



# SENSEİ FEDERAL

DJANGO İLE GELİŞTİRİLMİŞ WEB PROJESİ

## BLOG WEB SİTESİ

Sensei Federal ile Dövüşde çığır açmaya hazır mısın...

## Selçuk GÜNDÜÇ

(Tüm hakları saklıdır.)

## İÇİNDEKİLER

1. ÖZET
2. VERİ TABANI TABLO TİPLERİ
3. VERİTABANI ÖRNEK DATA GÖRÜNTÜLERİ
4. LOGİN SAYFASI ve VALIDATION
5. REGISTER SAYFASI ve VALIDATION
6. LOGOUT SAYFASI ve PROJE İMPORİT
7. ANASAYFA VE KODLARI
8. HAKKIMIZDA VE İLETİŞİM SAYFALARI VE KODLARI
9. PROGRAMLAR SAYFASI VE KODLARI

## ÖZET

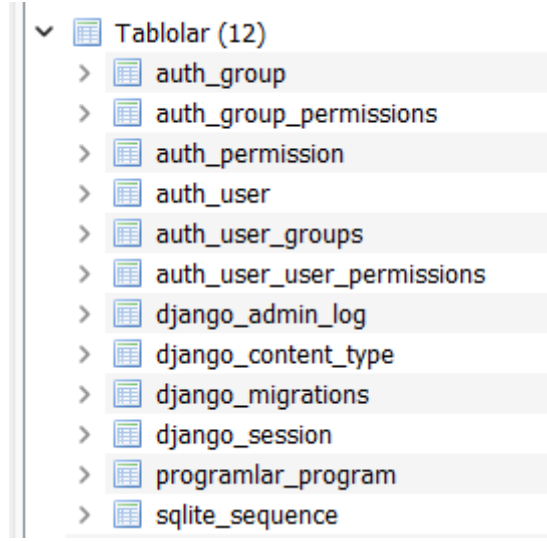
### SENSEİ FEDERAL

Bu proje bir blog web sitesi projesidir. İçeriği ise çoğunlukla Uzakdoğu sporları ile alakalıdır. Dövüş teknikleri öğrenmek isteyen, sağlık için spor yapan veya sporlar ile alakalı bilgi sahibi olmak isteyen herkes bizim sitemizi ziyaret edebilir. Doğal yollar ile veya spor salonunda spor yapan herkesin, içerisinde gerekli bilgiyi ve antrenman programlarını bulabileceği bir web site olma yolunda emin adımlarla ilerliyoruz. Her kullanıcıya özel farklı antrenman programları sunmayı hedefliyoruz. Kullanıcı verileri doğrultusunda kusursuz forma erişmesi için özel antrenman programlarımız ile kullanıcımızın mükemmel görünümüne ulaşması için sağlam bir köprü oluyoruz...

Projemde PYTHON dilinin bir uzantısı olan DJANGO dilini ve SQLite veritabanını kullanıyorum.

Projemi Visual Studio idesini kullanarak yazıyorum ve DB Browser for SQLite uygulamasını kullanıyorum.

## VERİ TABANI TABLO TİPLERİ

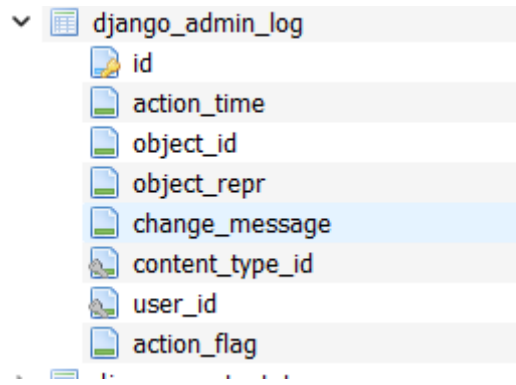


Tablolarımın genel görünümü yukarıdaki gibidir.

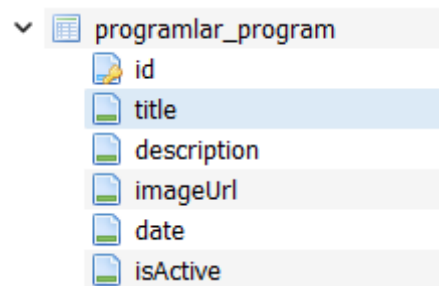
A screenshot of a database interface showing the structure of the 'auth\_user' table. The table has 11 columns with their respective data types. Each column name is preceded by a small icon of a document with a grid.

id	integer
password	varchar(128)
last_login	datetime
is_superuser	bool
username	varchar(150)
last_name	varchar(150)
email	varchar(254)
is_staff	bool
is_active	bool
date_joined	datetime
first_name	varchar(150)

Auth\_user ; Tablosu bize kullanıcının id sini, kullanıcı adını, şifresini(sansürlü bir biçimde), e mail adresini, programa ne zaman katıldığını, en son ne zaman giriş yaptığını ve ne zaman ayrıldığı gibi bir çok şeyi içerisinde saklayan tablodur.



Django\_admin\_log ; Adminin yani Superuserin yapmış olduğu eylemleri içerisinde barındıran bir veri tabanı tablosudur.



Programlar\_program ; Siteye yönetici tarafından yükleme yapılan verilerin saklandığı veri tabanı tablosudur.










## VERİ TABANI ÖRNEK DATA TİPLERİ

Tablo: auth_permission				
	id	content_type_id	codename	name
	Filtre	Filtre	Filtre	Filtre
1	1	1	add_logentry	Can add log entry
2	2	1	change_logentry	Can change log entry
3	3	1	delete_logentry	Can delete log entry
4	4	1	view_logentry	Can view log entry
5	5	2	add_permission	Can add permission
6	6	2	change_permission	Can change permission
7	7	2	delete_permission	Can delete permission
8	8	2	view_permission	Can view permission
9	9	3	add_group	Can add group
10	10	3	change_group	Can change group
11	11	3	delete_group	Can delete group
12	12	3	view_group	Can view group
13	13	4	add_user	Can add user
14	14	4	change_user	Can change user
15	15	4	delete_user	Can delete user










Auth\_permission ; Admin kullanıcısının site içerisinde yaptığı her hareketi kayıt altına alan bir veri saklama birimidir.

Tablo: auth_user											
	id	password	last_login	is_superuser	username	last_name	email	is_staff	is_active	date_joined	first_name
	Filtre	Filtre	Filtre	Filtre	Filtre	Filtre	Filtre	Filtre	Filtre	Filtre	Filtre
1	1	pbkdf2_sha256\$390000\$...	2024-03-21 01:24:58.238336	1	Federal		slckcetinoglu@gmail.com	1	1	2024-03-07 10:27:36.732941	
2	8	pbkdf2_sha256\$390000\$...	2024-03-21 03:51:21.278198	0	Selçuk		selcuk.gndc@gmail.com	0	1	2024-03-21 03:50:44.257443	

Auth\_user ; Siteye her kayıt olan kullanıcının verileri eş zamanlı olarak buraya düşer. Kayıt olmuş kullanıcıların username, userId, e-mail gibi kişisel verilerinin ve kullanıcının sitedeki hareketlerini kayıt alıp depoladığı alandır. Ne zaman giriş yapıp ne zaman çıkış yaptığı da dahil...

Tablo:	<div> <div>django_admin_log</div> <div>          </div> </div> <div>Filter in any column</div>							
	id	action_time	object_id	object_repr	change_message	content_type_id	user_id	action_flag
	Filtre	Filtre	Filtre	Filtre	Filtre	Filtre	Filtre	Filtre
1	1	2024-03-08 08:57:11.646138	2	beyazmed	[{"added": {}}]	4	1	1
2	2	2024-03-08 11:54:35.563358	3	DAYI	[{"added": {}}]	4	1	1
3	3	2024-03-09 10:31:17.356139	4	özgürbabapro	[{"added": {}}]	4	1	1
4	4	2024-03-20 10:58:23.855172	3	DAYI		4	1	3
5	5	2024-03-20 10:58:23.863180	2	beyazmed		4	1	3
6	6	2024-03-20 10:58:23.868080	4	özgürbabapro		4	1	3

Django\_admin\_log ; Admin kişinin eklediği , Düzenlediği ve sildiği kullanıcı verileri buraya depolanır.

Tablo:	<div> <div>django_migrations</div> <div>          </div> </div> <div>Filter in any</div>			
	id	app	name	applied
	Filtre	Filtre	Filtre	Filtre
1	1	contenttypes	0001_initial	2024-03-07 09:40:53.495824
2	2	auth	0001_initial	2024-03-07 09:40:53.510785
3	3	admin	0001_initial	2024-03-07 09:40:53.520221
4	4	admin	0002_logentry_remove_auto_add	2024-03-07 09:40:53.530139
5	5	admin	0003_logentry_add_action_flag_choices	2024-03-07 09:40:53.537969
6	6	contenttypes	0002_remove_content_type_name	2024-03-07 09:40:53.550523
7	7	auth	0002_alter_permission_name_max_length	2024-03-07 09:40:53.557400
8	8	auth	0003_alter_user_email_max_length	2024-03-07 09:40:53.566913
9	9	auth	0004_alter_user_username_opts	2024-03-07 09:40:53.581598
10	10	auth	0005_alter_user_last_login_null	2024-03-07 09:40:53.590341
11	11	auth	0006_require_contenttypes_0002	2024-03-07 09:40:53.593276
12	12	auth	0007_alter_validators_add_error_messages	2024-03-07 09:40:53.599307
13	13	auth	0008_alter_user_username_max_length	2024-03-07 09:40:53.606953
14	14	auth	0009_alter_user_last_name_max_length	2024-03-07 09:40:53.615394
15	15	auth	0010_alter_group_name_max_length	2024-03-07 09:40:53.621807
16	16	auth	0011_update_proxy_permissions	2024-03-07 09:40:53.628833
17	17	auth	0012_alter_user_first_name_max_length	2024-03-07 09:40:53.636339
18	18	sessions	0001_initial	2024-03-07 09:40:53.642849
19	19	programlar	0001_initial	2024-03-07 09:43:18.748540

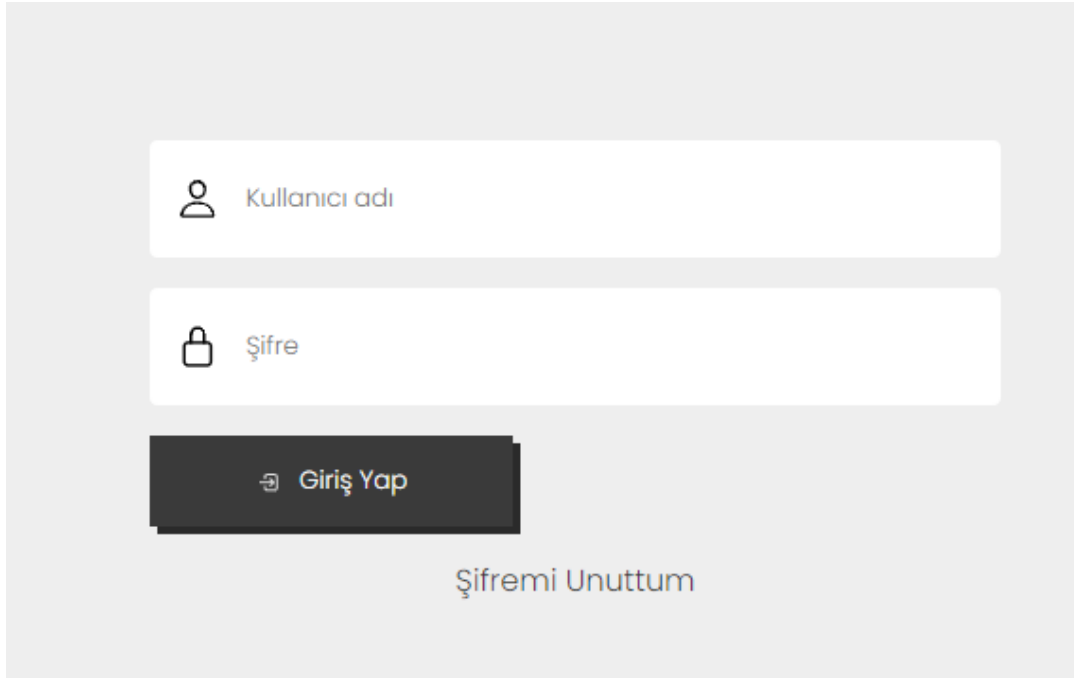
Django\_migrations ; Site tasarlayıcısı ve admin kullanıcısının site içerisinde yaptığı kapalı kaynak değişiklikleri depolayan yerdir. Örneğin : Site tasarlayıcısının koyduğu validation kuralları burada depolanır.

Tablo: programlar_program						
	id	title	description	imageUrl	date	isActive
	Filtre	Filtre	Filtre	Filtre	Filtre	Filtre
1	1	boks antrenmanı	Boks, iki kişinin bok...	1.jpg	2024-03-07	1
2	2	kickboks ...	Kick boks, yumruk, ...	2.jpg	2024-03-07	1
3	3	muay-thai ...	Muay-Thai diğer ...	3.jpg	2024-03-07	1

Programlar\_program ; Site tasarlayıcısı ve admin kullanıcısının siteye eklediği verilerin kayıt altına alındığı gruptur.



## LOGİN SAYFASI VE KODLARI



The image shows a login form with a light gray background. It contains two input fields: the first is labeled 'Kullanıcı adı' (Username) with a person icon, and the second is labeled 'Şifre' (Password) with a lock icon. Below these fields is a dark gray button with a right-pointing arrow and the text 'Giriş Yap' (Login). At the bottom, there is a link that says 'Şifremi Unuttum' (Forgot my password).

```
1  from django.shortcuts import redirect, render
2  from django.contrib.auth import authenticate, login, logout
3  from django.contrib.auth.models import User
4
5  def user_login(request):
6      if request.user.is_authenticated:
7          return redirect("index")
8
9      if request.method == "POST":
10         username = request.POST["username"]
11         password = request.POST["password"]
12
13         error = False
14         msg = ""
15
16         if username == "":
17             error = True
18             msg += "Kullanıcı Adı Zorunlu bir alan."
19
20         if error:
21             return render(request, "account/login.html", {"error": True, "msg": msg})
22
23         user = authenticate(request, username=username, password=password)
24
25         if user is not None:
26             login(request, user)
27             return redirect("index")
28         else:
29             return render(request, "account/login.html", {"error": "Username ya da Parola yanlış"})
30
31     else:
32         return render(request, "account/login.html")
```

Yukarıda gördüğünüz Login (Giriş Yapma) ekranımın backend kodlarıdır. Öncelikle kullanacağım metodları yukarıya import ediyorum. Sonrasında User\_login kodlarımı yazıyorum. Kodlara birkaç koşul belirterek aşağıdaki görsellerde gördüğünüz validation işlemlerimi satırlara işliyorum.

Kullanıcı boş bir form gönderdiği zaman aşağıdaki uyarı ile karşılaşır ve isteği geri çevrilir.

Kullanıcı Adı Zorunlu bir alan.

True

Şifremi Unuttum

Hatalı veri girişinde ise Yanlış veri uyarısı alır ve isteği geri çevrilir.

Username ya da Parola yanlış

Şifremi Unuttum

Başarılı girişte ise erişimin arttığı Anasayfa ya yönlendirilir.

```

<!DOCTYPE html>
<html lang="en">
> <head> ...
</head>
<body>

  <div class="login_forms">
    {% if error %}
    <div class="alert alert-danger">
      | {{msg}}
    </div>
    {% endif %}
    <form method="post" action="{% url 'user_login' %}">
      | {% csrf_token %}


    <div class="login_box">
      <ion-icon class="fs-20" name="person-outline"></ion-icon>
      <input type="text" name="username" placeholder="Kullanıcı Adı" autocomplete="off">
    </div>
    <div class="login_box">
      <ion-icon class="fs-20" name="lock-closed-outline"></ion-icon>
      <input type="password" name="password" placeholder="Şifre">
    </div>
    {% if error %}
    <div class="alert alert-danger">
      | {{ error }}
    </div>
    {% endif %}
    <button type="submit" class="btn-login">
      <ion-icon name="log-in-outline"></ion-icon>
      <span>Giriş Yap</span>
    </button>
  </form>
  <a class="forgot-password" href="#">Şifremi Unuttum</a>
</div>


  <script type="module" src="https://unpkg.com/ionicons@5.5.2/dist/ionicons/ionicons.esm.js"></script>
  <script nomodule src="https://unpkg.com/ionicons@5.5.2/dist/ionicons/ionicons.js"></script>
</body>
</html>


```


Yukarıdaki kodlar ise Login sayfamın Html kodlarının bir kısmıdır. Validation ise “{% if error %}” – “{% endif %}” blokları arasında msg bağlantısı sayesinde sağlanmakta.


## 4. REGİSTER SAYFASI

 Kullanıcı adı

 Email

 Şifre

 Şifreyi tekrarla

 Kayıt Ol

Hesabınız varsa buradan giriş yapın

```
def user_register(request):
    if request.method == "POST":
        username = request.POST.get("username")
        email = request.POST.get("email")
        password = request.POST.get("password")
        repassword = request.POST.get("repassword")

        error = False
        msg = ""

        if username == "":
            error = True
            msg += "Kullanıcı Adı Zorunlu bir alan."

        if len(password) < 5:
            error = True
            msg += " Şifre Minimum 5 karakter olmalı"

        if error:
            return render(request, "account/register.html", {"error": True, "msg": msg})

        if password == repassword:
            if User.objects.filter(username=username).exists():
                return render(request, "account/register.html", {"error": "Kullanıcı adı kullanılıyor."})

            elif User.objects.filter(email=email).exists():
                return render(request, "account/register.html", {"error": "Email hesabı kullanılıyor."})
            else:
                user = User.objects.create_user(username=username, email=email, password=password)
                user.save()
                return redirect("user_login")
        else:
            return render(request, "account/register.html", {"error": "Parolalar Eşleşmiyor."})
    else:
        return render(request, "account/register.html")
```

Yukarıda ise Register sayfamın görüntüsü ve Backend kodları bulunmakta. Hemen hemen Login kısmı ile aynı metodları kullandım. Doğal olarak Register kısmı daha çok veri istediğinden ötürü daha çok veri talebi ve doğrulama yapmak gerekti. Yine Login sayfasında yaptığım gibi “if – else – elif” metodlarıyla validation işlemlerimi entegre ettim.

Aşağıda Görmüş olduğunuz gibi Kullanıcı adı kısmını zorunlu kıldım ve kullanıcı adı olmadan gönderilen istekler sonuçsuz kalıyor.

Kullanıcı Adı Zorunlu bir alan.

Hesabınız varsa buradan giriş yapın

Aşağıda ise 3 haneli şifre girmek isteyen kullanıcının reddedilen isteğini görmekteyiz. Bu arada alt kısımda bulunan “Hesabınız varsa buradan giriş yapın” butonu aktif olarak çalışmakta.

Şifre Minimum 5 karakter olmalıdır

Hesabınız varsa buradan giriş yapın

```

1 <!DOCTYPE html>
2 <html lang="en">
3 <head> ...
83 </head>
84 <body>
85
86 <div class="register_forms">
87     {% if error %}
88     <div class="alert alert-danger">
89         {{msg}}
90     </div>
91     {% endif %}
92     <form method="post" action="{% url 'user_register' %}">
93         {% csrf_token %}
94         <div class="register_box">
95             <ion-icon class="fs-20" name="person-outline"></ion-icon>
96             <input type="text" name="username" placeholder="Kullanıcı Adı" autocomplete="off">
97         </div>
98         <div class="register_box email">
99             <ion-icon class="fs-20" name="mail-outline"></ion-icon> <!-- Email simgesi -->
100             <input type="email" name="email" placeholder="Email">
101         </div>
102         <div class="register_box">
103             <ion-icon class="fs-20" name="lock-closed-outline"></ion-icon>
104             <input type="password" name="password" placeholder="Şifre">
105         </div>
106         <div class="register_box">
107             <ion-icon class="fs-20" name="lock-closed-outline"></ion-icon>
108             <input type="password" name="repassword" placeholder="Şifreyi tekrarla">
109         </div>
110         <button type="submit" class="btn-register">
111             <ion-icon name="person-add-outline"></ion-icon>
112             <span>Kayıt Ol</span>
113         </button>
114     </form>
115     <div class="links">
116         <a href="{% url 'user_login' %}">Hesabınız varsa buradan giriş yapın</a>
117     </div>
118 </div>
119
120 <script type="module" src="https://unpkg.com/ionicons@5.5.2/dist/ionicons/ionicons.esm.js"></script>
121 <script nomodule src="https://unpkg.com/ionicons@5.5.2/dist/ionicons/ionicons.js"></script>
122 </body>
123 </html>
124

```

Burada ise Register sayfamızın Html kodlarını görmekteyiz. Yine Loginde yaptığımız gibi “{% if error %}-{% endif %}” bloğu arasında validation bağlantısı bulunmakta.

```

<div class="links">
    <a href="{% url 'user_login' %}">Hesabınız varsa buradan giriş yapın</a>
</div>

```

Bu kod ise bizi doğrudan Login sayfasına aktarıyor ve kullanıcıya büyük ölçüde kolaylık sağlıyor.

## LOGOUT SAYFASI ve PROJE İMPORT

```
def user_logout(request):  
    logout(request)  
    return redirect("index")
```

Logout metodu bizi doğrudan anasayfaya göndermekte.

```
healtylife > account > urls.py > ...  
1  from django.urls import path  
2  from . import views  
3  
4  
5  urlpatterns = [  
6      path('login', views.user_login, name="user_login"),  
7      path('register', views.user_register, name="user_register"),  
8      path('logout', views.user_logout, name="user_logout"),  
9  ]  
10
```

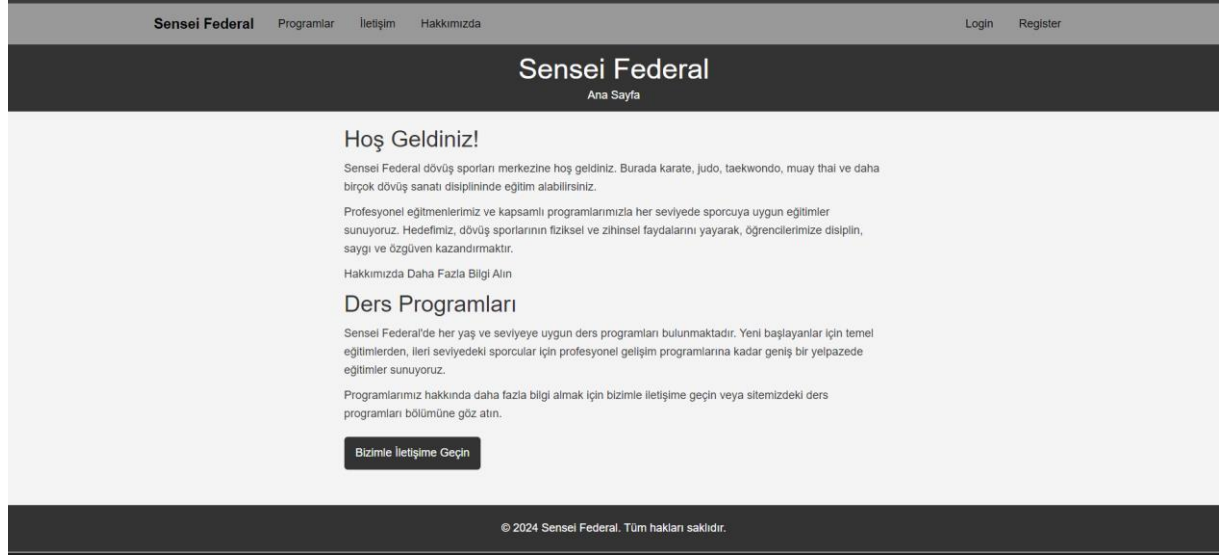
Bu gördüğünüz sayfaların çalışması için bütün sayfalara tanımladığım isim ve bağlantılarımı projemin “urls.py” dosyama import ediyorum.

```
INSTALLED_APPS = [  
    'django.contrib.admin',  
    'django.contrib.auth',  
    'django.contrib.contenttypes',  
    'django.contrib.sessions',  
    'django.contrib.messages',  
    'django.contrib.staticfiles',  
    'programlar',  
    'pages',  
    'account',  
    'templates'
```

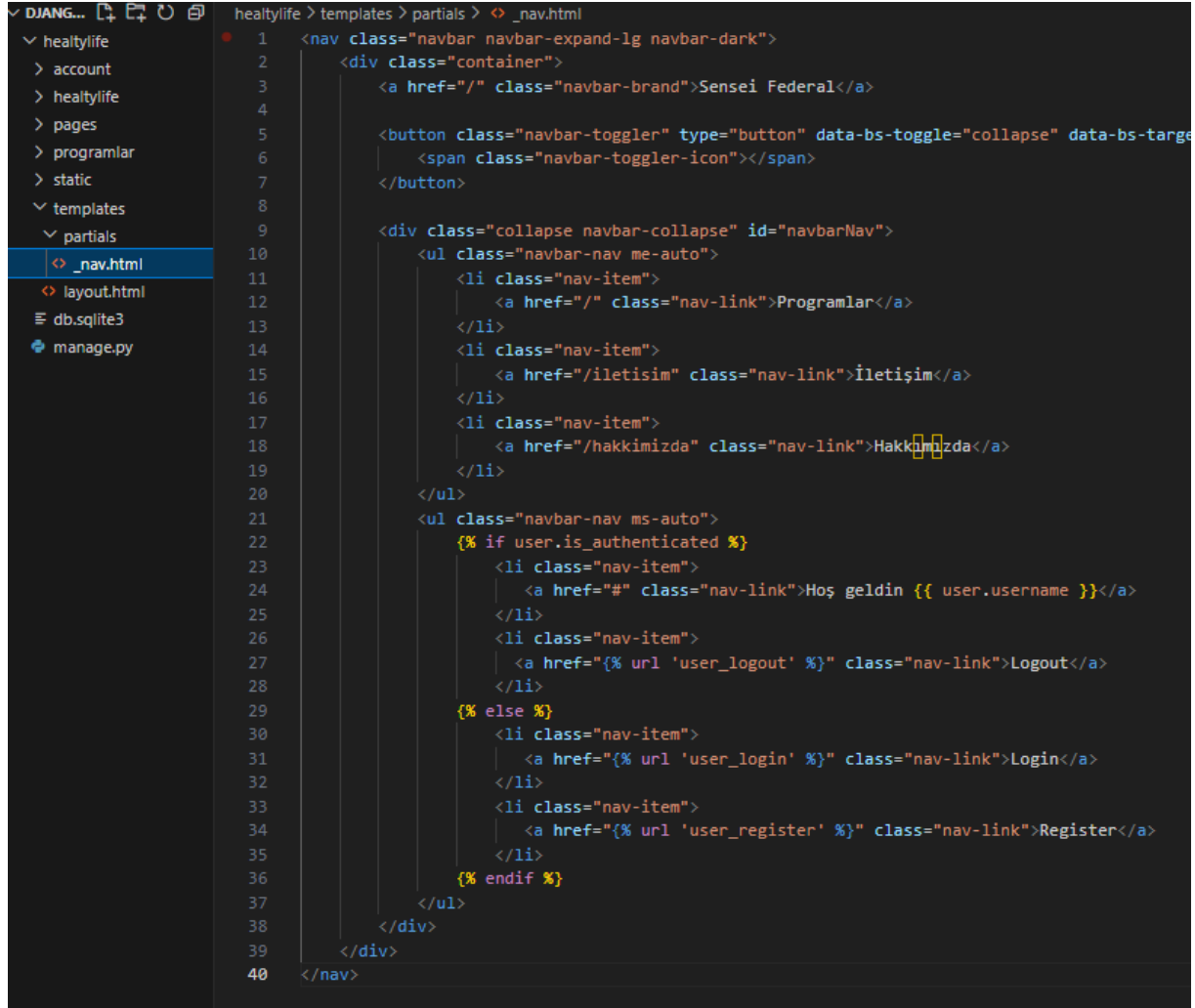
```
TEMPLATES = [  
    {  
        'BACKEND': 'django.template.backends.django.DjangoTemplates',  
        'DIRS': [  
            BASE_DIR / 'templates',  
            BASE_DIR / 'partials',  
            BASE_DIR / 'login.html',  
            BASE_DIR / 'register.html'
```

Ve yukarıdaki gibi ana projemin settings.py dosyasına da import ediyorum.

# ANASAYFA



Projeye girdikten sonra yukarıdaki anasayfa bizi karşılıyor. Yukarıda bulunan “navbar” bölümü girdiğim tüm sekmelerde görünsün diye “templates/layout” altında bir “navbar.html” dosyası yazıp projeme import ettim.





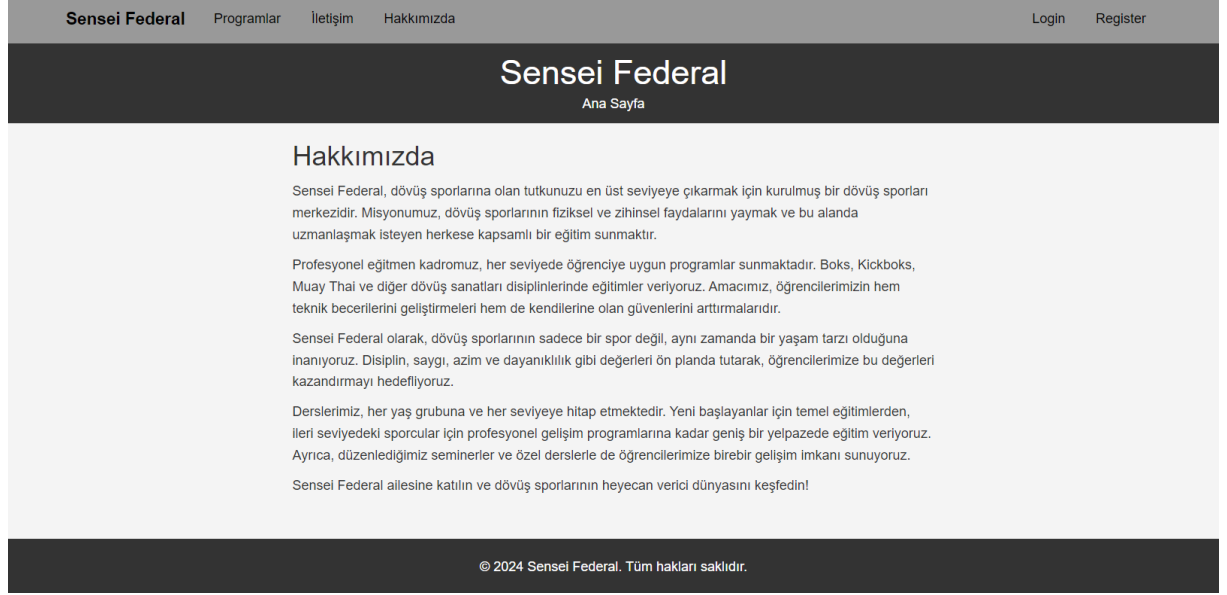
Yukarıda da navbar'ımın kodları bulunuyor.

```
healtylife > templates > layout.html
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Sensei Federal | {% block page_title %}{% endblock page_title %}</title>
7   <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/bootstrap/5.3.3/css/bootstrap.min.css">
8   <style>
9     /* Custom CSS styles */
10    body {
11      background-color: white; /* Beyaz arka plan rengi */
12      color: black; /* Siyah metin rengi */
13    }
14    .navbar {
15      background-color: #999; /* Gri tonlarından bir navbar arka plan rengi */
16    }
17    .navbar-brand {
18      font-weight: bold;
19      color: black; /* Siyah navbar marka metin rengi */
20    }
21    .navbar-nav .nav-link {
22      color: black; /* Siyah navbar link metin rengi */
23    }
24    .list-group-item {
25      background-color: #999; /* Gri tonlarından bir list group item arka plan rengi */
26      color: black; /* Siyah list group item metin rengi */
27      border-color: white; /* Belirgin beyaz çizgi rengi */
28      font-weight: bold; /* Kalın font */
29    }
30
31    .list-group-item:hover {
32      background-color: #777; /* List group item hover arka plan rengi */
33    }
34  </style>
35 </head>
36 <body>
37
38   <!-- Navbar -->
39   {% include 'partials/_nav.html' %}
40
41   {% block content %}{% endblock content %}
42
43   <script src="https://cdnjs.cloudflare.com/ajax/libs/bootstrap/5.3.3/js/bootstrap.bundle.min.js"></script>
44 </body>
45 </html>
46
```

Yukarıda ise anasayfamın ve navbar'ımın tasarım kodları bulunuyor. Rengini ve buton boyutlarını bu kodlar ile ayarladım.

# HAKKIMIZDA VE İLETİŞİM SAYFALARI VE KODLARI

## 1: HAKKIMIZDA SAYFASI



Yukarıda yazdığım Sensei Federal sitesine ait sayfayı görüyorsunuz.

Ve bu sayfanın kodları aşağıdaki şekildedir.

```
eaitylife > pages > templates > pages > <> hakkimizda.html
1  {% extends 'layout.html' %}
2
3  {% block page_title %}
4      Hakkımızda
5  {% endblock page_title %}
6
7  {% block content %}
8      <!DOCTYPE html>
9      <html lang="tr">
10         <head>
11             <meta charset="UTF-8">
12             <meta name="viewport" content="width=device-width, initial-scale=1.0">
13             <title>Hakkımızda - Sensei Federal</title>
14         </head>
15         <body>
16             <header>
17                 <h1>Sensei Federal</h1>
18                 <nav>
19                     <ul>
20                         <li><a href="{% url 'index' %}">Ana Sayfa</a></li>
21                         <!-- Diğer menü öğeleri eklenebilir -->
22                     </ul>
23                 </nav>
24             </header>
25
26             <main>
27                 <section>
28                     <h2>Hakkımızda</h2>
29                     <p>Sensei Federal, dövüş sporlarına olan tutkunuzu en üst seviyeye çıkarmak için kurulmuş bir dövüş sporları merkezidir. Misyonumuz profesyonel eğitmen kadromuz, her seviyede öğrenciye uygun programlar sunmaktır. Boks, Kickboks, Muay Thai ve diğer dövüş sanatları disiplinlerinde eğitimler veriyoruz. Amacımız, öğrencilerimizin hem teknik becerilerini geliştirmeleri hem de kendilerine olan güvenlerini arttırmalarıdır. Sensei Federal olarak, dövüş sporlarının sadece bir spor değil, aynı zamanda bir yaşam tarzı olduğuna inanıyoruz. Disiplin, saygı, azim ve dayanıklılık gibi değerleri ön planda tutarak, öğrencilerimize bu değerleri kazandırmayı hedefliyoruz. Derslerimiz, her yaş grubuna ve her seviyeye hitap etmektedir. Yeni başlayanlar için temel eğitimlerden, ileri seviyedeki sporcular için profesyonel gelişim programlarına kadar geniş bir yelpazede eğitim veriyoruz. Ayrıca, düzenlediğimiz seminerler ve özel derslerle öğrencilerimize birebir gelişim imkanı sunuyoruz. Sensei Federal ailesine katılın ve dövüş sporlarının heyecan verici dünyasını keşfedin!</p>
30                 </section>
31             </main>
32
33             <footer>
34                 <p>&copy; 2024 Sensei Federal. Tüm hakları saklıdır.</p>
35             </footer>
36         </body>
37     </html>
38  {% endblock content %}
```

## 2: İLETİŞİM SAYFASI

Sensei Federal

Programlar

İletişim

Hakkımızda

Login

Register

Sensei Federal

Ana Sayfa

İletişim

Adınız:

E-posta:

Mesajınız:

Gönder

© 2024 Sensei Federal. Tüm hakları saklıdır.

Yukarıda görüğünüz iletişim sayfası, siteyi kullananların her an site yöneticileri ile iletişime geçebilmesine imkan sağlayan bir iletişim bağlantı sayfasıdır.

Ve kodları aşağıdaki görseldedir.

```

healyife > pages > templates > pages > <i>iletisim.html</i>
1      {% extends "layout.html" %}
2
3      {% block page_title %}
4          iletisim
5      {% endblock page_title %}
6
7      {% block content %}
8      <i><!DOCTYPE html>
9      <i><html lang="tr">
10     <i><head>
11         <i><meta charset="UTF-8">
12         <i><meta name="viewport" content="width=device-width, initial-scale=1.0">
13         <i><title>iletisim - Sensei Federal</title>
14         <i><% load static %>
15         <i><link rel="stylesheet" href="{% static 'css/style.css' %}">
16     <i></head>
17     <i><style>--
112    <i>[?]style=[?]</i>
113     <i><body>
114         <i><header>
115             <i><h1>Sensei Federal</h1>
116             <i><nav>
117                 <i><ul>
118                     <i><li><a href="{% url 'index' %}">Ana Sayfa</a></li>
119                 <i></ul>
120             <i></nav>
121         <i></header>
122
123         <i><main>
124             <i><section>
125                 <i><h2>iletisim</h2>
126                 <i><% if success %>
127                 <i><p>Mesaj[?]iz başarıyla gönderildi. Teşekkür ederiz!</p>
128                 <i><% else %>
129                     <i><% csrf_token %>
130                     <i><div>
131                         <i><label form="name">Ad[?]iz:</label>
132                         <i><input type="text" id="name" name="name" required>
133                     <i></div>
134                     <i><div>
135                         <i><label form="email">E-posta:</label>
136                         <i><input type="email" id="email" name="email" required>
137                     <i></div>
138                     <i><div>
139                         <i><label form="message">Mesaj[?]iz:</label>
140                         <i><textarea id="message" name="message" rows="5" required></textarea>
141                     <i></div>
142                     <i><button type="submit">Gönder</button>
143                 <i></form>
144                 <i><% endif %>
145             <i></section>
146         <i></main>
147
148         <i><footer>
149             <i><p>©copy; 2024 Sensei Federal. Tüm hakları saklıdır.</p>
150         <i></footer>
151     <i></body>
152     <i></html>
153
154     <i><% endblock content %>
155
```


- PROGRAMLAR SAYFASI :

**Sensei Federal** Programlar İletişim Hakkımızda Login Register

Boks


Kickboks

Muay-thai



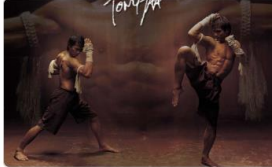
[boks antrenmanı](#)

Boks, iki kişinin boks hakemi gözetiminde karşılıklı yumruklaştıkları ve birbirlerini nakavtla ya da puanla yenmeye çalıştıkları spor müsabakasıdır.



[kickboks antrenmanı](#)

Kick boks, yumruk, tekme, diz ve sınırlı clinch uygulamalarının bir araya getirilmesiyle oluşturulmuş eklektik bir yakın dövüş stili ve dövüş sporudur.



[muay-thai antrenmanı](#)

Muay-Thai diğer adıyla Thai Boks, Kickboks sporuna göre daha sert ve saldırya yöneliktir.

Yukarıda ise antrenman derslerinin içerisinde bulunduğu programlar sayfasını görüyorsunuz. Her dala ayrı ayrı gruplanmış ve her spor dalı hakkında bilgi veren bir giriş sayfası bizi karşılıyor.

```
from datetime import date
from django.shortcuts import get_object_or_404, redirect, render
from django.http import Http404, HttpResponseNotFound
from django.urls import reverse
from .models import Program, Category

> data={...

db = {
  "programlar": [
    { ...
    { ...
    { ...
  ],
  "categories": [
    { "id":1,"name": "Boxing", "slug": "Boxing",},
    { "id":2,"name": "Kickboxing", "slug": "Kickboxing",},
    { "id":3,"name": "Muay-Thai", "slug": "Muay-Thai",},
  ]
}

def index(request):
    programlar = Program.objects.filter(isActive=1)
    kategoriler = Category.objects.all()

    return render(request, 'programlar/index.html', {
        'categories': kategoriler,
        'programlar': programlar
    })

def details(request, program_id):
    # try:
    #     program = Program.objects.get(pk=program_id)
    # except:
    #     raise Http404()
    program = get_object_or_404(Program, pk=program_id)

    context={
        'program': program
    }
    return render(request, 'programlar/details.html', context)

def getProgramlarByCategory(request, category_name):
    try:
        category_text = data[category_name];
        return render (request, 'programlar/programlar.html', {
            'category': category_name,
            'category_text': category_text
        })
    except:
        return HttpResponseNotFound("<h1>yanlış kategori seçimi</h1>")

def getProgramlarByCategoryId(request, category_id):
    category_list =list(data.keys())
    if(category_id > len(category_list)):
        return HttpResponseNotFound("Yanlış kategori seçimi")
    category_name = category_list[category_id - 1]

    redirect_url = reverse('programlar_by_category', args=[category_name])

    return redirect(redirect_url)
```

Yukarıda ise bu sayfanın backend kodlarını görüyorsunuz.


**Sensei Federal** Programlar İletişim Hakkımızda

Hoş geldin Federal Logout


**Boks**

Kickboks


Muay-thai



**[DİREKT VE KROŞE YUMRUK ANTRENMANI](#)**  
DİREKT VE KROŞE YUMRUKLAR EN ETKİLİ SALDIRI HAREKET TEKNİKLERİDİR



**[İSKİV VE ADIMLI KAÇMA](#)**  
İSKİV DEDİĞİMİZ YUMRUKDAN EĞİLME VE VÜCUT HAREKETLERİYLE KAÇMA MAÇ ESNASINDA SON DERECE ÖNEMLİ BİR TEKNİKTİR.



**[APARKAT](#)**  
YAKIN MESAFEDE GÜZEL BİR POZİSYON YAKALAMIŞSANIZ EĞER APARKATTAN DAHA TATLI BİR TEKNİK YOKTUR.TEK VURUŞTA TEMİZ BİR NAKAVT...

Yukarıda boks sekmesine ait eğitim program sayfasını görüyoruz. Boks Kickboks ve Muaythai ye ayrı ayrı teknikleri olan antrenman programları sayfalarında mevcuttur. Bu programları veritabanının yanı sıra Django Admin panelinden ekleyip çıkarabiliyoruz.

**Django yönetimi**

Giriş > Programlar > Programs

Süzmek için yazmaya başlayın...

**KİMLİK DOĞRULAMA VE YETKİLENDİRME**

**Gruplar** + Ekle

**Kullanıcılar** + Ekle

**PROGRAMLAR**

**Categorys** + Ekle

**Programs** + Ekle

Değiştirmek için program seçin

Eylem:   0 / 3 nesne seçildi

☐ PROGRAM

☐ APARKAT2024-03-07

☐ İSKİV VE ADIMLI KAÇMA2024-03-07

☐ DİREKT VE KROŞE YUMRUK ANTRENMANI2024-03-07

3 programs

Boks

Kickboks

Muay-thai



#### DİREKT - KROŞE - APARKAT

DİREKT - KROŞE - APARKAT YUMRUKLAR YAKIN MESAFEDE EN ETKİLİ SALDIRI HAREKET TEKNİKLERİDİR.



#### LOWKICK - MİDDLEKICK - HIGHKICK

KICKBOKSUN ANA HAREKETLERİ OLAN TEKME SON DERECE ÖNEMLİDİR. RAKİBE UZAK MESAFEDEYKEN BÜYÜK HASARLAR BIRAKIP NAKAVT ETMENİZE YARAR

Yukarıda kickboks sekmesine ait çalışma antrenmanlarını görüntülüyoruz.