

## 1 Proje İsmi

### 1 a. Projenin Açıklaması

Projemizin amacı isminden de anlaşılacağı üzere düşman tespit ve imhadır. Çalışma mantığına göre öncelikle ürün yerleştirildiği arandaki yüzleri kaydeder, veritabanı ile kıyaslar ve eşleşen görüntülerin herbiri için atış onayı istemi gönderir. Web uygulaması ve mobil uygulama üzerinde yetkili kullanıcılar atış onayı verebilirken Personeller sadece yetkililere atış onayı istemini yönlendirebilir. Ayrıca cihazın atış onayı almaksızın çalıştığı bir otomatik modu da bulunur.

### 1 b. Proje Ekibi

Tablo – 1: Proje Ekibi

Tablo – 1: Proje Ekibi

	Numara	Ad Soyad	Projedeki Görev Tanımı
1	160313065	Oğuzhan Akın	<ul style="list-style-type: none"><li>Veritabanı form uygulaması yapmaktır.</li><li>Raspberry pi kullanımına yardım etmektir.</li></ul>
2	160313004	Endercan Yılmaz	<ul style="list-style-type: none"><li>Mobil ön uç–arka uç geliştirmektir.</li><li>Veritabanı hazırlamaktır.</li><li>Raspberry pi kullanımına yardım etmektir.</li></ul>
3	160313003	Yasin Efe Eryılmaz	<ul style="list-style-type: none"><li>Web tasarım birimini geliştirmektir.</li><li>Raspberry pi kullanımına yardım etmektir.</li></ul>
4	160313024	Burak Yılmaz	<ul style="list-style-type: none"><li>Python geliştirmektir.</li><li>Raspberry pi kullanımına yardım etmektir.</li></ul>

### 1 c. Projenin Aşamaları

Tablo – 2: İş paketleri

No	İş Paketi Adı	Dönem	İçeriği
1	Planlama	17.10.2019 - 05.11.2019	
2	Analiz	08.11.2019 - 25.11.2019	
3	Tasarım	25.11.2019 - 22.12.2019	
4	Test Planları Oluşturulması	22.11.2019 - 17.01.2020	
6	Web Ön Uç ve Tasarım Geliştirilmesi	07.03.2020 – 20.03.2020	

7	Mobil Ön Uç Geliştirilmesi	07.03.2020 - 15.03.2020	
8	Yüz Tanıma Sistemi İlk Geliştirmeler	07.03.2020 - 20.03.2020	
9	Arka Uç Geliştirmelerin Yapılması	20.03.2020 – 15.04.2020	
10	Projenin Birleştirilmesi	30.04.2020 - 08.05.2020	
12	Karşılaşılan Sorunların Çözülmesi	12.05.2020 - 25.05.2020	
14	Yayına Alınma	27.05.2020 - 30.05.2020	

## 2 Dönem İçinde Yapılan İşler

### 2 a.

```

SqlConnection conn = new SqlConnection(@"Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\ASUS\source\repos\DTISVeritaba
nınaEkleme\DTIS_DB.mdf;Integrated Security=True");

private void button2_Click(object sender, EventArgs e)
{
    conn.Open();
    SqlCommand com = new SqlCommand("insert into dbo.dtis_db(adsoyad, kodadi, orgutu,
aranmaderecesi) values (@p1, @p2, @p3, @p4)", conn);
    com.Parameters.AddWithValue("@p1", textBox1.Text);
    com.Parameters.AddWithValue("@p2", textBox2.Text);
    com.Parameters.AddWithValue("@p3", textBox3.Text);
    com.Parameters.AddWithValue("@p4", textBox4.Text);
    //com.Parameters.AddWithValue("@p5", pictureBox1.Image);
    com.ExecuteNonQuery();
    conn.Close();
    MessageBox.Show("Kayıt Eklendi");
}

```

**Kod Örneği – 1: Veritabanına Ekleme**

## 2 b. Web Tasarım Geliştirme (Yasin Efe Eryılmaz)

Yetkili kullanıcı ve hedef ekleme sayfaları html ve css tarafları düzeltildi.Kullanıcı ekleme listbox ın tasarımında bazı

sorunlar ile karşılaşıldı üzerinde çalışılıyor.Rasperry pi donanım kısmında gerekli araçlar için araştırma yapıldı.

Talebinde sıkıntı çıkan bazı araçlar için araştırma devam ediyor.

```
<div class="field-wrap">
    <label>
        Hedef Adı Soyadı<span class="req">*</span>
    </label>
    <input type="Adısoyadı"required autocomplete="off"/>
</div>
<div class="field-wrap">
<label>Hedef Kod Adı<span class="req"></span></label>
<input type="kodadı"required autocomplete="off"/>
</div>
<div class="field-wrap">
<label>Suçlu Derecesi<span class=""></span></label>
<input type="derece"required autocomplete="off"/>
</div>
<button class="button button-block"/>Kaydet</button>
</form>
</div>
```

### Kod Örneği – 2: Web Tasarım

## 2 c. Veritabanı Form Application (Endercan Yılmaz)

Veritabanına, yüz tanımada kullanılacak fotoğrafların, Byte[] dizisiyle Binary olarak kayıt edilmesini sağlayan Form Application yapıldı.

```
OpenFileDialog file = new OpenFileDialog();
file.Title = "Resim";
file.Filter = "jpeg (*.jpg)/*.jpg|png (*.png)/*.png";
if (file.ShowDialog() == DialogResult.OK)
{
    pictureBox1.Image = Image.FromFile(file.FileName);
    resimpath = file.FileName.ToString();
}
```

```

FileStream fs = new FileStream(resimpath, FileMode.Open, FileAccess.Read);
BinaryReader br = new BinaryReader(fs);
byte[] resim = br.ReadBytes((int)fs.Length);
cmd.Parameters.Add("@image", SqlDbType.Image, resim.Length).Value = resim;
try
{
    bag.Open();
    cmd.ExecuteNonQuery();
    MessageBox.Show("Unsur Veritabanına Kaydedildi");
}

```

### Kod Örneği – 3: Veritabanı Image Ekleme

#### 2 d. Yüz Tanıma Sistemi Geliştirme (Burak Yıldırım)

Burada face detection için gerekli olan konsol yaratıldı. 3 aşamada yapmayı planlıyorduk bunlar; 1) Görseli yakalamak ve en az 15 tane screenshot loglamak. 2) Training; görsellerin eğitim kısmı. 3) Tracking; kadrja giren kişinin tespit edilmesi kısmı. Loglama kısmı tamamen bitti. Training ‘in büyük kısmı tamamlandı. Fakat şuan alınan ufak hatalar temizleniyor.

```

from tkinter import *

window = Tk()
window.title("Face Tracking System")
window.geometry("400x200")
idLabel = Label(window,text = "Enter ID")
idLabel.grid(row=0,column=0)
idText = Entry(window,width="8")
idText.grid(row=0,column=1)
nameLabel = Label(window,text="Enter the name")
nameText = Entry(window,width="15")
nameLabel.grid(row=1,column=0)
nameText.grid(row=1,column=1)
notfLabel = Label(window,text = "Notifications : ")
notfLabel.grid(row=3,column=0)
notf = Label(window,text="")
notf.grid(row=3,column=1)
takeImgs = Button(window,text="Take Images",command=TakeImages)
takeImgs.grid(row=0,column=9)
trainImgs = Button(window,text="Train Images",command=TrainingImages)

```

```
trainImgs.grid(row=1,column=9)
trackImgs = Button(window,text="Track Images",command="")
trackImgs.grid(row=2,column=9)
detectLabel = Label(window,text = "Detecting person : ")
detectLabel.grid(row=5,column=0)
detect = Label(window,text="")
detect.grid(row=5,column=1)
window.mainloop()
```

#### **Kod Örneği – 4: Yüz Tanıma**

### 3 Sonuç

#### 3 a. Başarı ile tamamlanan işler

Web tarafında yetkili hedef ekleme ve yetkili kullanıcı ekleme sayfası oluşturuldu. Ayarlar sayfası düzelti. Yüz tanıma tarafında ise hedefin sistemde loglanması ve loglanan dataların training kısmı yapıldı ve bunun için gerekli konsol kısmı hazırlandı. Masaüstü tarafında ise veritabanı oluşturuldu ve veritabanına fotoğraflar binary formatında eklenebilir ve yüz tanıma sistemi tarafından çekilebilir hale getirildi.

#### 3 b. Başarı ile tamamlanamayan işler

Face recognition için training tarafı yarım kaldı geliştirmeye devam ediliyor.

Veritabanının localhosttan web hostinge taşınması yarım kaldı.

.Kullanıcı ekleme listbox ın tasarımında bazı sorunlar ile karşılaşıldı üzerinde çalışılıyor.Rasperry pi donanım talebinde sıkıntı çıkan bazı araçlar için araştırma devam ediyor.

#### 3 c. Kişilerin görevlerini yerine getirme durumu

	Numara	Ad Soyad	Projedeki Görevini Yerine Getirme Durumu
1	160313065	Oğuzhan Akın	<ul style="list-style-type: none"><li>Veritabanı form uygulaması geliştirme görevinin projenin bu aşamasında yapması gereken kısmını %100 oranında yerine getirmiştir.</li></ul>
2	160313004	Endercan Yılmaz	<ul style="list-style-type: none"><li>Veritabanı form uygulaması geliştirme görevinin projenin bu aşamasında yapması gereken kısmını %100 oranında yerine getirmiştir.</li></ul>
3	1603130	Yasin Efe Eryılmaz	<ul style="list-style-type: none"><li>Web tasarım görevinin projenin bu aşamasında yapması gereken kısmını %100 oranında yerine getirmiştir.</li></ul>
4	1603130	Burak Yıldırım	<ul style="list-style-type: none"><li>Yüz tanıma sistemi geliştirme görevinin projenin bu aşamasında yapması gereken kısmını %100 oranında yerine getirmiştir.</li></ul>