



# İSTANBUL OKAN ÜNİVERSİTESİ

İSTANBUL OKAN ÜNİVERSİTESİ  
MESLEK YÜKSEK OKULU  
BİLGİSAYAR PROGRAMCILIĞI  
PROJE PLANLAMA DERSİ  
PROJE RAPORU

**HAZIRLAYAN**

**AHMET BURAK GÜNDÜZ**

# İÇİNDEKİLER

## 1. Giriş

- a. Projede çalışılan konunun açıklanması.
- b. İncelenen bu konuda daha önceden yapılan çalışmaların özeti.

## 2. Kullanılan araçlar

- a. Proje hazırlanırken kullanılan (yazılım & donanım) araç ve teknolojilerin açıklanması.
- b. Proje geliştirme ortamı ve araçlarının hazırlanması.

## 3. Projenin hazırlanması (yazılım projesi için)

- a. Projenin planlanması
  - i. Kullanıcı türlerinin tanımlanması,
  - ii. Kullanıcı bazlı iş akış planı,
  - iii. Kullanılacak nesne(class), arabirim(interface), koleksiyon ve diğer araçların planlanması,
  - iv. Ekranların tasarımı,
  - v. Veritabanının tasarımı,
- b. Projede iş katmanının hazırlanması,
- c. Projenin birleştirilmesi, varsa sunucuya yüklenmesi
- d. Projenin Test edilmesi

## 4. Sonuç

- a. Başlangıçtaki hedeflerine ulaşma durumu,
- b. Elde edilen performansın değerlendirilmesi,
- c. İleriye dönük olarak iyileştirme ve geliştirme hedefleri,
- d. ...

## 5. Yararlanılan Kaynaklar

## 6. Ekler

- a. Projenin bulunduğu Github adresi,
- b. Veritabanı yapısı (SQL veritabanını oluşturan betik/script)
- c. Program Arayüzleri,
- d. Program kodları,
- e. Hazırlanan uygulamanın kullanım kılavuzu,
- f. Hazırlanan uygulamanın tek sayfalık tanıtım broşürü.

# Giriş

## a. Projede çalışılan konunun açıklanması.

Yapmış olduğum proje fitness salonlarının kolayca kayıt yapmak ve kayıtları görmek için kullanabileceği bir projedir.

## İncelenen bu konuda daha önceden yapılan çalışmaların özeti.

Bu konuda daha önce gözlemleme yaptım. Kendi gittiğim spor salonunu baz alarak yaptığım gözlemler sonucunda bu projeyi oluşturdum.

# Kullanılan araçlar

## a. Proje hazırlanırken kullanılan (yazılım & donanım) araç ve teknolojilerin açıklanması.

Projemi hazırlama sürecimde bilgisayarımda kullandığım programlar Visual Studio ve SQL Server Management Studio'dur. Bu programlarda Kullandığım dil ise; C# ve SQL'dir.

# Projenin hazırlanması

## a. Projenin planlanması

### i. Kullanıcı türlerinin tanımlanması

Kullanıcı erişimi sınırlı olarak belirlendi. Sadece o işten sorumlu olan, ID ve şifreyi bilen kişi giriş yapabiliyor ve sisteme gerekli kayıtları yapabiliyor.

### ii. Kullanıcı bazlı iş akış planı

Programı, sadece programı kullanan şirketin ulaşabilmesi amaçlanmıştır.

### iii. Kullanılacak nesne(class), arabirim(interface), koleksiyon ve diğer araçların planlanması

Nesnemi ilk sayfadaki kullanıcı girişı sayfasında kullandım.

#### iv. Ekranların tasarımı

Ekran tasarımları oldukça sade ve basitleştirilmiş bir şekilde yapılmıştır. Aşağıda örnekleri mevcuttur.

#### v. Veritabanının tasarımı

Veritabanı tasarımı yapılan projeye uygun olarak tasarlanıp kullanıcıya sunulması planlamıştır. Aşağıda örnekleri mevcuttur.

### b. Projede iş katmanının hazırlanması

Projemde oluşturduğum katman ekle katmanıdır.

### c. Projenin birleştirilmesi, varsa sunucuya yüklenmesi

Projenin tasarımını ve yazılımını Visual Studio 2019 uygulamasında yapıp, SQL Server Management Studio'dan oluşturduğum veri dopalama kısmıyla birleştirdim.

### d. Projenin Test edilmesi

Proje kendim tarafından test edilmiş olup, hiçbir sorun çıkarmadan çalıştığı gözlemlenmiştir.

## Sonuç

### a. Başlangıçtaki hedeflerine ulaşma durumu

Başlangıçtaki hedeflerimin çoğuna ulaştım. Sadece silme ve güncelleme kısmına ulaşamadım

### b. Elde edilen performansın değerlendirilmesi

Uygulama yeterli akıcılıkta çalışmaktadır.

### c. İleriye dönük olarak iyileştirme ve geliştirme hedefleri

Projemi ileriye dönük olarak geliştirme hedefleri içerisinde silme ve güncelleme yeteneği getirmek, daha gösterişli bir ekran tasarlamak vardır.

## Yararlanılan Kaynaklar

<https://www.yazilimkodlama.com>

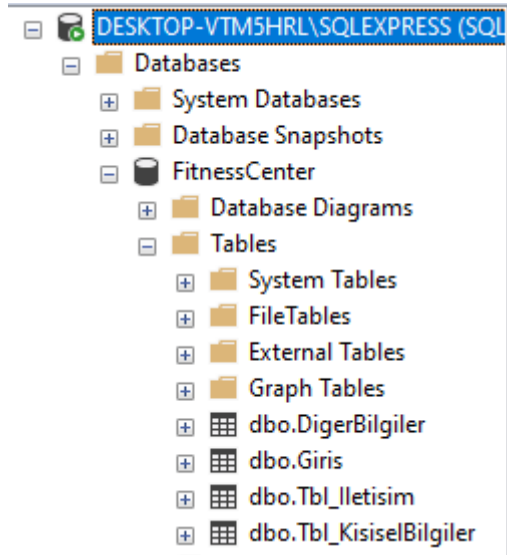
<http://www.btdersleri.com>

<https://www.youtube.com>

## Ekler

### a. Veritabanı yapısı (SQL veritabanını oluşturan betik/script)

SQL Server Management Studio uygulamasında bir adet FitnessCenter adında database oluşturup içerisinde 4 adet Tablo oluşturdum.



## b. Program Arayüzleri

MainWindow

Çalışırken Yeniden Yükleme kullanılabilir <

**GİRİŞ :**

ID :

Şifre :

**GİRİŞ**

KullanıcıGiris

Çalışırken Yeniden Yükleme kullanılabilir <

**KİŞİSEL BİLGİLER :**

ID :

Ad-Soyad :

Yaş :

Cinsiyet :

Boy-Kilo :

TC No :

**KAYDET** **DEVAM**



DigerGiris

Çalışırken Yeniden Yükleme kullanılabilir <

GERİ

DİĞER BİLGİLER :

ID :

Kayıt Tarihi :

Kayıt Süresi :

KAYDET

DEVAM

İletişimGiris

Çalışırken Yeniden Yükleme kullanılabilir

GERİ

**İLETİŞİM BİLGİLERİ :**

İC

Tel No :

E-mail :

KAYDET DEVAM

[illegible]



## c. Program kodları

### 1. Sayfanın kodları (MainWindow)

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows;
using System.Windows.Controls;
using System.Windows.Data;
using System.Windows.Documents;
using System.Windows.Input;
using System.Windows.Media;
using System.Windows.Media.Imaging;
using System.Windows.Shapes;
using System.Data.SqlClient;
using System.Data;

namespace FitnessCenter
{
    /// <summary>
    /// Giriş.xaml etkileşim mantığı
    /// </summary>
    2 bapvuru
    public partial class MainWindow : Window
    {
        0 bapvuru
        public MainWindow()
        {
            InitializeComponent();
        }

        public string conString = "Data Source=.\SQLEXPRESS;Initial Catalog=FitnessCenter;Integrated Security=True";

        1 bapvuru
        private void Giriş_Click(object sender, RoutedEventArgs e)
        {
            string ID = ID_txt.Text;
            string Sifre = Sifre_txt.Text;
            connection1 baglanti = new connection1();
            baglanti.GirisYap(ID, Sifre);
            this.Hide();
        }
    }
}
```

### 1. Sayfada kullanılan nesnenin class 1

```
using System.Threading.Tasks;
using System.Data.SqlClient;
using System.Data;
using System.Windows;

namespace FitnessCenter
{
    2 bapvuru
    class connection1
    {
        string conString = "Data Source=.\SQLEXPRESS;Initial Catalog=FitnessCenter;Integrated Security=True";

        1 bapvuru
        public void GirisYap (string ID , string Sifre)
        {
            SqlConnection con = new SqlConnection(conString);
            SqlDataAdapter sda = new SqlDataAdapter("Select Count(*) From Giris where ID='" + ID.ToString() + "' AND NAME ='" + Sifre.ToString() + "'", con);
            DataTable dt = new DataTable();
            sda.Fill(dt);
            if (dt.Rows[0][0].ToString() == "1")
            {
                KullanıcıGiris ss = new KullanıcıGiris();
                ss.Show();
            }
            else
            {
                MessageBox.Show("YANLIŞ ŞİFRE VEYA KULLANICI ADI GİRİŞİ!!! BİLGİLERİN DOĞRU OLDUĞUNA EMİN OLUP TEKRAR DENEYİNİZ!!!");
            }
        }
    }
}
```

## 2. Sayfa kodları(KullanıcıGiris)

```

public partial class KullaniciGiris : Window
{
    2 bagvuru
    public KullaniciGiris()
    {
        InitializeComponent();
    }

    public string conString = "Data Source=.\SQLEXPRESS;Initial Catalog=FitnessCenter;Integrated Security=True";

    1 bagvuru
    public void K_Bilgiler_Kaydet_Click(object sender, RoutedEventArgs e)
    {
        SqlConnection con = new SqlConnection(conString);
        con.Open();
        if (con.State == System.Data.ConnectionState.Open)
        {
            string q = "insert into Tbl_KisiselBilgiler (ID,AdSoyad,Yas,Cinsiyet,BoyKilo,TcNo) values ('" + ID_txt.Text.ToString() + "'" +
                ",'" + AdSoyad_txt.Text.ToString() + "','" + " +
                "'" + Yas_txt.Text.ToString() + "','" + " +
                "'" + Cinsiyet_txt.Text.ToString() + "'" +
                "','" + BoyKilo_txt.Text.ToString() + "','" + " +
                "'" + TcNo_txt.Text.ToString() + "')";
            SqlCommand cmd = new SqlCommand(q, con);
            cmd.ExecuteNonQuery();
            MessageBox.Show("Kayıt Başarıyla Olusturuldu!");
        }
    }

    1 bagvuru
    private void DEVAM_Click(object sender, RoutedEventArgs e)
    {
        this.Hide();
        IletisimGiris ss = new IletisimGiris();
        ss.Show();
    }
}

```

### 3. Sayfa Kodları(IletisimGiris)

```

1 2 bagyuru
public partial class IletisimGiris : Window
{
    2 bagyuru
    public IletisimGiris()
    {
        InitializeComponent();
    }
    public string conString = "Data Source=\\SQL EXPRESS;Initial Catalog=FitnessCenter;Integrated Security=True";

    1 bagyuru
    private void DEVAMI_Click(object sender, RoutedEventArgs e)
    {
        this.Hide();
        DigerGiris ss = new DigerGiris();
        ss.Show();
    }

    1 bagyuru
    private void Button_Click(object sender, RoutedEventArgs e)
    {
        SqlConnection con = new SqlConnection(conString);
        con.Open();
        if (con.State == System.Data.ConnectionState.Open)
        {
            string q = "insert into Tbl_Iletisim (ID1,TelNo,Email) values ('" + ID_txt1.Text.ToString() + "','"+ TelNo_txt.Text.ToString() + "','"+ Email_txt.Text.ToString() + "')";
            SqlCommand cmd = new SqlCommand(q, con);
            cmd.ExecuteNonQuery();
            MessageBox.Show("Kayıt Başarıyla Olusturuldu!");
        }
    }

    1 bagyuru
    private void I_Geri_Click(object sender, RoutedEventArgs e)
    {
        this.Hide();
        KullaniciGiris ss = new KullaniciGiris();
        ss.Show();
    }
}

```

## 4. Sayfa Kodları(DigerGiris)

```
8 bagirsu
public partial class DigerGiris : Window
{
    1 bagirsu
    public DigerGiris()
    {
        InitializeComponent();
    }
    public string conString = "Data Source=\\SQLEXPRESS;Initial Catalog=FitnessCenter;Integrated Security=True";

    1 bagirsu
    private void D_Bilgiler_Kaydet_Click(object sender, RoutedEventArgs e)
    {
        SqlConnection con = new SqlConnection(conString);
        con.Open();
        if (con.State == System.Data.ConnectionState.Open)
        {
            string q = "insert into DigerBilgiler (ID2,KayitTarihi,KayitSure) values ('" + ID_txt2.Text.ToString() + "','" + K_Tarihi_txt.Text.ToString() + "','" + K_Suresi_txt.Text.ToString() + "')";
            SqlCommand cmd = new SqlCommand(q, con);
            cmd.ExecuteNonQuery();
            MessageBox.Show("Kayıt Başarıyla Oluşturuldu!");
        }
    }

    1 bagirsu
    private void DEVAM2_Click(object sender, RoutedEventArgs e)
    {
        this.Hide();
        Kisi_Ara ss = new Kisi_Ara();
        ss.Show();
    }

    1 bagirsu
    private void Button_Click(object sender, RoutedEventArgs e)
    {
        this.Hide();
        IletisimGiris ss = new IletisimGiris();
        ss.Show();
    }
}
```

## 5. Sayfa kodları(Kisi\_Ara)

```
14 bagirsu
public partial class Kisi_Ara : Window
{
    1 bagirsu
    public Kisi_Ara()
    {
        InitializeComponent();
        FillDataGrid();
    }
    1 bagirsu
    private void FillDataGrid()
    {
        string ConString = ConfigurationManager.ConnectionStrings["ConString"].ConnectionString;
        string CmdString = string.Empty;
        using (SqlConnection con = new SqlConnection(ConString))
        {
            CmdString = "SELECT * FROM Tbl_KisiselBilgiler INNER JOIN Tbl_Iletisim ON Tbl_KisiselBilgiler.ID = Tbl_Iletisim.ID1 INNER JOIN DigerBilgiler ON Tbl_Iletisim.ID1 = DigerBilgiler.ID2";
            SqlCommand cmd = new SqlCommand(CmdString, con);
            SqlDataAdapter sda = new SqlDataAdapter(cmd);
            DataTable dt = new DataTable("tbl_iletisim");
            sda.Fill(dt);
            grdEmployee.ItemsSource = dt.DefaultView;
        }
    }
    1 bagirsu
    private void Button_Click(object sender, RoutedEventArgs e)
    {
        this.Hide();
        DigerGiris ss = new DigerGiris();
        ss.Show();
    }
}
```

[Başa dönmek için tıklayınız](#)

