



**CELAL BAYAR
ÜNİVERSİTESİ**

**YAZILIM MİMARİSİ VE TASARIMI
SEYAHAT UYGULAMASI**

HAZIRLAYANLAR

SAMET FURKAN KOPAR -202802041

ALİ EMRE KENDİRLİ -202802043

1.GİRİŞ

Projenin Amacı:

Projemizin amacı insanlara ulaşım ve konaklama imkanları sunmaktır. Çeşitli seçenekler olduğundan ulaşım için kişi hangi seçenek kendisi için uygunsa onu seçeर ardından konaklama içinde kendisine uygun olan seçeneği seçeر . Yani kullanıcı kolay bir şekilde ulaşım ve konaklama ihtiyaçlarını karşılar.

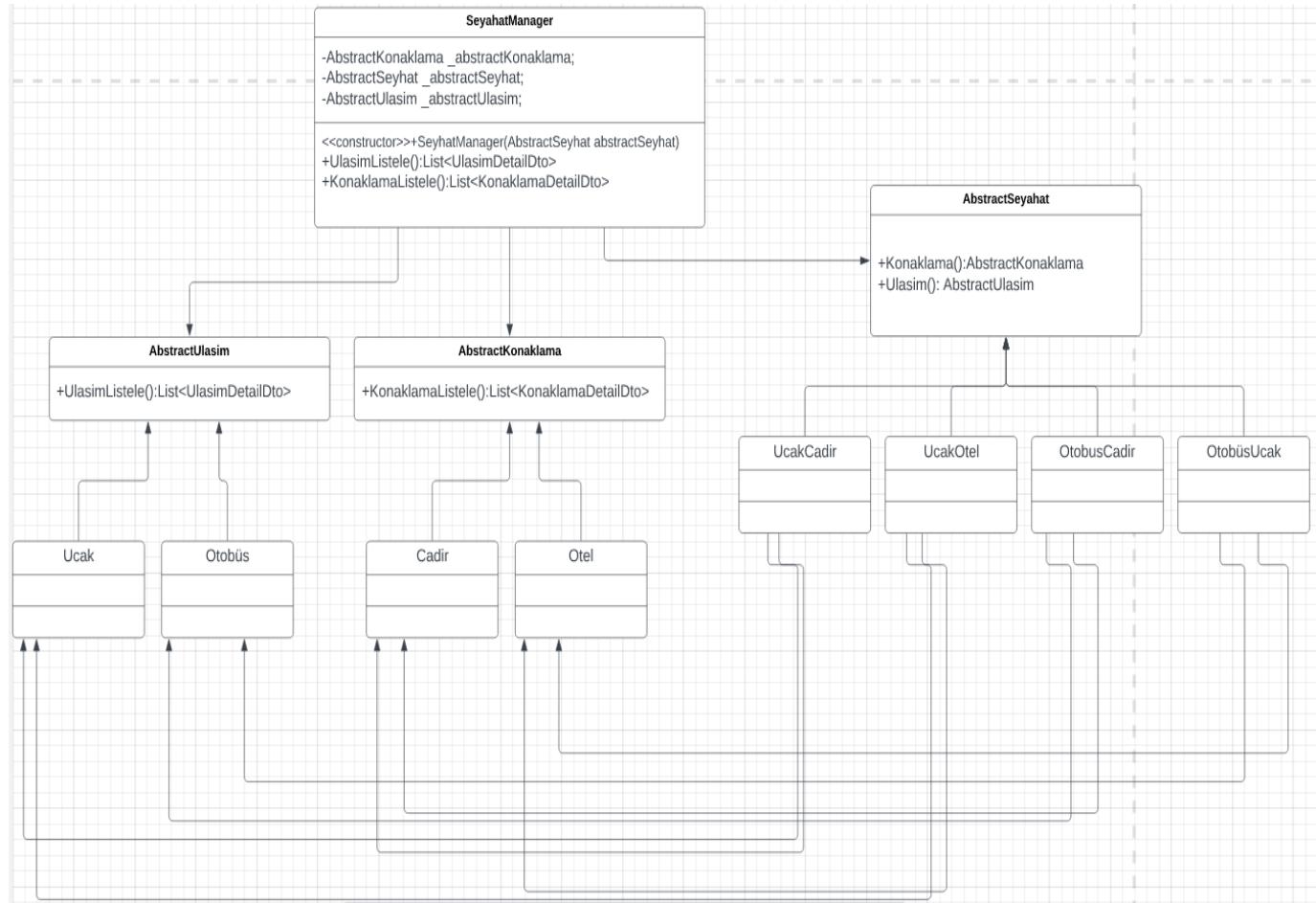
Projenin Kapsamı:

Seyahat yapmak isteyen kullanıcı bulunduğu il , gideceği il ,konaklama ve ulaşım türünü seçtikten sonra gidiş ve dönüş tarihini de seçtikten sonra önüne bir ekran gelir. Gelen ekranда seçtiğimiz konaklama ile ilgili şirketler ve fiyatları gelir ve birini seçmeniz istenir . Seçip devam edince gelen ekranдан ise ulaşım için şirketler ve fiyatları gelir. Buradan da birisini seçip devam edince rezervasyonumuz kayıt edilmiş olur. Rezervasyon kaydı tamamlanınca kullanıcı gereken bilgileri JSON, HTML veya XML formatta alabiliyor olacaktır.

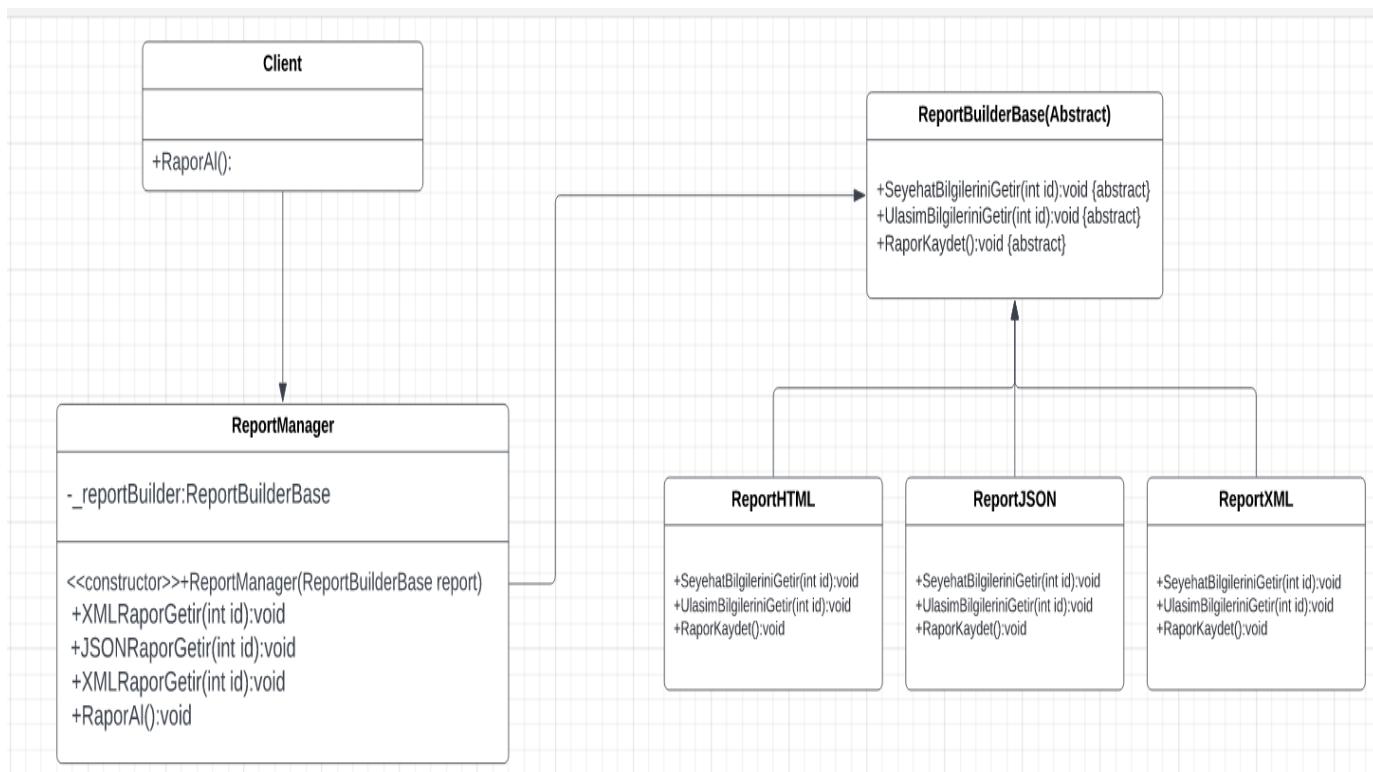
2-Proje

A-) UML Class Diyagramları

Abstract Factory Tasarım Deseni -Rezervasyon Yapma



Builder tasarım deseni -Rapor Alma Kısmı



B-) Proje Ekran Görüntüleri

The image displays five screenshots of a travel agency application interface, likely built with ASP.NET MVC:

- User Registration Screen:** Shows fields for First Name, Last Name, Username, Password, and a "Kayıt Ol" (Register) button.
- Login Screen:** Shows fields for Username and Password, a "Giriş" (Login) button, and a "Kayıt Ol" (Register) link.
- Trip Planning Screen:** Shows dropdowns for Departure City (Istanbul), Destination City (Izmir), Accommodation (Cadir), Transportation (Bus), and travel dates (May 25, 2022 - May 27, 2022). A "DEVAM EDİNİZ" (Continue) button is present.
- Accommodation Booking Screen:** Shows a table with columns: KonaklamaID, SirketAdı, KonaklamaTipi, ücret, TatilYeri. One row is selected (KonaklamaID 1, SirketAdı Buhra, KonaklamaTipi Cadir, ücret 50, TatilYeri Izmir). A "Devam Ediniz" (Continue) button is at the bottom.
- Vehicle Reservation Screen:** Shows a table with columns: UlasimId, SirketAdı, KalkisYeri, VarisYeri, KalkisSaatı, VarisSaatı, Ucret, AracTipi. One row is selected (UlasimId 4, SirketAdı Nur Turizm, KalkisYeri Istanbul, VarisYeri Izmir, KalkisSaatı 15:00, VarisSaatı 22:00, Ucret 225, AracTipi Otobus). A "Rezervasyonu Kaydet" (Save Reservation) button is at the bottom.

	UlasimId	SirketAdi	KalkisYeri	VarisYeri	KalkisSaatı	V
►	4	Nur Turizim	İstanbul	İzmir	15:00	22

	KonaklamaID	SirketAdi	KonaklamaTipi	ucret	TatilYeri
►	1	Buhra	Cadir	50	İzmir

Dikkat

i Ulaşım bilgileri silinirse konaklama bilgileride silinecek?

Evet Hayır

Sil

Rapor Türünüzü Seçiniz :

Rapor Al



TRAVEL

Rapor Çıktıları:

-HTML Raporu

SEYHAT BİLGİLERİ

Adı	Soyadı	Şirket Adı	Konaklama Tipi	Tatil Yeri	Rezervasyon Başlangıç Tarihi	Rezervasyon Bitiş Tarihi	Ucret
Emre	Kendirli	Buhra	Cadir	İzmir	25.05.2022 00:00:00	27.05.2022 00:00:00	100

Ulaşım BİLGİLERİ

Adı	Soyadı	Ulaşım Tipi	Kalkış yeri	Varış yeri	kalkış saatı	varış saatı	Ucret
Emre	Kendirli	Otobus	İstanbul	İzmir	15:00	22:00	450

-JSON Raporu

```
[{"UlasimAdi": "Konaklama Bilgileri", "Adi": "Emre", "Soyadi": "Kendirli", "Sirket_Adi": "Buhra", "Konaklama_Tipi": "Cadir", "Tatil_Yeri": "İzmir", "BaslangicTarihi": "2022-05-25T00:00:00", "BitisTarihi": "2022-05-27T00:00:00", "ucret": 100}, {"UlasimAdi": "Ulasim Bilgileri", "Adi": "Emre", "Soyadi": "Kendirli", "KalkisYeri": "İstanbul", "VarisYeri": "İzmir", "KalkisSaatı": "15:00", "VarisSaatı": "22:00", "ucret": 450}]
```

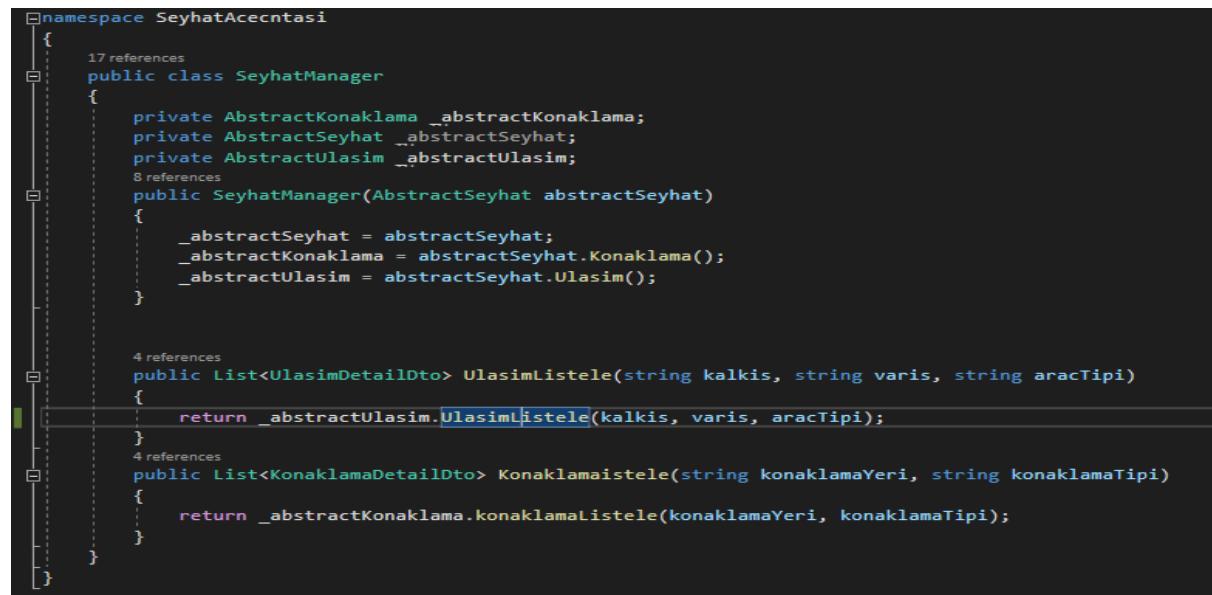
XML Raporu

```
<!-- sample XML -->
▼<SEYEHAT Bilgi="SEYEHAT BİLGİ">
  <title>-----KONAKLAMA BİLGİLERİ-----</title>
  <Adi>Emre</Adi>
  <Soyadi>Kendirli</Soyadi>
  <Sirket_Adi>Buhra</Sirket_Adi>
  <Konaklama_Tipi>Cadir</Konaklama_Tipi>
  <Tatil_Yeri>İzmir</Tatil_Yeri>
  <BaslangicTarihi>25.05.2022 00:00:00</BaslangicTarihi>
  <BitisTarihi>27.05.2022 00:00:00</BitisTarihi>
  <Ucret>100</Ucret>
  <title>-----Ulaşım BİLGİLERİ-----</title>
  <Adi>Emre</Adi>
  <Soyadi>Kendirli</Soyadi>
  <UlasimTipi>Otobus</UlasimTipi>
  <KalkisYeri>İstanbul</KalkisYeri>
  <VarisYeri>İzmir</VarisYeri>
  <KalkisSaatii>15:00</KalkisSaatii>
  <VarisSaatii>22:00</VarisSaatii>
  <Ucret>450</Ucret>
</SEYEHAT>
```

C-) Proje Kodları

Abstract Tasarım Deseni

Client Sınıfı



```
namespace SeyhatAcemntasi
{
    public class SeyhatManager
    {
        private AbstractKonaklama _abstractKonaklama;
        private AbstractSeyhat _abstractSeyhat;
        private AbstractUlasim _abstractUlasim;
        public SeyhatManager(AbstractSeyhat abstractSeyhat)
        {
            _abstractSeyhat = abstractSeyhat;
            _abstractKonaklama = abstractSeyhat.Konaklama();
            _abstractUlasim = abstractSeyhat.Ulasim();
        }

        public List<UlasimDetailDto> UlasimListele(string kalkis, string varis, string aracTipi)
        {
            return _abstractUlasim.UlasimListele(kalkis, varis, aracTipi);
        }
        public List<KonaklamaDetailDto> KonaklamaListele(string konaklamaYeri, string konaklamaTipi)
        {
            return _abstractKonaklama.konaklamaListele(konaklamaYeri, konaklamaTipi);
        }
    }
}
```

AbstractSeyhat sınıfı

```
namespace SeyhatAcentasi.AbstractKSe
{
    public abstract class AbstractSeyhat
    {
        public abstract AbstractKonaklama Konaklama();
        public abstract AbstractUlasim Ulasim();
    }
}
```

AbstractUlasim sınıfı

```
namespace SeyhatAcentasi.AbstractUlas
{
    public abstract class AbstractUlasim
    {
        public abstract List<UlasimDetailDto> UlasimListele(string kalkis, string varis, string aracTipi);
    }
}
```

AbstractKonaklama sınıfı

```
namespace SeyhatAcentasi.AbstractKonakla
{
    public abstract class AbstractKonaklama
    {
        public abstract List<KonaklamaDetailDto> konaklamaListele(string konaklamaYeri, string konaklamaTipi);
    }
}
```

Uçak Sınıfı

```
namespace SeyhatAcentasi.AbstractUlas
{
    public class Ucak : AbstractUlasim
    {
        UlasimAracManager ulasim = new UlasimAracManager(new EFUlasimAracDal());

        public override List<UlasimDetailDto> UlasimListele(string kalkis, string varis, string aracTipi)
        {
            return ulasim.GetUlasimDetailDtos(kalkis, varis, aracTipi);
        }
    }
}
```

Otobus Sınıfı

```
namespace SeyhatAcecntasi.AbstractUlas
{
    public class Otobus : AbstractUlasim
    {
        UlasimAracManager ulasim = new UlasimAracManager(new EFUlasimAracDal());
        public override List<UlasimDetailDto> UlasimListele(string kalkis, string varis, string aracTipi)
        {
            return ulasim.GetUlasimDetailDtos(kalkis, varis, aracTipi);
        }
    }
}
```

Cadır Sınıfı

```
namespace SeyhatAcecntasi.AbstractKonakla
{
    public class Cadir : AbstractKonaklama
    {
        KonaklamaBilgiManager konaklamaBilgiManager = new KonaklamaBilgiManager(new EFKonaklamaDal());
        public override List<KonaklamaDetailDto> konaklamaListele(string konaklamaYeri, string konaklamaTipi)
        {
            return konaklamaBilgiManager.konaklamaDetailDtos(konaklamaYeri, konaklamaTipi);
        }
    }
}
```

Otel Sınıfı

```
namespace SeyhatAcecntasi.AbstractKonakla
{
    public class Otel : AbstractKonaklama
    {
        KonaklamaBilgiManager konaklamaBilgiManager = new KonaklamaBilgiManager(new EFKonaklamaDal());
        public override List<KonaklamaDetailDto> konaklamaListele(string konaklamaYeri, string konaklamaTipi)
        {
            return konaklamaBilgiManager.konaklamaDetailDtos(konaklamaYeri, konaklamaTipi);
        }
    }
}
```

UcakCadir Sınıfı

```
namespace SeyhatAecntasi.AbstractSey
{
    public class UcakCadir : AbstractSeyhat
    {
        public override AbstractKonaklama Konaklama()
        {
            return new Cadir();
        }

        public override AbstractUlasim Ulasim()
        {
            return new Ucak();
        }
    }
}
```

UcakOtel Sınıfı

```
namespace SeyhatAecntasi.AbstractSey
{
    public class UcakOtel : AbstractSeyhat
    {
        public override AbstractKonaklama Konaklama()
        {
            return new Otel();
        }

        public override AbstractUlasim Ulasim()
        {
            return new Ucak();
        }
    }
}
```

OtobusCadir Sınıfı

```
namespace SeyhatAecntasi.AbstractSey
{
    public class OtobusCadir : AbstractSeyhat
    {
        public override AbstractKonaklama Konaklama()
        {
            return new Cadir();
        }

        public override AbstractUlasim Ulasim()
        {
            return new Otobus();
        }
    }
}
```

OtobusOtel Sınıfı

```
namespace SeyhatAcepntasi.AbstractSeyhat
{
    public class OtobusOtel : AbstractSeyhat
    {
        public override AbstractKonaklama Konaklama()
        {
            return new Otel();
        }

        public override AbstractUlasim Ulasim()
        {
            return new Otobus();
        }
    }
}
```

Builder Tasarım Deseni -Rapor Alma

Client

```
private void button1_Click(object sender, EventArgs e)
{
    int id = kullaniciManager.KullaniciAdiVeSifresiIleIdGetir(kullanici, Sifree).Id;
    if (comboBox1.Text == "HTML")
    {
        ReportManager reportManager = new ReportManager(new ReportHTML());
        reportManager.HTMLRaporGetir(id);
        reportManager.RaporAl();
        MessageBox.Show("HTML Raporunuz başarılı şekilde oluşturulmuştur");
    }
    else if (comboBox1.Text == "JSON")
    {
        ReportManager reportManager = new ReportManager(new ReportJSON());
        reportManager.JSONRaporGetir(id);
        reportManager.RaporAl();
        MessageBox.Show("JSON Raporunuz başarılı şekilde oluşturulmuştur");
    }
    else if (comboBox1.Text == "XML")
    {
        ReportManager reportManager = new ReportManager(new ReportXML());
        reportManager.JSONRaporGetir(id);
        reportManager.RaporAl();
        MessageBox.Show("XML Raporunuz başarılı şekilde oluşturulmuştur");
    }
    else
    {
        MessageBox.Show("Boş Geçmeyin");
    }
}
```

ReportManager sınıfı

```
namespace BuilderRaporlama
{
    public class ReportManager
    {
        private ReportBuilderBase _reportBuilder;
        public ReportManager(ReportBuilderBase report) //Kurucu metod
        {
            _reportBuilder = report;
        }
        public void HTMLRaporGetir(int id)
        {
            _reportBuilder.SeyehatBilgileriniGetir(id);
            _reportBuilder.UlasimBilgileriniGetir(id);
        }
        public void JSONRaporGetir(int id)
        {
            _reportBuilder.SeyehatBilgileriniGetir(id);
            _reportBuilder.UlasimBilgileriniGetir(id);
        }
        public void XMLRaporGetir(int id)
        {
            _reportBuilder.SeyehatBilgileriniGetir(id);
            _reportBuilder.UlasimBilgileriniGetir(id);
        }
        public void RaporAl()
        {
            //Soyut sınıfı bulunan (ReportKaytBuilderBase) Rapor kaydet
            _reportBuilder.RaporKaydet();
        }
    }
}
```

ReportBuilderBase Sınıfı

```
4 references
public override void SeyehatBilgileriniGetir(int id)
{
    var kaldigiGunSayisi = seyhatBilgiManager.GetId(id).RezervasyonBitis.Day - seyhatBilgiManager.GetId(id).RezervasyonBaslangic.Day;
    var konaklamaId = seyhatBilgiManager.GetId(id).KonaklamaID;

    string Adi = kullaniciManager.GetId(id).Adi;
    string Soyadi = kullaniciManager.GetId(id).Soyadi;
    string Sirket_Adi = konaklama.GetId(konaklamaId).SirketAdi;
    string Konaklama_Tipi = konaklama.GetId(konaklamaId).KonaklamaTipi;
    string Tatil_Yeri = konaklama.GetId(konaklamaId).TatilYeri;
    DateTime BaslangicTarihi = seyhatBilgiManager.GetId(id).RezervasyonBaslangic;
    DateTime BitisTarihi = seyhatBilgiManager.GetId(id).RezervasyonBitis;
    var ucret = konaklama.GetId(id).Ucret;

    writer1.Formatting = Formatting.Indented;
    writer1.WriteStartDocument();
    writer1.WriteDocType("SeyehatUlasim", null, null, "<!ENTITY h 'hardcover'>");
    writer1.WriteComment("sample XML");
    writer1.WriteStartElement("SEYEHAT");
    writer1.WriteAttributeString("Bilgi", "SEYEHAT BİLGİ");
    writer1.WriteElementString("title", "-----KONAKLAMA BİLGİLERİ-----");

    writer1.WriteElementString("Adi", Adi);
    writer1.WriteElementString("Soyadi", Soyadi);
    writer1.WriteElementString("Sirket_Adi", Sirket_Adi);
    writer1.WriteElementString("Konaklama_Tipi", Konaklama_Tipi);
    writer1.WriteElementString("Tatil_Yeri", Tatil_Yeri);
    writer1.WriteElementString("BaslangicTarihi", BaslangicTarihi.ToString());
    writer1.WriteElementString("BitisTarihi", BitisTarihi.ToString());
    writer1.WriteElementString("Ucret", ucret.ToString());
}
```

ReportXML Sınıfı

```
namespace BuilderRaporlama.Reports
{
    1 reference
    public class ReportXML : ReportBuilderBase
    {
        KullaniciManager kullaniciManager = new KullaniciManager(new EFKullaniciDal());
        SeyhatBilgiManager seyahatBilgiManager = new SeyhatBilgiManager(new EFSeyhatBilgiDal());
        UlasimAracManager ulasim = new UlasimAracManager(new EFUlasimAracDal());
        KonaklamaBilgiManager konaklama = new KonaklamaBilgiManager(new EFKonaklamaDal());
        private const string filename = @"C:\Users\EMRE\Desktop\xML.xml";
        XmlTextWriter writer1 = new XmlTextWriter(filename, null);

        2 references
        public override void RaporKaydet()
        {
            writer1.WriteEndElement();
            writer1.WriteEndDocument();
            writer1.Flush();
            writer1.Close();
            XmlDocument doc = new XmlDocument();

            doc.PreserveWhitespace = true;

            doc.Load(filename);
        }
    }
}
```

ReportHTML Sınıfı

```
1 reference
public class ReportHTML : ReportBuilderBase
{
    KullaniciManager kullaniciManager = new KullaniciManager(new EFKullaniciDal());
    SeyhatBilgiManager seyahatBilgiManager = new SeyhatBilgiManager(new EFSeyhatBilgiDal());
    UlasimAracManager ulasim = new UlasimAracManager(new EFUlasimAracDal());
    KonaklamaBilgiManager konaklama = new KonaklamaBilgiManager(new EFKonaklamaDal());
    references
    public override void RaporKaydet()
    {
        System.IO.File.WriteAllText(@"C:\Users\EMRE\Desktop\html.html", sb.ToString());
    }
    references
    public override void SeyhatBilgileriniGetir(int id)
    {
        var konaklamaID = seyahatBilgiManager.GetId(id).KonaklamaID;
        var adi = kullaniciManager.GetId(id).Adi;
        var soyadi = kullaniciManager.GetId(id).Soyadi;
        var SirkeAdi = konaklama.GetId(konaklamaID).SirkeAdi;
        var KonaklamaTipi = konaklama.GetId(konaklamaID).KonaklamaTipi;
        var uret = konaklama.GetId(konaklamaID).Uret;
        var baslangicTarihi = seyahatBilgiManager.GetId(id).RezervasyonBaslangicTarihi;
        var bitisTarihi = seyahatBilgiManager.GetId(id).RezervasyonBitis;
        int kaldigunSayisi = bitisTarihi.Day - baslangicTarihi.Day;

        sb.Append("<h3>SEYHAT BİLGİLERİ</h3>"); //Tablonun basılışı yazıldı
        sb.Append("<table class='display' id='dt' style='align: left; width: 99%;'>");
        sb.Append("rules='all' border='1px'\"");
        sb.Append("cellspacing='0px' cellpadding='4px'");

        sb.Append("<thead>");
        sb.Append("<tr><td>"); 
        sb.Append("<tr style='background-color: Silver; color: #1E1E1E;'>"); 
        sb.Append("font-weight: bold");
        sb.Append("<tr> <th>Adı</th> <th>Soyadı</th> <th>Sirke Adı</th> <th>Konaklama Tipi</th><th>Tatil Verisi</th><th>Rezervasyon Başlangıç Tarihi</th><th>Rezervasyon Bitiş Tarihi</th><th>Ucret</th><th></tr>"); //Tablonun sütun isimleri yazıldı
        sb.Append("<tr> <td>" + adi + "</td> <td>" + soyadi + "</td> <td>" + SirkeAdi + "</td><td>" + KonaklamaTipi + "</td><td>" + tatilVeri + "</td><td>" + baslangicTarihi + "</td><td>" + bitisTarihi + "</td><td>" + uret + "KaldigunSayisi + "</td></tr>"); //Tablonun sütun isimleri yazıldı
        sb.Append("</table>");

        4 references
        public override void UlasimBilgileriniGetir(int id)
        {
            var kalkisVeriID = seyahatBilgiManager.GetId(id).UlasimID;

            var adi = kullaniciManager.GetId(id).Adi;
            var soyadi = kullaniciManager.GetId(id).Soyadi;
            var ulasimTipi = ulasim.GetId(kalkisVeriID).UlasimTipi;
            var kalkisVeri = ulasim.GetId(kalkisVeriID).KalkisVeri;
            var VarisVeri = ulasim.GetId(kalkisVeriID).VarisVeri;
            var KalkisSaat = ulasim.GetId(kalkisVeriID).KalkisSaat;
            var VarisSaat = ulasim.GetId(kalkisVeriID).VarisSaat;
            var uret = ulasim.GetId(kalkisVeriID).Uret;
            sb.Append("<h3>ULASIM BİLGİLERİ</h3>"); //Tablonun basılışı yazıldı
            sb.Append("<table class='display' id='dt' style='align: left; width: 99%;'>");
            sb.Append("rules='all' border='1px'\"");
            sb.Append("cellspacing='0px' cellpadding='4px'");

            sb.Append("<thead>");
            sb.Append("<tr><td>"); 
            sb.Append("<tr style='background-color: Silver; color: #1E1E1E;'>"); 
            sb.Append("font-weight: bold");
            sb.Append("<tr> <th>Adı</th> <th>Soyadı</th> <th>Ulasim Tipi</th><th>Kalkis Verisi</th><th>Varis Verisi</th><th>kalkis saat</th><th>varis saat</th><th>Ucret</th><th></tr>"); //Tablonun sütun isimleri yazıldı
            sb.Append("<tr> <td>" + adi + "</td> <td>" + soyadi + "</td> <td>" + ulasimTipi + "</td> <td>" + kalkisVeri + "</td><td>" + VarisVeri + "</td><td>" + KalkisSaat + "</td><td>" + VarisSaat + "</td><td>" + uret + " 2 + "</td></tr>"); //Tablonun sütun isimleri yazıldı
            sb.Append("</table>");
```

ReportJSON Sınıfı

```
namespace BuilderRaporlama.Reports
{
    1 reference
    public class ReportJSON : ReportBuilderBase
    {
        UlasimBilgi ulasimBilgi = new UlasimBilgi();
        SeyhatBilgileri seyhatBilgi = new SeyhatBilgileri();
        KullaniciManager kullaniciManager = new KullaniciManager(new EFKullaniciDal());
        SeyhatBilgiManager seyhatBilgiManager = new SeyhatBilgiManager(new EFSeyhatBilgiDal());
        UlasimAracManager ulasim = new UlasimAracManager(new EFUlasimAracDal());
        KonaklamaBilgiManager konaklama = new KonaklamaBilgiManager(new EFKonaklamaDal());

        2 references
        public override void RaporKaydet()
        {
            System.IO.File.WriteAllText(@"C:\Users\EMRE\Desktop\json.json", sb.ToString());
        }
    }

    4 references
    public override void SeyhatBilgileriniGetir(int id)
    {
        var kaldigiGunSayisi = seyhatBilgiManager.GetId(id).RezervasyonBitis.Day - seyhatBilgiManager.GetId(id).RezervasyonBaslangic.Day;
        var konaklamaId = seyhatBilgiManager.GetId(id).KonaklamaID;

        seyhatBilgi.SeyhatAdi = "Konaklama Bilgileri";
        seyhatBilgi.Adi = kullaniciManager.GetId(id).Adi;
        seyhatBilgi.Soyadi = kullaniciManager.GetId(id).Soyadi;
        seyhatBilgi.Sirket_Adi = konaklama.GetId(konaklamaId).SirketAdi;
        seyhatBilgi.Konaklama_Tipi = konaklama.GetId(konaklamaId).KonaklamaTipi;
        seyhatBilgi.Tatil_Yeri = konaklama.GetId(konaklamaId).TatilYeri;
        seyhatBilgi.BaslangicTarihi = seyhatBilgiManager.GetId(id).RezervasyonBaslangic;
        seyhatBilgi.BitisTarihi = seyhatBilgiManager.GetId(id).RezervasyonBitis;
        seyhatBilgi.ucret = konaklama.GetId(konaklamaId).ucret*kaldigiGunSayisi;

        sb.Append(JsonConvert.SerializeObject(seyhatBilgi));
    }

    4 references
    public override void UlasimBilgileriniGetir(int id)
    {
        var kalkisYeriID = seyhatBilgiManager.GetId(id).UlasimID;
        ulasimBilgi.UlasimAdi = "Ulasim Bilgileri";
        ulasimBilgi.Adi = kullaniciManager.GetId(id).Adi;
        ulasimBilgi.Soyadi = kullaniciManager.GetId(id).Soyadi;
        ulasimBilgi.KalkisYeri = ulasim.GetId(kalkisYeriID).KalkisYeri;
        ulasimBilgi.VarisYeri = ulasim.GetId(kalkisYeriID).VarisYeri;
        ulasimBilgi.KalkisSaat = ulasim.GetId(kalkisYeriID).KalkisSaat;
        ulasimBilgi.VarisSaat = ulasim.GetId(kalkisYeriID).VarisSaat;
        ulasimBilgi.ucret = ulasim.GetId(kalkisYeriID).Ucret * 2;

        sb.Append(JsonConvert.SerializeObject(ulasimBilgi));
    }
}

class UlasimBilgi
{
    1 reference
    public string UlasimAdi { get; set; }
    1 reference
    public string Adi { get; set; }
    1 reference
    public string Soyadi { get; set; }
    1 reference
    public string KalkisYeri { get; set; }
    1 reference
    public string VarisYeri { get; set; }
    1 reference
    public string KalkisSaat { get; set; }
    1 reference
    public string VarisSaat { get; set; }
    1 reference
    public int ucret { get; set; }
}

2 references
class SeyhatBilgileri
{
    1 reference
    public string SeyhatAdi { get; set; }
    1 reference
    public string Adi { get; set; }
    1 reference
    public string Soyadi { get; set; }
    1 reference
    public string Sirket_Adi { get; set; }
    1 reference
    public string Konaklama_Tipi { get; set; }
    1 reference
    public string Tatil_Yeri { get; set; }
    1 reference
    public DateTime BaslangicTarihi { get; set; }
    1 reference
    public DateTime BitisTarihi { get; set; }
    1 reference
    public int ucret { get; set; }
}
```

