88

2.7.1. Giriş Formu Kodları 88-89

2.7.2. Yönetici Ekranı Formu Kodları 89-92

2.7.2.1. Yönetici Ekranı Öğretmenler Formu Kodları 92-96

2.7.2.2. Yönetici Ekranı Öğrenciler Formu Kodları 96-112

2.7.2.3. Yönetici Ekranı Veliler Formu Kodları 112-114

2.7.2.4. Yönetici Ekranı Davranışlar Formu Kodları 114-115

2.7.2.5. Yönetici Ekranı istatistikler Formu Kodları 166

2.7.2.6. Yönetici Ekranı Faturalar Formu Kodları 116-117

2.7.2.7. Yönetici Ekranı Giderler Formu Kodları 118-120

2.7.2.8. Yönetici Ekranı Kasa Formu Kodları 120-121

2.7.2.9. Yönetici Ekranı Raporlar Formu Kodları 122-125

2.7.2.10. Yönetici Ekranı Mesajlar Formu Kodları 121-122

2.7.2.11. Yönetici Ekranı Ayarlar Formu Kodları 125-128

2.7.3. Öğretmen Ekranı Ana Formu Kodları 128-130

2.7.3.1. Öğretmen Ekranı Öğretmenler Formu Kodları 130

2.7.3.2. Öğretmen Ekranı Öğrenciler Formu Kodları 130--131

2.7.3.3. Öğretmen Ekranı Veliler Formu Kodları 131-132

2.7.3.4. Öğretmen Ekranı İstatistikler Formu Kodları 132-135

2.7.3.5. Öğretmen Ekranı Davranışlar Formu Kodları 135-137

2.7.3.6. Öğretmen Ekranı Notlar Formu Kodları 137-139

2.7.4. Öğrenci Ekranı Ana Formu Kodları 139-140

2.7.5. Sql Bağlantı Kodu 140-141

2.7.6. Öğrenci Datagridview Kodları 141

2.7.7. Öğretmen Datagridview Kodları 141-142

Kaynakça 143

**TABLO LİSTESİ**

Tablo 1: Öğrenci Tablosu…………………………………………………….10

Tablo 2: Veli Tablosu………………………………………………………...10

Tablo 3: Öğretmen Tablosu…………………………………………………..11

Tablo 4: Ayarlar Tablosu……………………………………………………..13

Tablo 5: Öğrenci Tablosu…………………………………………………….13

Tablo 6: TBL\_AYARLAR Tablosu……………………………………….....16

Tablo 7: TBL\_BRANSLAR Tablosu…………………………………….…..17

Tablo 8: TBL\_ILCELER Tablosu…………………………………………....17

Tablo 9: TBL\_ILLER Tablosu……………………………………….………18

Tablo 10: TBL\_OGRAYARLAR Tablosu…………………………………...19

Tablo 11: TBL\_OGRENCILER Tablosu…………………………………….20

Tablo 12: TBL\_OGRETMENLER Tablosu………………………………….21

Tablo 13: TBL\_VELILER Tablosu……………………………….……….....22

# 

# **ŞEKİL LİSTESİ**

Şekil 1: Öğrenci Varlık Kümesi…………………………………………………5

Şekil 2: Ayarlar Varlık Kümesi………………………………………………….5

Şekil 3: OgrAyarlar Varlık Kümesi……………………………………………...5

Şekil 4: Öğretmen Varlık Kümesi……………………………………………….6

Şekil 5: Veli Varlık Kümesi……………………………………………………..7

Şekil 6: Branşlar Varlık Kümesi…………………………………………………8

Şekil 7: İller Varlık Kümesi……………………………………………………...8

Şekil 8: İlçeler Varlık Kümesi………………………………………………..…8

Şekil 9: Öğrenci-Veli İlişkisi…………………………………………………....9

Şekil 10: Öğretmen-Öğrenci İlişkisi…………………………………………...11

Şekil 11: Yönetici-Öğretmen İlişkisi……………………………………..……12

Şekil 12: Öğretmen-Ayarlar İlişkisi……………………………………..……..12

Şekil 13: Öğrenci-Ayarlar İlişkisi……………………………………..……….13

Şekil 14: Genel Gösterim……………………………………………...……….14

Şekil 15: Veritabanı Diyagramı………………………………………………..29

Şekil 16: Giriş Ekranı………….……………………………………………….52

Şekil 17: Yönetici Ana Sayfa………………………………………………….53

Şekil 18: Öğretmen Bilgi Kayıt Sekmesi………………………………………54

Şekil 19: Öğretmen Kaydet…...………………………………………………..55

Şekil 20: Öğretmen Bilgileri Sekmesi…...……………………………………..55

Şekil 21: Öğretmen Güncelle…………………………………………………..56

Şekil 22: Öğretmen Sil……………………………….……………………...…56

Şekil 23: Öğrenci Bilgi Kayıt Sekmesi………………………….………….…57

Şekil 24: Öğrenci Kayıt……………………………….………………………58

Şekil 25: Öğrenci Güncelle……………………….….……………………..…59

Şekil 26: Öğrenci Sil……………………………….……………………….…59

Şekil 27: Öğrenci Bilgileri…………………….…………………….…………60

Şekil 28: Veli Bilgi İşlem Sekmesi……………………………………………61

Şekil 29: Veli Kayıt……………………………….……………………..……61

Şekil 30: Veli Güncelle……………………………….………………………62

Şekil 31: Veli Sil…………………………………..….………………………62

Şekil 32: Yönetici Davranış Sekmesi……………...….………………………63

Şekil 33: Yönetici İstatistik Sekmesi……………...….………………………64

Şekil 34: Yönetici Fatura Bilgi İşlem Sekmesi…....….………………………65

Şekil 35: Fatura Kayıt……………………………...….………………………65

Şekil 36: Gider Bilgi İşlem Sekmesi……………...….…………………….…66

Şekil 37: Gider Öğretmen Bilgi Sekmesi…………….………………………67

Şekil 38: Gider Hesaplama……………..…………….………………………67

Şekil 39: Kasa Bilgi İşlem Sekmesi……………...….…………………..……68

Şekil 40: Kasa Öğrenci Bilgi Sekmesi…………..…………..…………..……68

Şekil 41: Kasa Hesaplama……………...….………………………….………69

Şekil 42: Rapor Bilgi İşlem Sekmesi……………...….…………………..…..70

Şekil 43: Rapor Kayıt İşlemi……………...….……………………..…..……70

Şekil 44: Rapor Öğretmen Bilgi Sekmesi……………...….……………..…..71

Şekil 45: Mesaj Bilgi İşlem Sekmesi……….……………………..…..…..…72

Şekil 46: Mesaj Kayıt İşlemi……………...….……………………..…..……72

Şekil 47: Öğretmen Ayarlar Bilgi İşlem Sekmesi…………………..…..……73

Şekil 48: Öğrenci Ayarlar Bilgi İşlem Sekmesi …………......................……74

Şekil 49: Öğretmen Şifre Kayıt ……………….…………......................……74

Şekil 50: Öğrenci Şifre Kayıt ……………….…………......................…...…75

Şekil 51: Öğretmen Şifre Güncelle ……………… …………..................…..76

Şekil 52: Öğrenci Şifre Güncelle ……………… …………..................…….76

Şekil 53: Web Sayfası ……………… …………...................................……77

Şekil 54: Öğretmen Ana Sayfa ……………… …………........................…..78

Şekil 55: Öğretmen Bilgi Sekmesi ……………… …………..................…..79

Şekil 56: Öğrenci Bilgi Sekmesi ……………… ………….....................…..80

Şekil 57: Veli Bilgi Sekmesi ……………… ………….……..................…..80

Şekil 58: İstatistik Bilgi İşlem Sekmesi ……………… …………..........…..81

Şekil 59: İstatistik Kayıt ……………… ……………………..................…..82

Şekil 60: İstatistik Öğrenci Bilgi Sekmesi ……………… …..................…..82

Şekil 61: Danışman Bilgi İşlem Sekmesi ……………… ……................…..83

Şekil 62: Danışman Öğrenci Bilgi Sekmesi…………………..................…..84

Şekil 63: Danışman Kayıt ………………………..…………..................…..84

Şekil 64: Not Bilgi İşlem Sekmesi ……………… ………….......................85

Şekil 65: Not Kayıt …………………………………………..................…..85

Şekil 66: Öğrenci Ekranı Bilgi Sekmesi ……………… ….....................…..87

Şekil 67: Öğrenci Ekranı Genel ………………….…………..................…..87

# **GİRİŞ**

Anaokulu Yönetimi projesinin amacı; okul öncesi eğitim için başlayan küçük çocukların hem yönetimin girebildiği hem de öğretmen ve öğrencinin girebildiği bir proje olup yönetimdeki kişinin öğrenci kayıt edebilme ve öğretmen kayıt edebilme yetkisinin olması. Öğretmenlerin ise öğrenci notu ve davranışlarını verme kısmına giriş yapabildiği yetkisinin olması. Öğrencilerin ise sadece öğretmenlerin verdikleri davranış ve not kısmına giriş yapabildikleri bir uygulamadır. Yönetim kısmının ise öğretmen ve öğrenciden hariç şifreleri düzenleme ve kayıt etme, okul gelir-giderlerinin de olduğu ayrı bir kısım a giriş yaparlar. Giriş yapacak kimselerin(Yönetici, Öğretmen, Öğrenci) kendilerine ait şifreleri vardır.

Projenin veri tabanı kısmında SQL kullanılmıştır. Kullanılan SQL programı Microsoft SQL Server Management Studio 18 ile yapılmıştır. Tasarım kısmı ise Visual Studio 19 ile tasarlanıp kodları ise ADO.NET ve Entity Framework kullanılmıştır. Tasarımda DevExpress araçları ve normal C# araçları kullanılmıştır. Gerektiğinde verilere hızlı bir şekilde ulaşabilmek, öğrencilerin kişisel ve okul ile ilgili bilgilerinin takibinin sağlamaktır. Program bu amaçla tasarlanmıştır.

# **VERİTABANI TASARIMI**

Düzgün tasarlanmış bir veri tabanı güncel ve doğru bilgilere erişmenizi sağlar. Veri tabanıyla çalışırken hedeflerinize ulaşmanız için doğru tasarım kritik önem taşıdığından, zaman ayırıp iyi bir tasarımın temel ilkelerini öğrenmek anlamlı bir yatırım olacaktır. Sonunda, ihtiyaçlarınıza uygun ve değişikliği kolayca kabul edebilecek bir veri tabanı elde etme olasılığınız çok daha yüksektir.

# **2.1. Gereklilik Analizi**

Program gereksinimleri ihtiyaçları yapılan araştırma sonucunda aşağıda belirtildiği şekilde tespit edilmiştir.

* Öğrencilerin kayıt altına alınması,
* Öğretmenlerin not girdilerinin kayıt altına alınması,
* Öğrenci veli bilgilerinin kayıt altına alınması,
* Öğrenci karne sisteminin kayıt altına alınması,
* Rehber bilgilerinin kayıt altına alınması,
* Yönetici girişi ve işlemlerinin kayıt altına alınması,
* Öğrenci girişi işlemlerinin kayıt altına alınması,
* Öğretmenlerin kayıt altına alınması,
* Öğrenci davranışlarının kayıt altına alınması,
* Ayarlar kısmının kayıt altına alınması.

# **2.2. Kavramsal Tasarım**

Kavramsal tasarım aşağıdaki ihtiyaçların giderilmesi maksadıyla tasarlanmıştır.

* Anaokulu Yönetimi programı yönetici, öğretmen ve öğrenci tarafından üç yönlü olarak kullanılması,
* Öğrenci ve bilgilerinin anlaşılır bir şekilde listelenmesi.
* Öğrenci bilgilerinin kolay ve hızlı bir şekilde girilebilmesi.
* Öğrenci bilgilerinin kayıt altına alınabilmesi.
* Öğrenci bilgilerinin takibinin sağlanması.
* Öğrenci not ve davranışları bilgilerin takibinin yapılması.
* Öğrencinin kendisine ait şifresi olup karnesini görebilmesi.
* Öğrenci için her türlü bilgiye ait ekleme, çıkarma, düzeltme, silme ve güncelleme işlemlerinin yapılabilmesi.
* Öğrenci veli bilgilerinin kayıt alınması ve hızlı şekilde ulaşılabilmesi.
* Öğretmen bilgilerinin ve branşının anlaşılır bir şekilde listelenmesi.
* Öğretmen bilgilerinin kolay ve hızlı bir şekilde girilebilmesi.
* Öğretmen için her türlü bilgiye ait ekleme, çıkarma, düzeltme, silme ve güncelleme işlemlerinin yapılabilmesi.
* Öğretmenin not girişi işlemlerini yapabilmesi.
* Veli için her türlü bilgiye ait ekleme, çıkarma, düzeltme, silme ve güncelleme işlemlerinin yapılabilmesi.
* Yöneticin kendisine ait paneli olup için her türlü bilgiye ait ekleme, çıkarma, düzeltme, silme ve güncelleme işlemlerinin yapılabilmesi.
* Yöneticin gelir-gider kontrolünün yapabilmesi.
* Veritabanı güvenirliğinin sağlanması.
* Tasarımın gözü yormaması ve kullanıcı dostu olması amaçlanmıştır.

Bu çerçevede; gereklilik analizinden de yola çıkılarak yapılan çalışma sonucunda aşağıdaki varlıkların planlanması gerekli bulunmuş planlama da buna yönelik olarak yapılmıştır. Buna göre Varlık Kümelerinin aşağıdaki maddelerden oluşması planlanmıştır.

* Öğrenci Varlık Kümesi,
* Ayarlar Varlık Kümesi,
* Öğretmen Varlık Kümesi,
* Veli Varlık Kümesi,
* Branş Varlık Kümesi,
* İl ve ilçeler Varlık Kümesi.

**2.2.1. Öğrenci Varlık Kümesi**

Öğrenci Varlık Kümesi tasarlanırken bir bu varlık kümesine ait nitelikler diğer varlıklar arasındaki ilişkileri belirleyen nitelikler ile birlikte tasarlanmıştır.

Buna göre Öğrenci Varlık Kümesi aşağıdaki şekilde oluşturulmuştur.

Öğrenci Varlık Kümesi;

* Öğrenci İd,
* Öğrenci No,
* Öğrenci Fotoğraf
* TC No,
* Adı,
* Soyadı,
* Cinsiyet,
* Öğrenci veli,
* Öğrenci sınıf,
* Öğrenci doğum tarihi,
* Öğrenci il,
* Öğrenci ilçe,
* Öğrenci adres.

Buna göre Öğrenci Varlık Kümesi aşağıdaki şekilde tasarlanmıştır. Tasarımda **Öğrenci İd’si** anahtar nitelik olarak tanımlanmıştır.

ÖĞRENCİ

UYRUĞU

**Şekil 1:** Öğrenci Varlık Kümesi

**2.2.2. Ayarlar Varlık Modeli:**

Ayarlar Varlık Kümesi tasarlanırken bir bu varlık kümesine ait nitelikler diğer varlıklar arasındaki ilişkileri belirleyen nitelikler ile birlikte tasarlanmıştır.

Buna göre Ayarlar Varlık Kümesi;

Öğretmenler;

* Ayarlar Id,
* Ogrt Sifre.

Öğrenciler;

* Ayarlar Ogr Id,
* Ogr Sifre.

Buna göre Ayarlar Varlık Kümesi aşağıdaki şekilde tasarlanmıştır. **Ayarlar Id ve Ayarlar Ogr Id** anahtar nitelik olarak tanımlanmıştır.

**Şekil 2:** Ayarlar Varlık Kümesi **Şekil 3:** OgrAyarlar Varlık Kümesi

AYARLAR

OGRAYARLAR

**2.2.3. Öğretmen Varlık Kümesi**

Öğretmen Varlık Kümesi tasarlanırken bir bu varlık kümesine ait nitelikler diğer varlıklar arasındaki ilişkileri belirleyen nitelikler ile birlikte tasarlanmıştır.

Buna göre Öğretmen Varlık Kümesi;

* Öğretmen İd,
* Öğretmen Ad,
* Öğretmen soyad,
* Öğretmen T.C,
* Öğretmen Telefon,
* Öğretmen mail,
* Öğretmen il,
* Öğretmen ilçe,
* Öğretmen adres,
* Öğretmen branş,
* Öğretmen fotoğraf

Buna göre Öğretmenler Varlık Kümesi aşağıdaki şekilde tasarlanmıştır. Tasarımda **Öğretmen İd** anahtar nitelik olarak tanımlanmıştır.

ÖĞRETMEN

**Şekil 4**: Öğretmenler Varlık Kümesi

**2.2.4. Veli Varlık Kümesi**

Veli varlık kümesi tasarlanırken bir bu varlık modeline ait nitelikler diğer varlıklar arasındaki ilişkileri belirleyen nitelikler ile birlikte tasarlanmıştır.

Buna göre Veli Varlık Kümesi;

* Veli id,
* Veli Anne,
* Veli Baba,
* Veli Telefon1,
* Veli Telefon2,
* Veli Mail.

Buna göre Veliler Varlık Kümesi aşağıdaki şekilde tasarlanmıştır. **Veli İd** anahtar nitelik olarak tanımlanmıştır.

VELİLER

**Şekil 5:** Veli Varlık Kümesi

**2.2.5. Branş Varlık Kümesi**

Branş Varlık Kümesi tasarlanırken bir bu varlık modeline ait nitelikler diğer varlıklar arasındaki ilişkileri belirleyen nitelikler ile birlikte tasarlanmıştır.

Buna göre Branş Varlık Kümesi aşağıdaki şekilde oluşturulmuştur.

Branş Varlık Kümesi;

* Sıra No
* Ders Adı,

Buna göre Branş Varlık Kümesi aşağıdaki şekilde tasarlanmıştır. **Branş İd** anahtar nitelik olarak tanımlanmıştır.

**Şekil 6:** Branş Varlık Kümesi

BRANS

VİZE

**ORTALAMA**

**HARF**

DÖNEMİ

ZORUNLU/SEÇMELİ

AKTES

SINIFI

KREDİ TOPLAMI

**2.2.6. İl ve İlçe Varlık Kümesi**

İl ve İlçeler varlık kümesi tasarlanırken bir bu varlık kümesine ait nitelikler diğer varlıklar arasındaki ilişkileri belirleyen nitelikler ile birlikte tasarlanmıştır.

Buna göre İl Varlık Kümesi;

* İd,
* Şehir.

Buna göre İlçe Varlık Kümesi;

* İd,
* İlçe,
* Şehir.

Buna göre İl ve İlçeler Varlık Kümesi aşağıdaki şekilde tasarlanmıştır. **İkisi içinde id** anahtar nitelik olarak tanımlanmıştır.

**Şekil 7:** İl Varlık Kümesi **Şekil 8:** İlçeler Varlık Kümesi

IL

ILCE

# **2.3. MANTIKSAL TASARIM**

Öğrenci İşleri veri tabanı tasarımında varlık kümeleri arasındaki ilişkilerin tasarlanmasından sonra mantıksal tasarım aşamasına geçilmiştir. Bu aşamada aşağıdaki konular işlenmiştir.

**2.3.1.Veli-Öğrenci İlişkisi**

Bir veya birden çok öğrencinin mutlaka bir velide yer almalıdır. Öğrenci olmadan velinin olması söz konusu değildir. Bir velide birçok öğrenci olabilir. Bu nedenle Veli-Öğrenci varlık kümeleri arasındaki ilişki 1:n ilişkidir. Veli-Öğrenci varlık kümeleri arasındaki ilişki ER şeması olarak Şekil 9'da gösterilmiştir.

VELİ

Öğrenci

Yazdırır

1

n

Şekil 9: Veli-Öğrenci İlişkisi

**Öğrenci Tablosu;**

* Öğrenci İd’si **Birincil Anahtar,**
* Veli tablosuna bağlantı için Veli İd’si Yabancı Anahtar,
* Ayarlar tablosuna bağlantı için Öğrenci İd’si Yabancı Anahtar olarak eklenmiştir.

Bu durumda Öğrenci Tablosu aşağıdaki şekilde tasarlanmıştır.

|  |
| --- |
| **ÖĞRENCİ TABLOSU** |
| ÖĞRENCİ İD  ÖĞRENCİ TC NO |
| ÖĞRENCİ NO |
| ADI |
| SOYADI |
| CİNSİYET  VELİİD |
| SINIF |
| ADRES  İL |
| İLÇE  ÖĞRENCİ FOTOĞRAF |
| VELIID |
| AYARLAROGRID |

Tablo 1: Öğrenci Tablosu

**Veli Tablosu:**

* VELIID Velı Tablosu ile ilişkili olacak şekilde **Birincil Anahtar** olarak tasarlanmıştır.

Tablo 2: Veli Tablosu

|  |
| --- |
| **VELİ TABLOSU** |
| VELIID |
| VELIANNE VELIBABA VELITEL1  VELITEL2  VELIMAIL |

**2.3.2. Öğretmen-Öğrenci İlişkisi**

Bir veya birden çok öğretmen birden fazla sınıfta yer almalıdır. Öğrenci olmadan öğretmen olması söz konusu değildir. Bir öğretmende birçok öğrenci olabilir. Bu nedenle Öğretmen-Öğrenci varlık kümeleri arasındaki ilişki m:n ilişkidir. Öğretmen-Öğrenci varlık kümeleri arasındaki ilişki ER şeması olarak Şekil 10 'da gösterilmiştir.

n

m

ÖĞRETMEN

ÖĞRENCI

SINIF

**Şekil 10.** Öğretmen-Öğrenci İlişkisi

**Öğretmen Tablosu:**

* OGRTID Öğrenci Tablosu ile ilişkili olacak şekilde **Birincil Anahtar** olarak tasarlanmıştır.

Tablo 3: Öğretmen Tablosu

|  |
| --- |
| **ÖĞRETMEN TABLOSU** |
| OGRTID |
| OGRTAD  OGRTSOYAD |
| OGRTTC |
| OGRTTEL  OGRTMAIL  OGRTIL  OGRTILCE  OGRTADRES  OGRTBRANS  OGRTFOTO |

**2.3.3. Yönetici-Öğretmen İlişkisi**

Kurumda bir yönetici olduğunu varsayarsak bir yönetici birden fazla öğretmen kurumda yer almalıdır. Öğretmen olmadan kurumun olması söz konusu değildir. Bir kurumda birçok öğretmen olabilir. Bu nedenle Yönetici-Öğretmen varlık kümeleri arasındaki ilişki 1:n ilişkidir. Yönetici-Öğretmen varlık kümeleri arasındaki ilişki ER şeması olarak Şekil 11 'da gösterilmiştir.

1

n

YNETİCİ

ÖĞRETMEN

ALIR

**Şekil 11:** Yönetici-Öğretmen İlişkisi

**2.3.4. Öğretmen-Ayarlar İlişkisi**

Bir öğretmenini sadece bir şifresi ayarlar kısmında yer almalıdır. Öğretmen olmadan şifre olması söz konusu değildir. Bir öğretmende bir şifre olabilir. Bu nedenle Öğretmen-Ayarlar varlık kümeleri arasındaki ilişki 1:1 ilişkidir. Öğretmen-Ayarlar varlık kümeleri arasındaki ilişki ER şeması olarak Şekil 12 'da gösterilmiştir.

1

1

ÖĞRETMEN

ŞİFRE

ALIR

**Şekil 12:** Öğretmen-Ayarlar İlişkisi

**Ayarlar Tablosu:**

* AYARLARID Öğretmen Tablosu ile ilişkili olacak şekilde **Birincil Anahtar** olarak tasarlanmıştır.

Tablo 4: Ayarlar Tablosu

|  |
| --- |
| **AYARLAR TABLOSU** |
| AYARLARID |
| OGRTSIFRE |

**2.3.5. Öğrenci-Ayarlar İlişkisi**

Bir öğrencinin sadece bir şifresi ayarlar kısmında yer almalıdır. Öğrenci olmadan şifre olması söz konusu değildir. Bir öğrencide bir şifre olabilir. Bu nedenle Öğrenci-Ayarlar varlık kümeleri arasındaki ilişki 1:1 ilişkidir. Öğrenci-Ayarlar varlık kümeleri arasındaki ilişki ER şeması olarak Şekil 13 'da gösterilmiştir.

**Şekil 13:** Öğrenci-Ayarlar İlişkisi

1

1

ÖĞRENCİ

ŞİFRE

ALIR

**OgrAyarlar Tablosu:**

* AYARLAROGRID Öğrenci Tablosu ile ilişkili olacak şekilde **Birincil Anahtar** olarak tasarlanmıştır.

|  |
| --- |
| **OGRTAYARLAR TABLOSU**  Tablo 5: ÖgrAyarlar Tablosu |
| AYARLAROGRID |
| OGRSIFRE |

**2.3.6. Genel Gösterim**

Varlık kümelerinin ve kümlere arasındaki var olan ilişkilerin toplu olarak gösterilmesi Şekil 14'de verilmiştir.

**Şekil 14:** Genel Gösterim

n

GİRER

YÖNETİCİ

ÖĞRETMEN

KARNE

VELİ

ALIR

ALIR

n

KAYIT

1

ALIR

n

1

1

n

m

ALIR

n

YAZDIRIR

n

n

1

ÖĞRENCİ

SINIF

SINIF

1

1

# **2.4. FİZİKSEL TASARIM**

# Proje MSSQL programı ile tasarlanmıştır. Projenin bazı bölümlerinde EntityFramework kullanılmış*tır*. Kayıtlı EntityFramework kısımları 6 adet Procedures kullanılmıştır. Kullanılan kısımlar öğrencilerin sınıflarını oluşturmak için kullanılmıştır. Öğretmen ve Öğrenci kayıt tablosu tüm tabloların bağlı olduğu ana tablodur. Projede toplamda 8 adet tablo bulunmaktadır. Projenin vize kısmını yetiştirmek olduğumuz için final zamanına kadar belirli eklemeler olacaktır.

# Bulunan tablo adları aşağıda listelenmiştir.

* **TBL\_AYARLAR** –Öğretmen Şifrelerinin kayıtlı olduğu tablo adıdır.
* **TBL\_BRANSLAR**– Öğretmen branşlarının kayıtlı olduğu tablo adıdır.
* **TBL\_ILCELER** – İlçelerin kayıtlı olduğu tablo adıdır.
* **TBL\_IL**–İllerin kayıtlı olduğu tablo adıdır.
* **TBL\_OGRAYARLAR-** Öğrenci şifrelerinin kayıtlı olduğu tablo adıdır.
* **TBL\_OGRENCILER** – Öğrenci Bilgilerinin kayıt olduğu tablo adıdır.
* **TBL\_OGRETMENLER-** Öğretmen Bilgilerinin kayıt olduğu tablo adıdır.
* **TBL\_VELILER-** Öğrenci Bilgilerinin kayıt olduğu tablo adıdır.

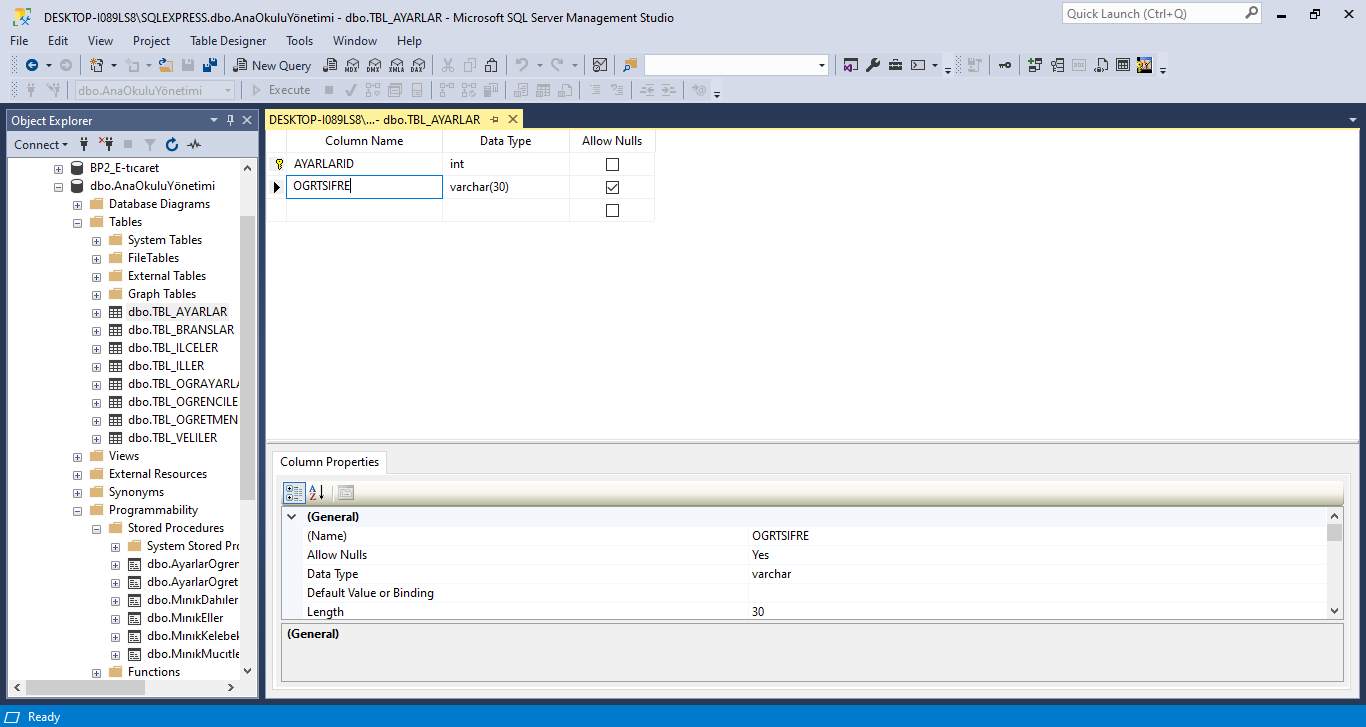
Kullanılan prodecure sadece öğrenci sınıflarını oluşturken kulanılmış. SQLKODALARI kısmındadır.

* **AyarlarOgrencıler** – Öğrenci ve şifresi için kullanılmıştır,
* **AyarlarOgretmenler** - Öğretmen ve şifresi için kullanılmıştır,
* **MınıkDahıler-** Öğrenci sınıfı için kullanılmıştır,
* **MınıkDahıler-** Öğrenci sınıfı için kullanılmıştır,
* **MınıkDahıler-** Öğrenci sınıfı için kullanılmıştır,
* **MınıkDahıler-** Öğrenci sınıfı için kullanılmıştır.

### **2.4.1. TBL\_AYARLAR Tablosu:**

Ayarlar Tablosu Alanları değişkenleri aşağıda belirtildiği şekilde tespit edilmiştir.

* AYARLARID, int veri tipi, Birincil Anahtar, Otomatik Artan ve Benzersiz
* OGRTSIFRE, nvarchar veri tipi.

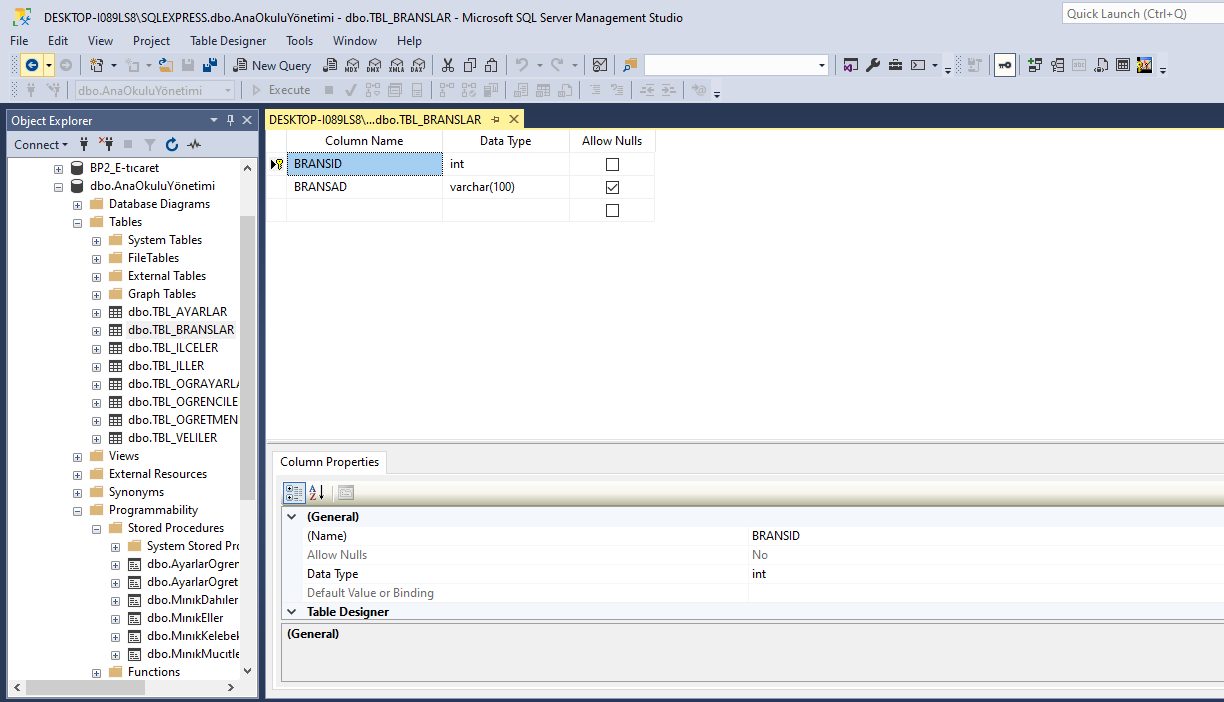


Tablo 6: TBL\_AYARLAR Tablosu

**2.4.2. TBL\_BRANSLAR Tablosu**

Branşlar Tablosu Alanları değişkenleri aşağıda belirtildiği şekilde tespit edilmiştir.

* BRANSID int veri tipi, Birincil Anahtar, Otomatik Artan ve Benzersiz
* BRANSAD, varchar veri tipi



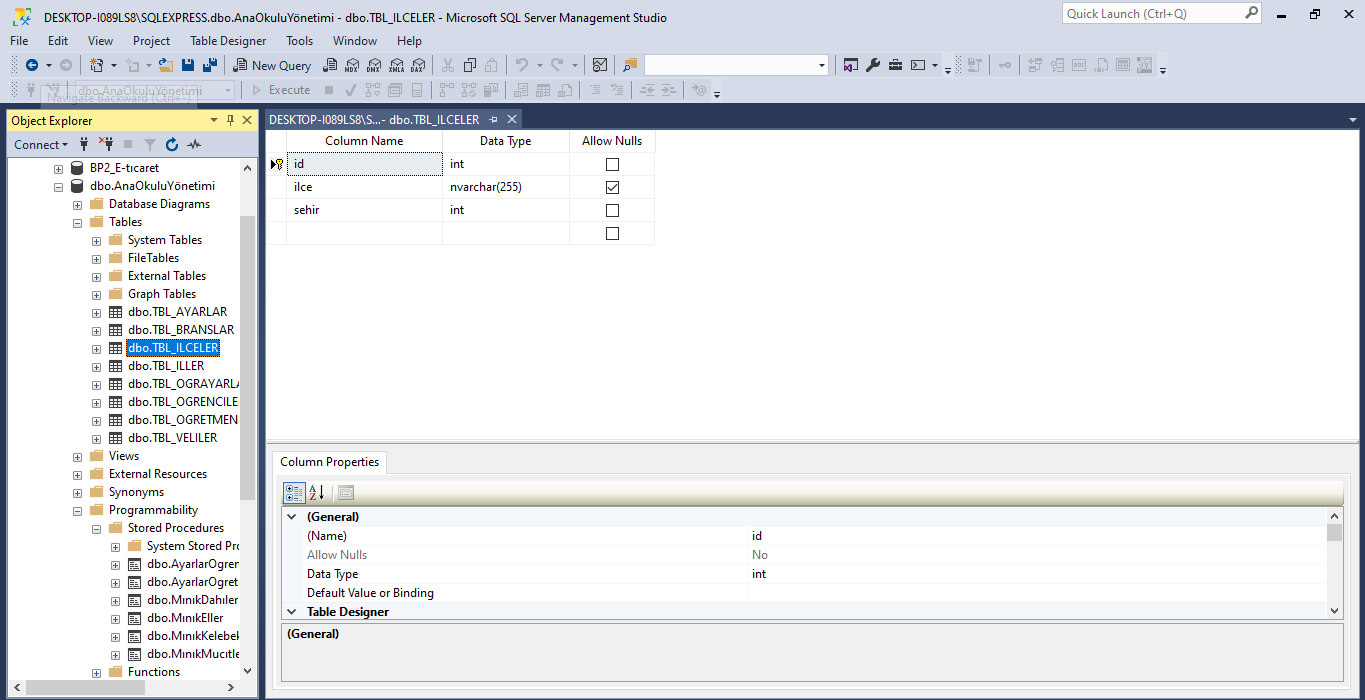
Tablo 7: TBL\_BRANSLAR Tablosu

### **2.4.3. TBL\_ILCELER Kayıt Tablosu**

Ilceler Tablosu Alanları değişkenleri aşağıda belirtildiği şekilde tespit edilmiştir.

* id int veri tipi, Birincil Anahtar, Otomatik Artan ve Benzersiz
* ilce varchar veri tipi nitelikleri olarak tasarlanmıştır.
* sehir int veri tipi.

Tablo 8: TBL\_ILCELER Tablosu

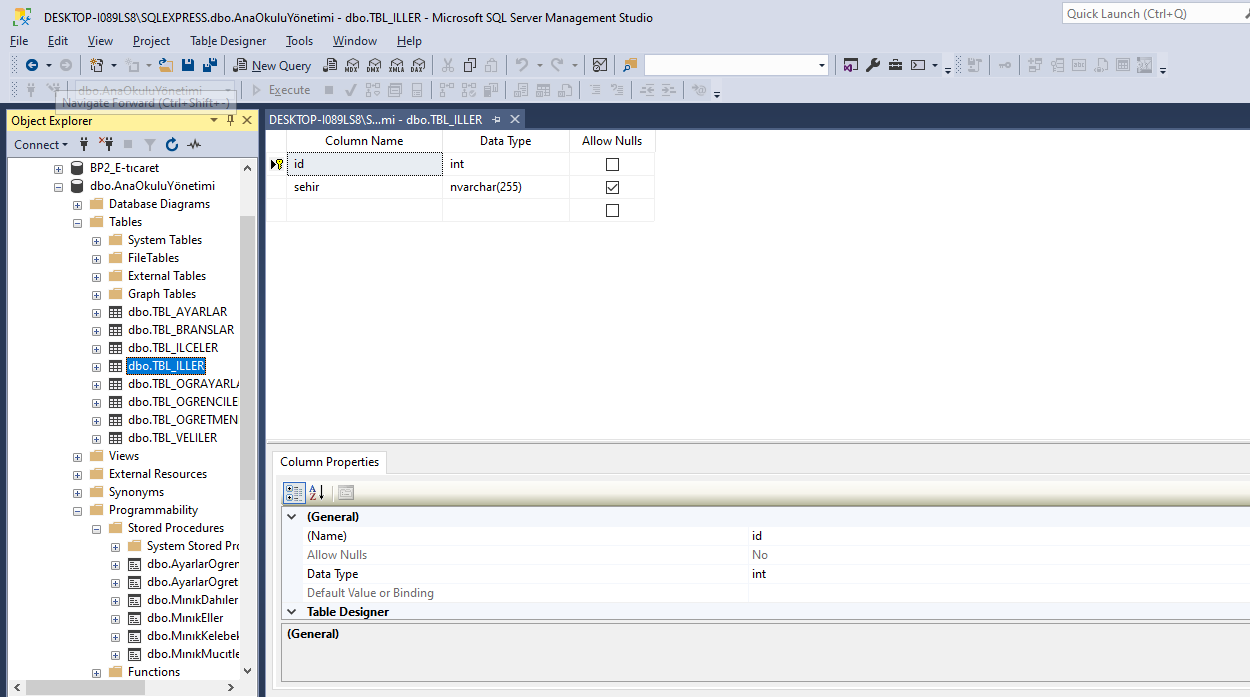


### **2.4.4. TBL\_ILLER Tablosu**

İller Tablosu Alanları değişkenleri aşağıda belirtildiği şekilde tespit edilmiştir.

* İd int veri tipi, Birincil Anahtar, Otomatik Artan ve Benzersiz
* sehir nvarchar.

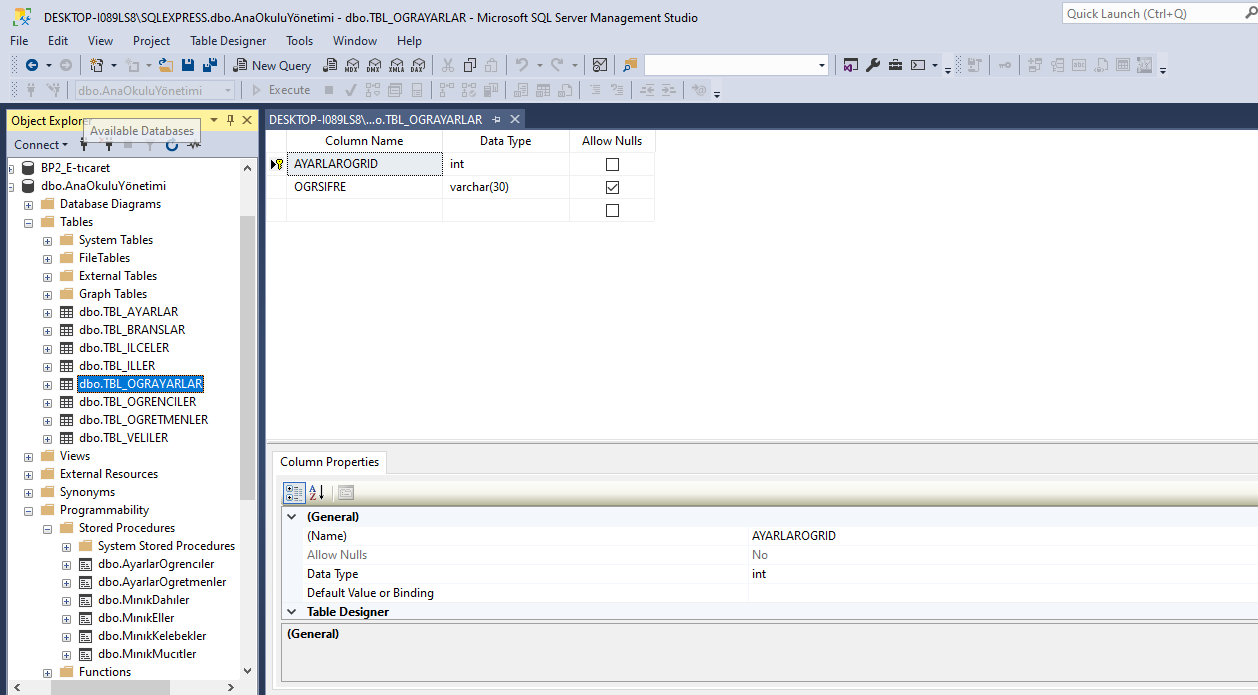
Tablo 9: TBL\_ILLER Tablosu



### **2.4.5. TBL\_OGRAYARLAR Tablosu**

Notlar Tablosu Alanları değişkenleri aşağıda belirtildiği şekilde tespit edilmiştir.

* AYARLAROGRID int veri tipi, Birincil Anahtar, Otomatik Artan ve Benzersiz
* OGRSIFRE varchar.

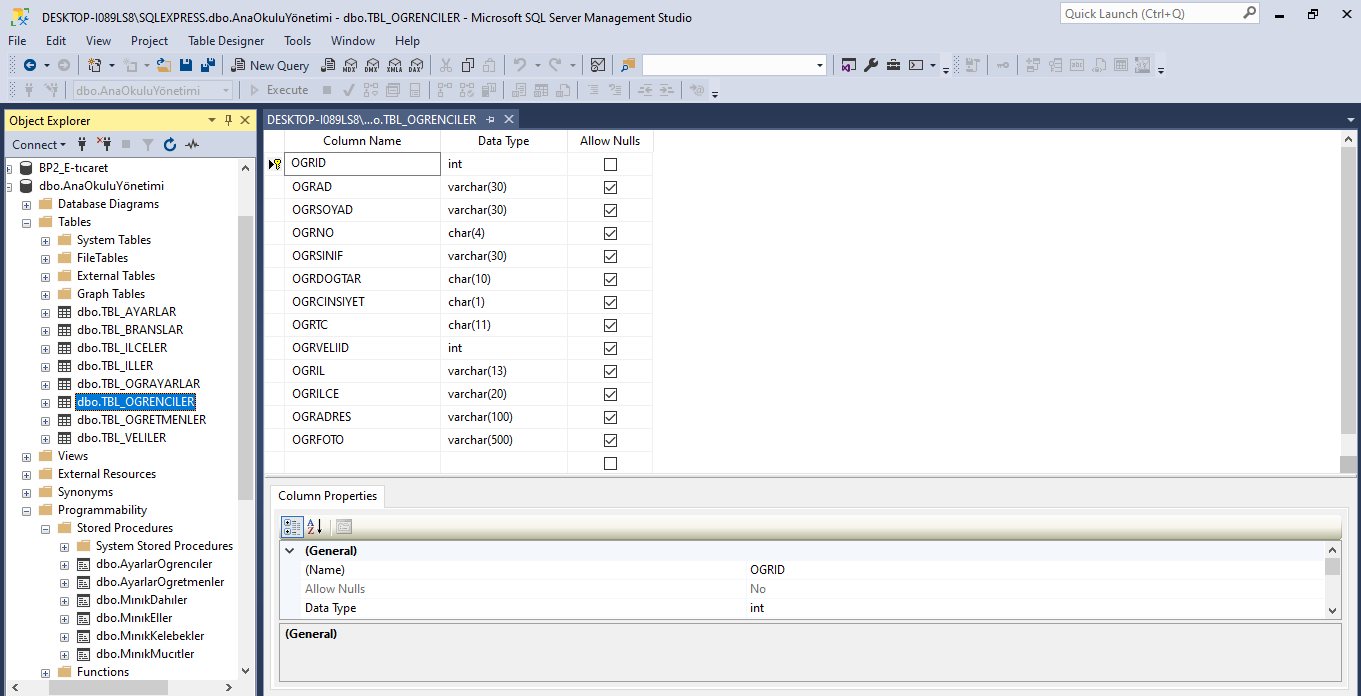


Tablo 10: TBL\_OGRAYARLAR Tablosu

### **2.4.6. TBL\_OGRENCILER Tablosu:**

Öğrenciler Tablosu Alanları değişkenleri aşağıda belirtildiği şekilde tespit edilmiştir.

* OGRID, int veri tipi, Birincil Anahtar, Otomatik Artan ve Benzersiz
* OGRAD, varchar veri tipi,
* OGRSOYAD, varchar veri tipi,
* OGRNO, char(4) veri tipi,
* OGRSINIF, varchar veri tipi,
* OGRDOGTAR, char(10) veri tipi,
* OGRCINSIYET, char(1) veri tipi
* OGRTC, char(11) veri tipi
* OGRVELIID, int veri tipi,
* OGRIL, varchar veri tipi,
* OGRILCE, varchar veri tipi,
* OGRADRES, varchar veri tipi,
* OGRFOTO, varchar veri tipi,

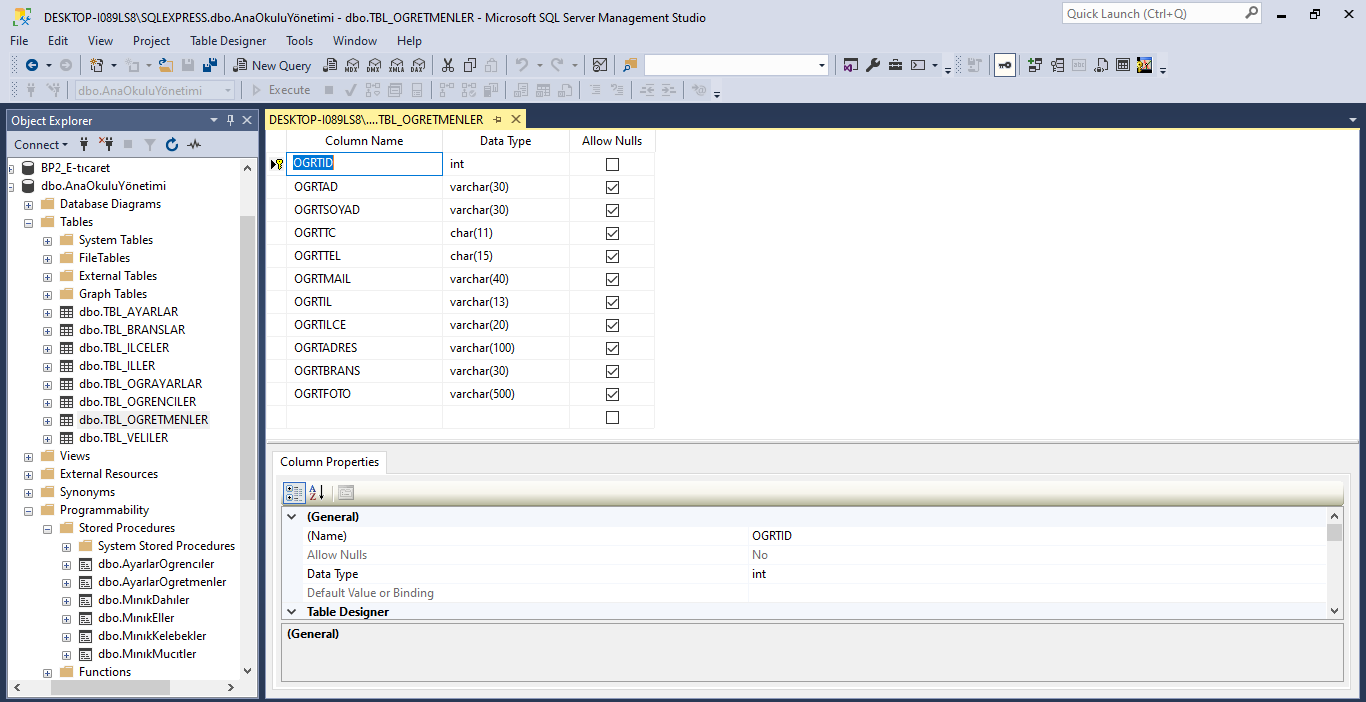
****

Tablo 11: TBL\_OGRENCILER Tablosu

### **2.4.7. TBL\_OGRETMENLER Tablosu:**

Öğretmenler Tablosu Alanları değişkenleri aşağıda belirtildiği şekilde tespit edilmiştir.

* OGRTID, int veri tipi, Birincil Anahtar, Otomatik Artan ve Benzersiz
* OGRTAD, varchar veri tipi,
* OGRTSOYAD, varchar veri tipi,
* OGRTTC, char(11) veri tipi,
* OGRTTEL,char(15),
* OGRTMAIL, varchar veri tipi,
* OGRTIL, varchar veri tipi,
* OGRTILCE, varchar veri tipi,
* OGRTADRES, varchar veri tipi,
* OGRTBRANS, varchar veri tipi,
* OGRTFOTO, varchar veri tipi.

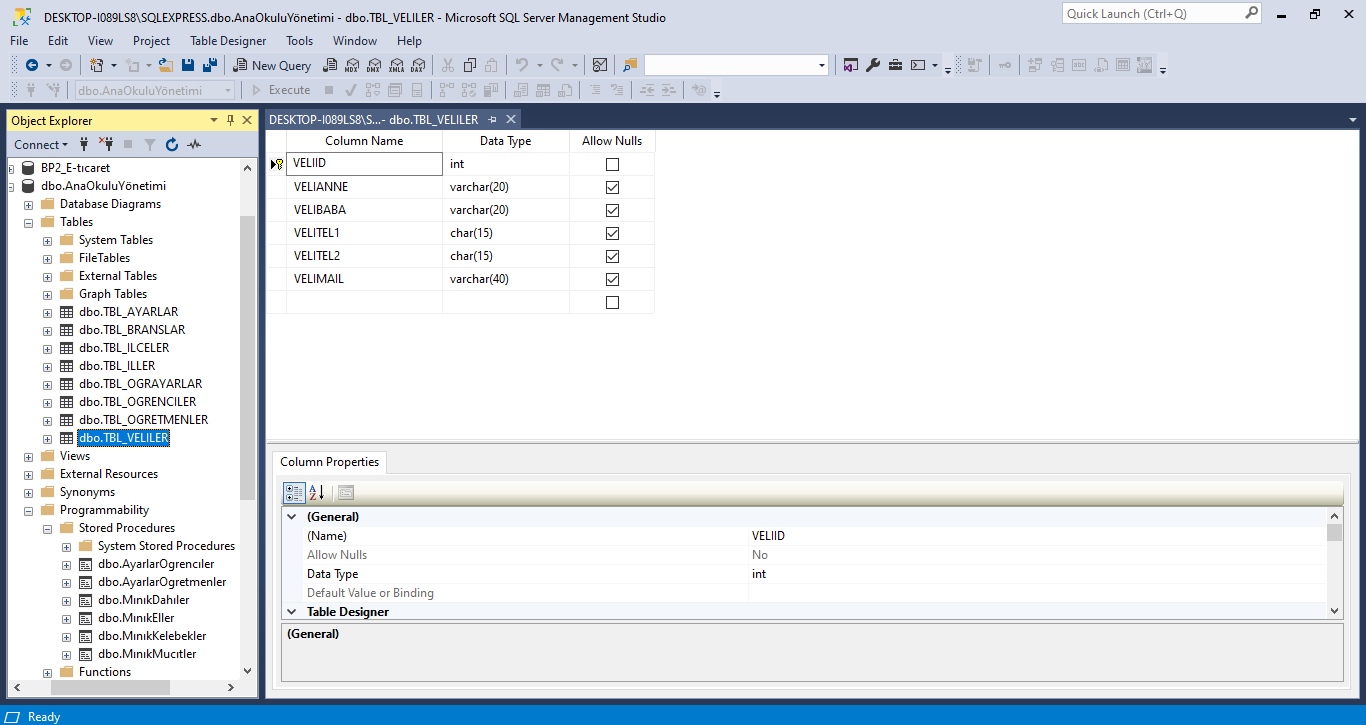
****

Tablo 12: TBL\_OGRETMENLER Tablosu

### **2.4.8. TBL\_VELILER Tablosu:**

Veliler Tablosu Alanları değişkenleri aşağıda belirtildiği şekilde tespit edilmiştir.

* VELIID, int veri tipi, Birincil Anahtar, Otomatik Artan ve Benzersiz
* VELIANNE, varchar veri tipi,
* VELIBABA, varchar veri tipi,
* VELITEL1, char(15) veri tipi,
* VELITEL2, char(15) veri tipi,
* VELIMAIL, varchar veri tipi.

****

Tablo 13: TBL\_VELILER Tablosu

**2.4.9. TABLOLARIN SQL KODLARI:**

**2.4.9.1. TBL\_AYARLAR:**

USE [dbo.AnaOkuluYönetimi]

GO

/\*\*\*\*\*\* Object: Table [dbo].[TBL\_AYARLAR] Script Date: 8.04.2020 19:11:21 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[TBL\_AYARLAR](

[AYARLARID] [int] NOT NULL,

[OGRTSIFRE] [varchar](30) NULL,

CONSTRAINT [PK\_TBL\_AYARLAR] PRIMARY KEY CLUSTERED

(

[AYARLARID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

ALTER TABLE [dbo].[TBL\_AYARLAR] WITH CHECK ADD CONSTRAINT [FK\_TBL\_AYARLAR\_TBL\_OGRETMENLER] FOREIGN KEY([AYARLARID])

REFERENCES [dbo].[TBL\_OGRETMENLER] ([OGRTID])

GO

ALTER TABLE [dbo].[TBL\_AYARLAR] CHECK CONSTRAINT [FK\_TBL\_AYARLAR\_TBL\_OGRETMENLER]

GO

**2.4.9.2. TBL\_BRANSLAR:**

USE [dbo.AnaOkuluYönetimi]

GO

/\*\*\*\*\*\* Object: Table [dbo].[TBL\_BRANSLAR] Script Date: 8.04.2020 19:12:18 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[TBL\_BRANSLAR](

[BRANSID] [int] IDENTITY(1,1) NOT NULL,

[BRANSAD] [varchar](100) NULL,

CONSTRAINT [PK\_TBL\_BRANSLAR] PRIMARY KEY CLUSTERED

(

[BRANSID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

**2.4.9.3. TBL\_ILCELER:**

USE [dbo.AnaOkuluYönetimi]

GO

/\*\*\*\*\*\* Object: Table [dbo].[TBL\_ILCELER] Script Date: 8.04.2020 19:13:00 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[TBL\_ILCELER](

[id] [int] NOT NULL,

[ilce] [nvarchar](255) NULL,

[sehir] [int] IDENTITY(1,1) NOT NULL,

CONSTRAINT [PK\_ilceler] PRIMARY KEY CLUSTERED

(

[id] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

**2.4.9.4. TBL\_ILLER:**

USE [dbo.AnaOkuluYönetimi]

GO

/\*\*\*\*\*\* Object: Table [dbo].[TBL\_ILLER] Script Date: 8.04.2020 19:13:39 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[TBL\_ILLER](

[id] [int] IDENTITY(1,1) NOT NULL,

[sehir] [nvarchar](255) NULL,

CONSTRAINT [PK\_iller] PRIMARY KEY CLUSTERED

(

[id] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

**2.4.9.5. TBL\_OGRAYARLAR:**

USE [dbo.AnaOkuluYönetimi]

GO

/\*\*\*\*\*\* Object: Table [dbo].[TBL\_OGRAYARLAR] Script Date: 8.04.2020 19:14:23 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[TBL\_OGRAYARLAR](

[AYARLAROGRID] [int] NOT NULL,

[OGRSIFRE] [varchar](30) NULL,

CONSTRAINT [PK\_TBL\_OGRAYARLAR] PRIMARY KEY CLUSTERED

(

[AYARLAROGRID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

ALTER TABLE [dbo].[TBL\_OGRAYARLAR] WITH CHECK ADD CONSTRAINT [FK\_TBL\_OGRAYARLAR\_TBL\_OGRENCILER] FOREIGN KEY([AYARLAROGRID])

REFERENCES [dbo].[TBL\_OGRENCILER] ([OGRID])

GO

ALTER TABLE [dbo].[TBL\_OGRAYARLAR] CHECK CONSTRAINT [FK\_TBL\_OGRAYARLAR\_TBL\_OGRENCILER]

GO

**2.4.9.6. TBL\_OGRENCILER:**

USE [dbo.AnaOkuluYönetimi]

GO

/\*\*\*\*\*\* Object: Table [dbo].[TBL\_OGRENCILER] Script Date: 8.04.2020 19:15:06 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[TBL\_OGRENCILER](

[OGRID] [int] IDENTITY(1,1) NOT NULL,

[OGRAD] [varchar](30) NULL,

[OGRSOYAD] [varchar](30) NULL,

[OGRNO] [char](4) NULL,

[OGRSINIF] [varchar](30) NULL,

[OGRDOGTAR] [char](10) NULL,

[OGRCINSIYET] [char](1) NULL,

[OGRTC] [char](11) NULL,

[OGRVELIID] [int] NULL,

[OGRIL] [varchar](13) NULL,

[OGRILCE] [varchar](20) NULL,

[OGRADRES] [varchar](100) NULL,

[OGRFOTO] [varchar](500) NULL,

CONSTRAINT [PK\_TBL\_OGRENCILER] PRIMARY KEY CLUSTERED

(

[OGRID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

ALTER TABLE [dbo].[TBL\_OGRENCILER] WITH CHECK ADD CONSTRAINT [FK\_TBL\_OGRENCILER\_TBL\_VELILER] FOREIGN KEY([OGRVELIID])

REFERENCES [dbo].[TBL\_VELILER] ([VELIID])

GO

ALTER TABLE [dbo].[TBL\_OGRENCILER] CHECK CONSTRAINT [FK\_TBL\_OGRENCILER\_TBL\_VELILER]

GO

**2.4.9.7. TBL\_OGRETMENLER:**

USE [dbo.AnaOkuluYönetimi]

GO

/\*\*\*\*\*\* Object: Table [dbo].[TBL\_OGRETMENLER] Script Date: 8.04.2020 19:15:39 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[TBL\_OGRETMENLER](

[OGRTID] [int] IDENTITY(1,1) NOT NULL,

[OGRTAD] [varchar](30) NULL,

[OGRTSOYAD] [varchar](30) NULL,

[OGRTTC] [char](11) NULL,

[OGRTTEL] [char](15) NULL,

[OGRTMAIL] [varchar](40) NULL,

[OGRTIL] [varchar](13) NULL,

[OGRTILCE] [varchar](20) NULL,

[OGRTADRES] [varchar](100) NULL,

[OGRTBRANS] [varchar](30) NULL,

[OGRTFOTO] [varchar](500) NULL,

CONSTRAINT [PK\_TBL\_OGRETMENLER] PRIMARY KEY CLUSTERED

(

[OGRTID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

**2.4.9.8. TBL\_VELILER:**

USE [dbo.AnaOkuluYönetimi]

GO

/\*\*\*\*\*\* Object: Table [dbo].[TBL\_VELILER] Script Date: 8.04.2020 19:16:28 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[TBL\_VELILER](

[VELIID] [int] IDENTITY(1,1) NOT NULL,

[VELIANNE] [varchar](20) NULL,

[VELIBABA] [varchar](20) NULL,

[VELITEL1] [char](15) NULL,

[VELITEL2] [char](15) NULL,

[VELIMAIL] [varchar](40) NULL,

CONSTRAINT [PK\_TBL\_VELI] PRIMARY KEY CLUSTERED

(

[VELIID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

**2.4.9.9. AyarlarOgrencıler(Prodecure):**

USE [dbo.AnaOkuluYönetimi]

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[AyarlarOgrencıler] Script Date: 8.04.2020 19:16:55 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

ALTER Procedure [dbo].[AyarlarOgrencıler]

as

Select AYARLAROGRID,(OGRAD+' '+OGRSOYAD) as OGRADSOYAD,OGRSINIF,OGRTC,OGRSIFRE,OGRFOTO From TBL\_OGRAYARLAR

inner join TBL\_OGRENCILER

on

TBL\_OGRAYARLAR.AYARLAROGRID=TBL\_OGRENCILER.OGRID

**2.4.9.10. AyarlarOgretmenler(Prodecure):**

USE [dbo.AnaOkuluYönetimi]

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[AyarlarOgretmenler] Script Date: 8.04.2020 19:18:37 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

ALTER Procedure [dbo].[AyarlarOgretmenler]

as

Select AYARLARID,(OGRTAD+' '+OGRTSOYAD) as OGRTADSOYAD,OGRTBRANS,OGRTTC,OGRTFOTO,OGRTSIFRE From TBL\_AYARLAR

inner join TBL\_OGRETMENLER

on

TBL\_AYARLAR.AYARLARID=TBL\_OGRETMENLER.OGRTID

**2.4.9.11. MınıkDahıler(Prodecure):**

USE [dbo.AnaOkuluYönetimi]

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[MınıkDahıler] Script Date: 8.04.2020 19:19:36 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

ALTER Procedure [dbo].[MınıkDahıler]

as

Select OGRID,OGRAD,OGRSOYAD,OGRTC,OGRSINIF,OGRNO,

OGRDOGTAR,OGRCINSIYET,OGRIL,OGRILCE,OGRADRES,OGRFOTO,

(VELIANNE+'|'+VELIBABA) as VELIANNEBABA

From TBL\_OGRENCILER

Inner join TBL\_VELILER

on

TBL\_OGRENCILER.OGRVELIID=TBL\_VELILER.VELIID

where OGRSINIF='Minik Dahiler'

**2.4.9.12. MınıkEller(Prodecure):**

USE [dbo.AnaOkuluYönetimi]

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[MınıkEller] Script Date: 8.04.2020 19:20:16 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

ALTER Procedure [dbo].[MınıkEller]

as

Select OGRID,OGRAD,OGRSOYAD,OGRTC,OGRSINIF,OGRNO,

OGRDOGTAR,OGRCINSIYET,OGRIL,OGRILCE,OGRADRES,OGRFOTO,

(VELIANNE+'|'+VELIBABA) as VELIANNEBABA

From TBL\_OGRENCILER

Inner join TBL\_VELILER

on

TBL\_OGRENCILER.OGRVELIID=TBL\_VELILER.VELIID

where OGRSINIF='Minik Eller'

**2.4.9.13. MınıkKelebekler(Prodecure):**

USE [dbo.AnaOkuluYönetimi]

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[MınıkKelebekler] Script Date: 8.04.2020 19:20:51 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

ALTER Procedure [dbo].[MınıkKelebekler]

as

Select OGRID,OGRAD,OGRSOYAD,OGRTC,OGRSINIF,OGRNO,

OGRDOGTAR,OGRCINSIYET,OGRIL,OGRILCE,OGRADRES,OGRFOTO,

(VELIANNE+'|'+VELIBABA) as VELIANNEBABA

From TBL\_OGRENCILER

Inner join TBL\_VELILER

on

TBL\_OGRENCILER.OGRVELIID=TBL\_VELILER.VELIID

where OGRSINIF='Minik Kelebekler'

**2.4.9.14. MınıkMucıtler(Prodecure):**

USE [dbo.AnaOkuluYönetimi]

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[MınıkMucıtler] Script Date: 8.04.2020 19:21:24 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

ALTER Procedure [dbo].[MınıkMucıtler]

as

Select OGRID,OGRAD,OGRSOYAD,OGRTC,OGRSINIF,OGRNO,

OGRDOGTAR,OGRCINSIYET,OGRIL,OGRILCE,OGRADRES,OGRFOTO,

(VELIANNE+'|'+VELIBABA) as VELIANNEBABA

From TBL\_OGRENCILER

Inner join TBL\_VELILER

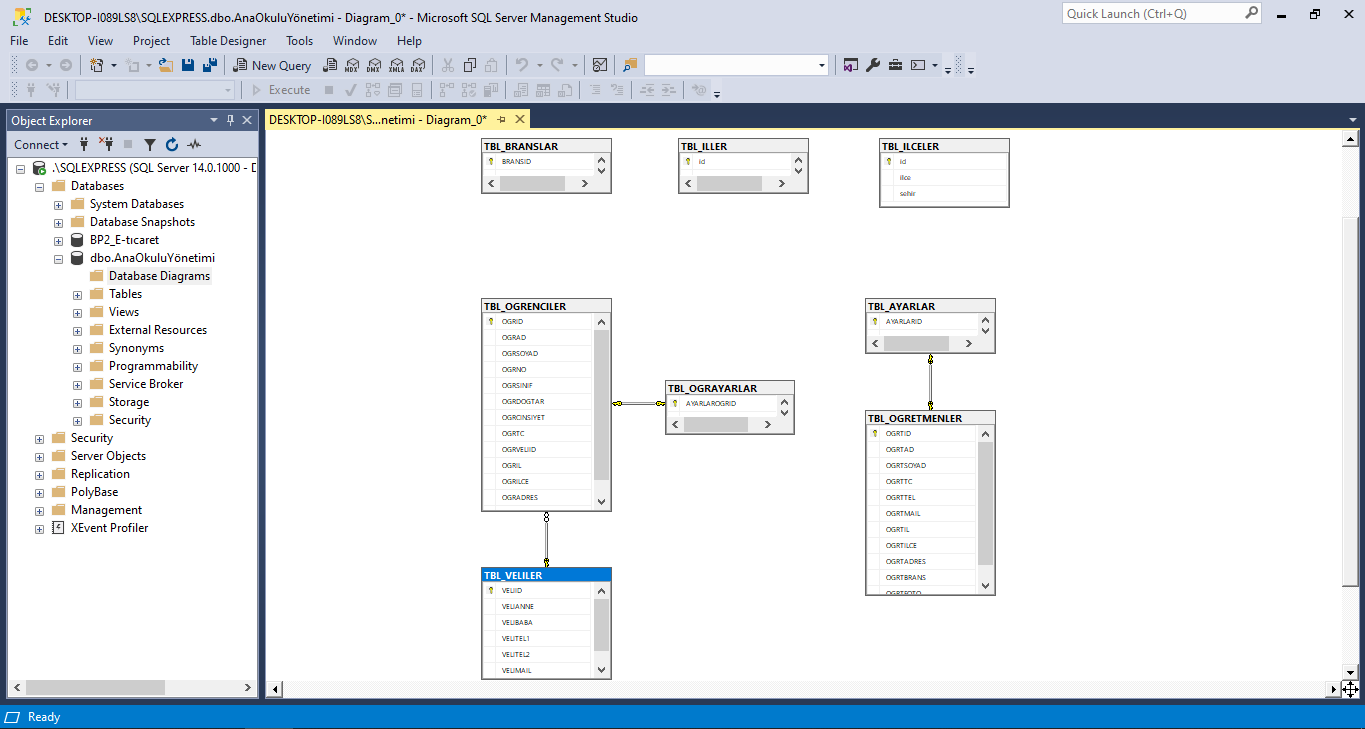
on

TBL\_OGRENCILER.OGRVELIID=TBL\_VELILER.VELIID

where OGRSINIF='Minik Mucitler'

**2.4.10. Veri Tabanı Diyagramı**

Tablolar yukarıda belirtilen analizler ışığında aşağıda belirtildiği şekilde birbirleri ile ilişkilendirilmiş ve diyagram oluşturulmuştur.

****

Şekil 15: Veritabanı Diyagramı

**2.5. Projenin Akış Şeması**

**2.5.1. Giriş Ekranı Akış Şeması**

**BAŞLA**

**Veri Tabanı Bağlantısını  
Kapat  
Close()  
  
MySqlCommand**

Tekrar Dene

**Uyarı  
Veri Tabanı ile Bağlantı Kurulamadı.**

Evet

**Veri Tabanı ile Bağlantı var mı? mı?**

**Şifre**

Hayır

Hayır

**Kullanıcı ve Şifre Doğru Evet ise**

**Yönetici veya Öğretmen veya Öğrenci Kullanıcı Adı ve Şifre Ekranı**

**Ana Menü (Öğrenci veya Yönetici Ekranı)**

**Uyarı  
Kullanıcı Adı Ve Şifresi Hatalı**  
**Hatalı veya Girilmedi**

Evet

**2.5.2. Yönetici Ekranı Öğrenci Ekleme Modülü Akış Şeması**

**Öğrenci Bilgileri  
Form Ekranı Gelir “Öğrenci No” sor**

**Bilgileri İlgili Araçlara Yükle**

**Bilgi Girişi**

**Öğrenci Bilgilerini Al**

**Veritabanına Bağlan**

**Doğruluğu onayla**

**Bilgileri Command Nesnesine Aktar**

**Uyarı  
Eklemeden Vazgeçildi.**

**Kaydet**

Hayır

Evet

**Öğrenci Bilgileri Kayıt Edildi.**

**2.5.3. Yönetici Ekranı Öğrenci Silme Modülü Akış Şeması**

**Bilgi Silme**

**Bilgileri Command Nesnesine Aktar**

**Bilgileri İlgili Araçlara Yükle**

**Öğrenci Adını Al**

**Veritabanına Bağlan**

**Doğruluğu onayla**

**Uyarı  
Silmeden Vazgeçildi.**

**Silmeyi Onayla**

Hayır

Evet

**Öğrenci Bilgileri Silindi**

**2.5.4. Yönetici Ekranı Öğrenci Güncelleme Modülü Akış Şeması**

**Bilgi Güncelleme**

**Bilgileri Command Nesnesine Aktar**

**Bilgileri İlgili Araçlara Yükle**

**Öğrenci Adını Al**

**Veritabanına Bağlan**

**Doğruluğu onayla**

**Uyarı  
Vazgeçildi.**

**Güncellemeyi Onayla**

Hayır

Evet

**Öğrenci Bilgileri Güncellendi**

**2.5.5. Yönetici Ekranı Öğrenci Görüntüleme Modülü Akış Şeması**

**Bilgi Görüntüleme**

**Bilgileri Command Nesnesine Aktar**

**Bilgileri İlgili Araçlara Yükle**

**Öğrenci Adını Al**

**Veritabanına Bağlan**

**Doğruluğu onayla**

**Öğrenci Bilgileri Görüntülendi**

**2.5.6. Yönetici Ekranı Öğretmen Ekleme Modülü Akış Şeması**

**Öğretmen Bilgileri  
Form Ekranı Gelir “Öğretmen Adı” sor**

**Bilgileri Command Nesnesine Aktar**

**Bilgileri İlgili Araçlara Yükle**

**Bilgi Girişi**

**Öğretmen Bilgilerini Al**

**Veritabanına Bağlan**

**Doğruluğu onayla**

**Uyarı  
Eklemeden Vazgeçildi.**

**Kaydet**

Hayır

Evet

**Öğretmen Bilgileri Kayıt Edildi.**

**2.5.7. Yönetici Ekranı Öğretmen Silme Modülü Akış Şeması**

**Bilgi Silme**

**Öğretmen Adını Al**

**Veritabanına Bağlan**

**Doğruluğu onayla**

**Bilgileri İlgili Araçlara Yükle**

**Bilgileri Command Nesnesine Aktar**

**Uyarı  
Silmeden Vazgeçildi.**

**Silmeyi Onayla**

Hayır

Evet

**Öğretmen Bilgileri Silindi**

**2.5.8. Yönetici Ekranı Öğretmen Güncelleme Modülü Akış Şeması**

**Bilgi Güncelleme**

**Öğretmen Adını Al**

**Veritabanına Bağlan**

**Doğruluğu onayla**

**Bilgileri İlgili Araçlara Yükle**

**Bilgileri Command Nesnesine Aktar**

**Uyarı  
Vazgeçildi.**

**Güncellemeyi Onayla**

Hayır

Evet

**Öğretmen Bilgileri Güncellendi**

**2.5.9. Yönetici Ekranı Veli Ekleme Modülü Akış Şeması**

**Veli Bilgileri  
Form Ekranı Gelir “Veli Adı” sor**

**Bilgileri İlgili Araçlara Yükle**

**Bilgi Girişi**

**Veli Bilgilerini Al**

**Veritabanına Bağlan**

**Doğruluğu onayla**

**Bilgileri Command Nesnesine Aktar**

**Uyarı  
Eklemeden Vazgeçildi.**

**Kaydet**

Hayır

Evet

**Veli Bilgileri Kayıt Edildi.**

**2.5.10. Yönetici Ekranı Veli Silme Modülü Akış Şeması**

**Bilgi Silme**

**Veli Adını Al**

**Veritabanına Bağlan**

**Doğruluğu onayla**

**Bilgileri İlgili Araçlara Yükle**

**Bilgileri Command Nesnesine Aktar**

**Uyarı  
Silmeden Vazgeçildi.**

**Silmeyi Onayla**

Hayır

Evet

**Veli Bilgileri Silindi**

**2.5.11. Yönetici Ekranı Veli Güncelleme Modülü Akış Şeması**

**Bilgi Güncelleme**

**Veli Adını Al**

**Veritabanına Bağlan**

**Doğruluğu onayla**

**Bilgileri İlgili Araçlara Yükle**

**Bilgileri Command Nesnesine Aktar**

**Uyarı  
Vazgeçildi.**

**Güncellemeyi Onayla**

Hayır

Evet

**Veli Bilgileri Güncellendi**

**2.5.12. Yönetici Ekranı Fatura Ekleme Modülü Akış Şeması**

**Faturalar Bilgileri  
Form Ekranı Gelir “Fatura Adı” sor**

**Bilgi Girişi**

**Fatura Bilgilerini Al**

**Veritabanına Bağlan**

**Doğruluğu onayla**

**Bilgileri İlgili Araçlara Yükle**

**Bilgileri Command Nesnesine Aktar**

**Uyarı  
Eklemeden Vazgeçildi.**

**Kaydet**

Hayır

Evet

**Fatura Bilgileri Kayıt Edildi.**

**2.5.13. Yönetici Ekranı Giderler Ekleme Modülü Akış Şeması**

**Gider Bilgileri  
Form Ekranı Gelir “Gider Adı” sor**

**Bilgileri İlgili Araçlara Yükle**

**Bilgi Girişi**

**Gider Bilgilerini Al**

**Veritabanına Bağlan**

**Doğruluğu onayla**

**Bilgileri Command Nesnesine Aktar**

**Uyarı  
Eklemeden Vazgeçildi.**

**Kaydet**

**Gider Bilgileri Kayıt Edildi.**

Evet

Hayır

**2.5.14. Yönetici Ekranı Kasa Ekleme Modülü Akış Şeması**

**Kasa Bilgileri  
Form Ekranı Gelir “Kasa Adı” sor**

**Bilgi Girişi**

**Kasa Bilgilerini Al**

**Veritabanına Bağlan**

**Doğruluğu onayla**

**Bilgileri İlgili Araçlara Yükle**

**Bilgileri Command Nesnesine Aktar**

**Uyarı  
Eklemeden Vazgeçildi.**

**Kasa Bilgileri Kayıt Edildi.**

**Kaydet**

Evet

Hayır

**2.5.15. Yönetici Ekranı Raporlar Ekleme Modülü Akış Şeması**

**Rapor Bilgileri  
Form Ekranı Gelir “Rapor Adı” sor**

**Bilgileri Command Nesnesine Aktar**

**Bilgileri İlgili Araçlara Yükle**

**Bilgi Girişi**

**Rapor Bilgilerini Al**

**Veritabanına Bağlan**

**Doğruluğu onayla**

**Uyarı  
Eklemeden Vazgeçildi.**

**Kaydet**

**Rapor Bilgileri Kayıt Edildi.**

Evet

Hayır

**2.5.16. Yönetici Ekranı Mesaj Ekleme Modülü Akış Şeması**

**Mesaj Bilgileri  
Form Ekranı Gelir “Mesaj Adı” sor**

**Bilgileri Command Nesnesine Aktar**

**Bilgileri İlgili Araçlara Yükle**

**Bilgi Girişi**

**Mesaj Bilgilerini Al**

**Veritabanına Bağlan**

**Doğruluğu onayla**

**Uyarı  
Eklemeden Vazgeçildi.**

**Kaydet**

**Mesaj Bilgileri Kayıt Edildi.**

Evet

Hayır

**2.5.17. Yönetici Ekranı Ayarlar Ekleme Modülü Akış Şeması**

**Ayarlar Bilgileri  
Form Ekranı Gelir “Ayarlar Adı” sor**

**Bilgileri Command Nesnesine Aktar**

**Bilgileri İlgili Araçlara Yükle**

**Bilgi Girişi**

**Ayarlar Bilgilerini Al**

**Veritabanına Bağlan**

**Doğruluğu onayla**

**Uyarı  
Eklemeden Vazgeçildi.**

**Kaydet**

**Ayarlar Bilgileri Kayıt Edildi.**

Evet

Hayır

**2.5.18. Yönetici Ekranı Ayalar Güncelleme Modülü Akış Şeması**

**Bilgi Güncelleme**

**Ayarlar Adını Al**

**Veritabanına Bağlan**

**Doğruluğu onayla**

**Bilgileri İlgili Araçlara Yükle**

**Bilgileri Command Nesnesine Aktar**

**Uyarı  
Vazgeçildi.**

**Güncellemeyi Onayla**

Hayır

Evet

**Ayarlar Bilgileri Güncellendi**

**2.5.19. Öğretmen Ekranı Davranış Ekleme Modülü Akış Şeması**

**Davranış Bilgileri  
Form Ekranı Gelir “Davranış Adı” sor**

**Bilgileri Command Nesnesine Aktar**

**Bilgileri İlgili Araçlara Yükle**

**Bilgi Girişi**

**Davranış Bilgilerini Al**

**Veritabanına Bağlan**

**Doğruluğu onayla**

**Uyarı  
Eklemeden Vazgeçildi.**

**Davranış Bilgileri Kayıt Edildi.**

**Kaydet**

Evet

Hayır

**2.5.20. Öğretmen Ekranı İstatistik Ekleme Modülü Akış Şeması**

**İstatistik Bilgileri  
Form Ekranı Gelir “Öğrenci Adı” sor**

**Bilgileri Command Nesnesine Aktar**

**Bilgileri İlgili Araçlara Yükle**

**Bilgi Girişi**

**İstatistik Bilgilerini Al**

**Veritabanına Bağlan**

**Doğruluğu onayla**

**Uyarı  
Eklemeden Vazgeçildi.**

**İstatistik Bilgileri Kayıt Edildi.**

**Kaydet**

Evet

Hayır

**2.5.21. Öğretmen Ekranı Not Ekleme Modülü Akış Şeması**

**Notlar Bilgileri  
Form Ekranı Gelir “Notlar Adı” sor**

**Bilgileri Command Nesnesine Aktar**

**Bilgileri İlgili Araçlara Yükle**

**Bilgi Girişi**

**Notlar Bilgilerini Al**

**Veritabanına Bağlan**

**Doğruluğu onayla**

**Uyarı  
Eklemeden Vazgeçildi.**

**Notlar Bilgileri Kayıt Edildi.**

**Kaydet**

Evet

Hayır

**2.5.22. Öğrenci Ekranı Davranış ve İstatistik Görüntüleme Modülü Akış Şeması**

**Bilgi Görüntüleme**

**Bilgileri Command Nesnesine Aktar**

**Bilgileri İlgili Araçlara Yükle**

**Öğrenci Adını Al**

**Veritabanına Bağlan**

**Doğruluğu onayla**

**Öğrenci Bilgileri Görüntülendi**

**2.6. Proje Uygulama Kısmı**

**2.6.1. Anaokulu Yönetim Sistemi Giriş Ekranı**

Öncelikli olarak MSSQL veri tabanı ile bağlantı kurulması sağlanmıştır. Giriş ekranında Yönetici, Öğretmen ve öğrencinin giriş yapması hedeflenmiştir. Yönetici kullanıcı adı ve şifresi doğru girildiğinde **Yönetici Ekranı** gelmektedir.

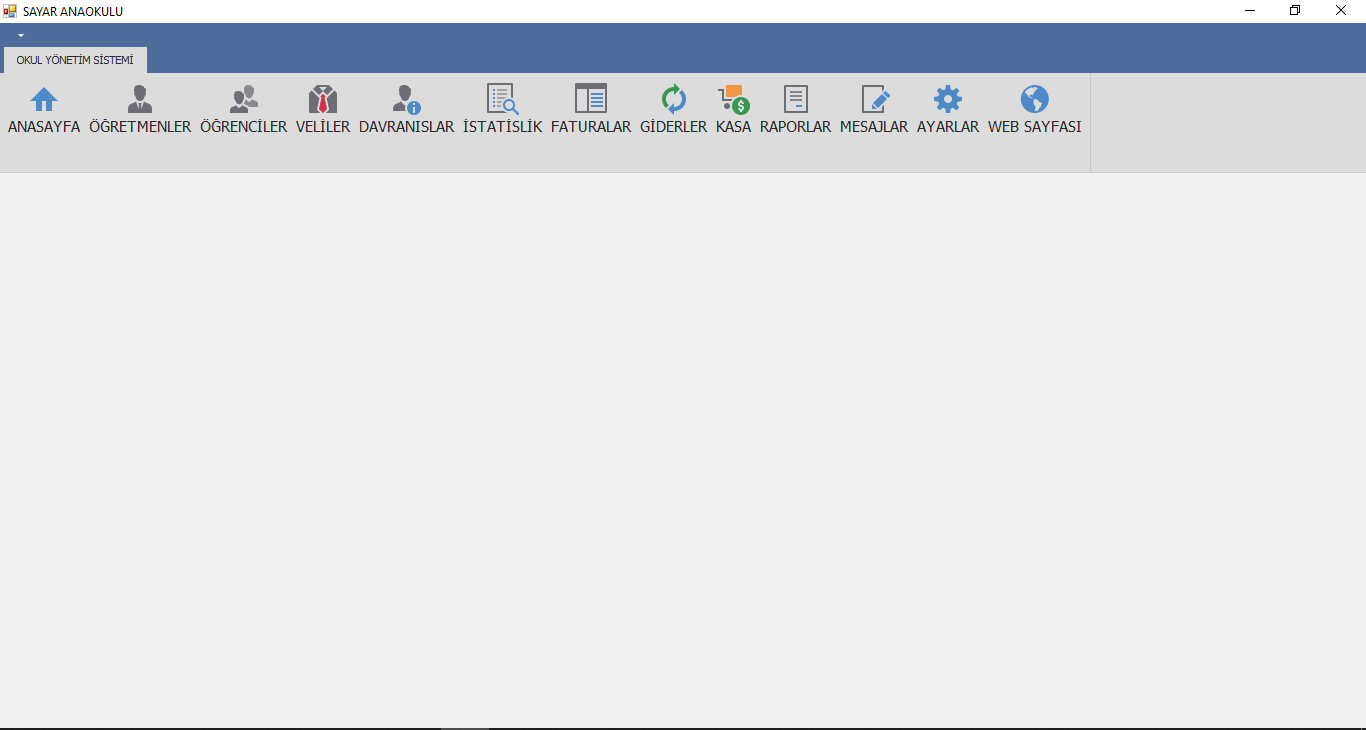


Şekil 16: Giriş Ekranı

## **Anaokulu Yönetim Sistemi Yönetici Ana Ekranı**

Yönetici olarak giriş yapıldığında **Yönetici Ana Ekranı** gelmektedir. Yönetici Ana Ekranında sırasıyla sekme olarak;

* Öğrenci Bilgi İşlemleri Sekmesi,
* Öğretmen Bilgi İşlemleri Sekmesi,
* Veli Bilgi İşlemleri Sekmesi,
* Davranış Bilgi Sekmesi,
* İstatislik Bilgi Sekmesi,
* Faturalar Bilgi İşlemleri Sekmesi,
* Giderler Bilgi İşlemler Sekmesi,
* Kasa Bilgi İşlemler Sekmesi,
* Raporlar Bilgi Sekmesi,
* Mesajlar Bilgi Sekmesi,
* Ayarlar Bilgi İşlem Sekmesi,
* Web Sayfası Bilgi Sekmesi.



Şekil 17: Yönetici Ana Sayfa

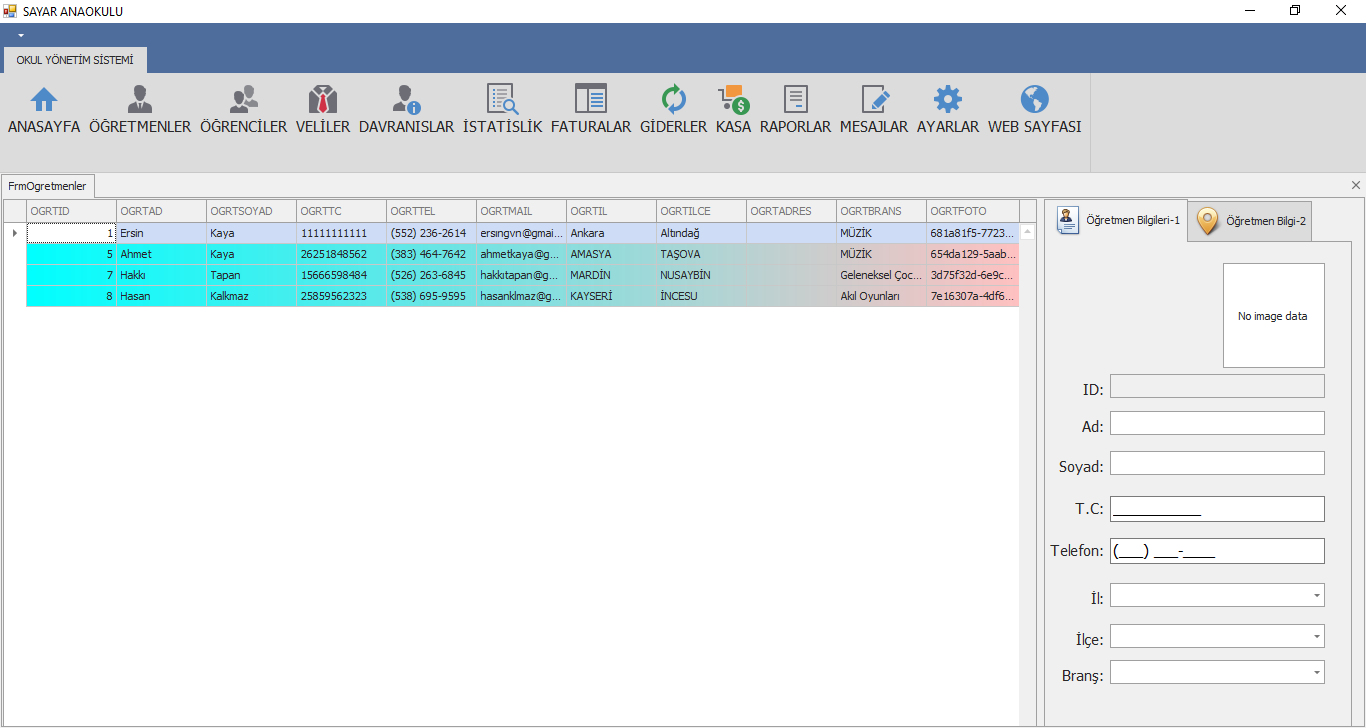
## **2.6.2.1 Öğretmen Bilgi İşlemleri Sekmesi:**

Öğretmen Bilgi İşlemleri Sekmesinde;

* Ekle,
* Sil,
* Güncelle,
* Resim Sec,
* Temizle.

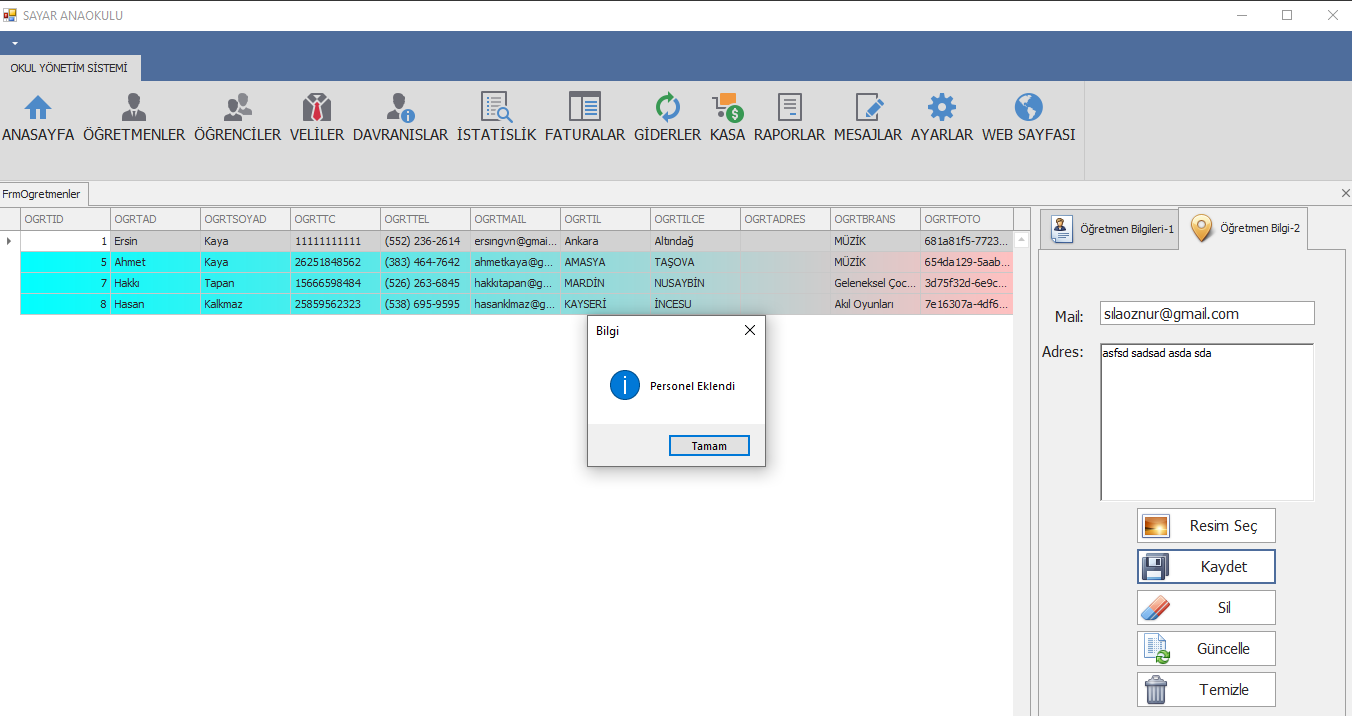
Bu işlemlere ait veriler TBL\_OGRETMENLER tablosundan getirilmekte. Grid Control ile gösterilmektedir. Bu bilgiler;

* Öğretmen İd,
* Adı,
* Soyadı,
* TC,
* Telefon,
* İl,
* İlçe,
* Branş,
* Mail,
* Adres,
* Resim.



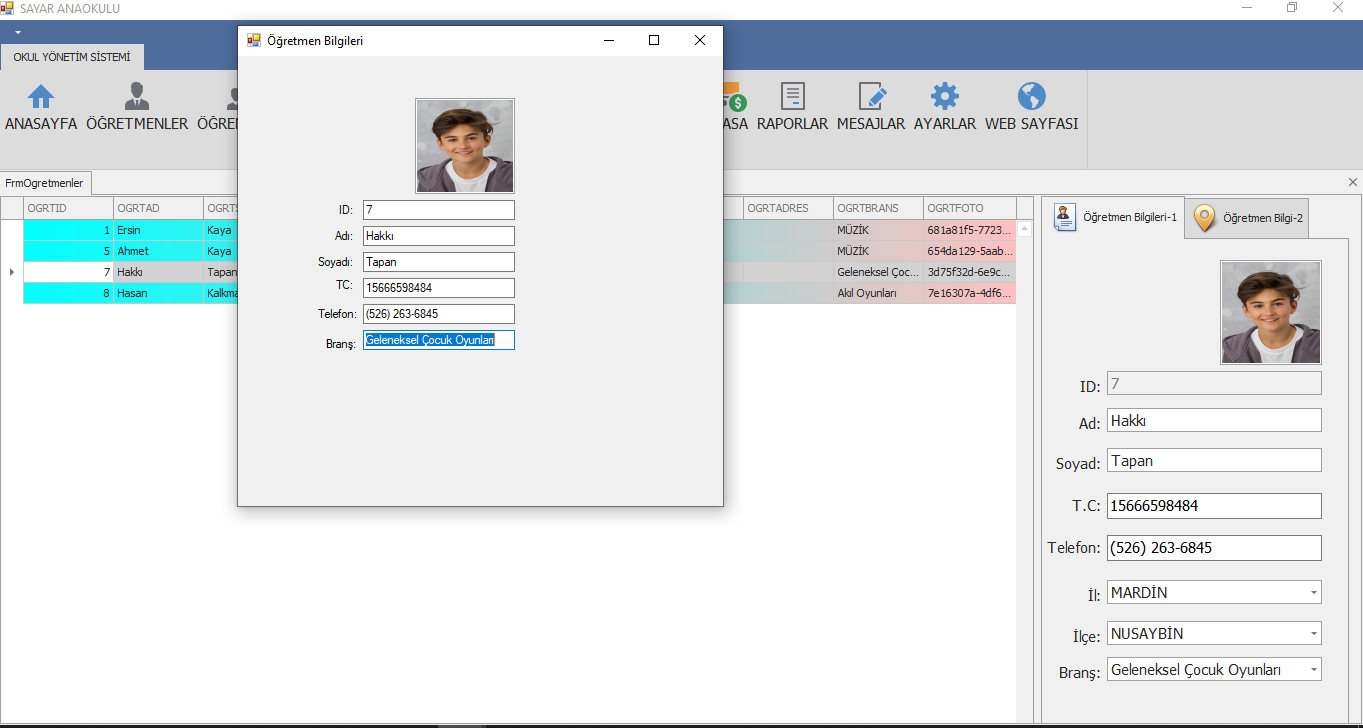
Şekil 18: Öğretmen Bilgi-Kayıt Sekmesi

Kayıt için yan kısımda ki bilgileri doldurarak yeni bir öğretmen kayıt etmiş oluyoruz. Öğretmen bilgileri girilmeyi müteakip Kaydet butonuna basılarak yeni kayıt eklenmektedir. Eklelenen kayıtlar TBL\_OGRETMENLER tablosuna kaydedilir. Öğretmen kayıt edilirken İl, İlçe ve Branş seçimi İl (TBL\_ILLER), İlçe (TBL\_ILCELER) ve Branş (TBL\_BRANSLAR) tablolarından ComboBox ile getirilen kayıtlardan seçilerek yapılmaktadır. Ayriyetten resim seçmek için masaüstünden seçilen resim Picture Edit ile kayıtlara eklenir.



Şekil 19: Öğretmen Kaydet

Burada kayıt olan öğrencinin Gridconrolde çift tıklanarak Öğrenci Bilgileri formunu acar ve öğrencinin bilgilerini aktarır.

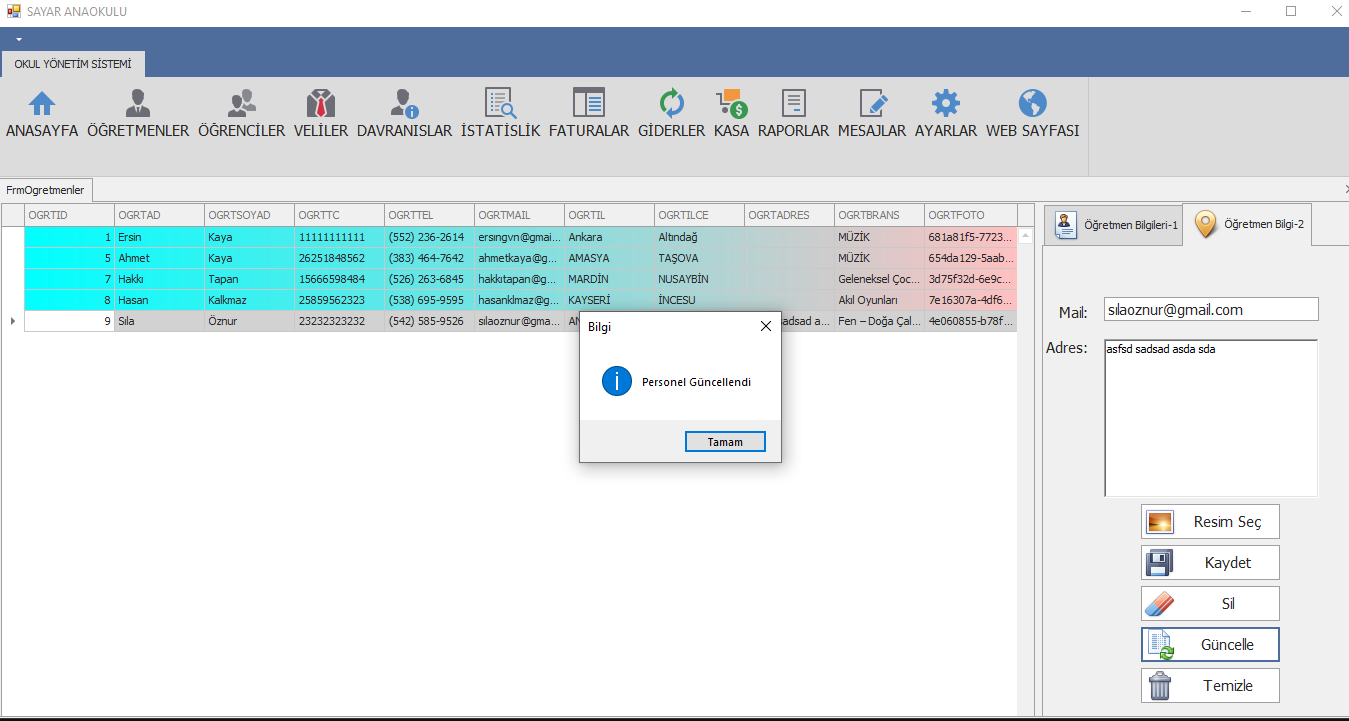


Şekil 20: Öğretmen Bilgileri

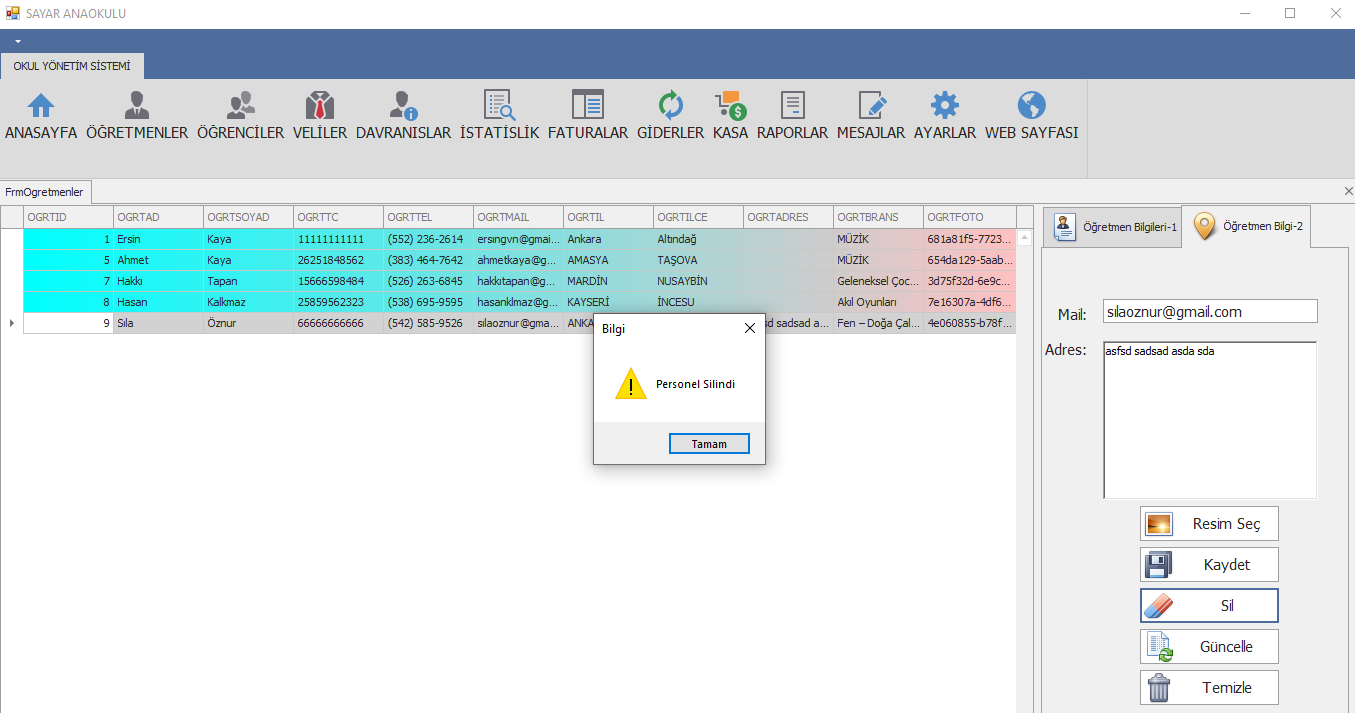
Öğretmen İd kısmına tıkladığımız zaman öğrencinin bilgileri yan kısımda otomatik olarak doldurur. Bu bilgiler getirildikten sonra bu öğretmene ait;

* **Sil**,
* **Güncelle** işlemleri yapılabilmektedir.

Öğretmene ait kayıt silinmek isteniyorsa **Sil** butonuna basılarak öğrenci kayıtları silinebilmektedir. Öğretmene ait bilgiler güncellenmek isteniyorsa güncellenmek istenen bilgiler değiştirilmeyi müteakip **Güncelle** butonuna tıklayarak bilgiler güncellenir.



Şekil 21: Öğretmen Güncelle



Şekil 22: Öğretmen Sil

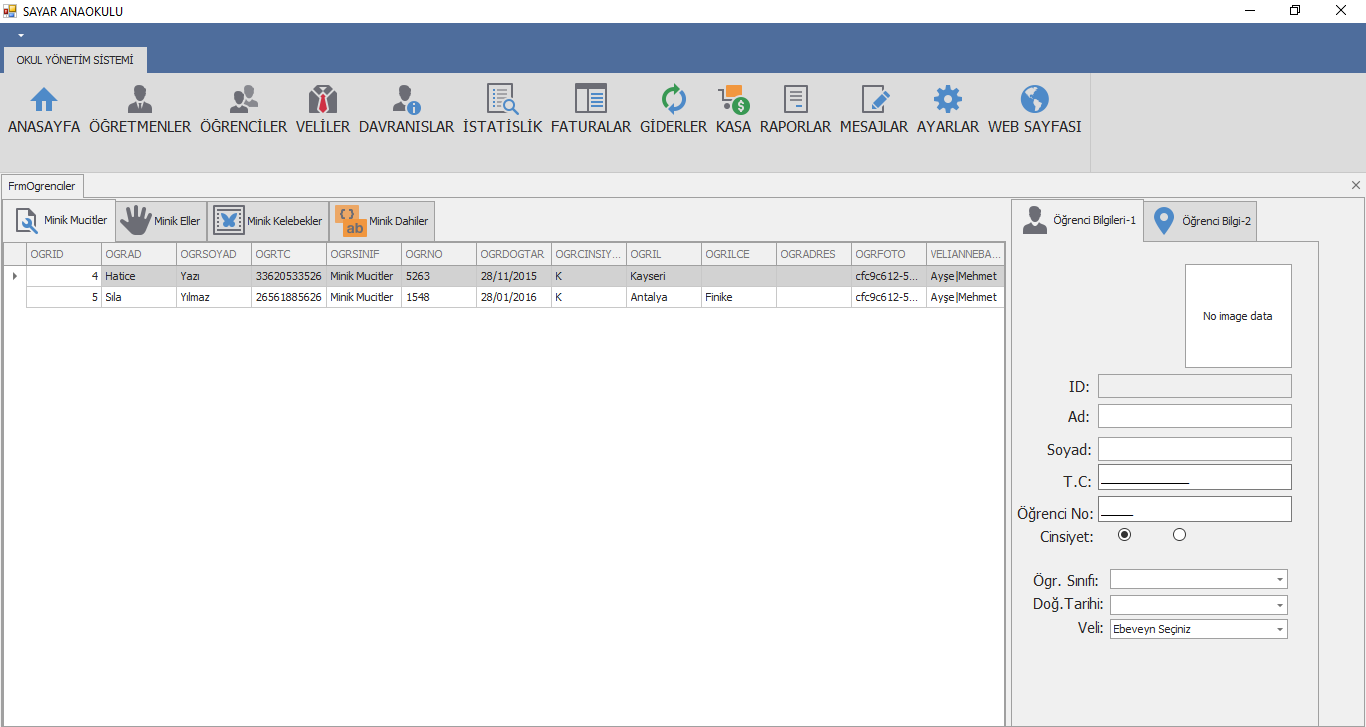
## **2.6.2.2. Öğrenci Bilgi İşlemleri Sekmesi:**

Öğrenci Bilgi İşlemleri Sekmesinde;

* Resim Sec,
* Kaydet,
* Güncelle,
* Sil,
* Temizle.

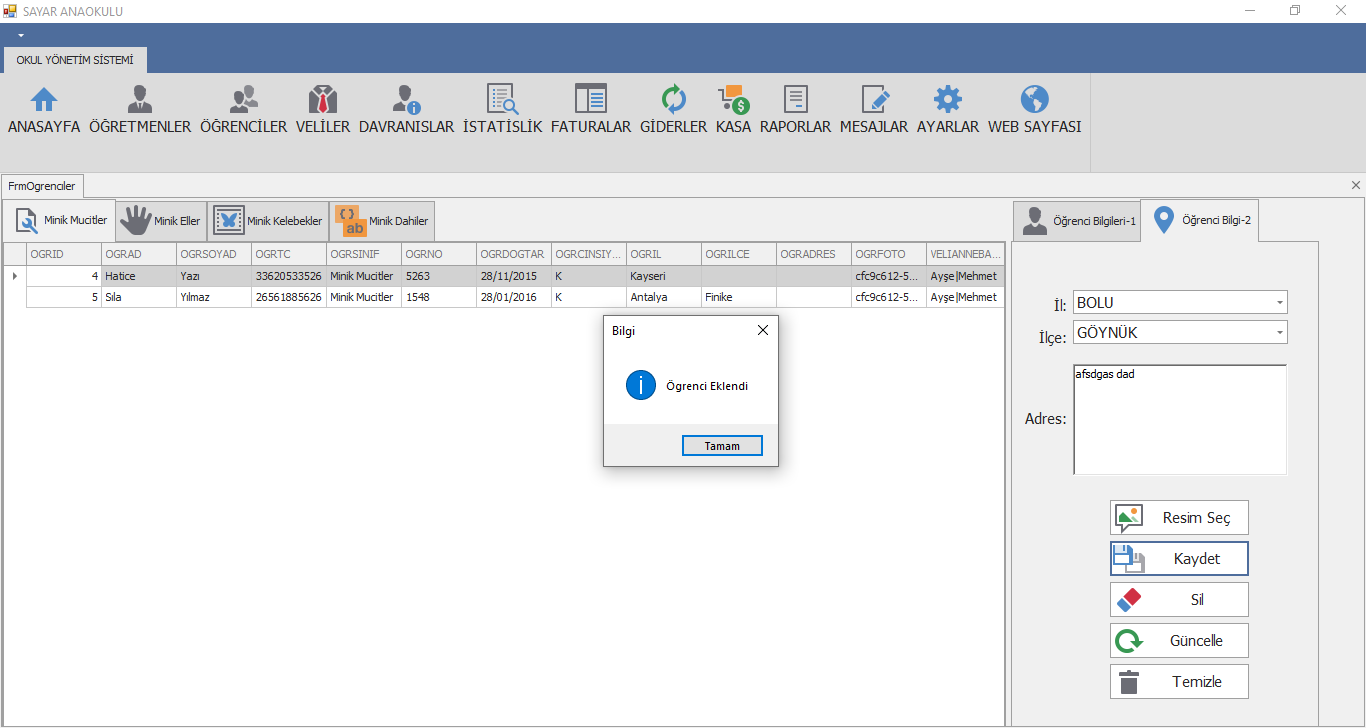
Bu işlemlere ait veriler TBL\_OGRENCILER tablosundan getirilmekte. Grid Control ile gösterilmektedir. Bu bilgiler;

* Öğrenci İd,
* Adı,
* Soyadı,
* TC,
* Öğrenci No,
* Cinsiyet,
* Veli Ad-Soyad,
* İl,
* İlçe,
* Sınıf,
* Doğum Tarihi,
* Adres,
* Resim.



Şekil 23: Öğrenci Bilgi-Kayıt Sekmesi

Kayıt için yan kısımda ki bilgileri doldurarak yeni bir öğrenci kayıt etmiş oluyoruz. Öğrenci bilgileri girilmeyi müteakip Kaydet butonuna basılarak yeni kayıt eklenmektedir. Yeni kaydedilen kayıtlar TBL\_OGRENCILER tablosuna kaydedilir. Öğrenci kayıt edilirken Veli, İl, İlçe ve Sınıf seçimi İl (TBL\_ILLER), İlçe (TBL\_ILCELER),Veli (TBL\_VELİLER) ve Sınıf (MınıkDahiler, MınıkKelebekler, MınıkEller, MınıkMucıtler) tablolarından ComboBox’a Velileri ise LookUpEdit aracına getirilen kayıtlardan seçilerek yapılmaktadır. Ayriyetten resim seçmek için masaüstünden seçilen resim Picture Edit ile kayıtlara eklenir.

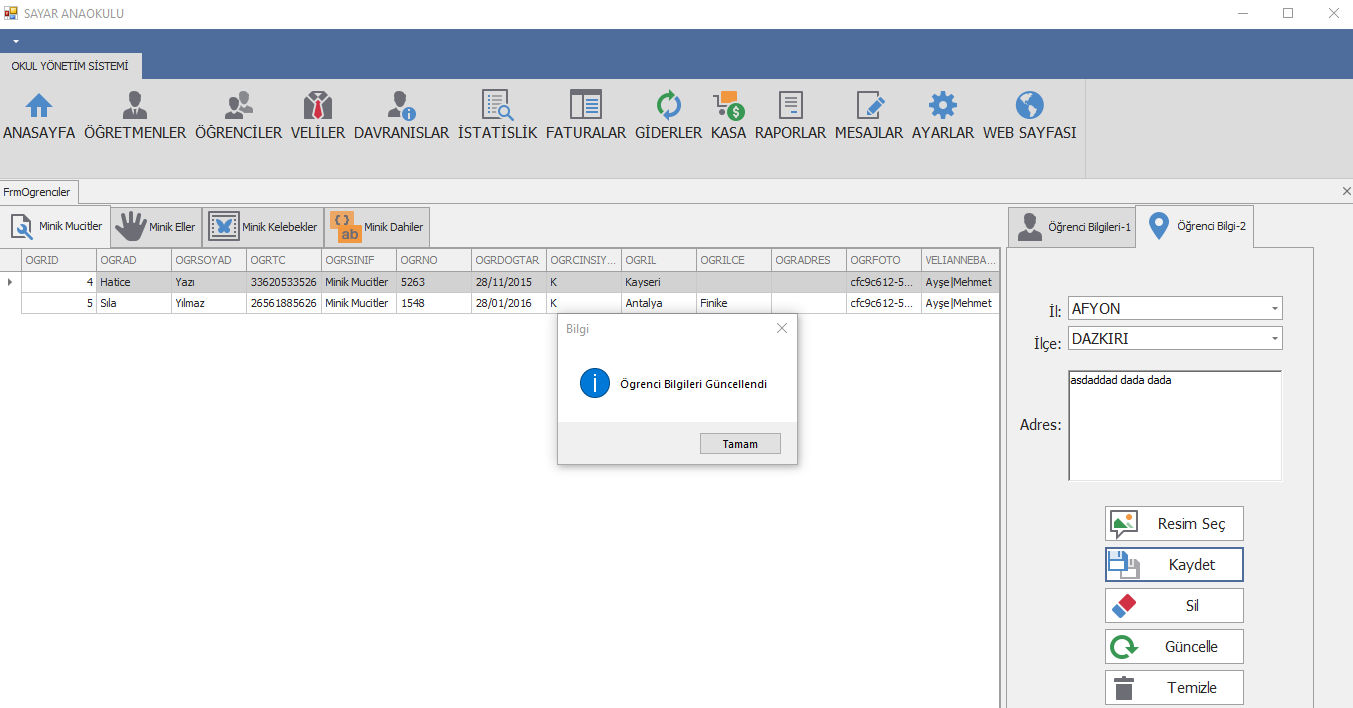


Şekil 24: Öğrenci Kayıt

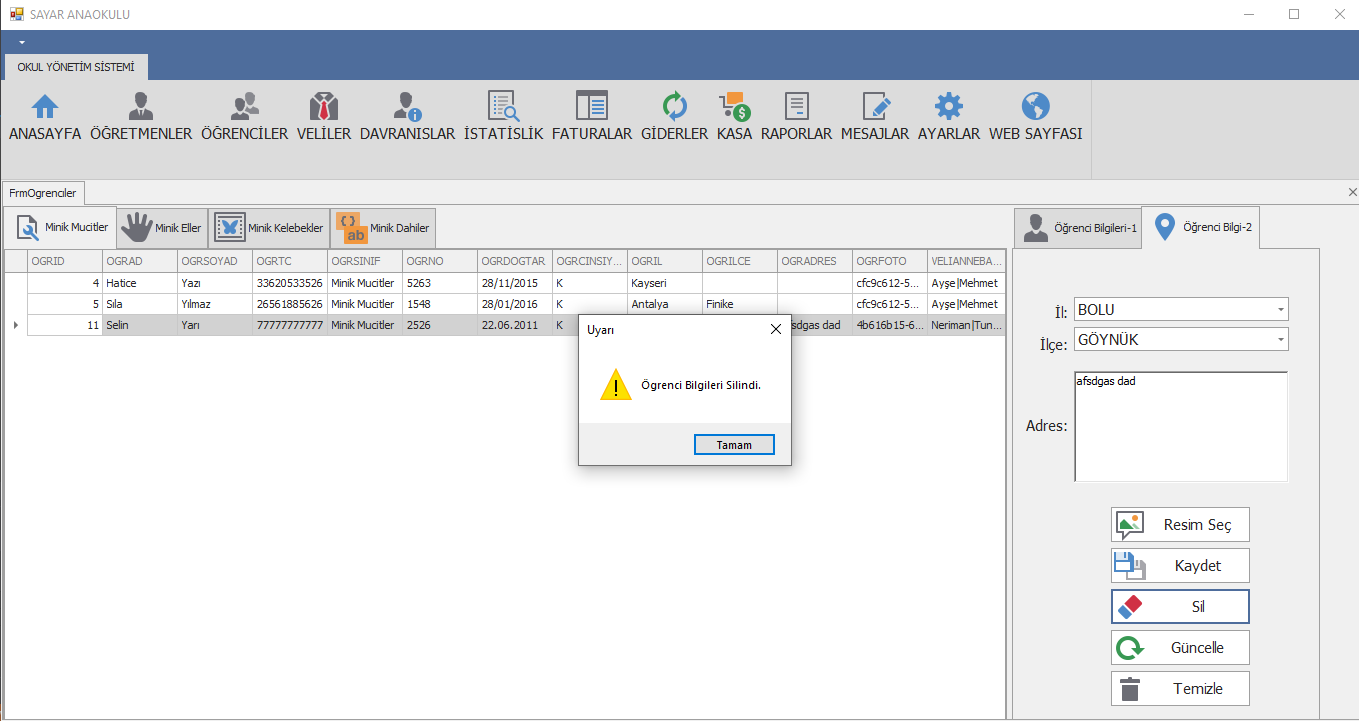
Öğrenci İd kısmına tıkladığımız zaman öğrencinin bilgileri yan kısımda otomatik olarak doldurur. Bu bilgiler getirildikten sonra bu öğrenciye ait;

* **Sil**,
* **Güncelle** işlemleri yapılabilmektedir.

Öğrenciye ait kayıt silinmek isteniyorsa **Sil** butonuna basılarak öğrenci kayıtları silinebilmektedir. Öğrenciye ait bilgiler güncellenmek isteniyorsa güncellenmek istenen bilgiler değiştirilmeyi müteakip **Güncelle** butonuna tıklayarak bilgiler güncellenir.

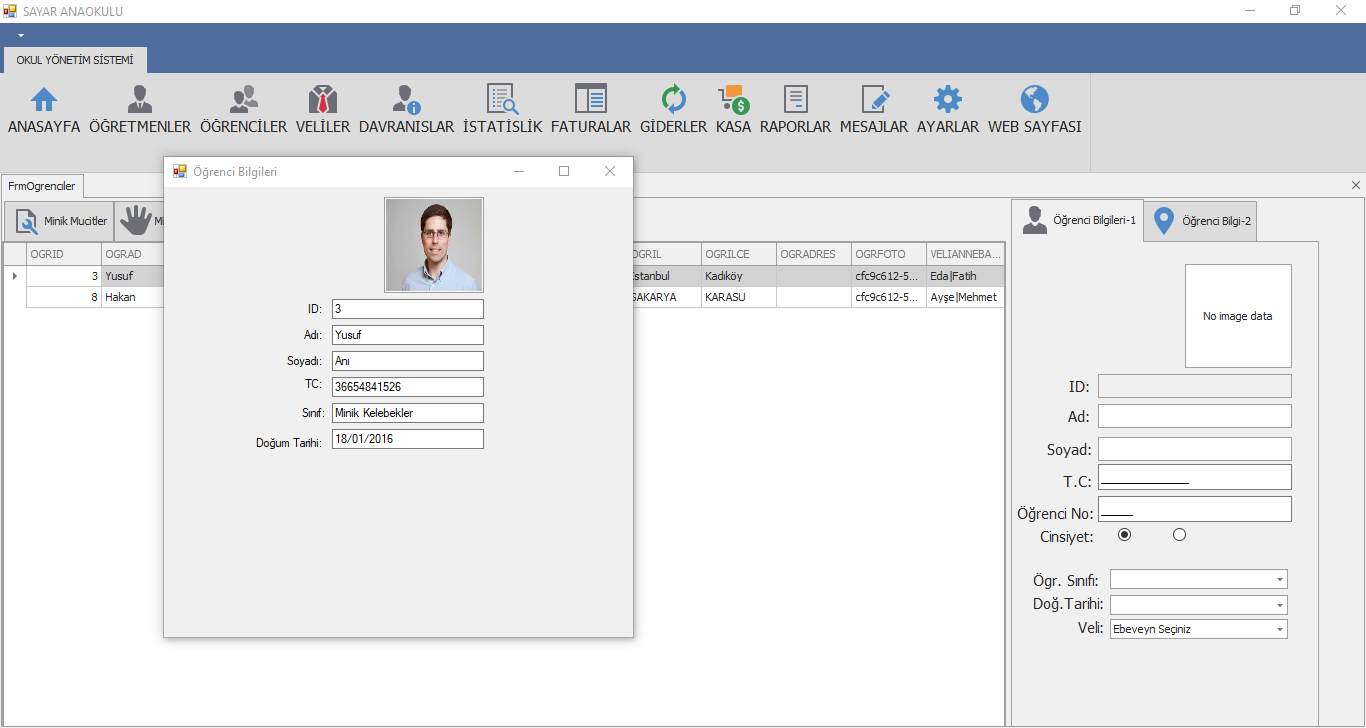


Şekil 25: Öğrenci Güncelle



Şekil 26: Öğrenci Sil

Burada kayıt olan öğrencinin Gridconrolde çift tıklanarak Öğrenci Bilgileri formunu acar ve öğrencinin bilgilerini aktarır.



Şekil 27: Öğrenci Bilgileri

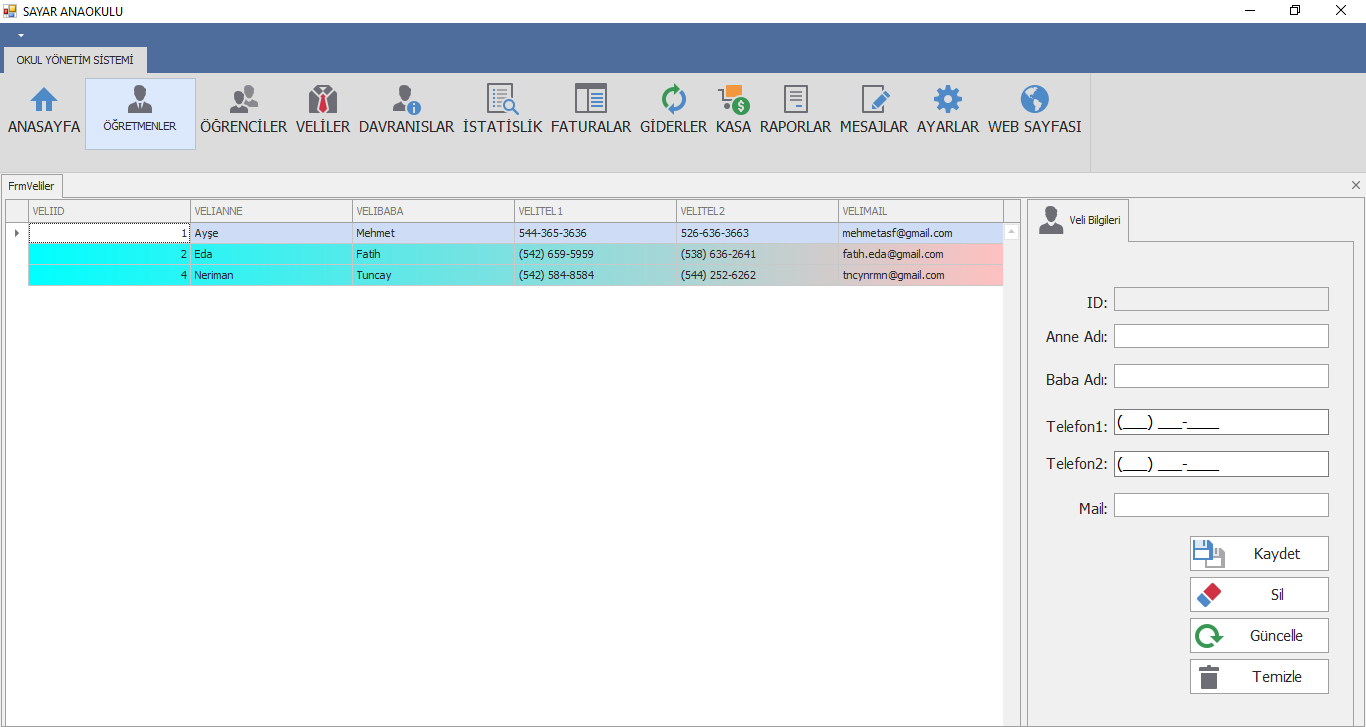
## **2.6.2.3. Veli Bilgi İşlemleri Sekmesi:**

Veli Bilgi İşlemleri Sekmesinde;

* Kaydet,
* Güncelle,
* Sil,
* Temizle.

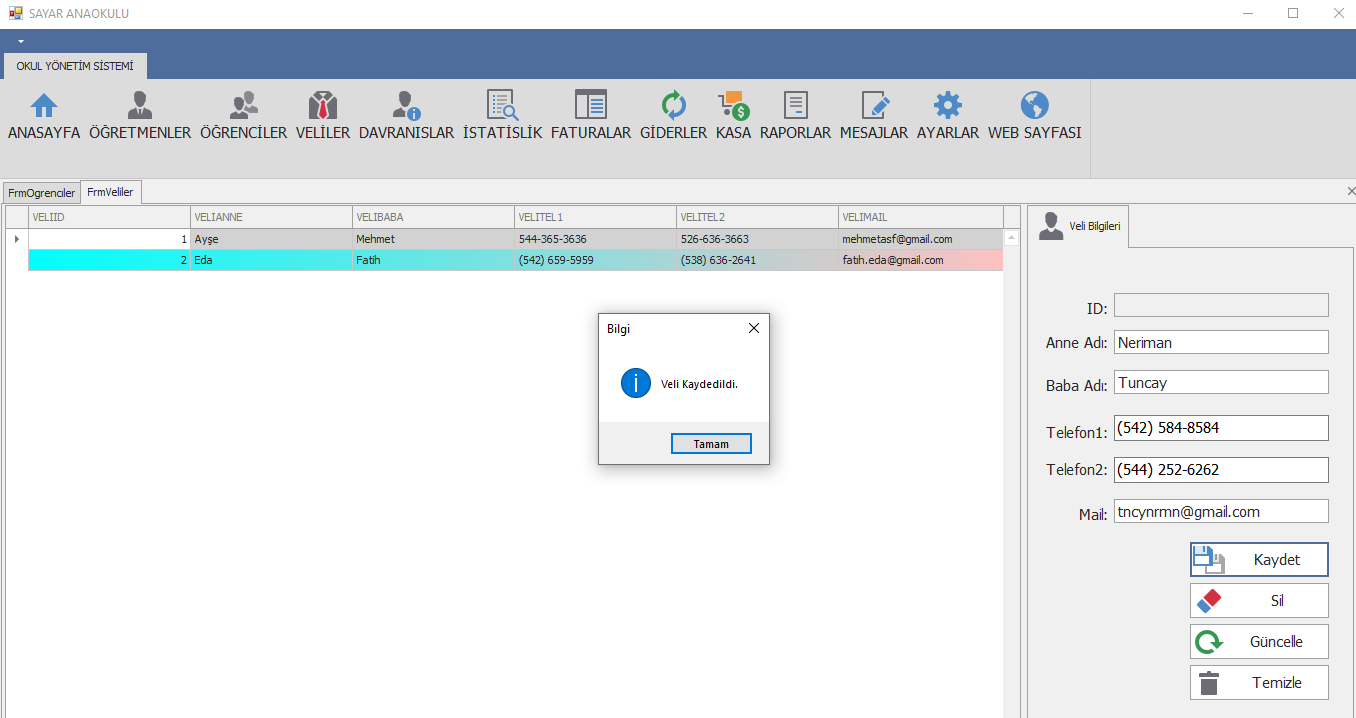
Bu işlemlere ait veriler TBL\_VELILER tablosundan getirilmekte. Grid Control ile gösterilmektedir. Bu bilgiler;

* Veli İd,
* Anne Adı,
* Baba Adı,
* Telefon 1,
* Telefon 2,
* Maıl.



Şekil 28: Veli Bilgi İşlem Sekmesi

Kayıt için yan kısımda ki bilgileri doldurarak yeni bir veli kayıt etmiş oluyoruz. Yeni kayıt eklediğimizde kayıtlar TBL\_VELILER tablosuna kaydedilir. Veli bilgileri girilmeyi müteakip Kaydet butonuna basılarak yeni kayıt eklenmektedir.

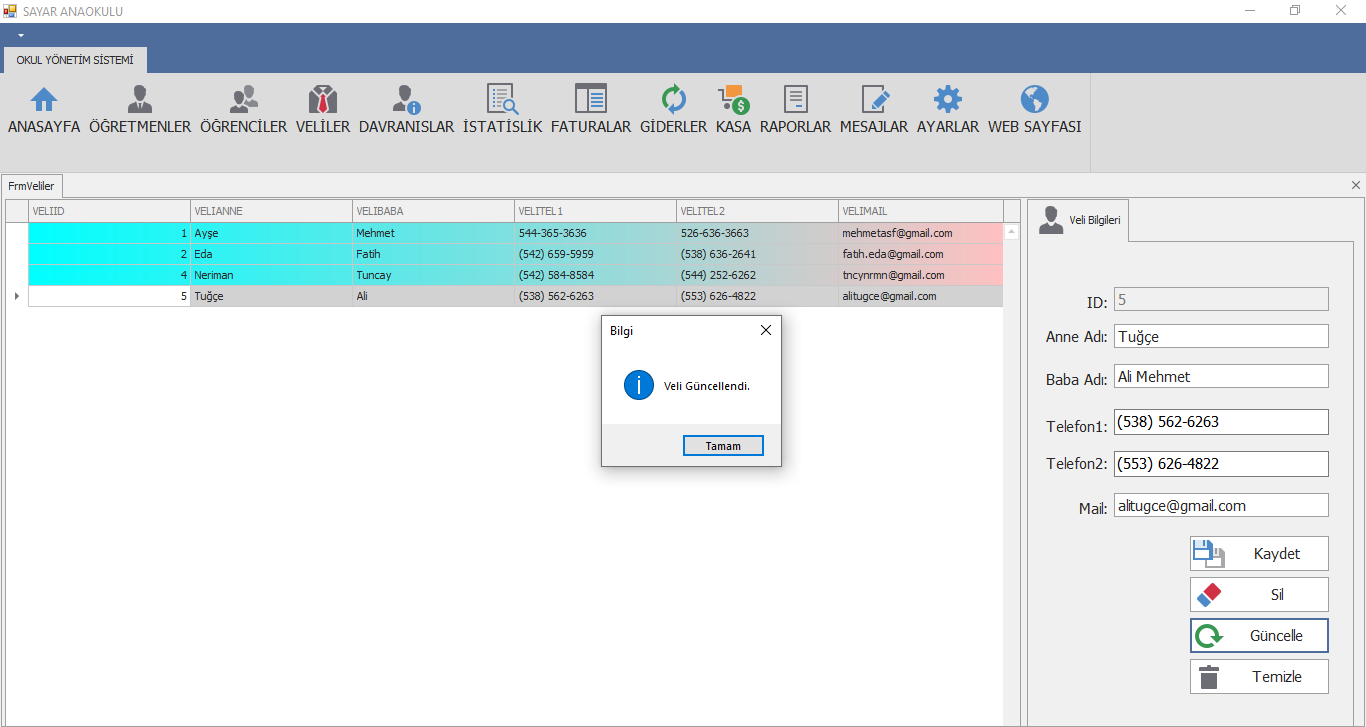


Şekil 29: Veli Kayıt

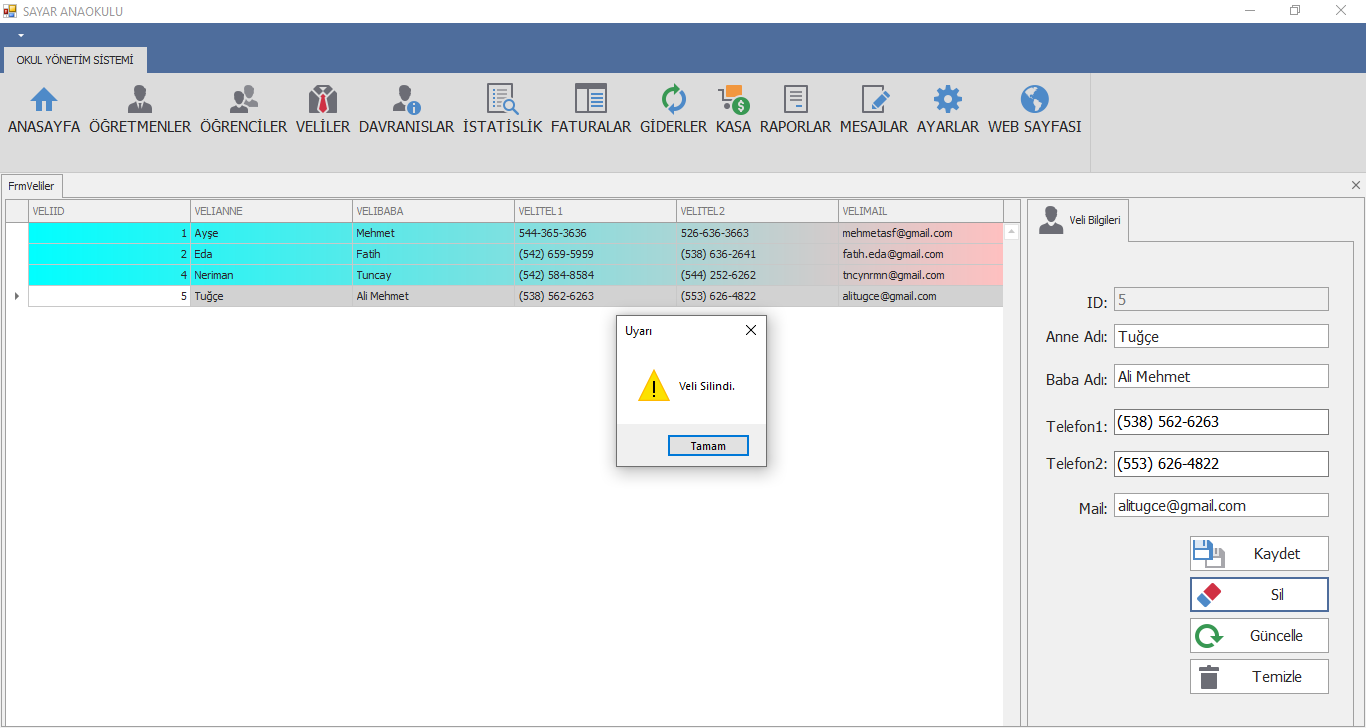
Veli İd kısmına tıkladığımız zaman velinin bilgileri yan kısımda otomatik olarak doldurur. Bu bilgiler getirildikten sonra bu veliye ait;

* **Sil**,
* **Güncelle** işlemleri yapılabilmektedir.

Veli ait kayıt silinmek isteniyorsa **Sil** butonuna basılarak öğrenci kayıtları silinebilmektedir. Veliye ait bilgiler güncellenmek isteniyorsa güncellenmek istenen bilgiler değiştirilmeyi müteakip **Güncelle** butonuna tıklayarak bilgiler güncellenir.

****

Şekil 30: Veli Güncelleme

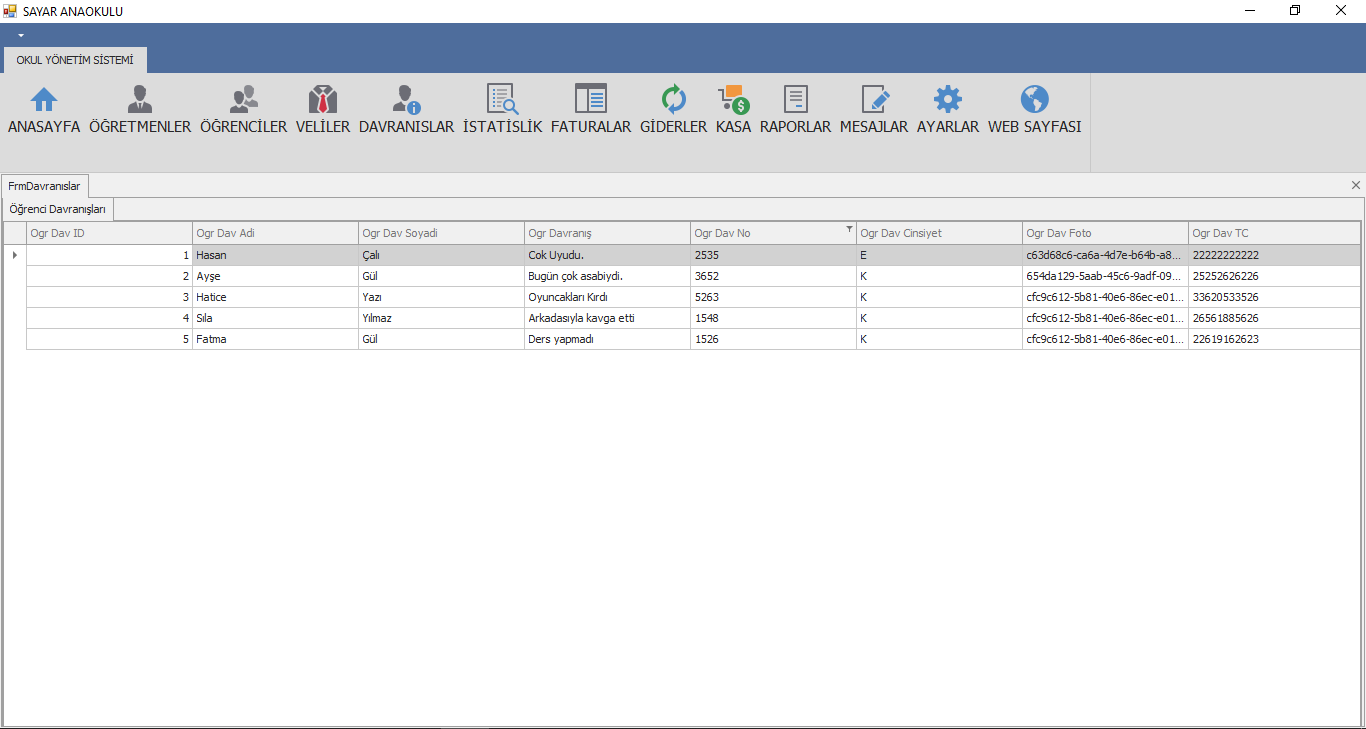
****

Şekil 31: Veli Sil

## **2.6.2.4. Davranış Bilgi Sekmesi:**

Bu işlemlere ait veriler TBL\_DAVRANISLAR tablosundan getirilmekte. Grid Control ile gösterilmektedir. Bu bölümde yönetici sadece öğretmen tarafından girilen bilgileri erişim sağlar. Bu bilgiler;

* Öğrenci Davranış İd,
* Öğrenci Adı,
* Öğrenci Soyadı,
* Öğrenci Davranış,
* Öğrenci Numarası,
* Öğrenci Cinsiyet,
* Öğrenci Fotoğraf,
* Öğrenci TC.

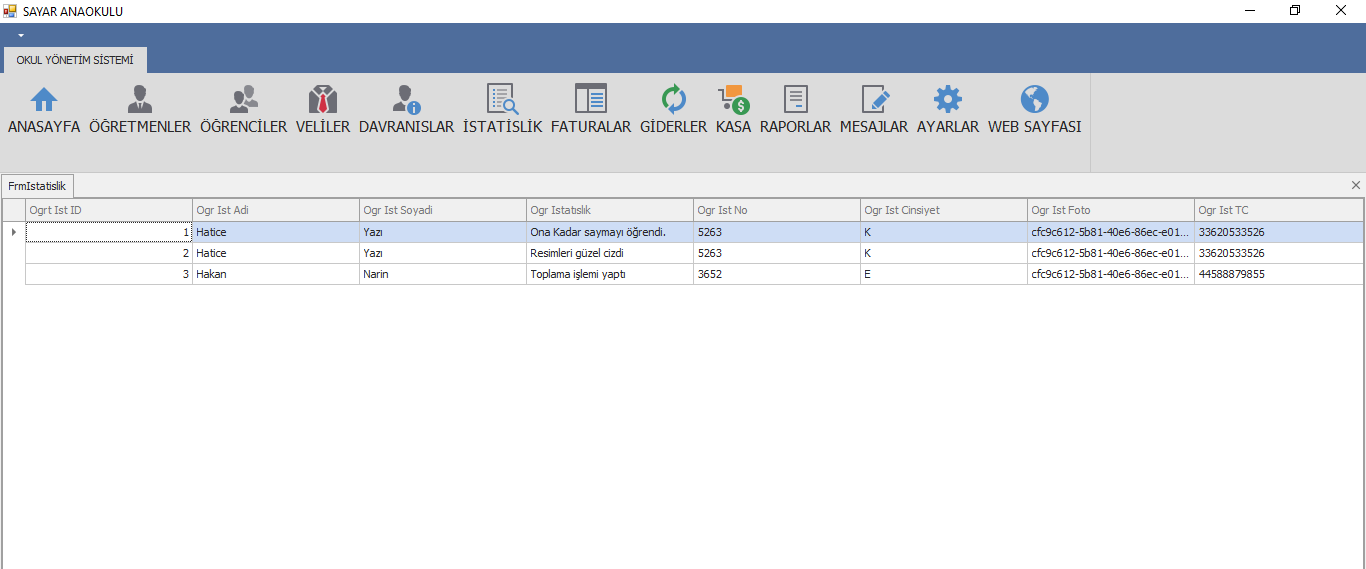


Şekil 32: Yönetici Davranış Sekmesi

## **2.6.2.5. İstatislik Bilgi Sekmesi:**

Bu işlemlere ait veriler TBL\_DAVRANISLAR tablosundan getirilmekte. Grid Control ile gösterilmektedir. Bu bölümde yönetici sadece öğretmen tarafından girilen bilgileri erişim sağlar. Bu bilgiler;

* Öğrenci İstatislik İd,
* Öğrenci Adı,
* Öğrenci Soyadı,
* Öğrenci Davranış,
* Öğrenci Numarası,
* Öğrenci Cinsiyet,
* Öğrenci Fotoğraf,
* Öğrenci TC.

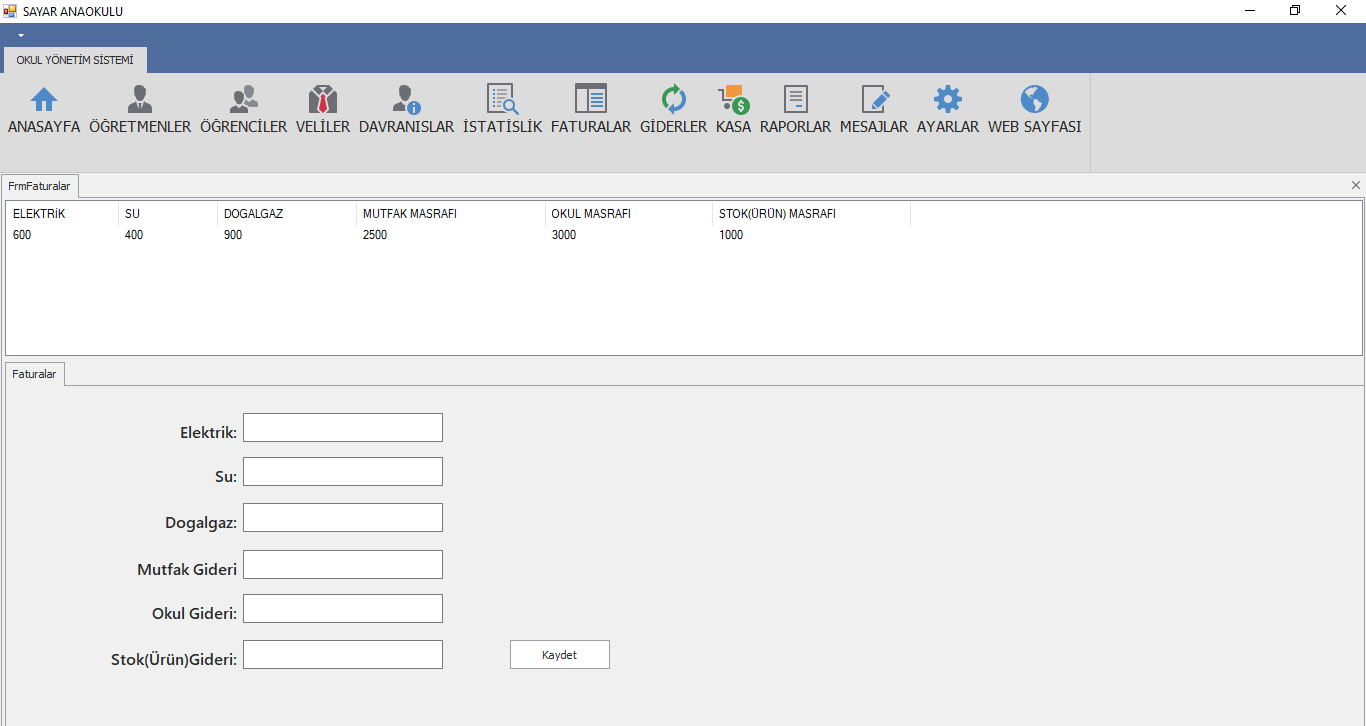


Şekil 33: Yönetici İstatislik Sekmesi

## **2.6.2.6. Faturalar Bilgi İşlem Sekmesi:**

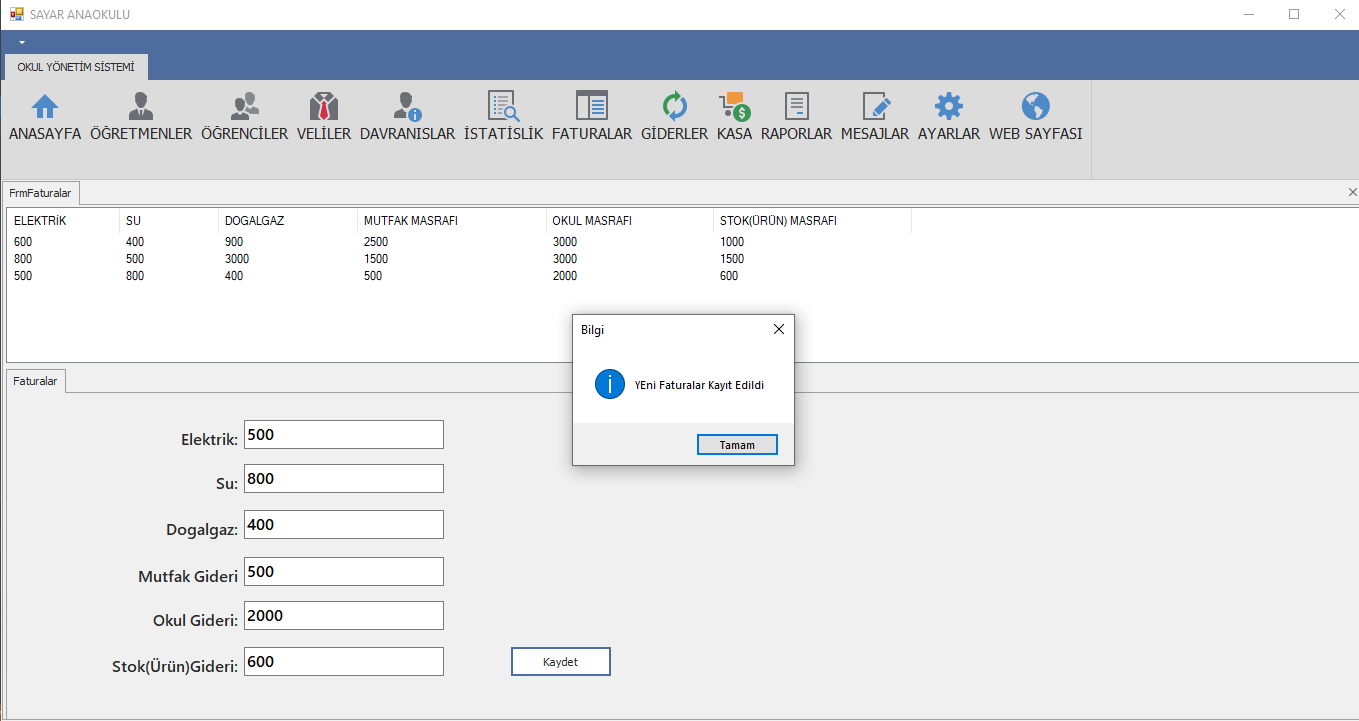
Bu işlemlere ait veriler TBL\_FATURALAR tablosundan getirilmekte. Listview ile gösterilmektedir. Bu bilgiler;

* Elektrik,
* Su,
* Dogalgaz,
* Mutfak Giderleri,
* Okul Giderleri,
* Stok Giderleri.



Şekil 34: Yönetici Fatura Sekmesi

Kayıt için yan kısımda ki bilgileri doldurarak yeni bir öğrenci kayıt etmiş oluyoruz. Yeni fatura girdigimiz değerler TBL\_FATURALAR tablosuna kaydedilir. Fatura bilgileri girilmeyi müteakip Kaydet butonuna basılarak yeni kayıt eklenmektedir.

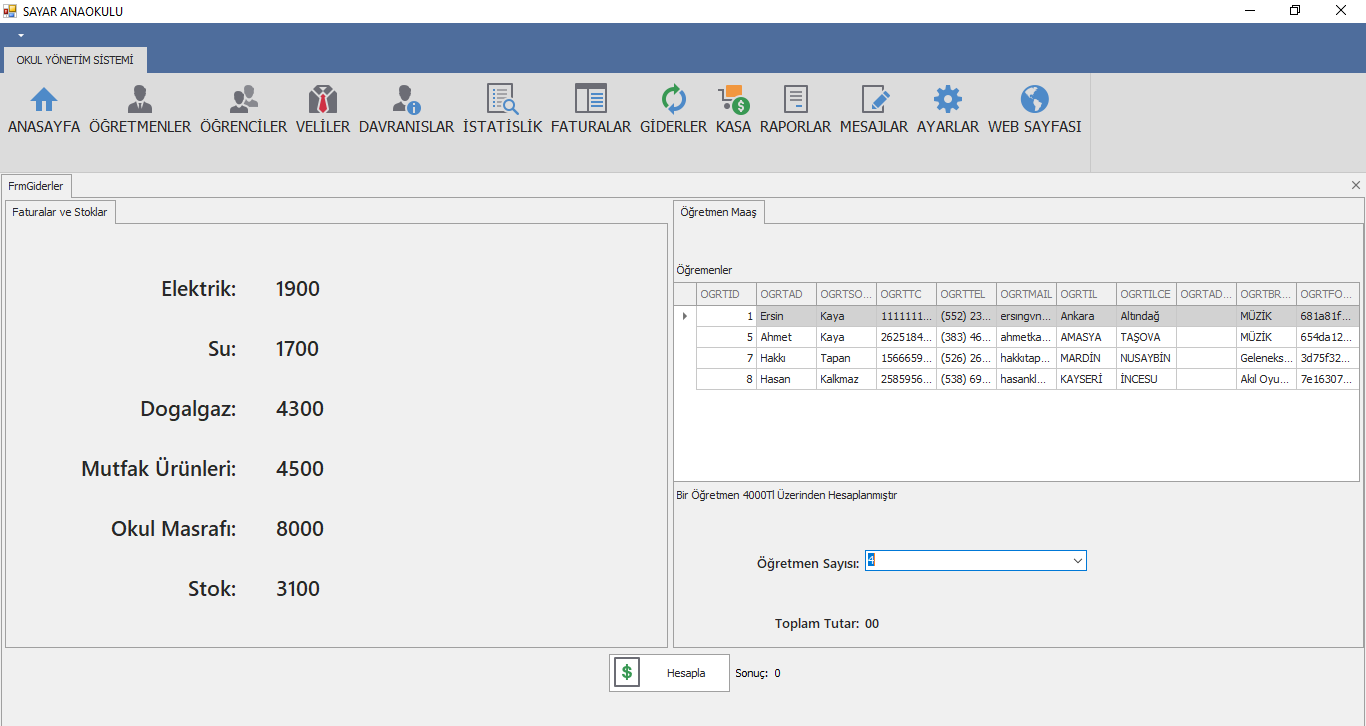
****

Şekil 35: Fatura Kayıt

## **2.6.2.7. Giderler Bilgi İşlem Sekmesi:**

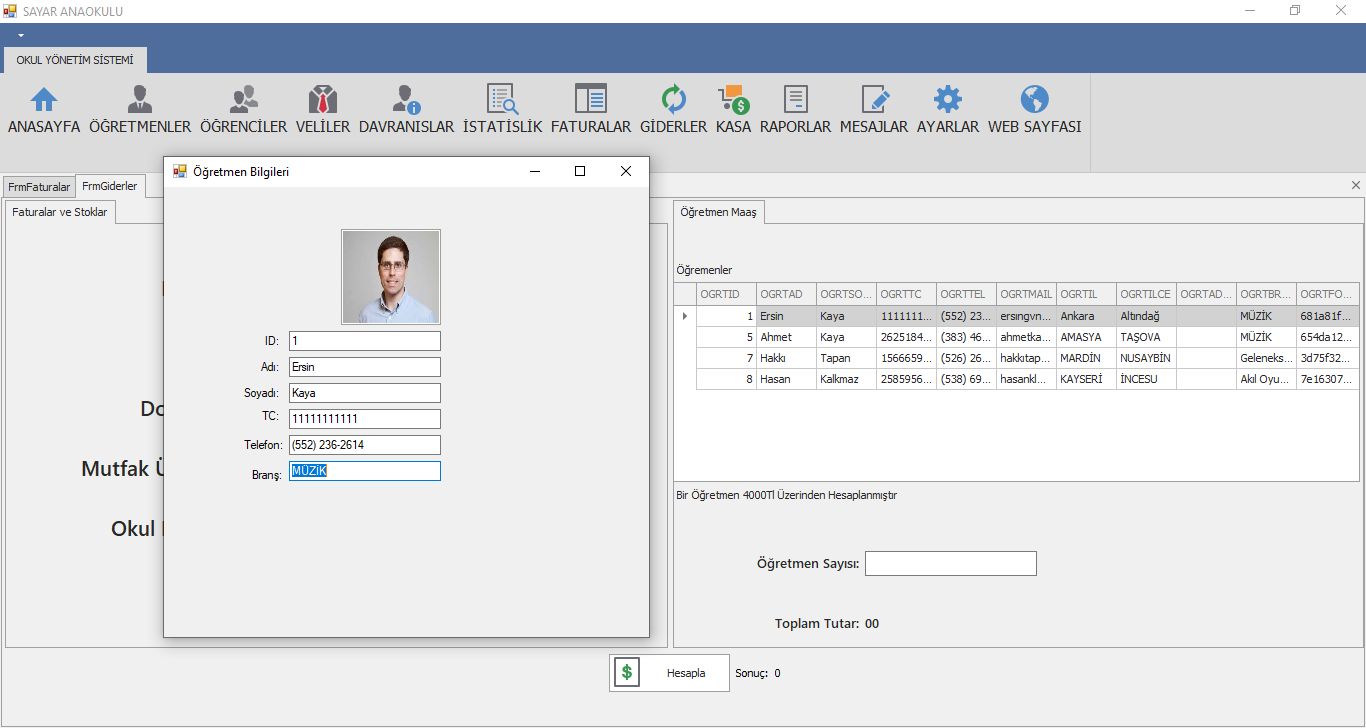
Bu işlemlere ait veriler TBL\_GIDERLER tablosundan ve TBL\_OGRETMENLER tablosundan getirilmekte. Bu bölümde faturalar kısmından faturalar gelir. GridControlde ise öğretmenler listelenerek bir öğretmen maaşı 4000 tl olarak hesaplanıp öğretmen toplam tutarı label da gösterilmiştir. Bu kısımda Faturalar ve Öğretmen maaşları Hesapla Butonuna basılmasıyla toplam gideri hesaplar. Bu bilgiler;

* Elektrik,
* Su,
* Dogalgaz,
* Mutfak Giderleri,
* Okul Giderleri,
* Stok Giderleri.
* Öğretmen Toplam Maaş.



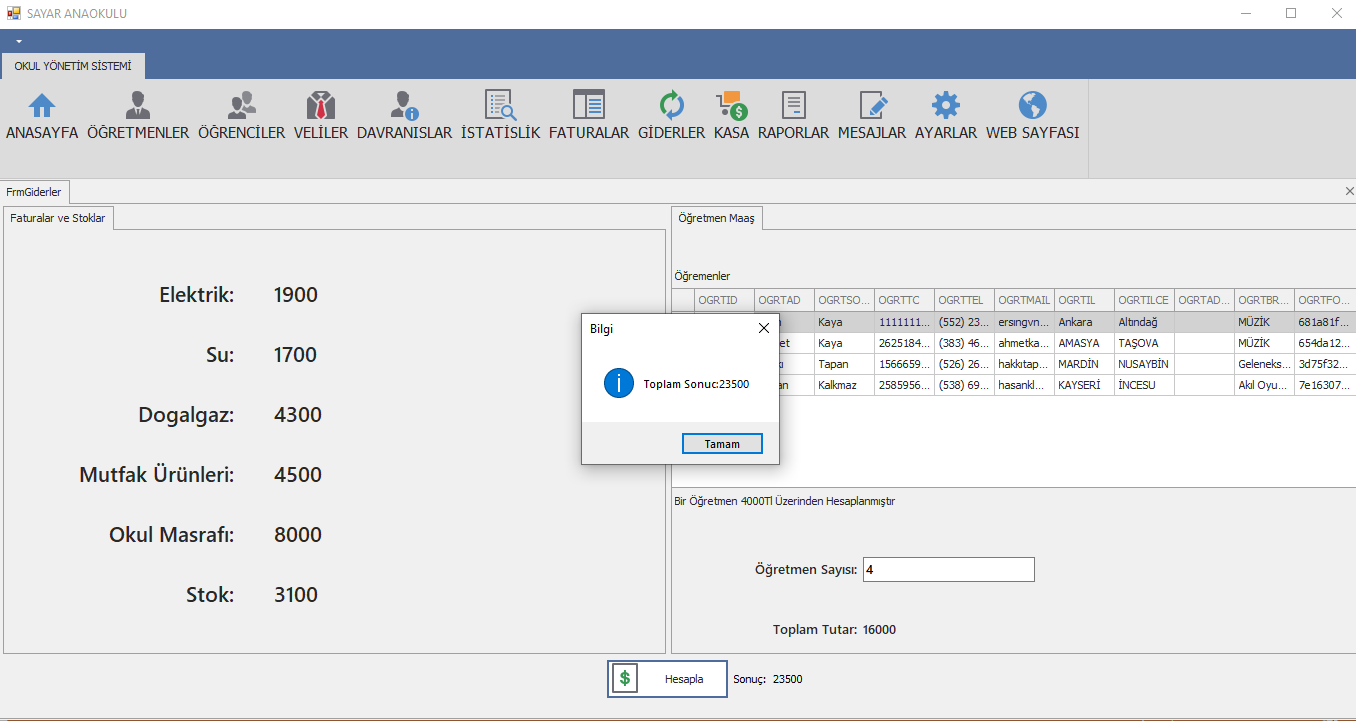
Şekil 36: Gider Bilgi Sekmesi

Burada kayıt olan öğrencinin Gridconrolde çift tıklanarak Öğrenci Bilgileri formunu açar ve öğrencinin bilgilerini aktarır.



Şekil 37: Gider Öğretmen Bilgi Sekmesi

Hesaplamak için butona basıldıgın sonuc mesaj ile ekrana yansıtılır.

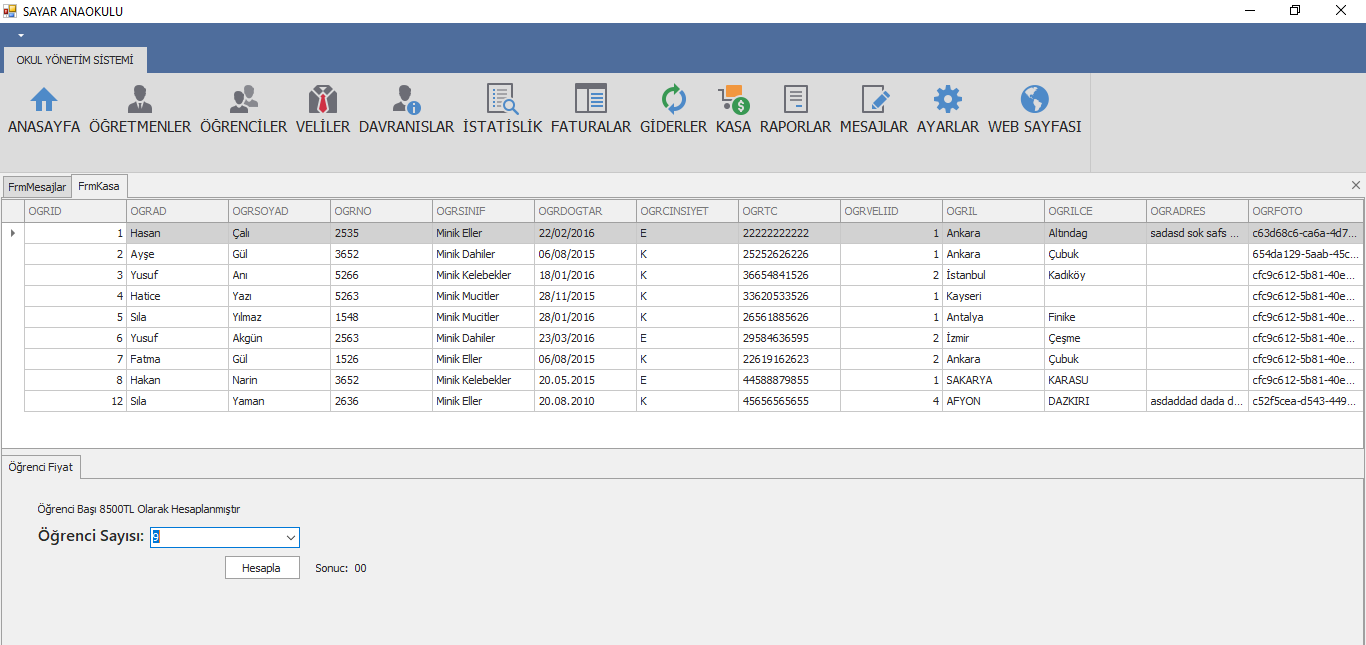
****

Şekil 38: Gider Hesaplama

## **2.6.2.8. Kasa Bilgi İşlem Sekmesi:**

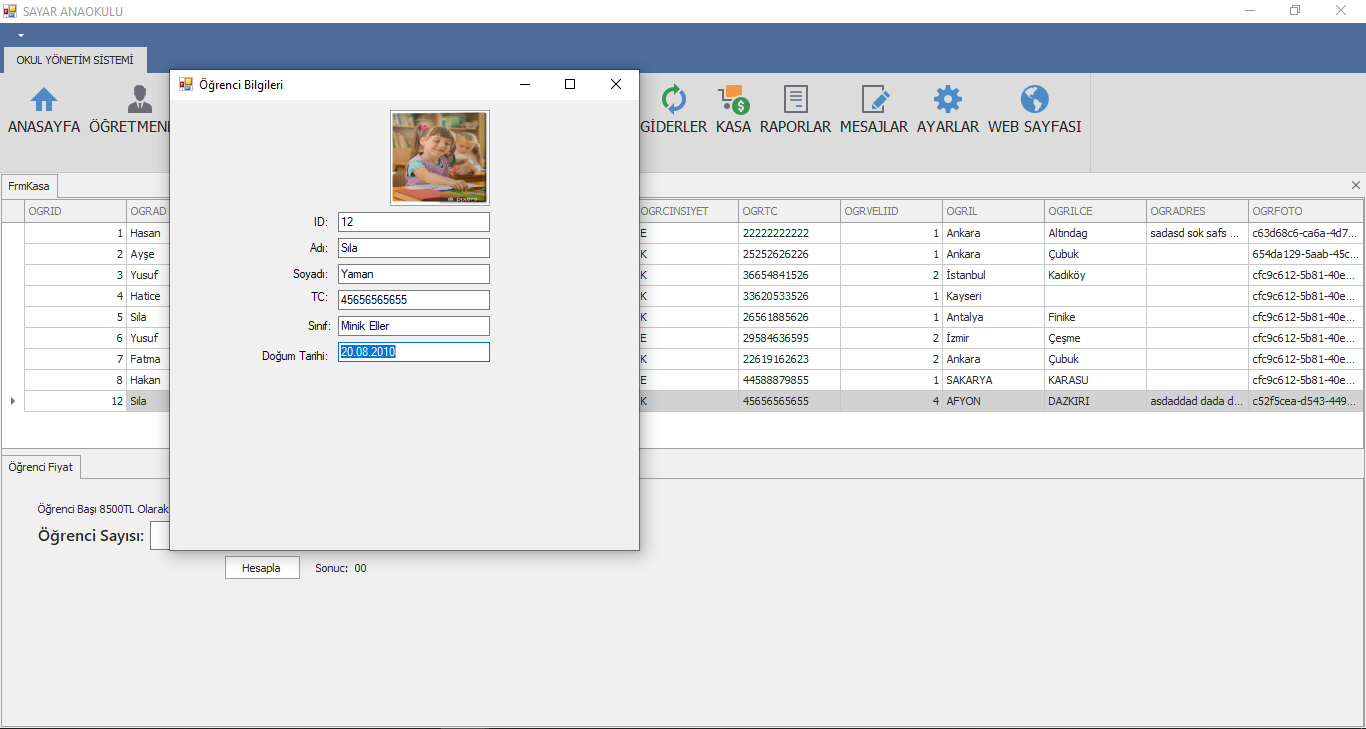
Bu işlemlere ait veriler TBL\_OGRENCILER tablosundan getirilmekte. GridControlde ise öğrenciler listelenerek TextBox’a girilen öğrenci başı sayısı kadar 8500 tl olarak hesaplanıp Öğrenci toplam tutarı label da gösterilmiştir. Bu kısımda öğrenci geliri Hesapla Butonuna basılmasıyla toplam kasayı hesaplar. Bu bilgiler;

* Öğrenci Bilgileri



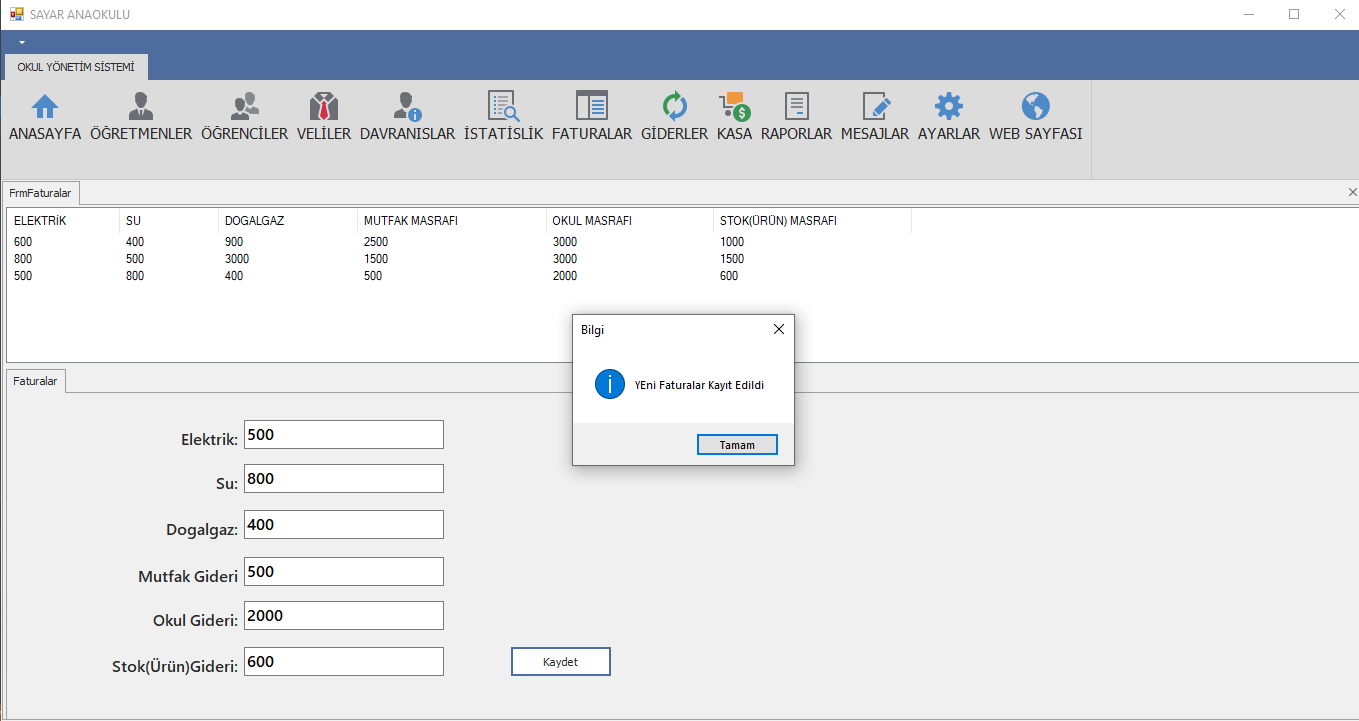
Şekil 39: Kasa Bilgi İşlem Sekmesi

Burada kayıt olan öğrencinin Gridconrolde çift tıklanarak Öğrenci Bilgileri formunu açar ve öğrencinin bilgilerini aktarır.

****

Şekil 40: Kasa Öğrenci Bilgi Sekmesi

Hesaplamak için butona basıldıgın sonuc mesaj ile ekrana yansıtılır.

****

Şekil 41: Kasa Hesaplama

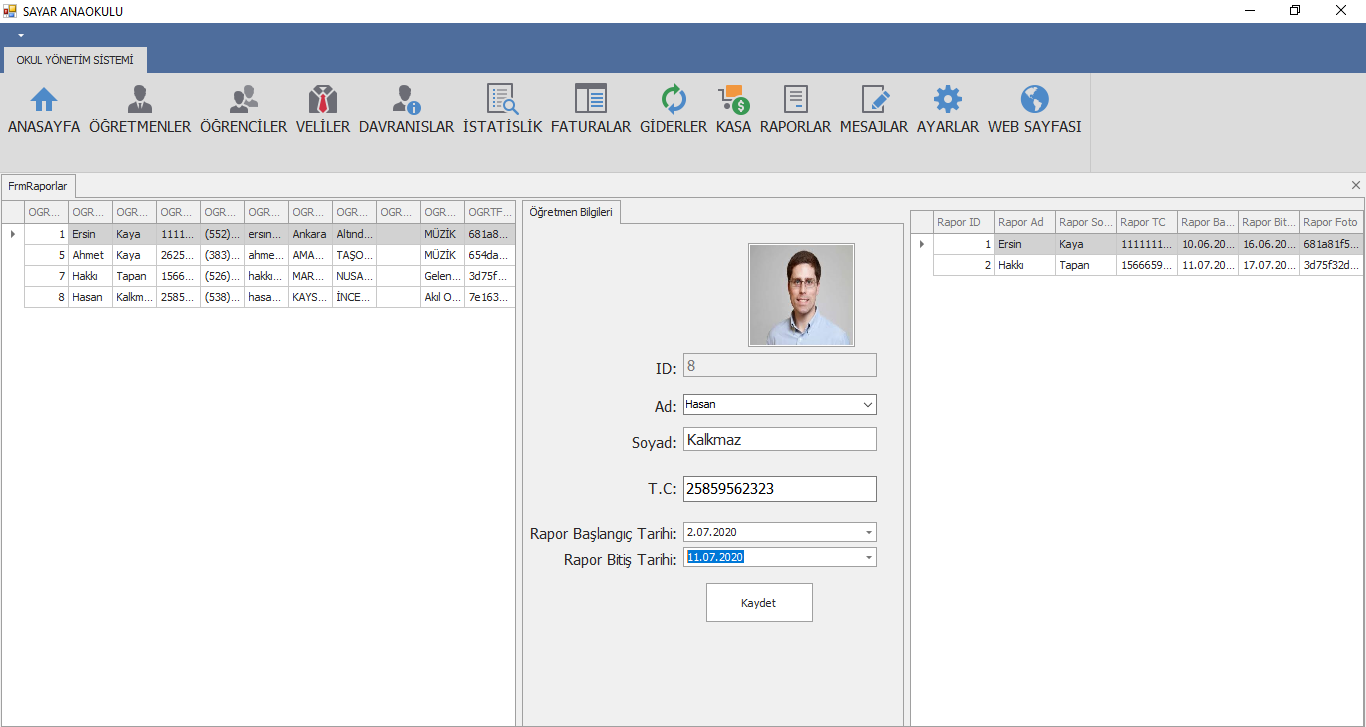
## **2.6.2.9. Rapor Bilgi İşlem Sekmesi:**

Rapor Bilgi İşlemleri Sekmesinde;

* Kaydet,
* Temizle.

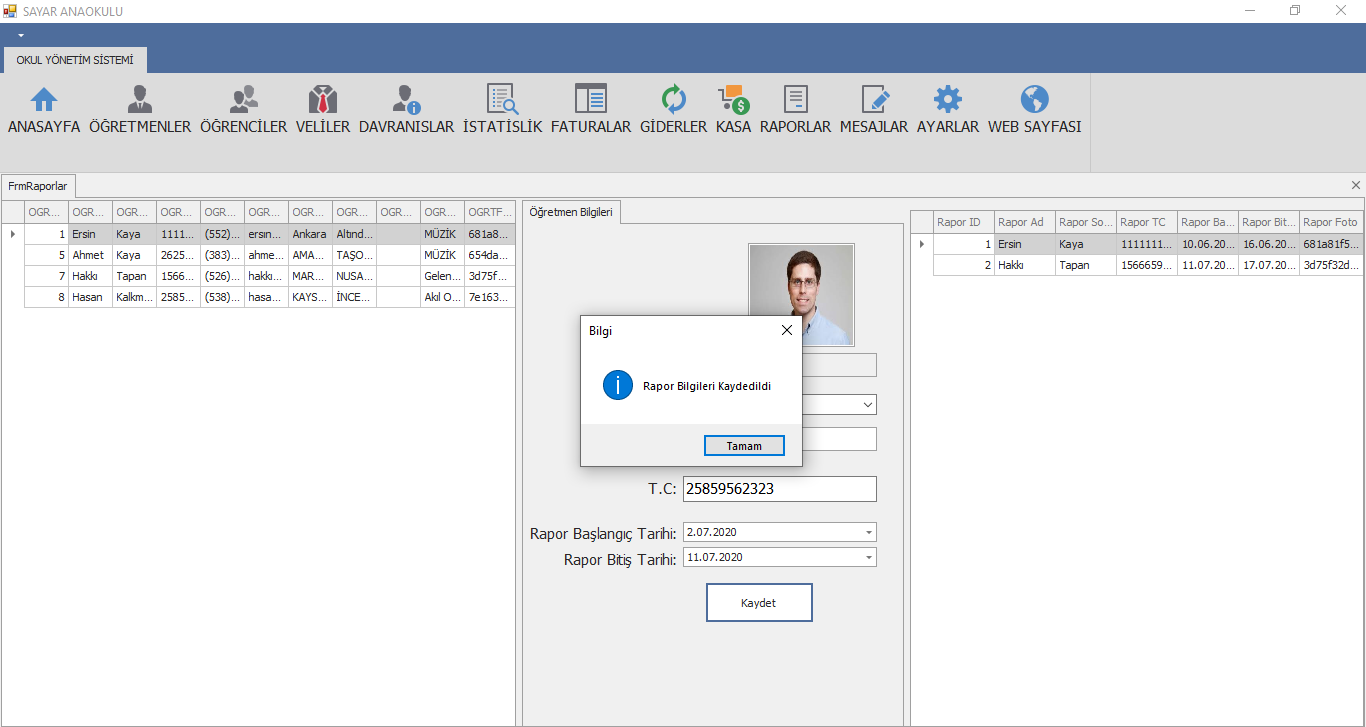
Bu işlemlere ait veriler TBL\_RAPORLAR tablosundan getirilmekte. Gridcontrol ile gösterilmektedir. Listelenen öğretmenlere tıklayarak otomatik olarak bilgiler doldurulur. Bu bilgiler;

* Rapor İd,
* Rapor Adı,
* Rapor Soyadı,
* Rapor Fotoğraf,
* Rapor TC,
* Rapor Başlangıc Tarih,
* Rapor Bitiş Tarih.

****

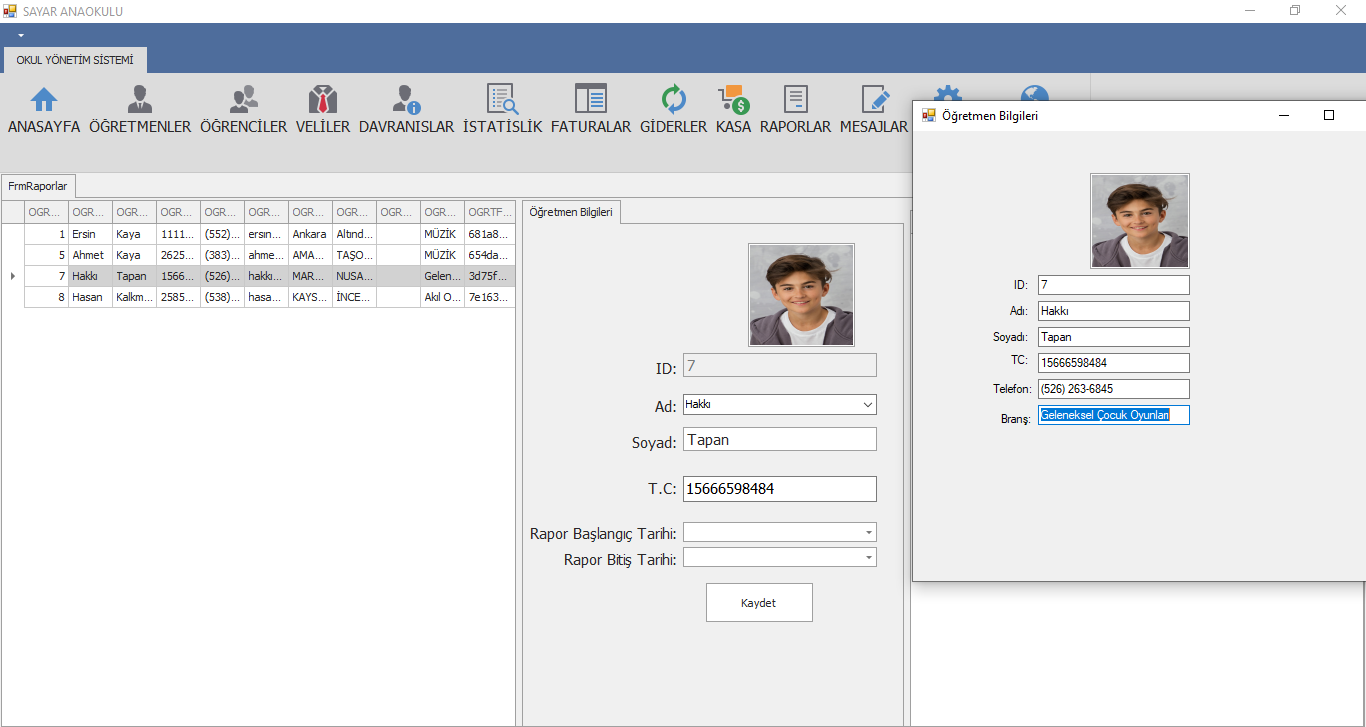
Şekil 42: Rapor Bilgi İşlem Sekmesi

Kayıt için yan kısımda ki bilgileri doldurarak yeni bir rapor tarihi girerek kayıt etmiş oluyoruz. Yeni gelen raporlorı girdikten sonra raporlar TBL\_RAPORLAR tablosuna kaydedilir.Comboboxtan seçilen öğretmen bilgileri otomatik olarak diger yerlere doldurulur. Öğretmen bilgileri girilmeyi müteakip Kaydet butonuna basılarak yeni kayıt eklenmektedir. Kayıt edilen bilgiler yeni bir Grid Controlde listelenmektedir.



Şekil 43: Rapor Kayıt İşlemi

Burada kayıt olan öğretmenlerin Gridconrolde çift tıklanarak Öğretmen Bilgileri formunu açar ve öğrencinin bilgilerini aktarır.



Şekil 44: Rapor Öğretmen Bilgi Sekmesi

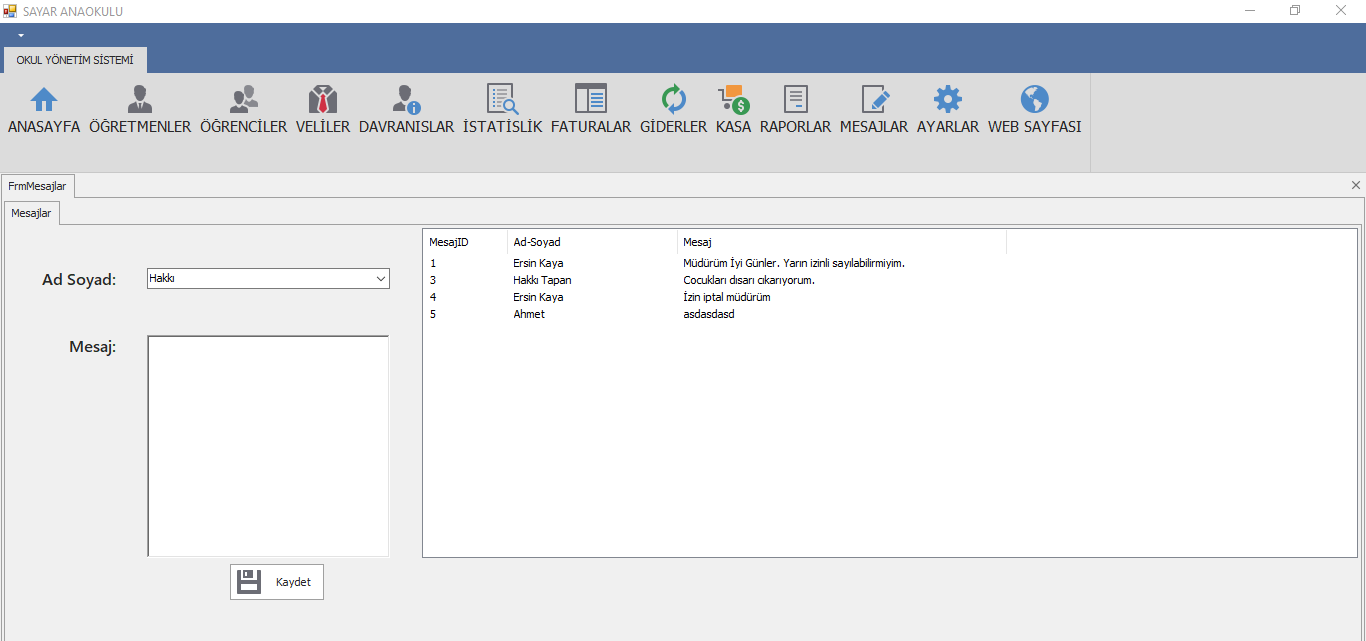
## **2.6.2.10. Mesaj Bilgi İşlem Sekmesi:**

Mesaj Bilgi İşlemleri Sekmesinde;

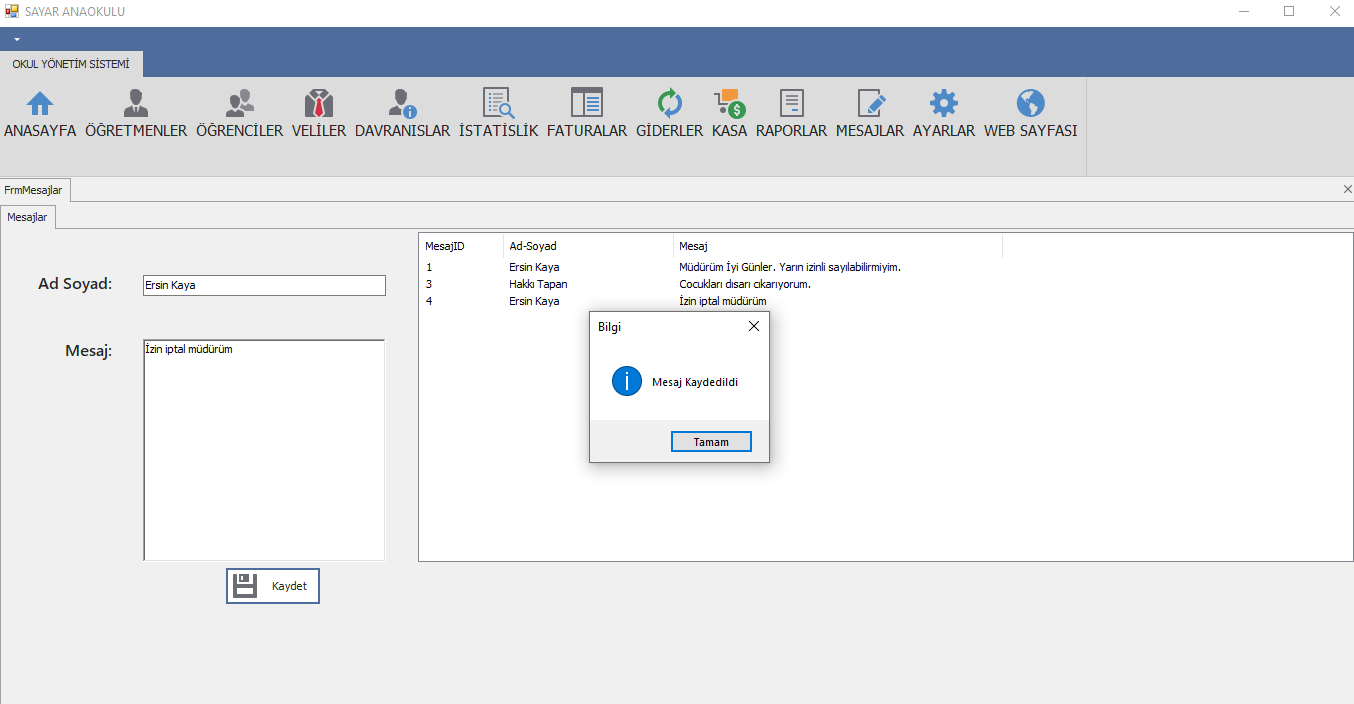
* Kaydet,

Bu işlemlere ait veriler TBL\_MESAJLAR tablosundan getirilmekte. Gridcontrol ile gösterilmektedir. Bu bilgiler;

* Mesa Adı Soyadı,
* Mesaj.



Şekil 45: Mesaj Bilgi İşlem Sekmesi

Kayıt için yan kısımda ki bilgileri doldurarak yeni bir mesaj girerek kayıt etmiş oluyoruz. Yeni bir mesaj yazdığımızda mesajlar TBL\_MESAJLAR tablosuna kaydedilir. Kaydet butonuna basılarak yeni kayıt eklenmektedir. 

Şekil 46: Mesaj Kayıt

## **2.6.2.11. Ayarlar Bilgi İşlem Sekmesi:**

Ayarlar Bilgi İşlemleri Sekmesi ikiye ayrılır. Birincisi Öğretmen Ayarlar İkinci Öğrenci Ayarlar. Bu İki kısımda da;

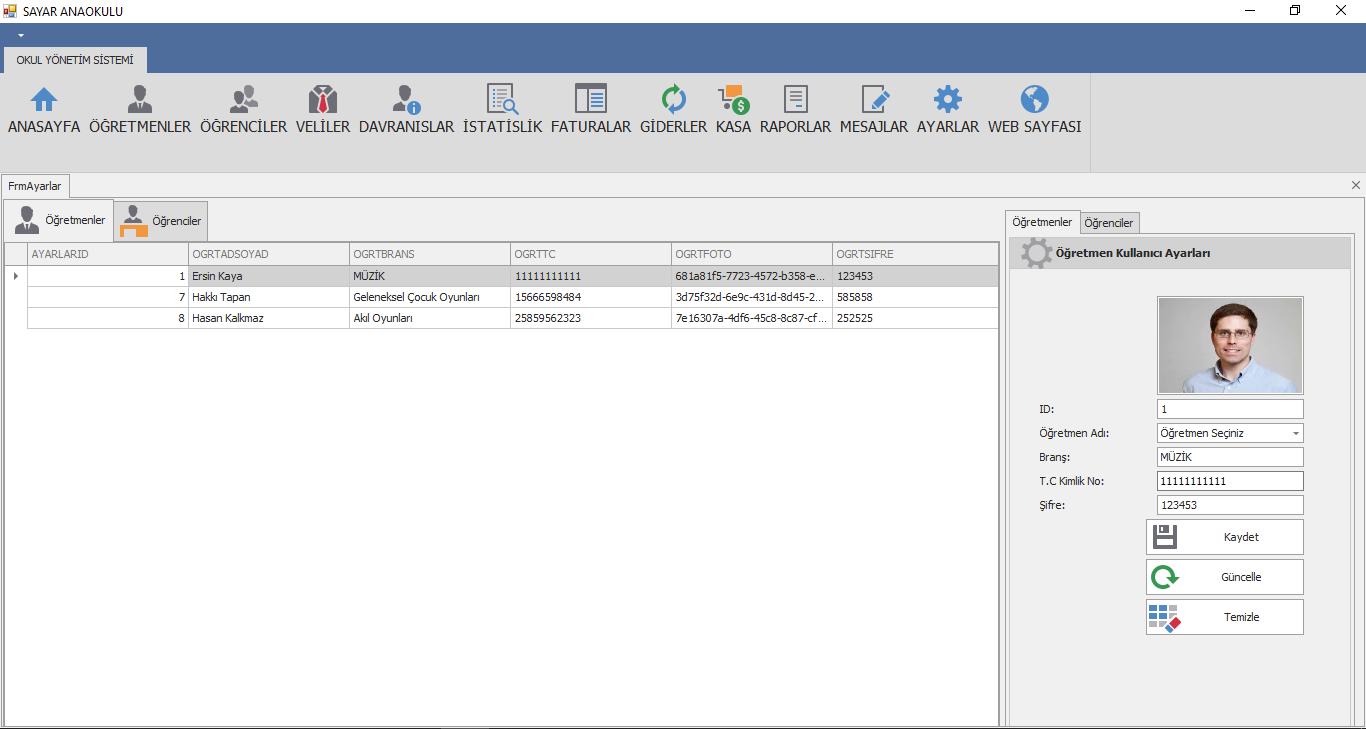
* Kaydet,
* Güncelle,
* Temizle.

Bu işlemlere ait veriler TBL\_OGRETMENLER ve TBL\_OGRENCILER tablosundan getirilmekte. Gridcontrol ile gösterilmektedir. Öğretmene ait bilgiler;

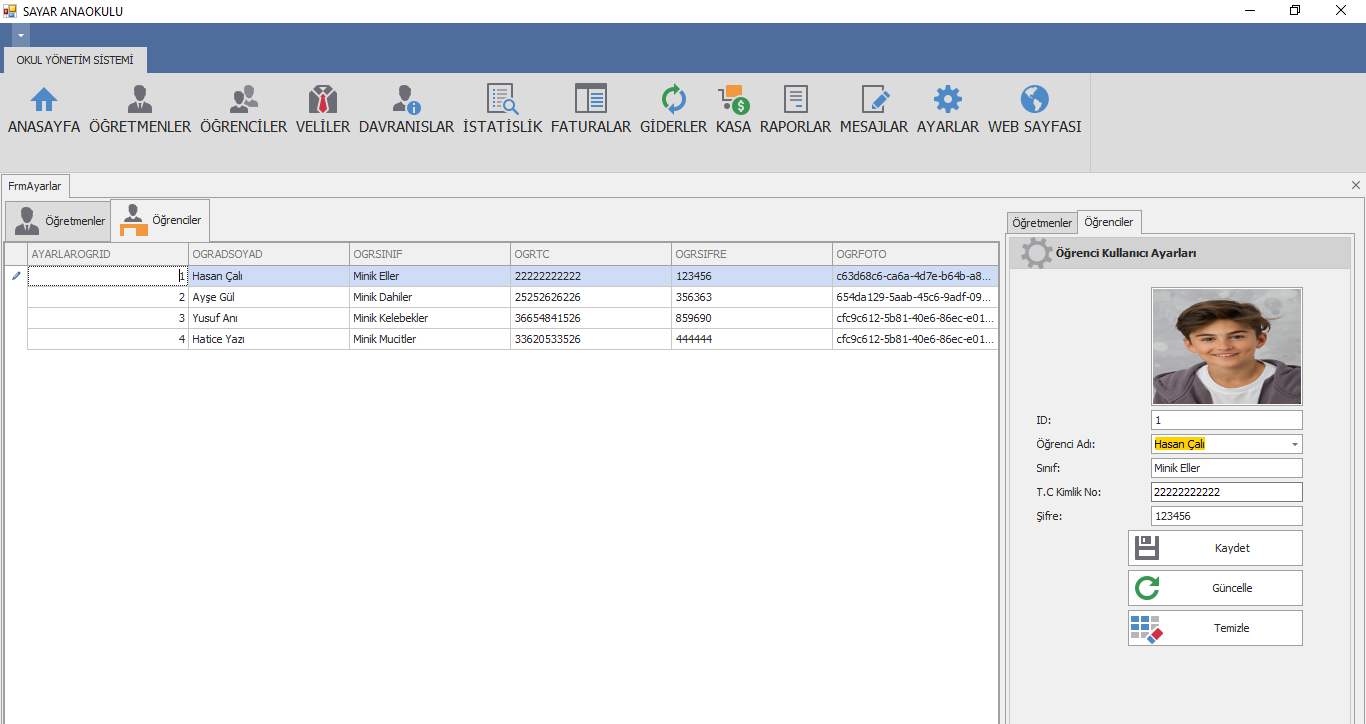
* Öğretmen İd,
* Öğretmen Fotoğraf,
* Öğretmen Adı,
* Öğretmen TC,
* Öğretmen Branşı,
* Öğretmen Şifre.

Öğrenciye ait bilgiler ise;

* Öğrenci İd,
* Öğrenci Foto,
* Öğrenci Adı,
* Öğrenci TC,
* Öğrenci Şifre.

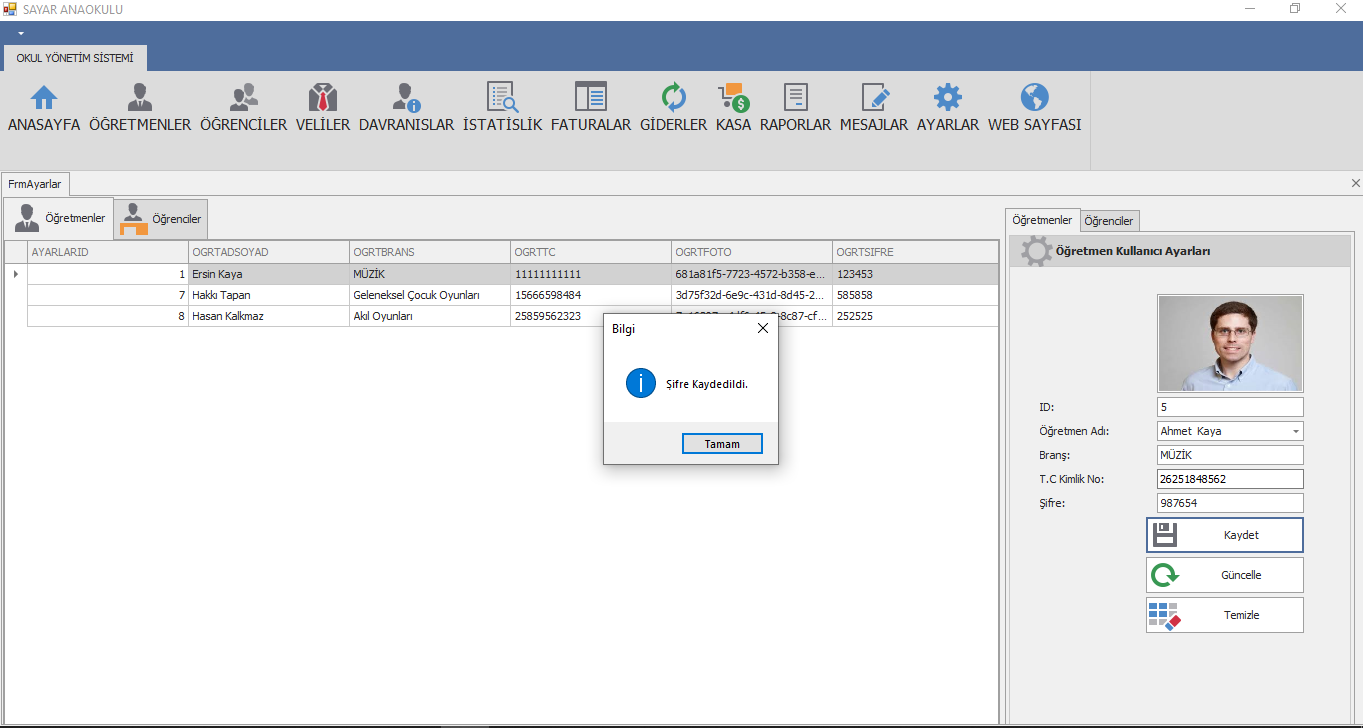


Şekil 47: Öğretmen Ayarlar Bilgi İşlem Sekmesi



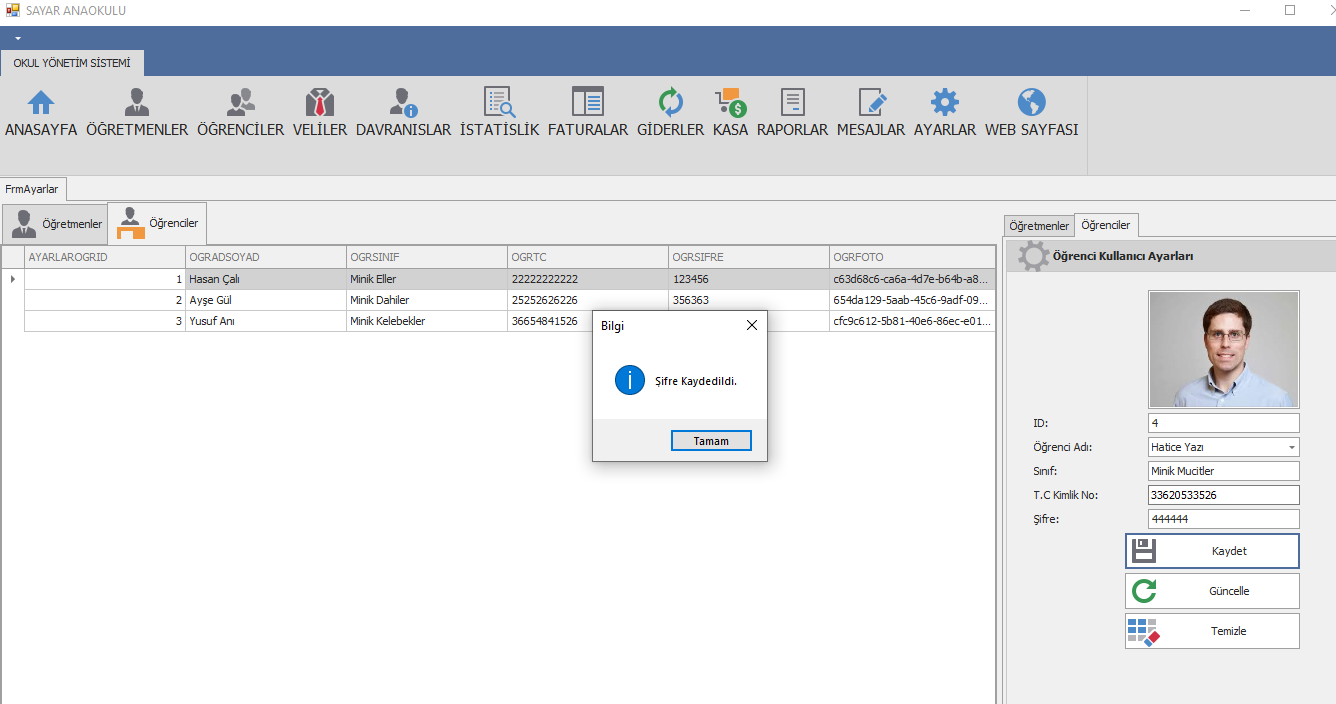
Şekil 48: Öğrenci Ayarlar Bilgi İşlem Sekmesi

Kayıt için yan kısımda ki bilgileri doldurarak yeni öğretmen şifre girerek kayıt etmiş oluyoruz. Yeni kayıtta bilgiler kaydedilirken TBL\_AYARLAR tanlosuna kaydedilir. ComboBoxtan şeçilen öğretmen bilgileri diger alanlara otomatik olarak doldurulur. Öğretmen bilgileri girilmeyi müteakip Kaydet butonuna basılarak yeni kayıt eklenmektedir.

****

Şekil 49: Öğretmen Şifre Kayıt

Kayıt için yan kısımda ki bilgileri doldurarak yeni öğretmen şifre girerek kayıt etmiş oluyoruz. Yeni kayıtlar kaydedilirken TBL\_OGRAYARLAR tablosuna kaydedilir. ComboBoxtan şeçilen öğretmen bilgileri diger alanlara otomatik olarak doldurulur. Öğrenci bilgileri girilmeyi müteakip Kaydet butonuna basılarak yeni kayıt eklenmektedir.



Şekil 50: Öğrenci Şifre Kayıt

Öğretmen İd kısmına tıkladığımız zaman öğrencinin bilgileri yan kısımda otomatik olarak doldurur. Bu bilgiler getirildikten sonra bu öğretmene ait;

* **Güncelle** işlemleri yapılabilmektedir.

Öğretmene ait bilgiler güncellenmek isteniyorsa güncellenmek istenen bilgiler değiştirilmeyi müteakip **Güncelle** butonuna tıklayarak bilgiler güncellenir.

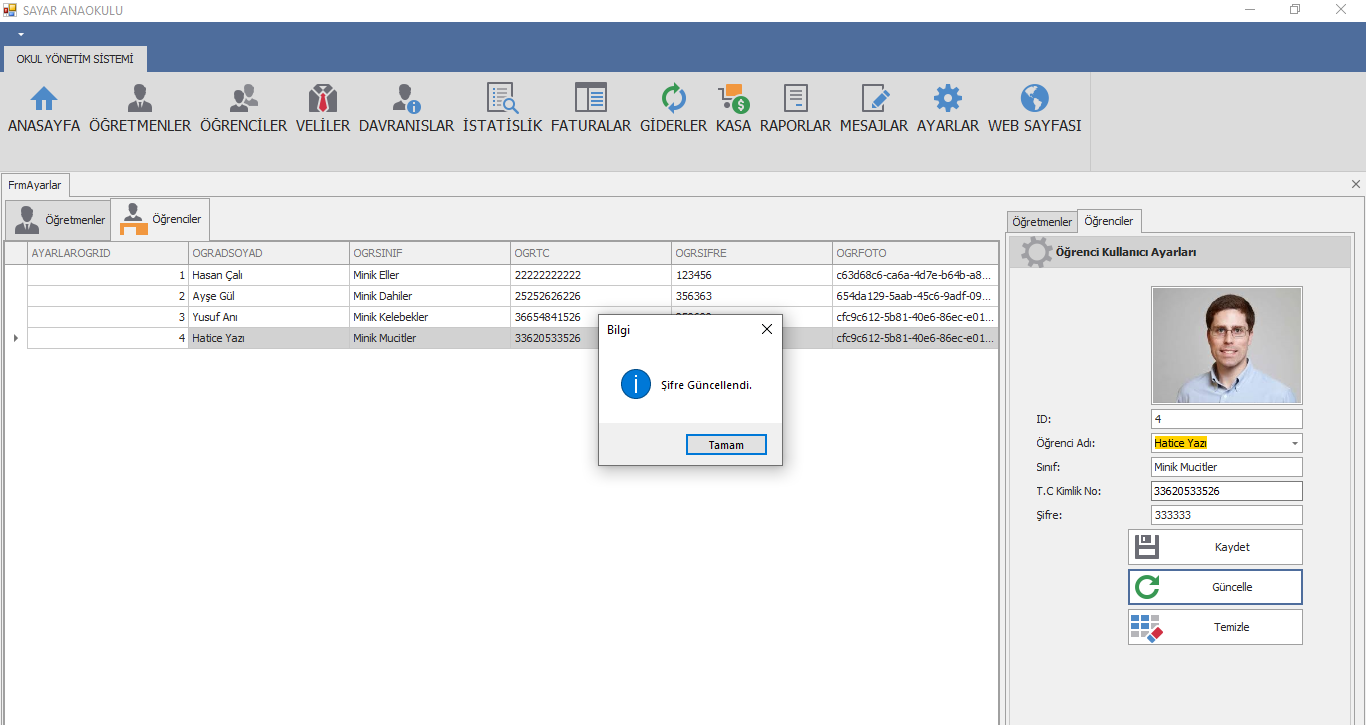


Şekil 51: Öğretmen Şifre Güncelle

Öğrenci İd kısmına tıkladığımız zaman öğrencinin bilgileri yan kısımda otomatik olarak doldurur. Bu bilgiler getirildikten sonra bu öğrenciye ait;

* **Güncelle** işlemleri yapılabilmektedir.

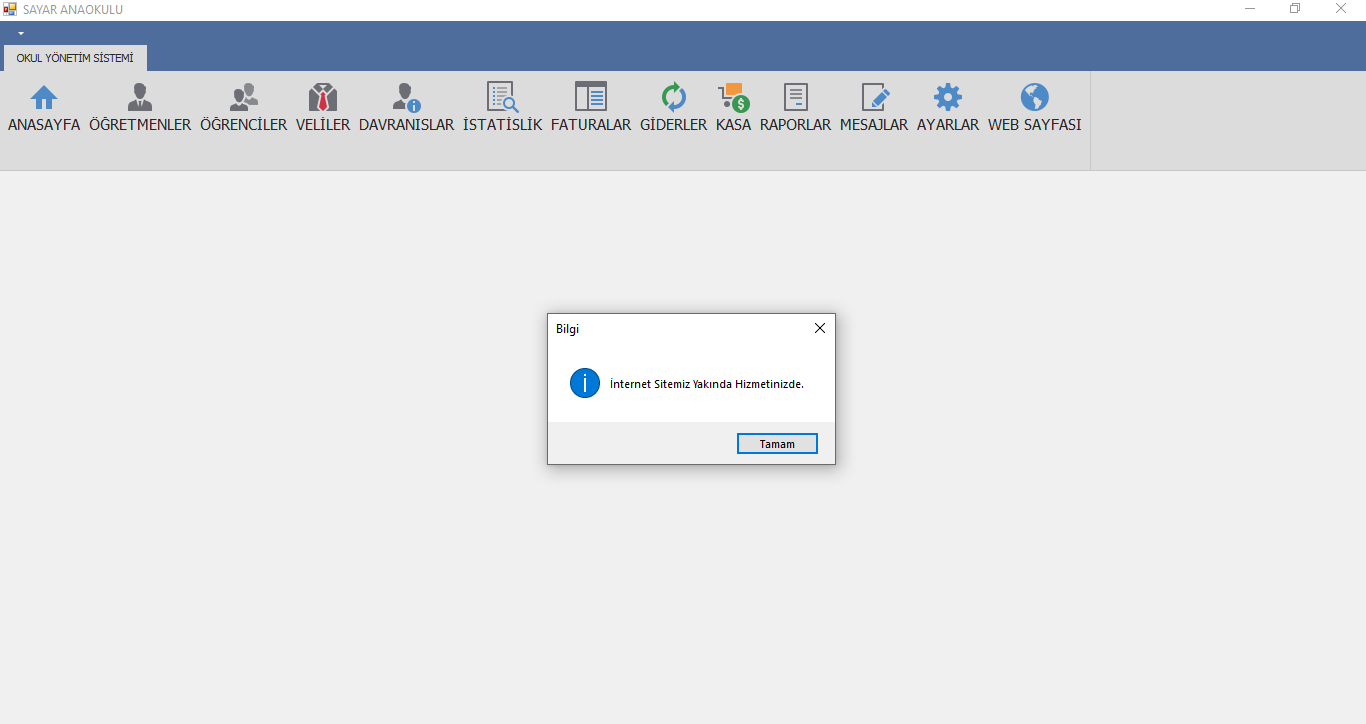
Öğrenci ait bilgiler güncellenmek isteniyorsa güncellenmek istenen bilgiler değiştirilmeyi müteakip **Güncelle** butonuna tıklayarak bilgiler güncellenir.



Şekil 52: Öğrenci Şifre Güncelle

## **2.6.2.12. Web Sayfası Sekmesi:**

Bu bölüm okulun web sayfası olduğu zaman açılacak kısım. Su anda bir web sayfası olmadığı için mesaj olarak olmadığı verilmiştir.

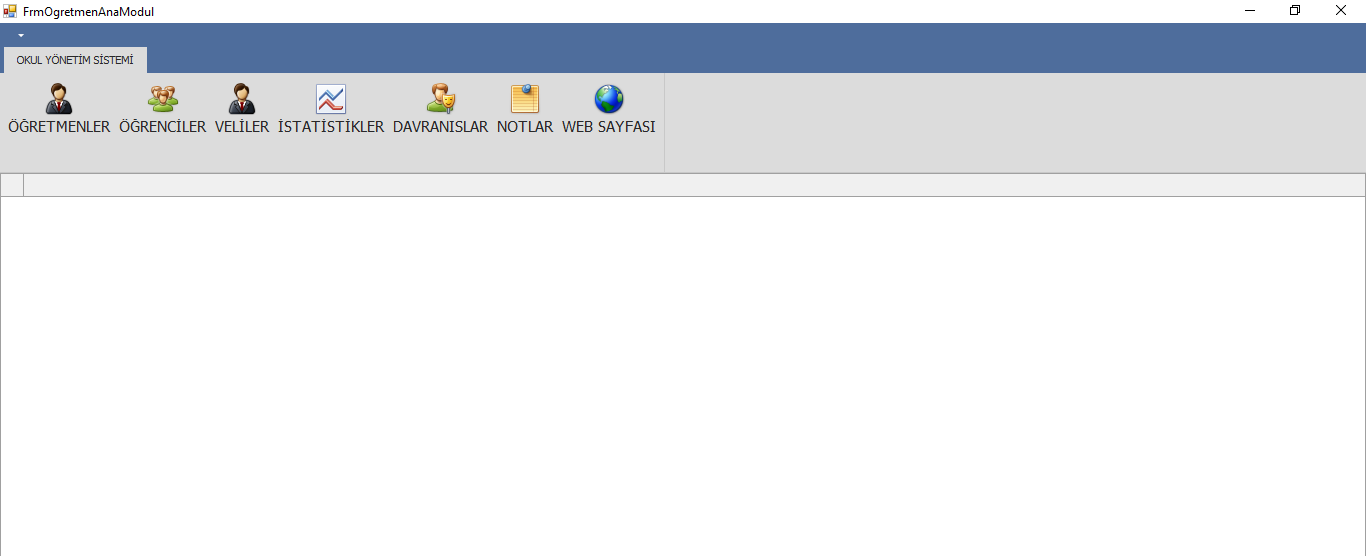


Şekil 53: Web Sayfası

## **Anaokulu Yönetim Sistemi Öğretmen Ana Ekranı**

Öğretmen olarak giriş yapıldığında **Öğretmen Ana Ekranı** gelmektedir. Öğretmen Ana Ekranında sırasıyla sekme olarak;

* Öğrenci Bilgi İşlemleri Sekmesi,
* Öğretmen Bilgi İşlemleri Sekmesi,
* Veli Bilgi İşlemleri Sekmesi,
* Davranış Bilgi Sekmesi,
* İstatislik Bilgi Sekmesi,
* Notlar Bilgi İşlem Sekmesi,
* Web Sayfası Sekmesi.

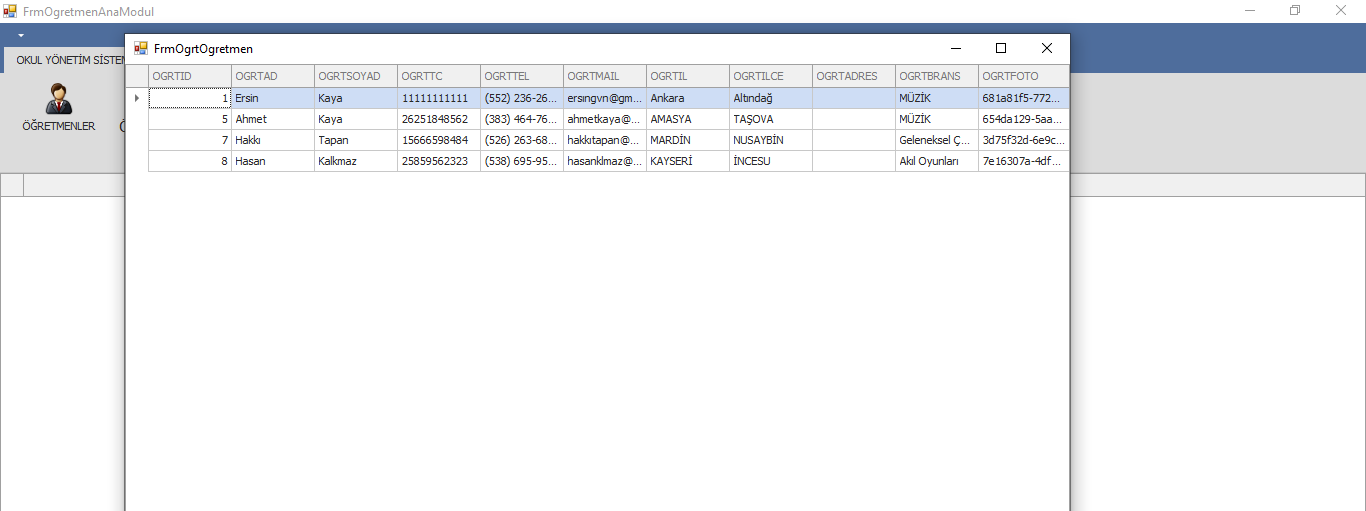


Şekil 54: Öğretmen Ana Sayfa

## **2.6.3.1. Öğretmen Bilgi Sekmesi**

Bu bölümde öğretmene ait veriler TBL\_OGRETMENLER tablosundan getirilmekte. Gridcontrol ile gösterilmektedir. Öğretmene ait bilgiler;

* Öğretmen İd,
* Adı,
* Soyadı,
* TC,
* Telefon,
* İl,
* İlçe,
* Branş,
* Mail,
* Adres,
* Resim.

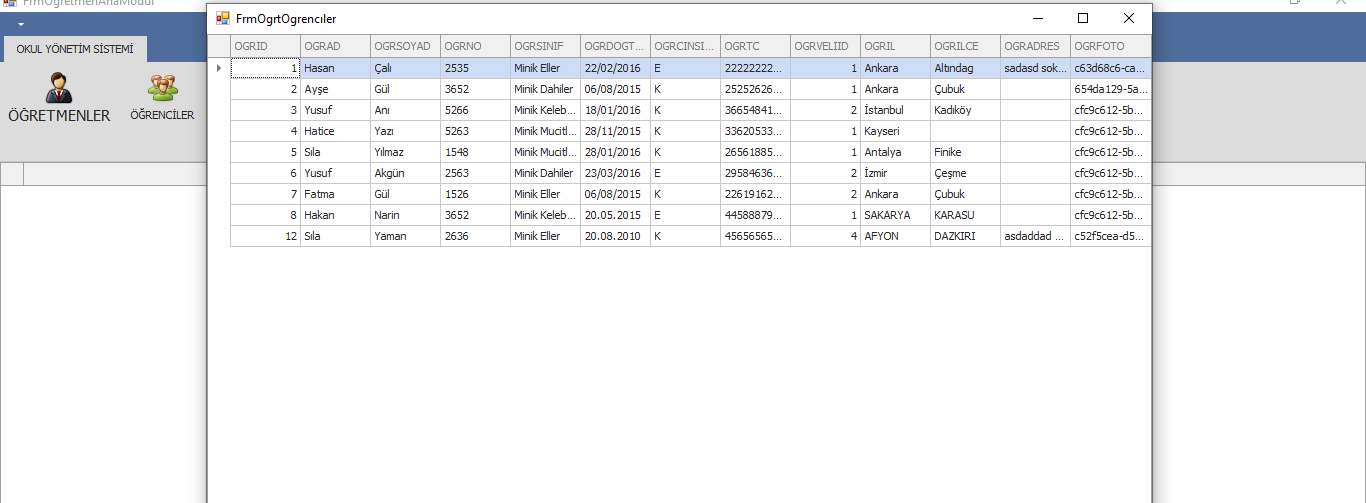
****

Şekil 55: Öğretmen Bilgi Sekmesi

## **2.6.3.2. Öğrenci Bilgi Sekmesi**

Bu bölümde öğrenciye ait veriler TBL\_OGRENCILER tablosundan getirilmekte. Gridcontrol ile gösterilmektedir. Öğrenciye ait bilgiler;

* Öğrenci İd,
* Adı,
* Soyadı,
* TC,
* Öğrenci No,
* Cinsiyet,
* Veli Ad-Soyad,
* İl,
* İlçe,
* Sınıf,
* Doğum Tarihi,
* Adres,
* Resim.

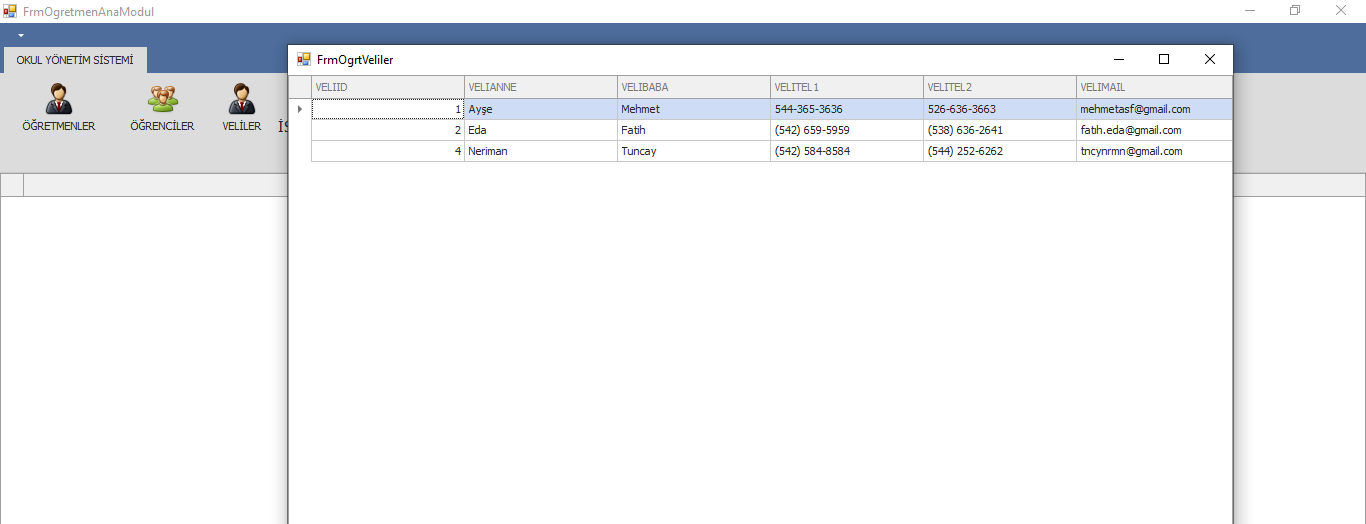
****

Şekil 56: Öğrenci Bilgi Sekmesi

## **2.6.3.3. Veli Bilgi Sekmesi**

Bu bölümde öğrenciye ait veriler TBL\_VELILER tablosundan getirilmekte. Gridcontrol ile gösterilmektedir. Veliye ait bilgiler;

* Veli İd,
* Anne Adı,
* Baba Adı,
* Telefon 1,
* Telefon 2,
* Maıl.



Şekil 57: Veli Bilgi Sekmesi

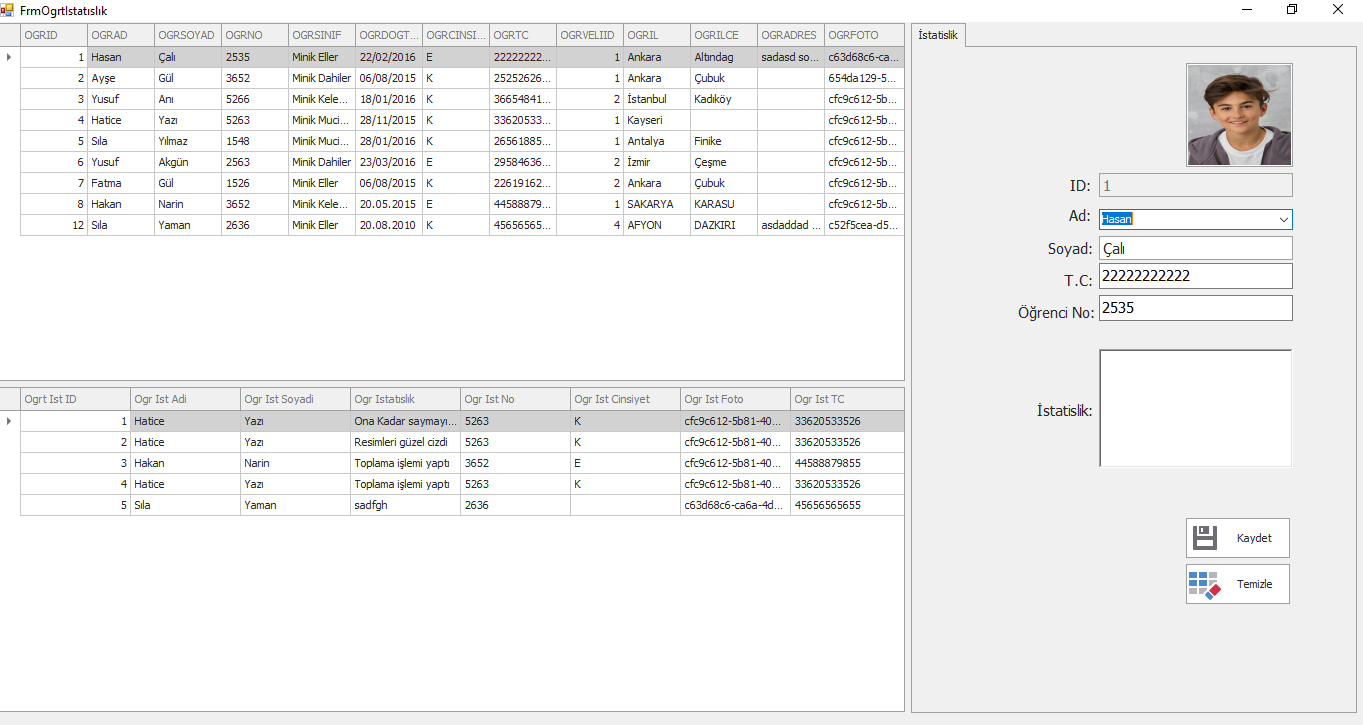
## **2.6.3.4. İstatistik Bilgi İşlem Sekmesi**

İstatistik Bilgi İşlemleri Sekmesinde;

* Kaydet,
* Temizle

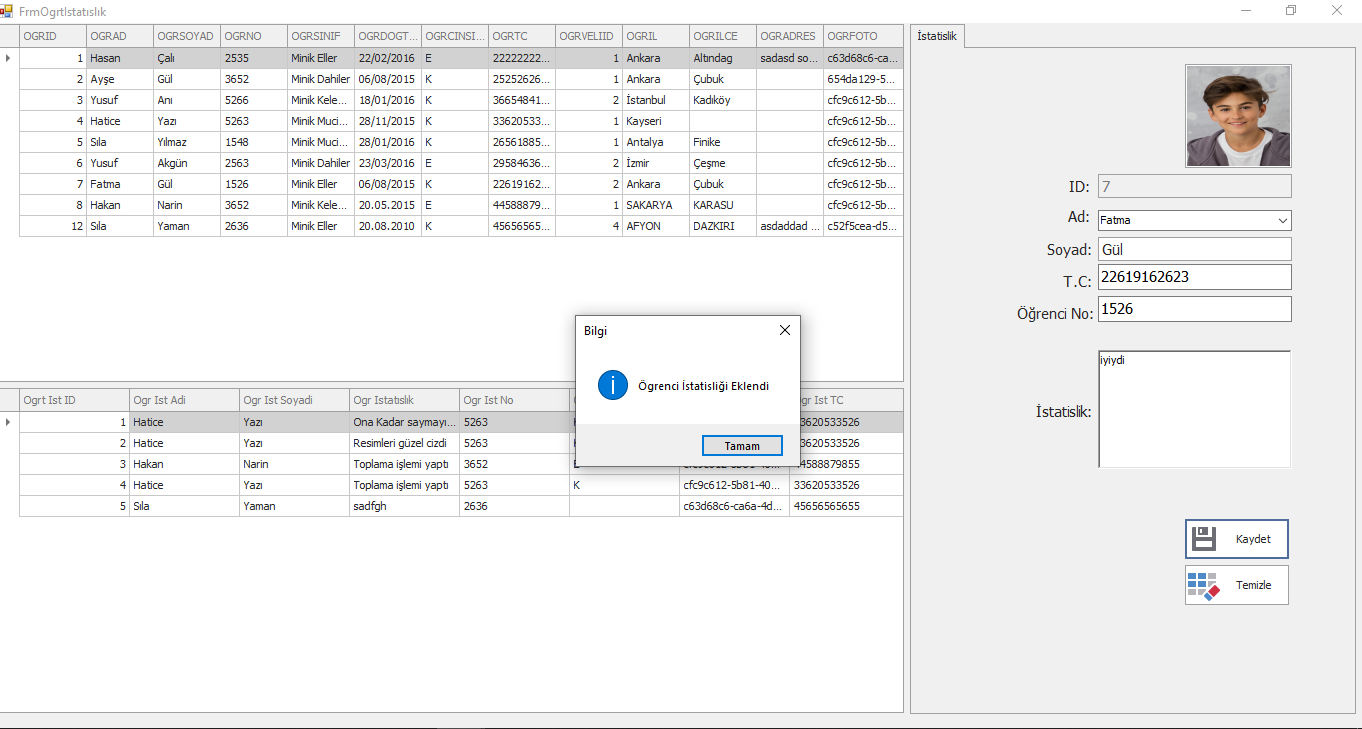
Bu bölümde öğrenciye ait veriler TBL\_OGRENCILER tablosundan getirilmekte. Gridcontrol ile gösterilmektedir. Öğrenciye ait bilgiler;

* Öğrenci İd,
* Adı,
* Soyadı,
* TC,
* Öğrenci No,
* Cinsiyet,
* Resim.



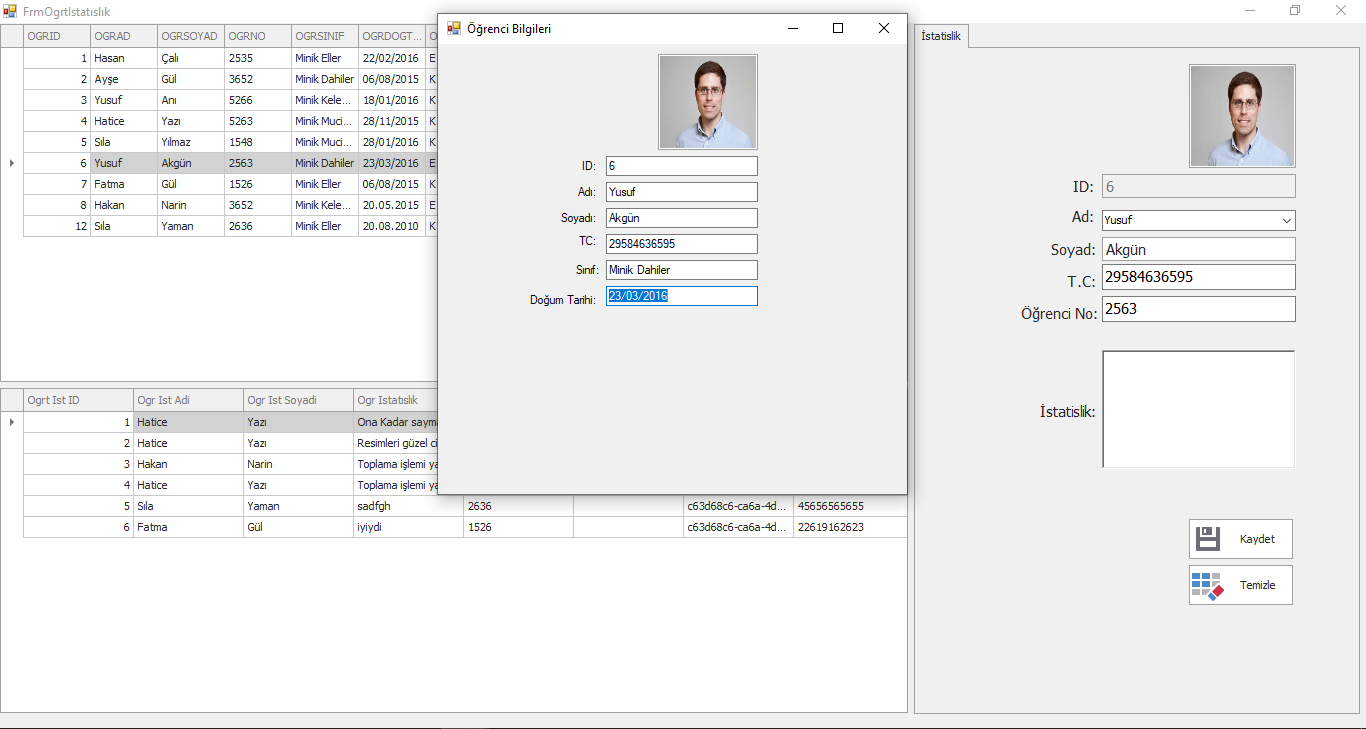
Şekil 58: İstatistik Bilgi İşlem Sekmesi

Kayıt için yan kısımda ki bilgileri doldurarak kayıt etmiş oluyoruz. Yeni kayıtlar kaydedilirken TBL\_ISTATISLIK tablosuna kaydedilir.ComboBoxtan seçtiğimiz isimdeki öğrenciye göre bilgiler diger yerlere aktarılır. Gelen bilgiler TBL\_OGRENCILERDEN gelir. Öğrenci bilgileri girilmeyi müteakip Kaydet butonuna basılarak yeni kayıt eklenmektedir. Kayıt edilen istatistikler GridControl’de tekrar listelenir.

****

Şekil 59: İstatistik Kayıt

Burada kayıt olan öğretmenlerin Gridconrolde çift tıklanarak Öğretmen Bilgileri formunu açar ve öğrencinin bilgilerini aktarır.

****

Şekil 60: İstatistik Öğrenci Bilgi Sekmesi

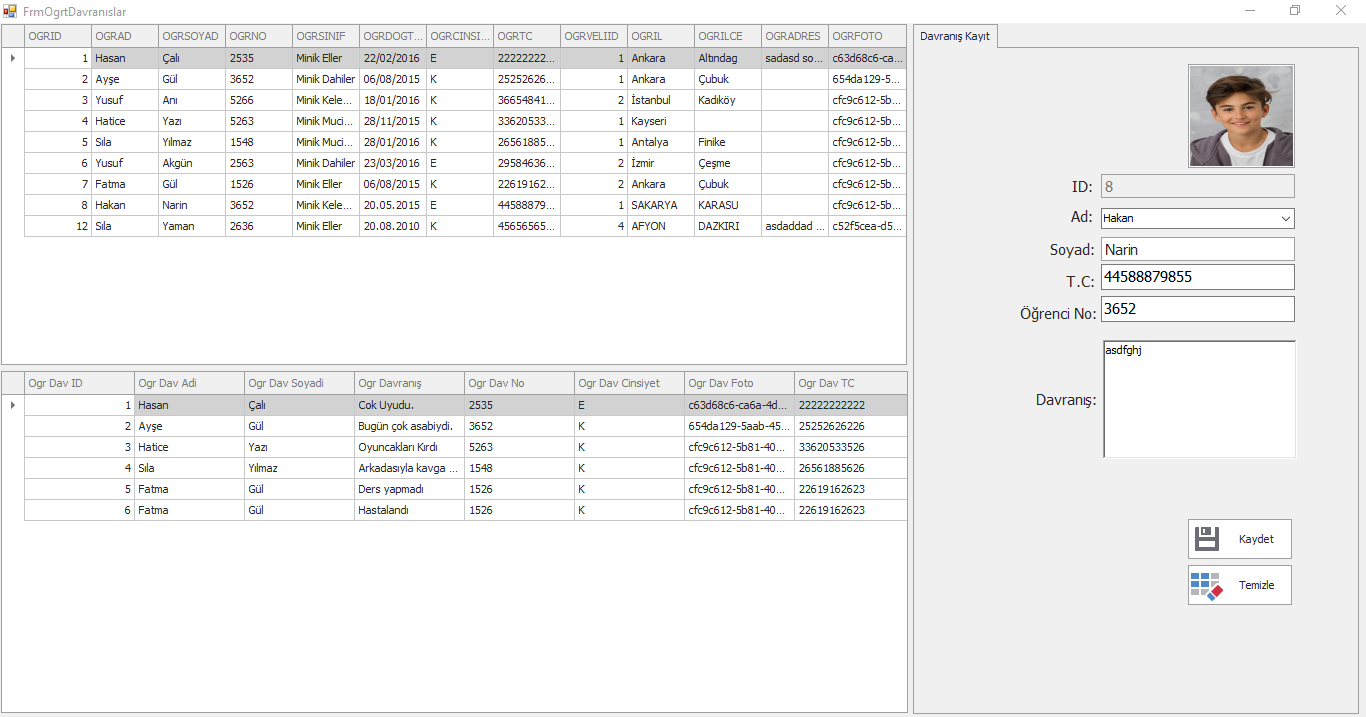
## **2.6.3.5. Davranış Bilgi İşlem Sekmesi**

Davranış Bilgi İşlemleri Sekmesinde;

* Kaydet,
* Temizle.

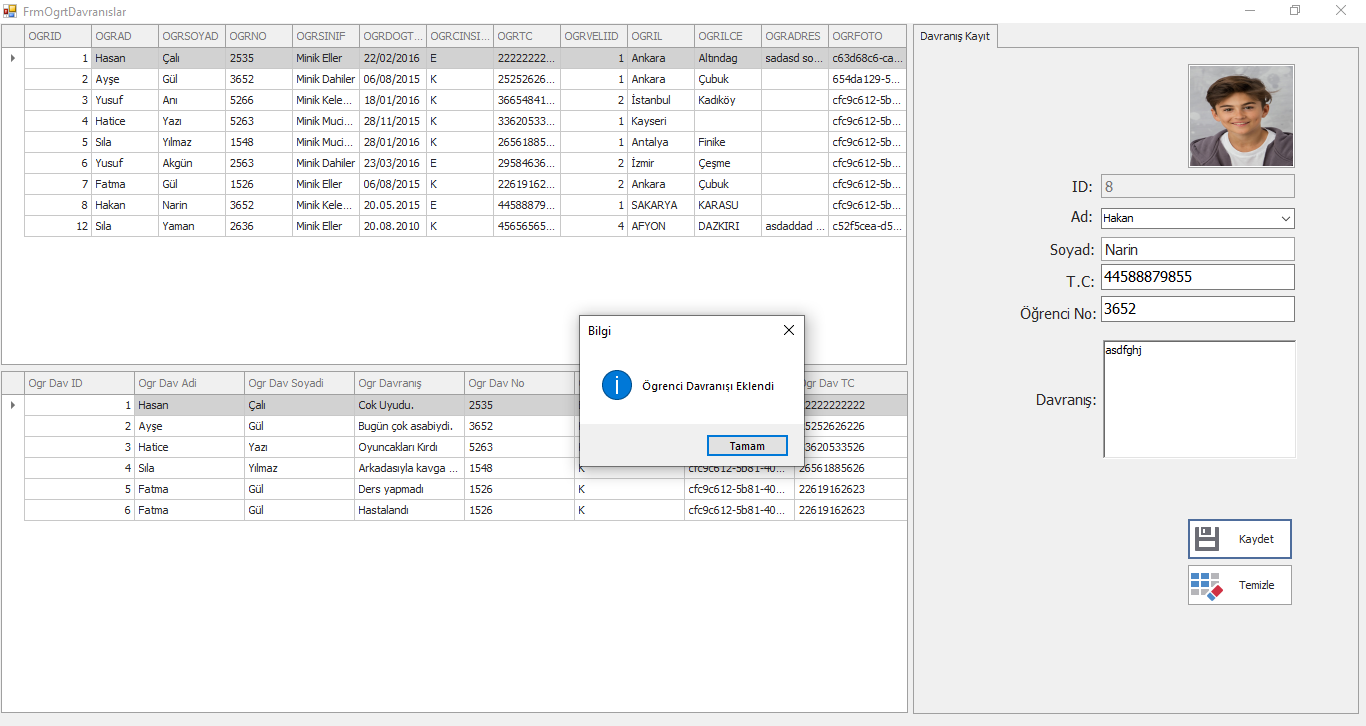
Bu bölümde öğrenciye ait veriler TBL\_OGRENCILER tablosundan getirilmekte. Gridcontrol ile gösterilmektedir. Öğrenciye ait bilgiler;

* Öğrenci İd,
* Adı,
* Soyadı,
* TC,
* Öğrenci No,
* Cinsiyet,
* Resim.



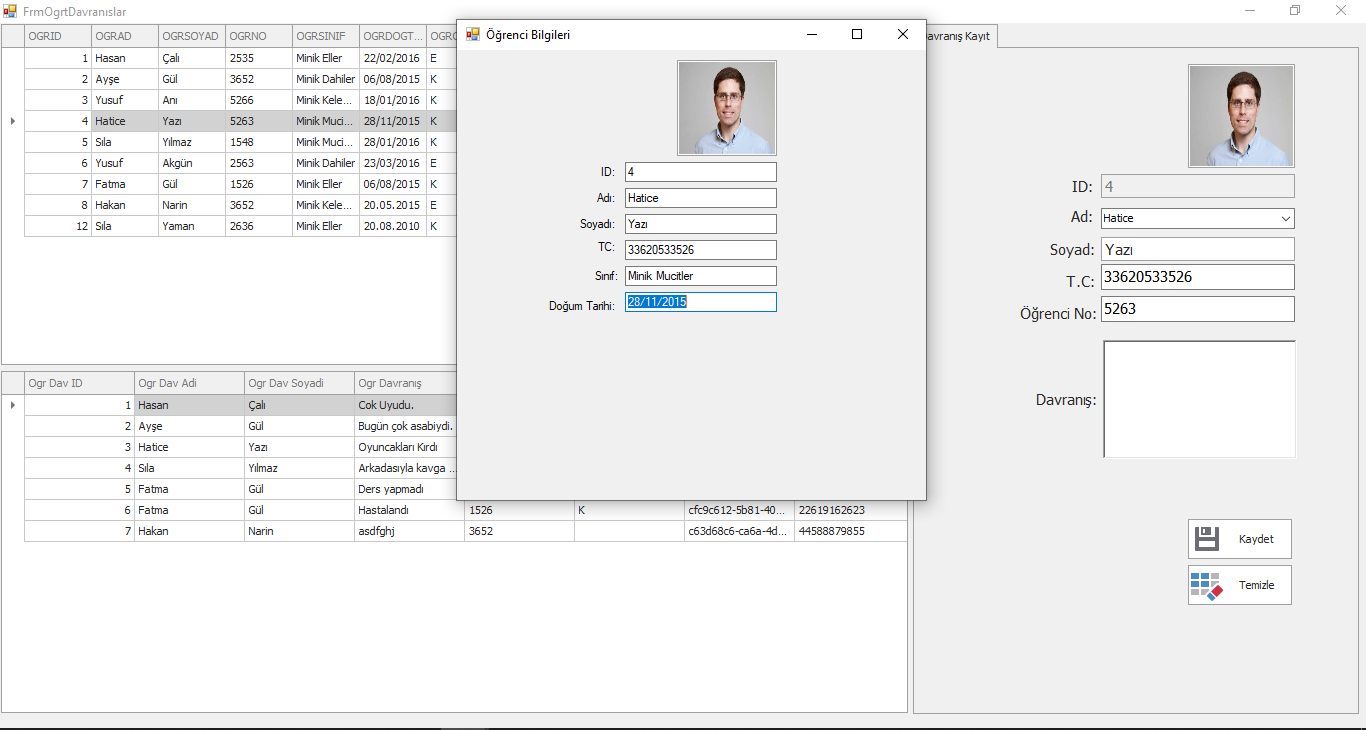
Şekil 61: Davranış Bilgi İşlem Sekmesi

Kayıt için yan kısımda ki bilgileri doldurarak kayıt etmiş oluyoruz. Yeni kayıtlar kaydedilirken TBL\_DAVRANISLAR tablosuna kaydedilir. ComboBoxtan seçtiğimiz isimdeki öğrenciye göre bilgiler diger yerlere aktarılır.Gelen bilgiler TBL\_OGRENCILERDEN gelir. Öğrenci bilgileri girilmeyi müteakip Kaydet butonuna basılarak yeni kayıt eklenmektedir. Kayıt edilen davranışlar GridControl’de tekrar listelenir.



Şekil 62: Davranış Kayıt

Burada kayıt olan öğretmenlerin Gridconrolde çift tıklanarak Öğretmen Bilgileri formunu açar ve öğrencinin bilgilerini aktarır.

****

Şekil 63: Davranış öğrenci Bilgi Sekmesi

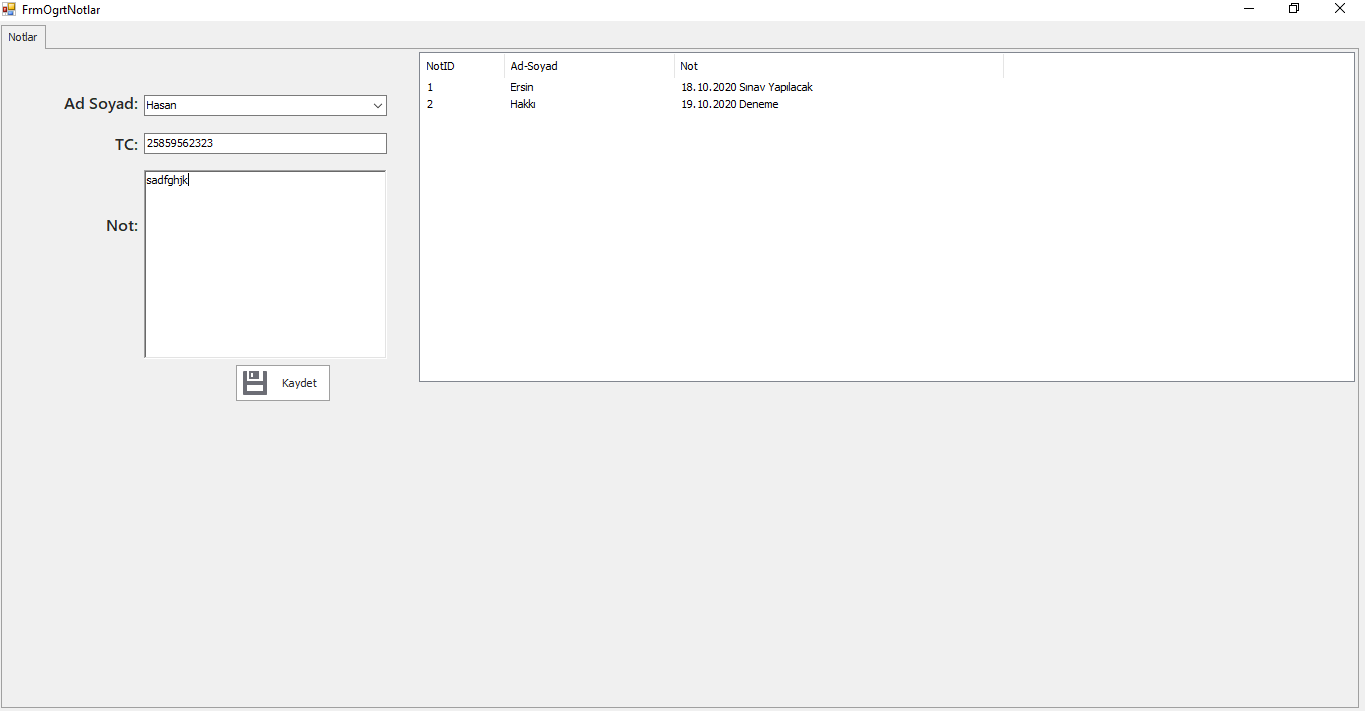
## **2.6.3.6. Notlar Bilgi İşlem Sekmesi**

Notlar Bilgi İşlemleri Sekmesinde;

* Kaydet,

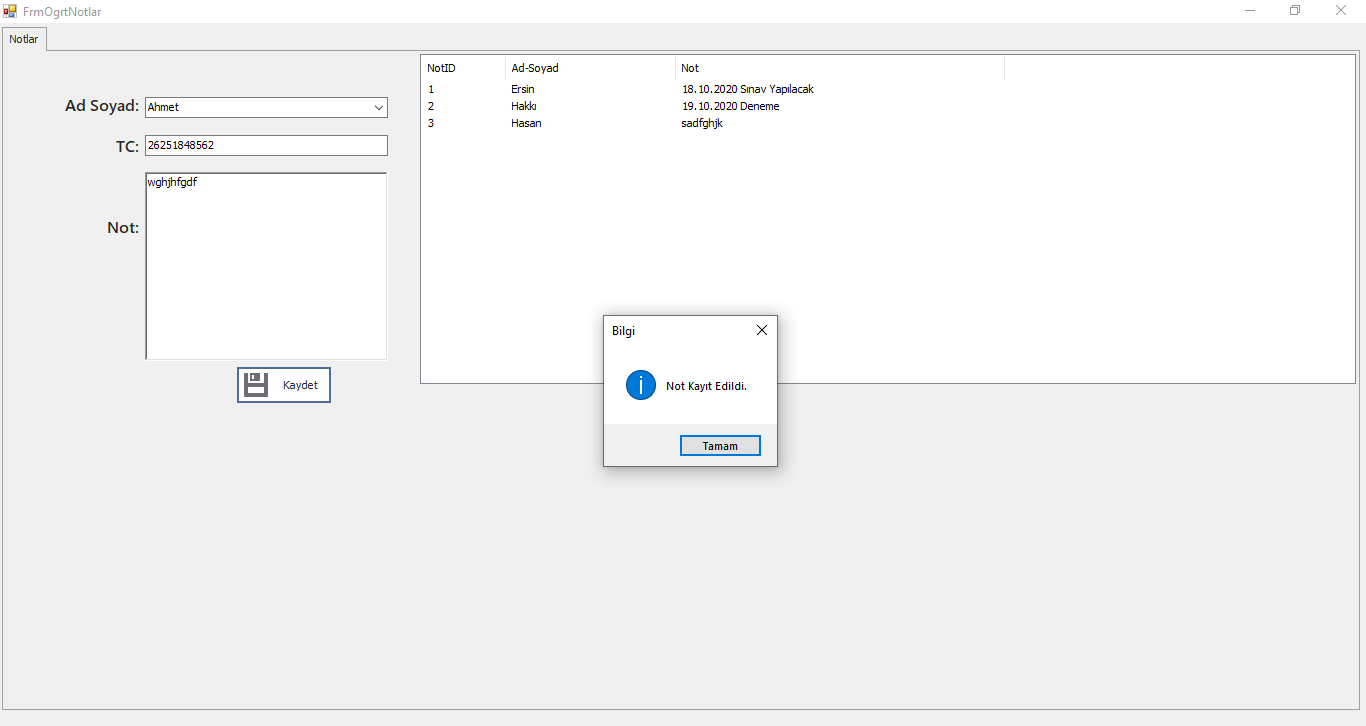
Bu bölümde öğretmen not girerek veriler TBL\_NOTLAR tablosundan getirilmekte. Gridcontrol ile gösterilmektedir. Notlara ait bilgiler;

* Ad-Soyad,
* Not.



Şekil 64: Not Bilgi İşlem Sekmesi

Kayıt için yan kısımda ki bilgileri doldurarak kayıt etmiş oluyoruz. Yeni kayıtlar kaydedilirken TBL\_NOTLAR tablosuna kaydedilir.ComboBoxtan seçtiğimiz isimdeki öğrenciye göre bilgiler diger yerlere aktarılır.Adlar ise TBL\_OGRETMENLER tablosundan gelir. Girilen notları girilmeyi müteakip Kaydet butonuna basılarak yeni kayıt eklenmektedir. Kayıt edilen notlar GridControl’de tekrar listelenir.

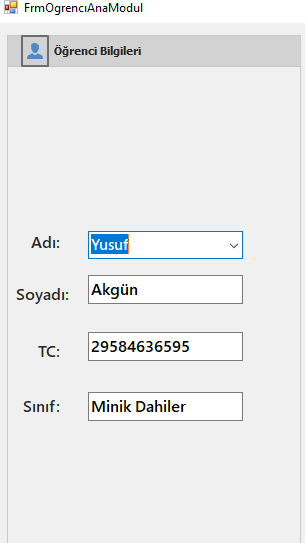
****

Şekil 65: Not Kayıt

## **Anaokulu Yönetim Sistemi Öğrenci Ana Ekranı**

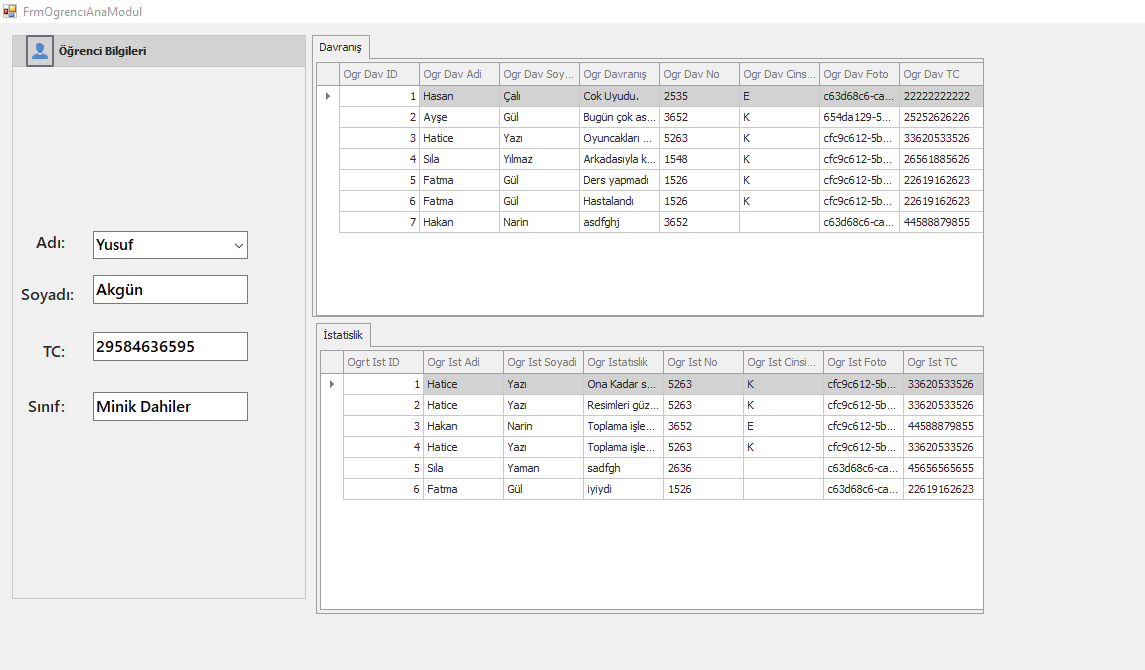
Bu bölümde öğretmen not girerek veriler TBL\_OGRENCILER tablosundan getirilmekte. GridControl ile gösterilmektedir. Öğrenciye ait bilgiler;

* Öğrenci Adı,
* Öğrenci Soyadı,
* TC,
* Sınıf.



Şekil 66: Öğrenci Ekranı Bilgi

Bu bölümde öğrencilerin davranışları ve istatistikleri GridControllerde listelenir.

****

Şekil 67: Öğrenci Ekranı Genel

# **2.7. Projenin Kodları**

**2.7.1. Giriş Formu Kodları**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.SqlClient;

namespace AnaOkuluYönetimi

{

public partial class FrmGiris : Form

{

public FrmGiris()

{

InitializeComponent();

}

sqlbaglantisi bgl = new sqlbaglantisi();

private void BtnYonetıcı\_Click(object sender, EventArgs e)

{

// yönetici giriş paneli

SqlCommand komut = new SqlCommand("select OGRTTC,OGRTSIFRE From TBL\_AYARLAR inner join TBL\_OGRETMENLER on TBL\_AYARLAR.AYARLARID=TBL\_OGRETMENLER.OGRTID where OGRTTC=@p1 and OGRTSIFRE=@p2", bgl.baglanti());

komut.Parameters.AddWithValue("@p1", MskTC.Text);

komut.Parameters.AddWithValue("@p2", TxtSıfre2.Text);

SqlDataReader dr = komut.ExecuteReader();

if (dr.Read())

{

FrmAnaModul frm1 = new FrmAnaModul();

frm1.Show();

this.Hide();

}

else

{

MessageBox.Show("Hatalı Kullanıcı veya Şifre.");

MskTC.Text = "";

TxtSıfre2.Text = "";

}

bgl.baglanti().Close();

}

private void BtnOgretmen\_Click(object sender, EventArgs e)

{

// öğretmen giriş panelı

SqlCommand komut = new SqlCommand("select OGRTTC,OGRTSIFRE From TBL\_AYARLAR inner join TBL\_OGRETMENLER on TBL\_AYARLAR.AYARLARID=TBL\_OGRETMENLER.OGRTID where OGRTTC=@p1 and OGRTSIFRE=@p2", bgl.baglanti());

komut.Parameters.AddWithValue("@p1", MskTC.Text);

komut.Parameters.AddWithValue("@p2", TxtSıfre2.Text);

SqlDataReader dr = komut.ExecuteReader();

if (dr.Read())

{

FrmOgretmenAnaModul frm2 = new FrmOgretmenAnaModul();

frm2.Show();

this.Hide();

}

else

{

MessageBox.Show("Hatalı Kullanıcı veya Şifre.");

MskTC.Text = "";

TxtSıfre2.Text = "";

}

bgl.baglanti().Close();

}

private void BtnOgrencı\_Click(object sender, EventArgs e)

{

// öğrenci giriş panelı

SqlCommand komut = new SqlCommand("select OGRTC,OGRSIFRE From TBL\_OGRAYARLAR inner join TBL\_OGRENCILER on TBL\_OGRAYARLAR.AYARLAROGRID=TBL\_OGRENCILER.OGRID where OGRTC=@p1 and OGRSIFRE=@p2", bgl.baglanti());

komut.Parameters.AddWithValue("@p1", MskTC.Text);

komut.Parameters.AddWithValue("@p2", TxtSıfre2.Text);

SqlDataReader dr = komut.ExecuteReader();

if (dr.Read())

{

FrmOgrencıAnaModul frm3 = new FrmOgrencıAnaModul();

frm3.Show();

this.Hide();

}

else

{

MessageBox.Show("Hatalı Kullanıcı veya Şifre.");

MskTC.Text = "";

TxtSıfre2.Text = "";

}

bgl.baglanti().Close();

}

}

}

**2.7.2. Yönetici Ekranı Formu Kodları**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace AnaOkuluYönetimi

{

public partial class FrmAnaModul : Form

{

public FrmAnaModul()

{

InitializeComponent();

}

FrmOgretmenler frm1;

FrmOgrencıler frm2;

FrmVeliler frm3;

FrmAyarlar frm4;

FrmDavranıslar frm5;

FrmGiderler frm6;

FrmFaturalar frm7;

FrmKasa frm8;

FrmMesajlar frm9;

FrmIstatislik frm10;

FrmRaporlar frm11;

private void BtnOgretmen\_ItemClick(object sender, DevExpress.XtraBars.ItemClickEventArgs e)

{

// ögretmenler kısmına tıkladıgında ogretmenler formunu acan kod

if (frm1 == null || frm1.IsDisposed)

{

frm1 = new FrmOgretmenler();

frm1.MdiParent = this;

frm1.Show();

}

}

private void BtnOgrencıler\_ItemClick(object sender, DevExpress.XtraBars.ItemClickEventArgs e)

{

// ögrencıler formuna gider

if (frm2 == null || frm2.IsDisposed)

{

frm2 = new FrmOgrencıler();

frm2.MdiParent = this;

frm2.Show();

}

}

private void BtnVelıler\_ItemClick(object sender, DevExpress.XtraBars.ItemClickEventArgs e)

{

// veliler formuna gider

if (frm3 == null || frm3.IsDisposed)

{

frm3 = new FrmVeliler();

frm3.MdiParent = this;

frm3.Show();

}

}

private void barButtonItem14\_ItemClick(object sender, DevExpress.XtraBars.ItemClickEventArgs e)

{

// ayarlar formuna gider

if (frm4 == null || frm4.IsDisposed)

{

frm4 = new FrmAyarlar();

frm4.MdiParent = this;

frm4.Show();

}

}

private void barButtonItem15\_ItemClick(object sender, DevExpress.XtraBars.ItemClickEventArgs e)

{

MessageBox.Show("İnternet Sitemiz Yakında Hizmetinizde.", "Bilgi", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

private void barButtonItem8\_ItemClick(object sender, DevExpress.XtraBars.ItemClickEventArgs e)

{

if (frm5 == null || frm5.IsDisposed)

{

frm5 = new FrmDavranıslar();

frm5.MdiParent = this;

frm5.Show();

}

}

private void barButtonItem10\_ItemClick(object sender, DevExpress.XtraBars.ItemClickEventArgs e)

{

frm6 = new FrmGiderler();

frm6.MdiParent = this;

frm6.Show();

}

private void barButtonItem7\_ItemClick(object sender, DevExpress.XtraBars.ItemClickEventArgs e)

{

frm7 = new FrmFaturalar();

frm7.MdiParent = this;

frm7.Show();

}

private void barButtonItem11\_ItemClick(object sender, DevExpress.XtraBars.ItemClickEventArgs e)

{

frm8 = new FrmKasa();

frm8.MdiParent = this;

frm8.Show();

}

private void barButtonItem13\_ItemClick(object sender, DevExpress.XtraBars.ItemClickEventArgs e)

{

frm9 = new FrmMesajlar();

frm9.MdiParent = this;

frm9.Show();

}

private void barButtonItem9\_ItemClick(object sender, DevExpress.XtraBars.ItemClickEventArgs e)

{

frm10 = new FrmIstatislik();

frm10.MdiParent = this;

frm10.Show();

}

private void barButtonItem12\_ItemClick(object sender, DevExpress.XtraBars.ItemClickEventArgs e)

{

frm11 = new FrmRaporlar();

frm11.MdiParent = this;

frm11.Show();

}

}

}

**2.7.2.1 Yönetici Ekranı Öğretmenler Formu Kodları**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.SqlClient;

using System.IO;

namespace AnaOkuluYönetimi

{

public partial class FrmOgretmenler : Form

{

public FrmOgretmenler()

{

InitializeComponent();

}

sqlbaglantisi bgl = new sqlbaglantisi();

void listele()

{

//Veritabından öğretmenler tablosunu getirir

DataTable dt = new DataTable();

SqlDataAdapter da = new SqlDataAdapter("Select \* from TBL\_OGRETMENLER",bgl.baglanti());

da.Fill(dt);

gridControl1.DataSource = dt;

}

void ilekle()

{

// Veritabanındaki iller tablosunu getirir

SqlCommand komut = new SqlCommand("select \* from TBL\_ILLER",bgl.baglanti());

SqlDataReader dr =komut.ExecuteReader();

while (dr.Read())

{

Cmbil.Properties.Items.Add(dr[1]);

}

bgl.baglanti().Close();

}

private void FrmOgretmenler\_Load(object sender, EventArgs e)

{

listele();

ilekle();

bransgetir();

temizle();

}

void bransgetir()

{

// veritabanından braşları combobox getiren kod

SqlCommand komut = new SqlCommand("select \* from TBL\_BRANSLAR", bgl.baglanti());

SqlDataReader dr = komut.ExecuteReader();

while (dr.Read())

{

CmbBrans.Properties.Items.Add(dr[1]);

}

bgl.baglanti().Close();

}

void temizle()

{

TxtID.Text = "";

TxtAd.Text = "";

TxtSoyad.Text = "";

MskTC.Text ="";

MskTelefon.Text = "";

TxtMail.Text = "";

Cmbil.Text = "";

Cmbilce.Text = "";

RchAdres.Text = "";

CmbBrans.Text = "";

pictureEdit.Text = "";

}

private void Cmbil\_SelectedIndexChanged(object sender, EventArgs e)

{

// Veritabanındaki ilçeler tablosunu getirir ve seçilen ilçeyi temizler

Cmbilce.Properties.Items.Clear();

Cmbilce.Text = "";

SqlCommand komut = new SqlCommand("select \* from TBL\_ILCELER where sehir=@p1", bgl.baglanti());

komut.Parameters.AddWithValue("@p1", Cmbil.SelectedIndex+1);

SqlDataReader dr = komut.ExecuteReader();

while (dr.Read())

{

Cmbilce.Properties.Items.Add(dr[1]);

}

bgl.baglanti().Close();

}

private void BtnKaydet\_Click(object sender, EventArgs e)

{

// ögretmen tablosundan veritabnına kayıt ekleme kaydetme işlemi

SqlCommand komut = new SqlCommand("Insert into TBL\_OGRETMENLER (OGRTAD,OGRTSOYAD,OGRTTC,OGRTTEL,OGRTMAIL,OGRTIL,OGRTILCE,OGRTADRES,OGRTBRANS,OGRTFOTO) values(@p1,@p2,@p3,@p4,@p5,@p6,@p7,@p8,@p9,@p10)", bgl.baglanti());

komut.Parameters.AddWithValue("@p1", TxtAd.Text);

komut.Parameters.AddWithValue("@p2", TxtSoyad.Text);

komut.Parameters.AddWithValue("@p3", MskTC.Text);

komut.Parameters.AddWithValue("@p4", MskTelefon.Text);

komut.Parameters.AddWithValue("@p5", TxtMail.Text);

komut.Parameters.AddWithValue("@p6", Cmbil.Text);

komut.Parameters.AddWithValue("@p7", Cmbilce.Text);

komut.Parameters.AddWithValue("@p8", RchAdres.Text);

komut.Parameters.AddWithValue("@p9", CmbBrans.Text);

komut.Parameters.AddWithValue("@p10", Path.GetFileName(yeniyol));

komut.ExecuteNonQuery();

bgl.baglanti().Close();

MessageBox.Show("Personel Eklendi", "Bilgi", MessageBoxButtons.OK, MessageBoxIcon.Information);

listele();

}

private void gridView1\_FocusedRowObjectChanged(object sender, DevExpress.XtraGrid.Views.Base.FocusedRowObjectChangedEventArgs e)

{

// datagridview e tıklayınca bilgileri gösterme

DataRow dr = gridView1.GetDataRow(gridView1.FocusedRowHandle);

if (dr!=null)

{

TxtID.Text = dr["OGRTID"].ToString();

TxtAd.Text = dr["OGRTAD"].ToString();

TxtSoyad.Text = dr["OGRTSOYAD"].ToString();

MskTC.Text = dr["OGRTTC"].ToString();

MskTelefon.Text = dr["OGRTTEL"].ToString();

TxtMail.Text = dr["OGRTMAIL"].ToString();

Cmbil.Text = dr["OGRTIL"].ToString();

Cmbilce.Text = dr["OGRTILCE"].ToString();

RchAdres.Text = dr["OGRTADRES"].ToString();

CmbBrans.Text = dr["OGRTBRANS"].ToString();

yeniyol = "C:\\Users\\hüsnü\\Desktop\\C# AnaOkulu\\AnaOkuluYönetimi\\AnaOkuluYönetimi" + "\\resimler\\" + dr["OGRTFOTO"].ToString();

pictureEdit.Image = Image.FromFile(yeniyol);

}

}

private void BtnGuncelle\_Click(object sender, EventArgs e)

{

// Ogretmenler tablosundaki bilgileri günceller

SqlCommand komut = new SqlCommand("Update TBL\_OGRETMENLER set OGRTAD=@p1,OGRTSOYAD=@p2,OGRTTC=@p3,OGRTTEL=@p4,OGRTMAIL=@p5,OGRTIL=@p6,OGRTILCE=@p7,OGRTADRES=@p8,OGRTBRANS=@p9,OGRTFOTO=@p10 where OGRTID=@p11",bgl.baglanti());

komut.Parameters.AddWithValue("@p1", TxtAd.Text);

komut.Parameters.AddWithValue("@p2", TxtSoyad.Text);

komut.Parameters.AddWithValue("@p3", MskTC.Text);

komut.Parameters.AddWithValue("@p4", MskTelefon.Text);

komut.Parameters.AddWithValue("@p5", TxtMail.Text);

komut.Parameters.AddWithValue("@p6", Cmbil.Text);

komut.Parameters.AddWithValue("@p7", Cmbilce.Text);

komut.Parameters.AddWithValue("@p8", RchAdres.Text);

komut.Parameters.AddWithValue("@p9", CmbBrans.Text);

komut.Parameters.AddWithValue("@p10", Path.GetFileName(yeniyol));

komut.Parameters.AddWithValue("@p11", TxtID.Text);

komut.ExecuteNonQuery();

bgl.baglanti().Close();

MessageBox.Show("Personel Güncellendi", "Bilgi", MessageBoxButtons.OK, MessageBoxIcon.Information);

listele();

}

private void BtnSil\_Click(object sender, EventArgs e)

{

// İd ye göre silme işlemi

SqlCommand komut = new SqlCommand("Delete from TBL\_OGRETMENLER where OGRTID=@p1", bgl.baglanti());

komut.Parameters.AddWithValue("@p1", TxtID.Text);

komut.ExecuteNonQuery();

bgl.baglanti().Close();

MessageBox.Show("Personel Silindi", "Bilgi", MessageBoxButtons.OK, MessageBoxIcon.Warning);

listele();

}

private void BtnTemizle\_Click(object sender, EventArgs e)

{

temizle();

}

public string yeniyol;

private void BtnResimSec\_Click(object sender, EventArgs e)

{

OpenFileDialog dosya = new OpenFileDialog();

dosya.Filter = "Resim dosyası |\*.jpg;\*.png;\*.nef |Tüm Dosyalar |\*.\*";

dosya.ShowDialog();

string dosyayolu = dosya.FileName;

yeniyol = "C:\\Users\\hüsnü\\Desktop\\C# AnaOkulu\\AnaOkuluYönetimi\\AnaOkuluYönetimi" + "\\resimler\\" + Guid.NewGuid().ToString() + ".jpg";

File.Copy(dosyayolu, yeniyol);

pictureEdit.Image = Image.FromFile(yeniyol);

}

private void gridControl1\_DoubleClick(object sender, EventArgs e)

{

DataRow dr = gridView1.GetDataRow(gridView1.FocusedRowHandle);

FrmOgretmenDataGrıpView frm = new FrmOgretmenDataGrıpView();

if (dr != null)

{

frm.ıd = dr["OGRTID"].ToString();

frm.ad = dr["OGRTAD"].ToString();

frm.soyad = dr["OGRTSOYAD"].ToString();

frm.tc = dr["OGRTTC"].ToString();

frm.brans = dr["OGRTBRANS"].ToString();

frm.telefon= dr["OGRTTEL"].ToString();

frm.uzantı = "C:\\Users\\hüsnü\\Desktop\\C# AnaOkulu\\AnaOkuluYönetimi\\AnaOkuluYönetimi" + "\\resimler\\" + dr["OGRTFOTO"].ToString();

}

frm.Show();

}

}

}

**2.7.2.2. Yönetici Ekranı Öğrenciler Formu Kodları**

**using System;**

**using System.Collections.Generic;**

**using System.ComponentModel;**

**using System.Data;**

**using System.Drawing;**

**using System.Linq;**

**using System.Text;**

**using System.Threading.Tasks;**

**using System.Windows.Forms;**

**using System.Data.SqlClient;**

**using System.IO;**

**namespace AnaOkuluYönetimi**

**{**

**public partial class FrmOgrencıler : Form**

**{**

**public FrmOgrencıler()**

**{**

**InitializeComponent();**

**}**

**sqlbaglantisi bgl = new sqlbaglantisi();**

**void listele()**

**{**

**// mibik mucitler**

**DataTable dt1 = new DataTable();**

**SqlDataAdapter da1 = new SqlDataAdapter("Execute MınıkMUcıtler", bgl.baglanti());**

**da1.Fill(dt1);**

**GrdCtrl5.DataSource = dt1;**

**//mink eller**

**DataTable dt2 = new DataTable();**

**SqlDataAdapter da2 = new SqlDataAdapter("Execute MınıkEller", bgl.baglanti());**

**da2.Fill(dt2);**

**GrdCtrl6.DataSource = dt2;**

**//minik kelebekler**

**DataTable dt3= new DataTable();**

**SqlDataAdapter da3 = new SqlDataAdapter("Execute MınıkKelebekler", bgl.baglanti());**

**da3.Fill(dt3);**

**GrdCtrl7.DataSource = dt3;**

**//minik dahiler**

**DataTable dt4 = new DataTable();**

**SqlDataAdapter da4 = new SqlDataAdapter("Execute MınıkDahıler", bgl.baglanti());**

**da4.Fill(dt4);**

**GrdCtrl8.DataSource = dt4;**

**}**

**void velilistesi()**

**{**

**DataTable dt = new DataTable();**

**SqlDataAdapter da = new SqlDataAdapter("select VELIID,(VELIANNE+' | '+VELIBABA) as 'VELI ANNE BABA' from TBL\_VELILER", bgl.baglanti());**

**da.Fill(dt);**

**lookUpEdit1.Properties.ValueMember = "VELIID";**

**lookUpEdit1.Properties.DisplayMember = "VELI ANNE BABA";**

**lookUpEdit1.Properties.DataSource = dt;**

**}**

**void sehirekle()**

**{**

**// veritabanındaki illeri cmb ekler**

**SqlCommand komut = new SqlCommand("select \* from TBL\_ILLER", bgl.baglanti());**

**SqlDataReader dr = komut.ExecuteReader();**

**while (dr.Read())**

**{**

**Cmbil.Properties.Items.Add(dr[1]);**

**}**

**bgl.baglanti().Close();**

**}**

**void temizle()**

**{**

**TxtID.Text = "";**

**TxtAd.Text = "";**

**TxtSoyad.Text = "";**

**MskOgrencıNo.Text = "";**

**MskTC.Text = "";**

**RdBtnErkek.Text = "";**

**RdBtnKız.Text = "";**

**CmbSinif.Text = "";**

**Cmbil.Text = "";**

**Cmbilce.Text = "";**

**dateEdit1.Text = "";**

**RchAdres.Text = "";**

**pictureEdit1.Text = "";**

**}**

**private void FrmOgrencıler\_Load(object sender, EventArgs e)**

**{**

**listele();**

**sehirekle();**

**temizle();**

**velilistesi();**

**}**

**private void Cmbil\_SelectedIndexChanged(object sender, EventArgs e)**

**{**

**// scilern ilin ilcelerini ekler ve il degiştiginde ilcelerde degişir**

**Cmbilce.Properties.Items.Clear();**

**Cmbilce.Text = "";**

**SqlCommand komut = new SqlCommand("select \* from TBL\_ILCELER where sehir=@p1", bgl.baglanti());**

**komut.Parameters.AddWithValue("@p1", Cmbil.SelectedIndex + 1);**

**SqlDataReader dr = komut.ExecuteReader();**

**while (dr.Read())**

**{**

**Cmbilce.Properties.Items.Add(dr[1]);**

**}**

**bgl.baglanti().Close();**

**}**

**private void gridView0\_FocusedRowObjectChanged(object sender, DevExpress.XtraGrid.Views.Base.FocusedRowObjectChangedEventArgs e)**

**{**

**DataRow dr = gridView0.GetDataRow(gridView0.FocusedRowHandle);**

**if (dr != null)**

**{**

**TxtID.Text = dr["OGRID"].ToString();**

**TxtAd.Text = dr["OGRAD"].ToString();**

**TxtSoyad.Text = dr["OGRSOYAD"].ToString();**

**MskTC.Text = dr["OGRTC"].ToString();**

**MskOgrencıNo.Text = dr["OGRNO"].ToString();**

**CmbSinif.Text = dr["OGRSINIF"].ToString();**

**if (dr["OGRCINSIYET"].ToString() == "E")**

**{**

**RdBtnErkek.Checked = true;**

**}**

**else**

**{**

**RdBtnKız.Checked = true;**

**}**

**Cmbil.Text = dr["OGRIL"].ToString();**

**Cmbilce.Text = dr["OGRILCE"].ToString();**

**dateEdit1.Text = dr["OGRDOGTAR"].ToString();**

**RchAdres.Text = dr["OGRADRES"].ToString();**

**yeniyol = "C:\\Users\\hüsnü\\Desktop\\C# AnaOkulu\\AnaOkuluYönetimi\\AnaOkuluYönetimi" + "\\resimler\\" + dr["OGRFOTO"].ToString();**

**pictureEdit1.Image = Image.FromFile(yeniyol);**

**lookUpEdit1.Text = dr["VELIANNEBABA"].ToString();**

**}**

**}**

**private void gridView1\_FocusedRowObjectChanged(object sender, DevExpress.XtraGrid.Views.Base.FocusedRowObjectChangedEventArgs e)**

**{**

**DataRow dr = gridView1.GetDataRow(gridView1.FocusedRowHandle);**

**if (dr != null)**

**{**

**TxtID.Text = dr["OGRID"].ToString();**

**TxtAd.Text = dr["OGRAD"].ToString();**

**TxtSoyad.Text = dr["OGRSOYAD"].ToString();**

**MskTC.Text = dr["OGRTC"].ToString();**

**MskOgrencıNo.Text = dr["OGRNO"].ToString();**

**CmbSinif.Text = dr["OGRSINIF"].ToString();**

**if (dr["OGRCINSIYET"].ToString() == "E")**

**{**

**RdBtnErkek.Checked = true;**

**}**

**else**

**{**

**RdBtnKız.Checked = true;**

**}**

**Cmbil.Text = dr["OGRIL"].ToString();**

**Cmbilce.Text = dr["OGRILCE"].ToString();**

**dateEdit1.Text = dr["OGRDOGTAR"].ToString();**

**RchAdres.Text = dr["OGRADRES"].ToString();**

**yeniyol = "C:\\Users\\hüsnü\\Desktop\\C# AnaOkulu\\AnaOkuluYönetimi\\AnaOkuluYönetimi" + "\\resimler\\" + dr["OGRFOTO"].ToString();**

**pictureEdit1.Image = Image.FromFile(yeniyol);**

**lookUpEdit1.Text = dr["VELIANNEBABA"].ToString();**

**}**

**}**

**private void gridView3\_FocusedRowObjectChanged(object sender, DevExpress.XtraGrid.Views.Base.FocusedRowObjectChangedEventArgs e)**

**{**

**DataRow dr = gridView3.GetDataRow(gridView3.FocusedRowHandle);**

**if (dr != null)**

**{**

**TxtID.Text = dr["OGRID"].ToString();**

**TxtAd.Text = dr["OGRAD"].ToString();**

**TxtSoyad.Text = dr["OGRSOYAD"].ToString();**

**MskTC.Text = dr["OGRTC"].ToString();**

**MskOgrencıNo.Text = dr["OGRNO"].ToString();**

**CmbSinif.Text = dr["OGRSINIF"].ToString();**

**if (dr["OGRCINSIYET"].ToString() == "E")**

**{**

**RdBtnErkek.Checked = true;**

**}**

**else**

**{**

**RdBtnKız.Checked = true;**

**}**

**Cmbil.Text = dr["OGRIL"].ToString();**

**Cmbilce.Text = dr["OGRILCE"].ToString();**

**dateEdit1.Text = dr["OGRDOGTAR"].ToString();**

**RchAdres.Text = dr["OGRADRES"].ToString();**

**yeniyol = "C:\\Users\\hüsnü\\Desktop\\C# AnaOkulu\\AnaOkuluYönetimi\\AnaOkuluYönetimi" + "\\resimler\\" + dr["OGRFOTO"].ToString();**

**pictureEdit1.Image = Image.FromFile(yeniyol);**

**lookUpEdit1.Text = dr["VELIANNEBABA"].ToString();**

**}**

**}**

**private void gridView4\_FocusedRowObjectChanged(object sender, DevExpress.XtraGrid.Views.Base.FocusedRowObjectChangedEventArgs e)**

**{**

**DataRow dr = gridView4.GetDataRow(gridView4.FocusedRowHandle);**

**if (dr != null)**

**{**

**TxtID.Text = dr["OGRID"].ToString();**

**TxtAd.Text = dr["OGRAD"].ToString();**

**TxtSoyad.Text = dr["OGRSOYAD"].ToString();**

**MskTC.Text = dr["OGRTC"].ToString();**

**MskOgrencıNo.Text = dr["OGRNO"].ToString();**

**CmbSinif.Text = dr["OGRSINIF"].ToString();**

**if (dr["OGRCINSIYET"].ToString() == "E")**

**{**

**RdBtnErkek.Checked = true;**

**}**

**else**

**{**

**RdBtnKız.Checked = true;**

**}**

**Cmbil.Text = dr["OGRIL"].ToString();**

**Cmbilce.Text = dr["OGRILCE"].ToString();**

**dateEdit1.Text = dr["OGRDOGTAR"].ToString();**

**RchAdres.Text = dr["OGRADRES"].ToString();**

**yeniyol = "C:\\Users\\hüsnü\\Desktop\\C# AnaOkulu\\AnaOkuluYönetimi\\AnaOkuluYönetimi" + "\\resimler\\" + dr["OGRFOTO"].ToString();**

**pictureEdit1.Image = Image.FromFile(yeniyol);**

**lookUpEdit1.Text = dr["VELIANNEBABA"].ToString();**

**}**

**}**

**public string cinsiyet;**

**private void BtnKaydet\_Click(object sender, EventArgs e)**

**{**

**// veritabanına ögrencılerı kaydeden kod**

**SqlCommand komut = new SqlCommand("Insert into TBL\_OGRENCILER (OGRAD,OGRSOYAD,OGRNO,OGRSINIF,OGRDOGTAR,OGRCINSIYET,OGRTC,OGRIL,OGRILCE,OGRADRES,OGRFOTO,OGRVELIID) values (@p1,@p2,@p3,@p4,@p5,@p6,@p7,@p8,@p9,@p10,@p11,@p12)",bgl.baglanti());**

**komut.Parameters.AddWithValue("@p1", TxtAd.Text);**

**komut.Parameters.AddWithValue("@p2",TxtSoyad.Text);**

**komut.Parameters.AddWithValue("@p3",MskOgrencıNo.Text);**

**komut.Parameters.AddWithValue("@p4",CmbSinif.Text);**

**komut.Parameters.AddWithValue("@p5",dateEdit1.Text);**

**if (RdBtnErkek.Checked==true)**

**{**

**komut.Parameters.AddWithValue("@p6",cinsiyet="E");**

**}**

**else**

**{**

**komut.Parameters.AddWithValue("@p6",cinsiyet="K");**

**}**

**komut.Parameters.AddWithValue("@p7",MskTC.Text);**

**komut.Parameters.AddWithValue("@p8",Cmbil.Text);**

**komut.Parameters.AddWithValue("@p9",Cmbilce.Text);**

**komut.Parameters.AddWithValue("@p10",RchAdres.Text);**

**komut.Parameters.AddWithValue("@p11", Path.GetFileName(yeniyol));**

**komut.Parameters.AddWithValue("@p12", lookUpEdit1.EditValue);**

**komut.ExecuteNonQuery();**

**bgl.baglanti().Close();**

**MessageBox.Show("Ögrenci Bilgileri Kaydedildi", "Bilgi", MessageBoxButtons.OK, MessageBoxIcon.Information);**

**listele();**

**}**

**public string yeniyol;**

**private void BtnResimSec\_Click(object sender, EventArgs e)**

**{**

**OpenFileDialog dosya = new OpenFileDialog();**

**dosya.Filter = "Resim dosyası |\*.jpg;\*.png;\*.nef |Tüm Dosyalar |\*.\*";**

**dosya.ShowDialog();**

**string dosyayolu = dosya.FileName;**

**yeniyol = "C:\\Users\\hüsnü\\Desktop\\C# AnaOkulu\\AnaOkuluYönetimi\\AnaOkuluYönetimi" + "\\resimler\\"+Guid.NewGuid().ToString()+".jpg";**

**File.Copy(dosyayolu, yeniyol);**

**pictureEdit1.Image = Image.FromFile(yeniyol);**

**}**

**private void BtnGuncelle\_Click(object sender, EventArgs e)**

**{**

**SqlCommand komut = new SqlCommand("Update TBL\_OGRENCILER set OGRAD=@p1,OGRSOYAD=@p2,OGRNO=@p3,OGRSINIF=@p4,OGRDOGTAR=@p5,OGRCINSIYET=@p6,OGRTC=@p7,OGRIL=@p8,OGRILCE=@p9,OGRADRES=@p10,OGRFOTO=@p11,OGRVELIID=@p12 where OGRID=@p13", bgl.baglanti());**

**komut.Parameters.AddWithValue("@p1", TxtAd.Text);**

**komut.Parameters.AddWithValue("@p2", TxtSoyad.Text);**

**komut.Parameters.AddWithValue("@p3", MskOgrencıNo.Text);**

**komut.Parameters.AddWithValue("@p4", CmbSinif.Text);**

**komut.Parameters.AddWithValue("@p5", dateEdit1.Text);**

**if (RdBtnErkek.Checked == true)**

**{**

**komut.Parameters.AddWithValue("@p6", cinsiyet = "E");**

**}**

**else**

**{**

**komut.Parameters.AddWithValue("@p6", cinsiyet = "K");**

**}**

**komut.Parameters.AddWithValue("@p7", MskTC.Text);**

**komut.Parameters.AddWithValue("@p8", Cmbil.Text);**

**komut.Parameters.AddWithValue("@p9", Cmbilce.Text);**

**komut.Parameters.AddWithValue("@p10", RchAdres.Text);**

**komut.Parameters.AddWithValue("@p11", Path.GetFileName(yeniyol));**

**komut.Parameters.AddWithValue("@p13", TxtID.Text);**

**komut.Parameters.AddWithValue("@p12", lookUpEdit1.EditValue);**

**komut.ExecuteNonQuery();**

**bgl.baglanti().Close();**

**MessageBox.Show("Ögrenci Bilgileri Güncellendi.", "Bilgi", MessageBoxButtons.OK, MessageBoxIcon.Information);**

**listele();**

**}**

**private void BtnSil\_Click(object sender, EventArgs e)**

**{**

**SqlCommand komut = new SqlCommand("Delete from TBL\_OGRENCILER where OGRID=@p1", bgl.baglanti());**

**komut.Parameters.AddWithValue("@p1", TxtID.Text);**

**komut.ExecuteNonQuery();**

**bgl.baglanti().Close();**

**MessageBox.Show("Ögrenci Bilgileri Silindi.", "Uyarı", MessageBoxButtons.OK, MessageBoxIcon.Warning);**

**listele();**

**}**

**private void BtnTemizle\_Click(object sender, EventArgs e)**

**{**

**temizle();**

**}**

**private void gridView0\_DoubleClick(object sender, EventArgs e)**

**{**

**// Burası grıd vıewsde kişi üzerine çift tıklandıgında öğrenci bilgileri ayrı bir formda gösterecek.**

**DataRow dr = gridView0.GetDataRow(gridView0.FocusedRowHandle);**

**FrmOgrencıDataGrıdVıew frm = new FrmOgrencıDataGrıdVıew();**

**if (dr!=null)**

**{**

**frm.ıd = dr["OGRID"].ToString();**

**frm.ad = dr["OGRAD"].ToString();**

**frm.soyad = dr["OGRSOYAD"].ToString();**

**frm.tc = dr["OGRTC"].ToString();**

**frm.sınıf = dr["OGRSINIF"].ToString();**

**frm.dogtarihi = dr["OGRDOGTAR"].ToString();**

**frm.uzantı= "C:\\Users\\hüsnü\\Desktop\\C# AnaOkulu\\AnaOkuluYönetimi\\AnaOkuluYönetimi" + "\\resimler\\" + dr["OGRFOTO"].ToString();**

**}**

**frm.Show();**

**}**

**private void gridView1\_DoubleClick(object sender, EventArgs e)**

**{**

**DataRow dr = gridView1.GetDataRow(gridView1.FocusedRowHandle);**

**FrmOgrencıDataGrıdVıew frm = new FrmOgrencıDataGrıdVıew();**

**if (dr != null)**

**{**

**frm.ıd = dr["OGRID"].ToString();**

**frm.ad = dr["OGRAD"].ToString();**

**frm.soyad = dr["OGRSOYAD"].ToString();**

**frm.tc = dr["OGRTC"].ToString();**

**frm.sınıf = dr["OGRSINIF"].ToString();**

**frm.dogtarihi = dr["OGRDOGTAR"].ToString();**

**frm.uzantı = "C:\\Users\\hüsnü\\Desktop\\C# AnaOkulu\\AnaOkuluYönetimi\\AnaOkuluYönetimi" + "\\resimler\\" + dr["OGRFOTO"].ToString();**

**}**

**frm.Show();**

**}**

**private void gridView3\_DoubleClick(object sender, EventArgs e)**

**{**

**DataRow dr = gridView3.GetDataRow(gridView3.FocusedRowHandle);**

**FrmOgrencıDataGrıdVıew frm = new FrmOgrencıDataGrıdVıew();**

**if (dr != null)**

**{**

**frm.ıd = dr["OGRID"].ToString();**

**frm.ad = dr["OGRAD"].ToString();**

**frm.soyad = dr["OGRSOYAD"].ToString();**

**frm.tc = dr["OGRTC"].ToString();**

**frm.sınıf = dr["OGRSINIF"].ToString();**

**frm.dogtarihi = dr["OGRDOGTAR"].ToString();**

**frm.uzantı = "C:\\Users\\hüsnü\\Desktop\\C# AnaOkulu\\AnaOkuluYönetimi\\AnaOkuluYönetimi" + "\\resimler\\" + dr["OGRFOTO"].ToString();**

**}**

**frm.Show();**

**}**

**private void gridView4\_DoubleClick(object sender, EventArgs e)**

**{**

**DataRow dr = gridView4.GetDataRow(gridView4.FocusedRowHandle);**

**FrmOgrencıDataGrıdVıew frm = new FrmOgrencıDataGrıdVıew();**

**if (dr != null)**

**{**

**frm.ıd = dr["OGRID"].ToString();**

**frm.ad = dr["OGRAD"].ToString();**

**frm.soyad = dr["OGRSOYAD"].ToString();**

**frm.tc = dr["OGRTC"].ToString();**

**frm.sınıf = dr["OGRSINIF"].ToString();**

**frm.dogtarihi = dr["OGRDOGTAR"].ToString();**

**frm.uzantı = "C:\\Users\\hüsnü\\Desktop\\C# AnaOkulu\\AnaOkuluYönetimi\\AnaOkuluYönetimi" + "\\resimler\\" + dr["OGRFOTO"].ToString();**

**}**

**frm.Show();**

**}**

**}**

**}**

**2.7.2.3. Yönetici Ekranı Veliler Formu Kodları**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace AnaOkuluYönetimi

{

public partial class FrmVeliler : Form

{

public FrmVeliler()

{

InitializeComponent();

}

DbOkulEntities db = new DbOkulEntities();

void listele()

{

// bu tbl ogrencıler ıle ilişkilendirdigimiz için tbl ögrencıler formunu göstermemek.

var query = from item in db.TBL\_VELILER

select new { item.VELIID, item.VELIANNE, item.VELIBABA, item.VELITEL1, item.VELITEL2, item.VELIMAIL };

gridControl1.DataSource = query.ToList();

// gridView1.Columns[6].Visible = false;

}

void temizle()

{

TxtID.Text = "";

TxtAnneAd.Text = "";

TxtBabaAd.Text = "";

MskTelefon1.Text = "";

MskTelefon2.Text = "";

TxtMail.Text = "";

}

private void FrmVeliler\_Load(object sender, EventArgs e)

{

listele();

temizle();

}

private void BtnKaydet\_Click(object sender, EventArgs e)

{

TBL\_VELILER veli = new TBL\_VELILER();

veli.VELIANNE = TxtAnneAd.Text;

veli.VELIBABA = TxtBabaAd.Text;

veli.VELITEL1 = MskTelefon1.Text;

veli.VELITEL2 = MskTelefon2.Text;

veli.VELIMAIL = TxtMail.Text;

db.TBL\_VELILER.Add(veli);

db.SaveChanges();

MessageBox.Show("Veli Kaydedildi.", "Bilgi", MessageBoxButtons.OK, MessageBoxIcon.Information);

listele();

temizle();

}

private void gridView1\_FocusedRowObjectChanged(object sender, DevExpress.XtraGrid.Views.Base.FocusedRowObjectChangedEventArgs e)

{

// otomatik olarak tabcontrol la bilgileri getiri ve tıklandıgında değiştirir

TxtID.Text = gridView1.GetRowCellValue(gridView1.FocusedRowHandle, "VELIID").ToString();

TxtAnneAd.Text= gridView1.GetRowCellValue(gridView1.FocusedRowHandle, "VELIANNE").ToString();

TxtBabaAd.Text= gridView1.GetRowCellValue(gridView1.FocusedRowHandle, "VELIBABA").ToString();

MskTelefon1.Text= gridView1.GetRowCellValue(gridView1.FocusedRowHandle, "VELITEL1").ToString();

MskTelefon2.Text= gridView1.GetRowCellValue(gridView1.FocusedRowHandle, "VELITEL2").ToString();

TxtMail.Text= gridView1.GetRowCellValue(gridView1.FocusedRowHandle, "VELIMAIL").ToString();

}

private void BtnGuncelle\_Click(object sender, EventArgs e)

{

// güncelleme işlemi

int id = Convert.ToInt32(gridView1.GetRowCellValue(gridView1.FocusedRowHandle, "VELIID").ToString());

var item = db.TBL\_VELILER.Find(id);

item.VELIANNE = TxtAnneAd.Text;

item.VELIBABA = TxtBabaAd.Text;

item.VELITEL1 = MskTelefon1.Text;

item.VELITEL2 = MskTelefon2.Text;

item.VELIMAIL = TxtMail.Text;

MessageBox.Show("Veli Güncellendi.", "Bilgi", MessageBoxButtons.OK, MessageBoxIcon.Information);

db.SaveChanges();

listele();

temizle();

}

private void BtnSil\_Click(object sender, EventArgs e)

{

// silme işlemi

int id = Convert.ToInt32(gridView1.GetRowCellValue(gridView1.FocusedRowHandle, "VELIID").ToString());

var item = db.TBL\_VELILER.Find(id);

db.TBL\_VELILER.Remove(item);

MessageBox.Show("Veli Silindi.", "Uyarı", MessageBoxButtons.OK, MessageBoxIcon.Warning);

db.SaveChanges();

listele();

}

private void BtnTemizle\_Click(object sender, EventArgs e)

{

temizle();

}

}

}

**2.7.2.4. Yönetici Ekranı Davranışlar Formu Kodları**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.Sql;

using System.Data.SqlClient;

namespace AnaOkuluYönetimi

{

public partial class FrmDavranıslar : Form

{

public FrmDavranıslar()

{

InitializeComponent();

}

sqlbaglantisi bgl = new sqlbaglantisi();

private void FrmDavranıslar\_Load(object sender, EventArgs e)

{

davraisListele();

}

void davraisListele()

{

//Veritabından Davranışları tablosunu getirir

DataTable dt = new DataTable();

SqlDataAdapter da = new SqlDataAdapter("Select \* from TBL\_DAVRANISLAR", bgl.baglanti());

da.Fill(dt);

gridControl1.DataSource = dt;

}

}

}

**2.7.2.5. Yönetici Ekranı İstatistikler Formu Kodları**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.Sql;

using System.Data.SqlClient;

namespace AnaOkuluYönetimi

{

public partial class FrmIstatislik : Form

{

public FrmIstatislik()

{

InitializeComponent();

}

sqlbaglantisi bgl = new sqlbaglantisi();

private void FrmIstatislik\_Load(object sender, EventArgs e)

{

ıstatislikisListele();

}

void ıstatislikisListele()

{

//Veritabından istatislik tablosunu getirir

DataTable dt = new DataTable();

SqlDataAdapter da = new SqlDataAdapter("Select \* from TBL\_OGRTISTATISLIK", bgl.baglanti());

da.Fill(dt);

gridControl1.DataSource = dt;

}

}

}

**2.7.2.6. Yönetici Ekranı Faturalar Formu Kodları**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.SqlClient;

using System.Data.Sql;

namespace AnaOkuluYönetimi

{

public partial class FrmFaturalar : Form

{

public FrmFaturalar()

{

InitializeComponent();

}

sqlbaglantisi bgl = new sqlbaglantisi();

private void veriler()

{

listView1.Items.Clear();

SqlCommand komut = new SqlCommand("Select \* from TBL\_GIDERLER", bgl.baglanti());

SqlDataReader oku = komut.ExecuteReader();

while (oku.Read())

{

ListViewItem ekle = new ListViewItem();

ekle.Text = oku["Elektrik"].ToString();

ekle.SubItems.Add(oku["Su"].ToString());

ekle.SubItems.Add(oku["Dogalgaz"].ToString());

ekle.SubItems.Add(oku["Mutfak"].ToString());

ekle.SubItems.Add(oku["Okul"].ToString());

ekle.SubItems.Add(oku["Stok"].ToString());

listView1.Items.Add(ekle);

}

bgl.baglanti().Close();

}

private void FrmFaturalar\_Load(object sender, EventArgs e)

{

veriler();

}

private void BtnFaturaKaydet\_Click(object sender, EventArgs e)

{

SqlCommand komut = new SqlCommand("insert into TBL\_GIDERLER(Elektrik,Su,Dogalgaz,Mutfak,Okul,Stok) values ('" + TxtElektrik.Text + "','" + TxtSu.Text + "','" + TxtDogalgaz.Text + "','"+TxtMutfak.Text+"','"+TxtOkul.Text+"','"+TxtStok.Text+"')", bgl.baglanti());

komut.ExecuteNonQuery();

bgl.baglanti().Close();

veriler();

MessageBox.Show("YEni Faturalar Kayıt Edildi", "Bilgi", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

}

}

**2.7.2.7. Yönetici Ekranı Giderler Formu Kodları**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.Sql;

using System.Data.SqlClient;

namespace AnaOkuluYönetimi

{

public partial class FrmGiderler : Form

{

public FrmGiderler()

{

InitializeComponent();

}

sqlbaglantisi bgl = new sqlbaglantisi();

void listele()

{

//Veritabından öğretmenler tablosunu getirir

DataTable dt = new DataTable();

SqlDataAdapter da = new SqlDataAdapter("Select \* from TBL\_OGRETMENLER", bgl.baglanti());

da.Fill(dt);

gridControl1.DataSource = dt;

}

private void FrmGiderler\_Load(object sender, EventArgs e)

{

listele();

//Elektrik Gideri

SqlCommand komut4 = new SqlCommand("select sum(Elektrik) as toplam4 from TBL\_GIDERLER", bgl.baglanti());

SqlDataReader oku4 = komut4.ExecuteReader();

while (oku4.Read())

{

LblElektrik.Text = oku4["toplam4"].ToString();

}

bgl.baglanti().Close();

// Su giideri

SqlCommand komut5 = new SqlCommand("select sum(Su) as toplam5 from TBL\_GIDERLER", bgl.baglanti());

SqlDataReader oku5 = komut5.ExecuteReader();

while (oku5.Read())

{

LblSu.Text = oku5["toplam5"].ToString();

}

bgl.baglanti().Close();

// Dogalgaz gideri

SqlCommand komut6 = new SqlCommand("select sum(Dogalgaz) as toplam6 from TBL\_GIDERLER", bgl.baglanti());

SqlDataReader oku6 = komut6.ExecuteReader();

while (oku6.Read())

{

LblDogalgaz.Text = oku6["toplam6"].ToString();

}

bgl.baglanti().Close();

// Mutfak gideri

SqlCommand komut7 = new SqlCommand("select sum(Mutfak) as toplam7 from TBL\_GIDERLER", bgl.baglanti());

SqlDataReader oku7 = komut7.ExecuteReader();

while (oku7.Read())

{

LblMutfak.Text = oku7["toplam7"].ToString();

}

bgl.baglanti().Close();

//Okul gideri

SqlCommand komut8 = new SqlCommand("select sum(Okul) as toplam8 from TBL\_GIDERLER", bgl.baglanti());

SqlDataReader oku8 = komut8.ExecuteReader();

while (oku8.Read())

{

LblOkulMasrafı.Text = oku8["toplam8"].ToString();

}

bgl.baglanti().Close();

// Stok Gider

SqlCommand komut9 = new SqlCommand("select sum(Stok) as toplam9 from TBL\_GIDERLER", bgl.baglanti());

SqlDataReader oku9 = komut9.ExecuteReader();

while (oku9.Read())

{

LblStok.Text = oku9["toplam9"].ToString();

}

bgl.baglanti().Close();

}

private void BtnHesapla\_Click(object sender, EventArgs e)

{

int pernonel;

pernonel = Convert.ToInt32(TxtÖğretmenSayısı.Text);

LblOgrtTutar.Text = (pernonel \* 4000).ToString();

int sonuc;

sonuc = (Convert.ToInt32(LblElektrik.Text) + Convert.ToInt32(LblSu.Text) + Convert.ToInt32(LblDogalgaz.Text) + Convert.ToInt32(LblMutfak.Text) + Convert.ToInt32(LblOkulMasrafı.Text) + Convert.ToInt32(LblStok.Text) );

LblSonuc.Text = sonuc.ToString();

MessageBox.Show("Toplam Sonuc:" +sonuc,"Bilgi", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

private void gridControl1\_DoubleClick(object sender, EventArgs e)

{

DataRow dr = gridView1.GetDataRow(gridView1.FocusedRowHandle);

FrmOgretmenDataGrıpView frm = new FrmOgretmenDataGrıpView();

if (dr != null)

{

frm.ıd = dr["OGRTID"].ToString();

frm.ad = dr["OGRTAD"].ToString();

frm.soyad = dr["OGRTSOYAD"].ToString();

frm.tc = dr["OGRTTC"].ToString();

frm.brans = dr["OGRTBRANS"].ToString();

frm.telefon = dr["OGRTTEL"].ToString();

frm.uzantı = "C:\\Users\\hüsnü\\Desktop\\C# AnaOkulu\\AnaOkuluYönetimi\\AnaOkuluYönetimi" + "\\resimler\\" + dr["OGRTFOTO"].ToString();

}

frm.Show();

}

}

}

**2.7.2.8. Yönetici Ekranı Kasa Formu Kodları**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.Sql;

using System.Data.SqlClient;

namespace AnaOkuluYönetimi

{

public partial class FrmKasa : Form

{

public FrmKasa()

{

InitializeComponent();

}

sqlbaglantisi bgl = new sqlbaglantisi();

void listele()

{

//Veritabından öğretmenler tablosunu getirir

DataTable dt = new DataTable();

SqlDataAdapter da = new SqlDataAdapter("Select \* from TBL\_OGRENCILER", bgl.baglanti());

da.Fill(dt);

gridControl1.DataSource = dt;

}

private void FrmKasa\_Load(object sender, EventArgs e)

{

listele();

}

private void BtnOgrHesapla\_Click(object sender, EventArgs e)

{

int ogrencı;

ogrencı = Convert.ToInt32(TxtOgrencı.Text);

LblSonuc.Text = (ogrencı \* 8500).ToString();

MessageBox.Show("Toplam:" + ogrencı \* 8500, "Bilgi", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

private void gridControl1\_DoubleClick(object sender, EventArgs e)

{

DataRow dr = gridView1.GetDataRow(gridView1.FocusedRowHandle);

FrmOgrencıDataGrıdVıew frm = new FrmOgrencıDataGrıdVıew();

if (dr != null)

{

frm.ıd = dr["OGRID"].ToString();

frm.ad = dr["OGRAD"].ToString();

frm.soyad = dr["OGRSOYAD"].ToString();

frm.tc = dr["OGRTC"].ToString();

frm.sınıf = dr["OGRSINIF"].ToString();

frm.dogtarihi = dr["OGRDOGTAR"].ToString();

frm.uzantı = "C:\\Users\\hüsnü\\Desktop\\C# AnaOkulu\\AnaOkuluYönetimi\\AnaOkuluYönetimi" + "\\resimler\\" + dr["OGRFOTO"].ToString();

}

frm.Show();

}

}

}

**2.7.2.9. Yönetici Ekranı Mesajlar Formu Kodları**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.Sql;

using System.Data.SqlClient;

namespace AnaOkuluYönetimi

{

public partial class FrmMesajlar : Form

{

public FrmMesajlar()

{

InitializeComponent();

}

sqlbaglantisi bgl = new sqlbaglantisi();

private void BtnMesajKaydet\_Click(object sender, EventArgs e)

{

SqlCommand komut = new SqlCommand("insert into TBL\_MESAJLAR (AdSoyad,Mesaj) values ('" + TxtAdSoyad.Text + "','" + RchTxtMesaj.Text + "')", bgl.baglanti());

komut.ExecuteNonQuery();

bgl.baglanti().Close();

verilerigoster();

MessageBox.Show("Mesaj Kaydedildi", "Bilgi", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

private void verilerigoster()

{

listView1.Items.Clear();

SqlCommand komut = new SqlCommand("select\*from TBL\_MESAJLAR", bgl.baglanti());

SqlDataReader oku = komut.ExecuteReader();

while (oku.Read())

{

ListViewItem ekle = new ListViewItem();

ekle.Text = oku["MesajID"].ToString();

ekle.SubItems.Add(oku["AdSoyad"].ToString());

ekle.SubItems.Add(oku["Mesaj"].ToString());

listView1.Items.Add(ekle);

}

bgl.baglanti().Close();

}

private void FrmMesajlar\_Load(object sender, EventArgs e)

{

verilerigoster();

}

int id = 0;

private void listView1\_DoubleClick(object sender, EventArgs e)

{

id = int.Parse(listView1.SelectedItems[0].SubItems[0].Text);

TxtAdSoyad.Text = listView1.SelectedItems[0].SubItems[1].Text;

RchTxtMesaj.Text = listView1.SelectedItems[0].SubItems[2].Text;

}

}

}

**2.7.2.10. Yönetici Ekranı Raporlar Formu Kodları**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.Sql;

using System.Data.SqlClient;

using System.IO;

namespace AnaOkuluYönetimi

{

public partial class FrmRaporlar : Form

{

public FrmRaporlar()

{

InitializeComponent();

}

sqlbaglantisi bgl = new sqlbaglantisi();

void listele()

{

//Veritabından öğretmenler tablosunu getirir

DataTable dt = new DataTable();

SqlDataAdapter da = new SqlDataAdapter("Select \* from TBL\_OGRETMENLER", bgl.baglanti());

da.Fill(dt);

gridControl1.DataSource = dt;

}

void ogrtsec()

{

// veritabanındaki öğretmneleri cmb ekler

SqlCommand komut = new SqlCommand("select \* from TBL\_OGRETMENLER", bgl.baglanti());

SqlDataReader dr = komut.ExecuteReader();

while (dr.Read())

{

comboBox1.Items.Add(dr[1]);

}

bgl.baglanti().Close();

}

private void FrmRaporlar\_Load(object sender, EventArgs e)

{

listele();

listelerapor();

ogrtsec();

}

public string yeniyol;

private void gridView1\_FocusedRowObjectChanged(object sender, DevExpress.XtraGrid.Views.Base.FocusedRowObjectChangedEventArgs e)

{

DataRow dr = gridView1.GetDataRow(gridView1.FocusedRowHandle);

if (dr != null)

{

TxtID.Text = dr["OGRTID"].ToString();

comboBox1.Text = dr["OGRTAD"].ToString();

TxtSoyad.Text = dr["OGRTSOYAD"].ToString();

MskTC.Text = dr["OGRTTC"].ToString();

yeniyol = "C:\\Users\\hüsnü\\Desktop\\C# AnaOkulu\\AnaOkuluYönetimi\\AnaOkuluYönetimi" + "\\resimler\\" + dr["OGRTFOTO"].ToString();

pictureEdit1.Image = Image.FromFile(yeniyol);

}

}

private void BtnRaporKaydet\_Click(object sender, EventArgs e)

{

SqlCommand komut = new SqlCommand("Insert into TBL\_RAPORLAR (RaporAd,RaporSoyad,RaporTC,RaporBasTarih,RaporFoto,RaporBitTarih) values (@p1,@p2,@p3,@p4,@p5,@p6)", bgl.baglanti());

komut.Parameters.AddWithValue("@p1", comboBox1.Text);

komut.Parameters.AddWithValue("@p2", TxtSoyad.Text);

komut.Parameters.AddWithValue("@p3", MskTC.Text);

komut.Parameters.AddWithValue("@p4", dateEdit1.Text);

komut.Parameters.AddWithValue("@p5", Path.GetFileName(yeniyol));

komut.Parameters.AddWithValue("@p6", dateEdit2.Text);

komut.ExecuteNonQuery();

bgl.baglanti().Close();

MessageBox.Show("Rapor Bilgileri Kaydedildi", "Bilgi", MessageBoxButtons.OK, MessageBoxIcon.Information);

listelerapor();

}

void listelerapor()

{

//Veritabından rapor tablosunu getirir

DataTable dt = new DataTable();

SqlDataAdapter da = new SqlDataAdapter("Select \* from TBL\_RAPORLAR", bgl.baglanti());

da.Fill(dt);

gridControl2.DataSource = dt;

}

private void comboBox1\_SelectedIndexChanged(object sender, EventArgs e)

{

SqlCommand komut = new SqlCommand("Select OGRTID,OGRTSOYAD,OGRTTC From TBL\_OGRETMENLER Where OGRTAD= '" + comboBox1.Text + "'", bgl.baglanti());

SqlDataReader oku = komut.ExecuteReader();

while (oku.Read())

{

TxtID.Text = oku["OGRTID"].ToString();

TxtSoyad.Text = oku["OGRTSOYAD"].ToString();

MskTC.Text = oku["OGRTTC"].ToString();

}

bgl.baglanti().Close();

}

private void gridControl1\_DoubleClick(object sender, EventArgs e)

{

DataRow dr = gridView1.GetDataRow(gridView1.FocusedRowHandle);

FrmOgretmenDataGrıpView frm = new FrmOgretmenDataGrıpView();

if (dr != null)

{

frm.ıd = dr["OGRTID"].ToString();

frm.ad = dr["OGRTAD"].ToString();

frm.soyad = dr["OGRTSOYAD"].ToString();

frm.tc = dr["OGRTTC"].ToString();

frm.brans = dr["OGRTBRANS"].ToString();

frm.telefon = dr["OGRTTEL"].ToString();

frm.uzantı = "C:\\Users\\hüsnü\\Desktop\\C# AnaOkulu\\AnaOkuluYönetimi\\AnaOkuluYönetimi" + "\\resimler\\" + dr["OGRTFOTO"].ToString();

}

frm.Show();

}

}

}

**2.7.2.11. Yönetici Ekranı Ayarlar Formu Kodları**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.SqlClient;

using System.IO;

namespace AnaOkuluYönetimi

{

public partial class FrmAyarlar : Form

{

public FrmAyarlar()

{

InitializeComponent();

}

sqlbaglantisi bgl = new sqlbaglantisi();

DbOkulEntities db = new DbOkulEntities();

void ogrtlistele()

{

// ögretmen listeleme

gridControl1.DataSource = db.AyarlarOgretmenler();

}

void ogrlistele()

{

//öğrenci listeleme

gridControl2.DataSource = db.AyarlarOgrencıler();

}

void ogretmenlistesi()

{

var deger1 = from item in db.TBL\_OGRETMENLER

select new

{

OGRTID = item.OGRTID,

OGRTADSOYAD = item.OGRTAD + " " + item.OGRTSOYAD,

OGRTBRANS = item.OGRTBRANS,

};

lookUpEdit1.Properties.ValueMember = "OGRTID";

lookUpEdit1.Properties.DisplayMember = "OGRTADSOYAD";

lookUpEdit1.Properties.NullText = "Öğretmen Seçiniz";

lookUpEdit1.Properties.DataSource = deger1.ToList();

}

void ogrencılıstesı()

{

var deger2=from item in db.TBL\_OGRENCILER

select new

{

OGRID = item.OGRID,

OGRADSOYAD = item.OGRAD + " " + item.OGRSOYAD,

OGRSINIF=item.OGRSINIF,

};

lookUpEdit2.Properties.ValueMember = "OGRID";

lookUpEdit2.Properties.DisplayMember = "OGRADSOYAD";

lookUpEdit2.Properties.NullText = "Öğrenci Seçiniz";

lookUpEdit2.Properties.DataSource = deger2.ToList();

}

void temizle()

{

TxtOgrtID.Text = "";

TxtOgrID.Text = "";

TxtBrans.Text = "";

TxtOgrSınıf.Text = "";

TxtOgrtSıfre.Text = "";

TxtSıfre.Text = "";

MskOgrtTC.Text = "";

MskOgrTC.Text = "";

pictureEdit1.Text = "";

pictureEdit2.Text = "";

lookUpEdit1.Text = "";

lookUpEdit2.Text = "";

lookUpEdit1.Properties.NullText = "Öğretmen Şeçiniz";

lookUpEdit2.Properties.NullText = "Öğrenci Seçiniz";

}

private void FrmAyarlar\_Load(object sender, EventArgs e)

{

ogrtlistele();

ogrlistele();

ogretmenlistesi();

ogrencılıstesı();

}

public string yeniyol;

private void gridView2\_FocusedRowObjectChanged(object sender, DevExpress.XtraGrid.Views.Base.FocusedRowObjectChangedEventArgs e)

{

//secilen ögrencinin gridview gösterimi

TxtOgrID.Text = gridView2.GetRowCellValue(gridView2.FocusedRowHandle, "AYARLAROGRID").ToString();

lookUpEdit2.Text = gridView2.GetRowCellValue(gridView2.FocusedRowHandle, "OGRADSOYAD").ToString();

TxtOgrSınıf.Text = gridView2.GetRowCellValue(gridView2.FocusedRowHandle, "OGRSINIF").ToString();

MskOgrTC.Text = gridView2.GetRowCellValue(gridView2.FocusedRowHandle, "OGRTC").ToString();

TxtSıfre.Text = gridView2.GetRowCellValue(gridView2.FocusedRowHandle, "OGRSIFRE").ToString();

string uzantı = gridView2.GetRowCellValue(gridView2.FocusedRowHandle, "OGRFOTO").ToString();

yeniyol = "C:\\Users\\hüsnü\\Desktop\\C# AnaOkulu\\AnaOkuluYönetimi\\AnaOkuluYönetimi" + "\\resimler\\" + uzantı;

pictureEdit2.Image = Image.FromFile(yeniyol);

}

private void gridView1\_FocusedRowObjectChanged(object sender, DevExpress.XtraGrid.Views.Base.FocusedRowObjectChangedEventArgs e)

{

//secilen ögretmenin gridview gösterimi

TxtOgrtID.Text = gridView1.GetRowCellValue(gridView1.FocusedRowHandle, "AYARLARID").ToString();

lookUpEdit1.Text = gridView1.GetRowCellValue(gridView1.FocusedRowHandle, "OGRTADSOYAD").ToString();

TxtBrans.Text = gridView1.GetRowCellValue(gridView1.FocusedRowHandle, "OGRTBRANS").ToString();

MskOgrtTC.Text = gridView1.GetRowCellValue(gridView1.FocusedRowHandle, "OGRTTC").ToString();

TxtOgrtSıfre.Text = gridView1.GetRowCellValue(gridView1.FocusedRowHandle, "OGRTSIFRE").ToString();

string uzantı = gridView1.GetRowCellValue(gridView1.FocusedRowHandle, "OGRTFOTO").ToString();

yeniyol = "C:\\Users\\hüsnü\\Desktop\\C# AnaOkulu\\AnaOkuluYönetimi\\AnaOkuluYönetimi" + "\\resimler\\" + uzantı;

pictureEdit1.Image = Image.FromFile(yeniyol);

}

private void lookUpEdit2\_Properties\_EditValueChanged(object sender, EventArgs e)

{

TxtSıfre.Text = "";

//lookupedıtede secilen kişilerin bilgilerini gösterir öğrenciler kısmı

using(DbOkulEntities db=new DbOkulEntities())

{

TBL\_OGRENCILER sorgu2 = db.TBL\_OGRENCILER.Find(lookUpEdit2.ItemIndex + 1);

TxtOgrID.Text = sorgu2.OGRID.ToString();

TxtOgrSınıf.Text = sorgu2.OGRSINIF;

MskOgrTC.Text = sorgu2.OGRTC;

yeniyol = "C:\\Users\\hüsnü\\Desktop\\C# AnaOkulu\\AnaOkuluYönetimi\\AnaOkuluYönetimi" + "\\resimler\\" + sorgu2.OGRFOTO;

pictureEdit2.Image = Image.FromFile(yeniyol);

}

}

private void BtnOgrKaydet\_Click(object sender, EventArgs e)

{

//ogrencı sıfre kaydetme

TBL\_OGRAYARLAR komut = new TBL\_OGRAYARLAR();

komut.AYARLAROGRID = Convert.ToInt32(TxtOgrID.Text);

komut.OGRSIFRE = TxtSıfre.Text;

db.TBL\_OGRAYARLAR.Add(komut);

db.SaveChanges();

MessageBox.Show("Şifre Kaydedildi.", "Bilgi", MessageBoxButtons.OK, MessageBoxIcon.Information);

ogrlistele();

temizle();

}

private void BtnOgrtKaydet\_Click(object sender, EventArgs e)

{

//ogrt şifre kaydetme

TBL\_AYARLAR komut = new TBL\_AYARLAR();

komut.AYARLARID= Convert.ToInt32(TxtOgrtID.Text);

komut.OGRTSIFRE = TxtOgrtSıfre.Text;

db.TBL\_AYARLAR.Add(komut);

db.SaveChanges();

MessageBox.Show("Şifre Kaydedildi.", "Bilgi", MessageBoxButtons.OK, MessageBoxIcon.Information);

ogrtlistele();

temizle();

}

private void BtnOgrGuncelle\_Click(object sender, EventArgs e)

{

// ogrencıler güncelleme işlememi

int id =Convert.ToInt32( gridView2.GetRowCellValue(gridView2.FocusedRowHandle, "AYARLAROGRID"));

var item = db.TBL\_OGRAYARLAR.FirstOrDefault(x => x.AYARLAROGRID == id);

item.OGRSIFRE = TxtSıfre.Text;

db.SaveChanges();

MessageBox.Show("Şifre Güncellendi.", "Bilgi", MessageBoxButtons.OK, MessageBoxIcon.Information);

ogrlistele();

temizle();

}

private void BtnOgrtGuncelle\_Click(object sender, EventArgs e)

{

// ogretmenler güncelleme işlememi

int id = Convert.ToInt32(gridView1.GetRowCellValue(gridView1.FocusedRowHandle, "AYARLARID"));

var item = db.TBL\_AYARLAR.FirstOrDefault(x => x.AYARLARID == id);

item.OGRTSIFRE = TxtOgrtSıfre.Text;

db.SaveChanges();

MessageBox.Show("Şifre Güncellendi.", "Bilgi", MessageBoxButtons.OK, MessageBoxIcon.Information);

ogrtlistele();

temizle();

}

private void BtnOgrtTemızle\_Click(object sender, EventArgs e)

{

temizle();

}

private void BtnOgrTemızle\_Click(object sender, EventArgs e)

{

temizle();

}

}

}

**2.7.3. Öğretmen Ekranı Ana Formun Kodları**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace AnaOkuluYönetimi

{

public partial class FrmOgretmenAnaModul : Form

{

public FrmOgretmenAnaModul()

{

InitializeComponent();

}

FrmOgrtOgretmen frm1;

FrmOgrtOgrencıler frm2;

FrmOgrtVeliler frm3;

FrmOgrtDavranıslar frm4;

FrmOgrtNotlar frm5;

FrmOgrtIstatıslık frm6;

private void barButtonItem15\_ItemClick(object sender, DevExpress.XtraBars.ItemClickEventArgs e)

{

MessageBox.Show("İnternet Sitemiz Yakında Hizmetinizde.", "Bilgi", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

private void BtnOgrencıler\_ItemClick(object sender, DevExpress.XtraBars.ItemClickEventArgs e)

{

// ögrencıler formuna gider

frm2 = new FrmOgrtOgrencıler();

frm2.Show();

}

private void BtnOgrtOgretmen\_ItemClick(object sender, DevExpress.XtraBars.ItemClickEventArgs e)

{

frm1 = new FrmOgrtOgretmen();

frm1.Show();

}

private void BtnVelıler\_ItemClick(object sender, DevExpress.XtraBars.ItemClickEventArgs e)

{

frm3 = new FrmOgrtVeliler();

frm3.Show();

}

private void barButtonItem8\_ItemClick(object sender, DevExpress.XtraBars.ItemClickEventArgs e)

{

frm4 = new FrmOgrtDavranıslar();

frm4.Show();

}

private void barButtonItem13\_ItemClick(object sender, DevExpress.XtraBars.ItemClickEventArgs e)

{

frm5 = new FrmOgrtNotlar();

frm5.Show();

}

private void barButtonItem7\_ItemClick(object sender, DevExpress.XtraBars.ItemClickEventArgs e)

{

frm6 = new FrmOgrtIstatıslık();

frm6.Show();

}

}

**2.7.3.1. Öğretmen Ekranı Öğretmen Formun Kodları**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.Sql;

using System.Data.SqlClient;

namespace AnaOkuluYönetimi

{

public partial class FrmOgrtOgretmen : Form

{

public FrmOgrtOgretmen()

{

InitializeComponent();

}

sqlbaglantisi bgl = new sqlbaglantisi();

void listele()

{

//Veritabından öğretmenler tablosunu getirir

DataTable dt = new DataTable();

SqlDataAdapter da = new SqlDataAdapter("Select \* from TBL\_OGRETMENLER", bgl.baglanti());

da.Fill(dt);

GrdCntrl1.DataSource = dt;

}

private void FrmOgrtOgretmen\_Load(object sender, EventArgs e)

{

listele();

}

}

}

**2.7.3.2. Öğretmen Ekranı Öğrenciler Formun Kodları**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.Sql;

using System.Data.SqlClient;

namespace AnaOkuluYönetimi

{

public partial class FrmOgrtOgrencıler : Form

{

public FrmOgrtOgrencıler()

{

InitializeComponent();

}

sqlbaglantisi bgl = new sqlbaglantisi();

void listeleogr()

{

//Veritabından öğrenciler tablosunu getirir

DataTable dt = new DataTable();

SqlDataAdapter da = new SqlDataAdapter("Select \* from TBL\_OGRENCILER", bgl.baglanti());

da.Fill(dt);

gridControl1.DataSource = dt;

}

private void FrmOgrtOgrencıler\_Load(object sender, EventArgs e)

{

listeleogr();

}

}

}

**2.7.3.3. Öğretmen Ekranı Veliler Formun Kodları**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.Sql;

using System.Data.SqlClient;

namespace AnaOkuluYönetimi

{

public partial class FrmOgrtVeliler : Form

{

public FrmOgrtVeliler()

{

InitializeComponent();

}

sqlbaglantisi bgl = new sqlbaglantisi();

void velilistele()

{

//Veritabından Veliler tablosunu getirir

DataTable dt = new DataTable();

SqlDataAdapter da = new SqlDataAdapter("Select \* from TBL\_VELILER", bgl.baglanti());

da.Fill(dt);

gridControl1.DataSource = dt;

}

private void FrmOgrtVeliler\_Load(object sender, EventArgs e)

{

velilistele();

}

}

}

**2.7.3.4. Öğretmen Ekranı İstatistikler Formun Kodları**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.Sql;

using System.Data.SqlClient;

using System.IO;

namespace AnaOkuluYönetimi

{

public partial class FrmOgrtIstatıslık : Form

{

public FrmOgrtIstatıslık()

{

InitializeComponent();

}

sqlbaglantisi bgl = new sqlbaglantisi();

void listele()

{

//Veritabından ögrenciler tablosunu getirir

DataTable dt = new DataTable();

SqlDataAdapter da = new SqlDataAdapter("Select \* from TBL\_OGRENCILER", bgl.baglanti());

da.Fill(dt);

gridControl1.DataSource = dt;

}

void istatisliklistele()

{

//Veritabından istatislik tablosunu getirir

DataTable dt = new DataTable();

SqlDataAdapter da = new SqlDataAdapter("Select \* from TBL\_OGRTISTATISLIK", bgl.baglanti());

da.Fill(dt);

gridControl2.DataSource = dt;

}

void ogrsec()

{

// veritabanındaki illeri cmb ekler

SqlCommand komut = new SqlCommand("select \* from TBL\_OGRENCILER", bgl.baglanti());

SqlDataReader dr = komut.ExecuteReader();

while (dr.Read())

{

comboBox1.Items.Add(dr[1]);

}

bgl.baglanti().Close();

}

private void FrmOgrtIstatıslık\_Load(object sender, EventArgs e)

{

listele();

istatisliklistele();

ogrsec();

}

public string yeniyol;

private void gridView1\_FocusedRowObjectChanged(object sender, DevExpress.XtraGrid.Views.Base.FocusedRowObjectChangedEventArgs e)

{

DataRow dr = gridView1.GetDataRow(gridView1.FocusedRowHandle);

if (dr != null)

{

TxtID.Text = dr["OGRID"].ToString();

comboBox1.Text = dr["OGRAD"].ToString();

TxtSoyad.Text = dr["OGRSOYAD"].ToString();

MskTC.Text = dr["OGRTC"].ToString();

MskOgrencıNo.Text = dr["OGRNO"].ToString();

yeniyol = "C:\\Users\\hüsnü\\Desktop\\C# AnaOkulu\\AnaOkuluYönetimi\\AnaOkuluYönetimi" + "\\resimler\\" + dr["OGRFOTO"].ToString();

pictureEdit1.Image = Image.FromFile(yeniyol);

}

}

public string cinsiyet;

private void BtnIstKaydet\_Click(object sender, EventArgs e)

{

// veritabanına davranış kaydeden kod

SqlCommand komut = new SqlCommand("Insert into TBL\_OGRTISTATISLIK(OgrIstAdi,OgrIstSoyadi,OgrIstNo,OgrIstTC,Ogrıstatıslık,OgrIstFoto) values (@p1,@p2,@p3,@p7,@p8,@p11)", bgl.baglanti());

komut.Parameters.AddWithValue("@p1", comboBox1.Text);

komut.Parameters.AddWithValue("@p2", TxtSoyad.Text);

komut.Parameters.AddWithValue("@p3", MskOgrencıNo.Text);

komut.Parameters.AddWithValue("@p7", MskTC.Text);

komut.Parameters.AddWithValue("@p8", richTextBox1.Text);

komut.Parameters.AddWithValue("@p11", Path.GetFileName(yeniyol));

komut.ExecuteNonQuery();

bgl.baglanti().Close();

MessageBox.Show("Ögrenci İstatisliği Eklendi", "Bilgi", MessageBoxButtons.OK, MessageBoxIcon.Information);

listele();

}

void temizle()

{

TxtID.Text = "";

comboBox1.Text = "";

TxtSoyad.Text = "";

MskOgrencıNo.Text = "";

MskTC.Text = "";

richTextBox1.Text = "";

pictureEdit1.Text = "";

}

private void BtnIstTemizle\_Click(object sender, EventArgs e)

{

temizle();

}

private void comboBox1\_SelectedIndexChanged(object sender, EventArgs e)

{

SqlCommand komut = new SqlCommand("Select OGRID,OGRSOYAD,OGRTC,OGRNO From TBL\_OGRENCILER Where OGRAD= '" + comboBox1.Text+"'", bgl.baglanti());

SqlDataReader oku = komut.ExecuteReader();

while (oku.Read())

{

TxtID.Text = oku["OGRID"].ToString();

TxtSoyad.Text = oku["OGRSOYAD"].ToString();

MskTC.Text = oku["OGRTC"].ToString();

MskOgrencıNo.Text = oku["OGRNO"].ToString();

}

bgl.baglanti().Close();

}

private void gridControl1\_DoubleClick(object sender, EventArgs e)

{

DataRow dr = gridView1.GetDataRow(gridView1.FocusedRowHandle);

FrmOgrencıDataGrıdVıew frm = new FrmOgrencıDataGrıdVıew();

if (dr != null)

{

frm.ıd = dr["OGRID"].ToString();

frm.ad = dr["OGRAD"].ToString();

frm.soyad = dr["OGRSOYAD"].ToString();

frm.tc = dr["OGRTC"].ToString();

frm.sınıf = dr["OGRSINIF"].ToString();

frm.dogtarihi = dr["OGRDOGTAR"].ToString();

frm.uzantı = "C:\\Users\\hüsnü\\Desktop\\C# AnaOkulu\\AnaOkuluYönetimi\\AnaOkuluYönetimi" + "\\resimler\\" + dr["OGRFOTO"].ToString();

}

frm.Show();

}

}

}

**2.7.3.5. Öğretmen Ekranı Davranışlar Formun Kodları**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.Sql;

using System.Data.SqlClient;

using System.IO;

namespace AnaOkuluYönetimi

{

public partial class FrmOgrtDavranıslar : Form

{

public FrmOgrtDavranıslar()

{

InitializeComponent();

}

sqlbaglantisi bgl = new sqlbaglantisi();

void listele()

{

//Veritabından ögrenciler tablosunu getirir

DataTable dt = new DataTable();

SqlDataAdapter da = new SqlDataAdapter("Select \* from TBL\_OGRENCILER", bgl.baglanti());

da.Fill(dt);

gridControl1.DataSource = dt;

}

void davranıslistele()

{

//Veritabından Davranıslartablosunu getirir

DataTable dt = new DataTable();

SqlDataAdapter da = new SqlDataAdapter("Select \* from TBL\_DAVRANISLAR", bgl.baglanti());

da.Fill(dt);

gridControl2.DataSource = dt;

}

void ogrsec()

{

// veritabanındaki illeri cmb ekler

SqlCommand komut = new SqlCommand("select \* from TBL\_OGRENCILER", bgl.baglanti());

SqlDataReader dr = komut.ExecuteReader();

while (dr.Read())

{

comboBox1.Items.Add(dr[1]);

}

bgl.baglanti().Close();

}

private void FrmOgrtDavranıslar\_Load(object sender, EventArgs e)

{

listele();

davranıslistele();

ogrsec();

}

public string yeniyol;

private void gridView1\_FocusedRowObjectChanged(object sender, DevExpress.XtraGrid.Views.Base.FocusedRowObjectChangedEventArgs e)

{

DataRow dr = gridView1.GetDataRow(gridView1.FocusedRowHandle);

if (dr != null)

{

TxtID.Text = dr["OGRID"].ToString();

comboBox1.Text = dr["OGRAD"].ToString();

TxtSoyad.Text = dr["OGRSOYAD"].ToString();

MskTC.Text = dr["OGRTC"].ToString();

MskOgrencıNo.Text = dr["OGRNO"].ToString();

yeniyol = "C:\\Users\\hüsnü\\Desktop\\C# AnaOkulu\\AnaOkuluYönetimi\\AnaOkuluYönetimi" + "\\resimler\\" + dr["OGRFOTO"].ToString();

pictureEdit1.Image = Image.FromFile(yeniyol);

}

}

public string cinsiyet;

private void BtnDavKaydet\_Click(object sender, EventArgs e)

{

// veritabanına davranış kaydeden kod

SqlCommand komut = new SqlCommand("Insert into TBL\_DAVRANISLAR(OgrDavAdi,OgrDavSoyadi,OgrDavNo,OgrDavTC,OgrDavranış,OgrDavFoto) values (@p1,@p2,@p3,@p7,@p8,@p11)", bgl.baglanti());

komut.Parameters.AddWithValue("@p1", comboBox1.Text);

komut.Parameters.AddWithValue("@p2", TxtSoyad.Text);

komut.Parameters.AddWithValue("@p3", MskOgrencıNo.Text);

komut.Parameters.AddWithValue("@p7", MskTC.Text);

komut.Parameters.AddWithValue("@p8", richTextBox1.Text);

komut.Parameters.AddWithValue("@p11", Path.GetFileName(yeniyol));

komut.ExecuteNonQuery();

bgl.baglanti().Close();

MessageBox.Show("Ögrenci Davranışı Eklendi", "Bilgi", MessageBoxButtons.OK, MessageBoxIcon.Information);

listele();

}

void temizle()

{

TxtID.Text = "";

TxtSoyad.Text = "";

MskOgrencıNo.Text = "";

MskTC.Text = "";

richTextBox1.Text = "";

pictureEdit1.Text = "";

}

private void BtnDavTemizle\_Click(object sender, EventArgs e)

{

temizle();

}

private void comboBox1\_SelectedIndexChanged(object sender, EventArgs e)

{

SqlCommand komut = new SqlCommand("Select OGRID,OGRSOYAD,OGRTC,OGRNO From TBL\_OGRENCILER Where OGRAD= '" + comboBox1.Text + "'", bgl.baglanti());

SqlDataReader oku = komut.ExecuteReader();

while (oku.Read())

{

TxtID.Text = oku["OGRID"].ToString();

TxtSoyad.Text = oku["OGRSOYAD"].ToString();

MskTC.Text = oku["OGRTC"].ToString();

MskOgrencıNo.Text = oku["OGRNO"].ToString();

}

bgl.baglanti().Close();

}

private void gridControl1\_DoubleClick(object sender, EventArgs e)

{

DataRow dr = gridView1.GetDataRow(gridView1.FocusedRowHandle);

FrmOgrencıDataGrıdVıew frm = new FrmOgrencıDataGrıdVıew();

if (dr != null)

{

frm.ıd = dr["OGRID"].ToString();

frm.ad = dr["OGRAD"].ToString();

frm.soyad = dr["OGRSOYAD"].ToString();

frm.tc = dr["OGRTC"].ToString();

frm.sınıf = dr["OGRSINIF"].ToString();

frm.dogtarihi = dr["OGRDOGTAR"].ToString();

frm.uzantı = "C:\\Users\\hüsnü\\Desktop\\C# AnaOkulu\\AnaOkuluYönetimi\\AnaOkuluYönetimi" + "\\resimler\\" + dr["OGRFOTO"].ToString();

}

frm.Show();

}

}

}

**2.7.3.6. Öğretmen Ekranı Notlar Formun Kodları**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.Sql;

using System.Data.SqlClient;

namespace AnaOkuluYönetimi

{

public partial class FrmOgrtNotlar : Form

{

public FrmOgrtNotlar()

{

InitializeComponent();

}

sqlbaglantisi bgl = new sqlbaglantisi();

private void verilerigoster()

{

listView1.Items.Clear();

SqlCommand komut = new SqlCommand("select\*from TBL\_OGRTNOTLAR", bgl.baglanti());

SqlDataReader oku = komut.ExecuteReader();

while (oku.Read())

{

ListViewItem ekle = new ListViewItem();

ekle.Text = oku["NotID"].ToString();

ekle.SubItems.Add(oku["AdSoyad"].ToString());

ekle.SubItems.Add(oku["Notlar"].ToString());

listView1.Items.Add(ekle);

}

bgl.baglanti().Close();

}

private void FrmOgrtNotlar\_Load(object sender, EventArgs e)

{

verilerigoster();

ogrtsec();

}

void ogrtsec()

{

// veritabanındaki ogretmenleri cmb ekler

SqlCommand komut = new SqlCommand("select \* from TBL\_OGRETMENLER", bgl.baglanti());

SqlDataReader dr = komut.ExecuteReader();

while (dr.Read())

{

comboBox1.Items.Add(dr[1]);

}

bgl.baglanti().Close();

}

private void BtnOgrtNotKaydet\_Click(object sender, EventArgs e)

{

SqlCommand komut = new SqlCommand("insert into TBL\_OGRTNOTLAR (AdSoyad,Notlar) values ('" + comboBox1.Text + "','" + RchTxtNot.Text + "')", bgl.baglanti());

komut.ExecuteNonQuery();

bgl.baglanti().Close();

MessageBox.Show("Not Kayıt Edildi.", "Bilgi", MessageBoxButtons.OK, MessageBoxIcon.Information);

verilerigoster();

}

int id = 0;

private void listView1\_DoubleClick(object sender, EventArgs e)

{

id = int.Parse(listView1.SelectedItems[0].SubItems[0].Text);

comboBox1.Text = listView1.SelectedItems[0].SubItems[1].Text;

RchTxtNot.Text = listView1.SelectedItems[0].SubItems[2].Text;

}

private void comboBox1\_SelectedIndexChanged(object sender, EventArgs e)

{

SqlCommand komut = new SqlCommand("Select OGRTTC From TBL\_OGRETMENLER Where OGRTAD= '" + comboBox1.Text + "'", bgl.baglanti());

SqlDataReader oku = komut.ExecuteReader();

while (oku.Read())

{

textBox1.Text = oku["OGRTTC"].ToString();

}

bgl.baglanti().Close();

}

}

}

**2.7.4. Öğrenci Ekranı Ana Formun Kodları**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.Sql;

using System.Data.SqlClient;

namespace AnaOkuluYönetimi

{

public partial class FrmOgrencıAnaModul : Form

{

public FrmOgrencıAnaModul()

{

InitializeComponent();

}

sqlbaglantisi bgl = new sqlbaglantisi();

private void FrmOgrencıAnaModul\_Load(object sender, EventArgs e)

{

davraisListele();

ıstatislikisListele();

}

void davraisListele()

{

//Veritabından Davranışları tablosunu getirir

DataTable dt = new DataTable();

SqlDataAdapter da = new SqlDataAdapter("Select \* from TBL\_DAVRANISLAR ", bgl.baglanti());

da.Fill(dt);

gridControl1.DataSource = dt;

}

void ıstatislikisListele()

{

//Veritabından istatislik tablosunu getirir

DataTable dt = new DataTable();

SqlDataAdapter da = new SqlDataAdapter("Select \* from TBL\_OGRTISTATISLIK", bgl.baglanti());

da.Fill(dt);

gridControl2.DataSource = dt;

}

public string ad, soyad, tc, cinsiyet, sinif, uzantı;

private void FrmNufusCuzdani\_Load(object sender, EventArgs e)

{

TxtAdi.Text = ad;

TxtSoyad.Text = soyad;

TxtTC.Text = tc;

TxtSınıf.Text = sinif;

pictureBox1.Image = Image.FromFile(uzantı);

}

}

}

**2.7.5. Sql Bağlantısı Kodu**

Bu kod ile sql bağlantımızı sürekli açık tutup ve sürekli bağlantı adresini yazmaktan kurtulduk. Oluşturduğumuz “baglanti” adındaki değişken ile veri tabanına bağlamamız gerektiği yerde bu kodu yazarak bağladım.

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Data.SqlClient;

namespace AnaOkuluYönetimi

{

class sqlbaglantisi

{

public SqlConnection baglanti()

{

SqlConnection baglan = new SqlConnection(@"Data Source=.\SQLEXPRESS;Initial Catalog=dbo.AnaOkuluYönetimi;Integrated Security=True");

baglan.Open();

return baglan;

}

}

}

**2.7.6. Datagridview Öğrenci Kodu**

Bu kod ile datagridview öğrenci kısımlarına çift tıkladığımızda öğrenciye ait bilgiler gelmektedir.

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace AnaOkuluYönetimi

{

public partial class FrmOgrencıDataGrıdVıew : Form

{

public FrmOgrencıDataGrıdVıew()

{

InitializeComponent();

}

public string ıd,ad, soyad, tc, sınıf, dogtarihi, uzantı;

private void FrmOgrencıDataGrıdVıew\_Load(object sender, EventArgs e)

{

textBox1.Text=ıd;

textBox2.Text = ad;

textBox3.Text = soyad;

textBox4.Text = tc;

textBox5.Text = sınıf;

textBox7.Text = dogtarihi;

pictureEdit1.Image = Image.FromFile(uzantı);

}

}

}

**2.7.7. Datagridview Öğretmen Kodu**

Bu kod ile datagridview öğretmen kısımlarına çift tıkladığımızda öğretmene ait bilgiler gelmektedir.

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace AnaOkuluYönetimi

{

public partial class FrmOgretmenDataGrıpView : Form

{

public FrmOgretmenDataGrıpView()

{

InitializeComponent();

}

public string ıd, ad, soyad, tc, telefon, brans, uzantı;

private void FrmOgretmenDataGrıpView\_Load(object sender, EventArgs e)

{

textBox1.Text = ıd;

textBox2.Text = ad;

textBox3.Text = soyad;

textBox4.Text = tc;

textBox5.Text = telefon;

textBox7.Text = brans;

pictureEdit1.Image = Image.FromFile(uzantı);

}

}

}

**KAYNAKÇA**

<http://www.kodkaynagi.com/>

<https://www.heryerdeyazilim.com/>

<https://www.sqlkodlari.com/>

<https://medium.com/t%C3%BCrkiye/entity-framework-nedir-bff3943bec72>

<https://osmancelik.com.tr/>

<https://www.bilisimkonulari.com/tag/c-windows-form-application-ornekleri-pdf>

<https://medium.com/software-development-turkey/nedir-bu-c-b49087c1d788>

<https://medium.com/sql-t%C3%BCrkiye>

<https://medium.com/swlh/basics-of-sql-7f1e72772e63>

<https://medium.com/gokhanyavas/tagged/sql>