

**“BİLGE NESİL TEKNOLOJİ MERKEZİ" Takip Sistemi  
(Asya-yı Miyane Örneği)**

**بیلگه نسل تکنولوژی مرکزى**

**FAZIL AHMED AZİZİ**

**23670310067**

**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

**BSM218/BSM303**

**Veritabanı Yönetim Sistemleri**

**Final Ödevi**

**Haziran 2025 Bartın**

## ADIM-1: Senaryo

### Bilge Nesil Teknoloji Merkezi - Takip Sistemi (Asya-yı Miyane Örneği) Senaryomuz:

Önce Tanımı ile başlayalım; **Bilge Nesil Teknoloji Merkezi** proje idealist bir teknoloji ve eğitim merkezi için geliştirilmiştir. Bu merkez; Türkiye’de yükseköğrenim görmüş mühendis ve teknoloji uzmanı gençlerin, Afganistan gibi teknolojiye erişimi kısıtlı bölgelerde kurmayı hedeflediği bir kalkınma projesidir. Merkezde verilen eğitimlerin ve öğrenci gelişimlerinin dijital ortamda takibini sağlayacak bir yönetim ve takip sistemi oluşturulmuştur. Uygulamamız; PHP programlama diliyle hazırlanmış web tabanlı bir admin paneline sahip olup ve MySQL veri tabanı sunucusu kullanılarak yapılmıştır.

Sistem olan ihtiyaç, Merkez büyüdükçe, kağıt-kalem ile yapılan takip işlemleri yetersiz kalmıştır. Öğrenci kayıtları, kurs yönetimi, devam takibi ve sertifika süreçlerini dijital ortamda yönetecek bir Web Tabanlı Takip Sistemi geliştirilmesi ihtiyacıyla ortaya çıkmış bir proje fikridir.

### Sistemin Esas Kısımları

#### 1. Öğrenci Yönetimi

- Öğrenci kayıt işlemleri (kişisel bilgiler, veli bilgileri, teknoloji seviyesi)
- Öğrenci listesi görüntüleme ve düzenleme
- Öğrenci profil sayfaları
- Durum takibi (Aktif/Pasif/Mezun/Ayrılmış)

#### 2. Eğitimci Yönetimi

- Türkiye'den dönen mühendis/öğretmenlerin kaydı
- Uzmanlık alanları ve deneyim bilgileri
- Mezun oldukları üniversite ve bölüm bilgileri
- Pozisyon ve maaş bilgileri yönetimi

#### 3. Kurs/Eğitim Yönetimi

- Kurs oluşturma ve düzenleme (Temel Bilgisayar, Kodlama, Yapay Zeka, Siber Güvenlik)
- Kurs kategorileri ve seviye belirleme
- Eğitimci atama ve kapasite yönetimi
- Kurs takvimi ve kayıt dönemleri

#### 4. Kayıt Yönetimi

- Öğrenci-kurs eşleştirme
- Kayıt onay süreçleri
- Ön değerlendirme notları
- Seviye uygunluk kontrolü

#### 5. Devam Takibi

- Günlük devam girişi
- Devamsızlık raporları
- Katılım durumu analizi
- Geç kalma ve izin takibi

## 6. Sertifika Yönetimi

- Başarılı öğrenciler için sertifika oluşturma
- Benzersiz sertifika numarası üretimi
- Sertifika onay süreçleri
- Sertifika geçmişi kayıtları

## Database Yapısı

### Tablolar

1. **ogrenciler** : Öğrenci kişisel bilgileri
2. **egitmenler**: Eğitimci profil ve yetkinlik bilgileri
3. **kurslar** : Kurs detayları ve program bilgileri
4. **kayıtlar** : Öğrenci-kurs kayıt ilişkileri
5. **devam** : Günlük devam kayıtları
6. **sertifikalar**: Verilen sertifika kayıtları

## ADIM-2: Varlıklar ve Nitelikler

### ○ Kurslar

- **kurs\_id**: Benzersiz Tanımlayıcı
- **kurs\_kodu**: Kursun benzersiz kodu
- **kurs\_adi**: Kursun adı
- **seviye**: Kursun zorluk seviyesi
- **toplam\_ders\_saati**: Kursun toplam ders saati
- **haftalik\_ders\_saati**: Kursun haftalık ders saati (Haftalık kaç saat ders olduğu)
- **kurs\_suresi\_hafta**: Kursun toplam süresi (hafta cinsinden)
- **sorumlu\_egitmen\_id**: Bu kursu veren eğitmenin ID'si
- **durum**: Kursun mevcut durumu
- **baslangic\_tarihi**: Kursun başlangıç tarihi

### ○ **Eđitmenler**

- **egitmen\_id**: Benzersiz Tanımlayıcı
- **ad**: Eđitmenin adı
- **soyad**: Eđitmenin soyadı
- **e\_posta**: Eđitmenin e-posta adresi
- **telefon**: Eđitmenin telefon numarası
- **uzmanlik\_alani**: Eđitmenin uzmanlık alanı

### ○ **Öđrenciler**

- **ogrenci\_id**: Benzersiz Tanımlayıcı
- **ad**: Öđrencinin adı
- **soyad**: Öđrencinin soyadı
- **e\_posta**: Öđrencinin e-posta adresi
- **telefon**: Öđrencinin telefon numarası
- **dogum\_tarihi**: Öđrencinin doğum tarihi
- **kayit\_tarihi**: Öđrencinin sisteme kayıt tarihi

### ○ **Kayıtlar**

- **kayit\_id**: Benzersiz Tanımlayıcı
- **ogrenci\_id**: Kaydolan öđrencinin ID'si
- **kurs\_id**: Kaydolunan kursun ID'si
- **kayit\_tarihi**: Öđrencinin kursa kaydolduđu tarih
- **tamamlanma\_durumu**: Kurstaki ilerleme durumu

### ○ **Kullanıcılar** (Giriş Yönetimi İçin)

- **user\_id**: Benzersiz Tanımlayıcı
- **kullanici\_adi**: Sisteme giriş için kullanılan kullanıcı adı
- **parola**: Kullanıcının şifresi
- **rol**: Kullanıcının sistemdeki rolü
- **iliskili\_id**: Bu kullanıcının egitmen\_id veya ogrenci\_id ile ilişkisi

## ADIM-3: Varlıklar Arası İlişkiler

- **Kurslar - Eğitimciler**

- Bir kursun bir sorumlu eğitmeni vardır.
- Bir eğitmen birden fazla kursa sorumlu olabilir.
- **N:1** (Çoktan-Bire)

- **Öğrenciler - Kayıtlar**

- Bir öğrenci birden fazla kursa kayıt olabilir-yapabilir.
- Bir kayıt tek bir öğrenciye aittir.
- **1:N** (Bire-Çok ilişkiden bahsedebiliriz)

- **Kurslar - Kayıtlar**

- Bir kursa birden fazla öğrenci kaydolabilir (birden fazla kayıt olabilir).
- Bir kayıt tek bir kursa aittir.
- **1:N** (Bire-Çok)

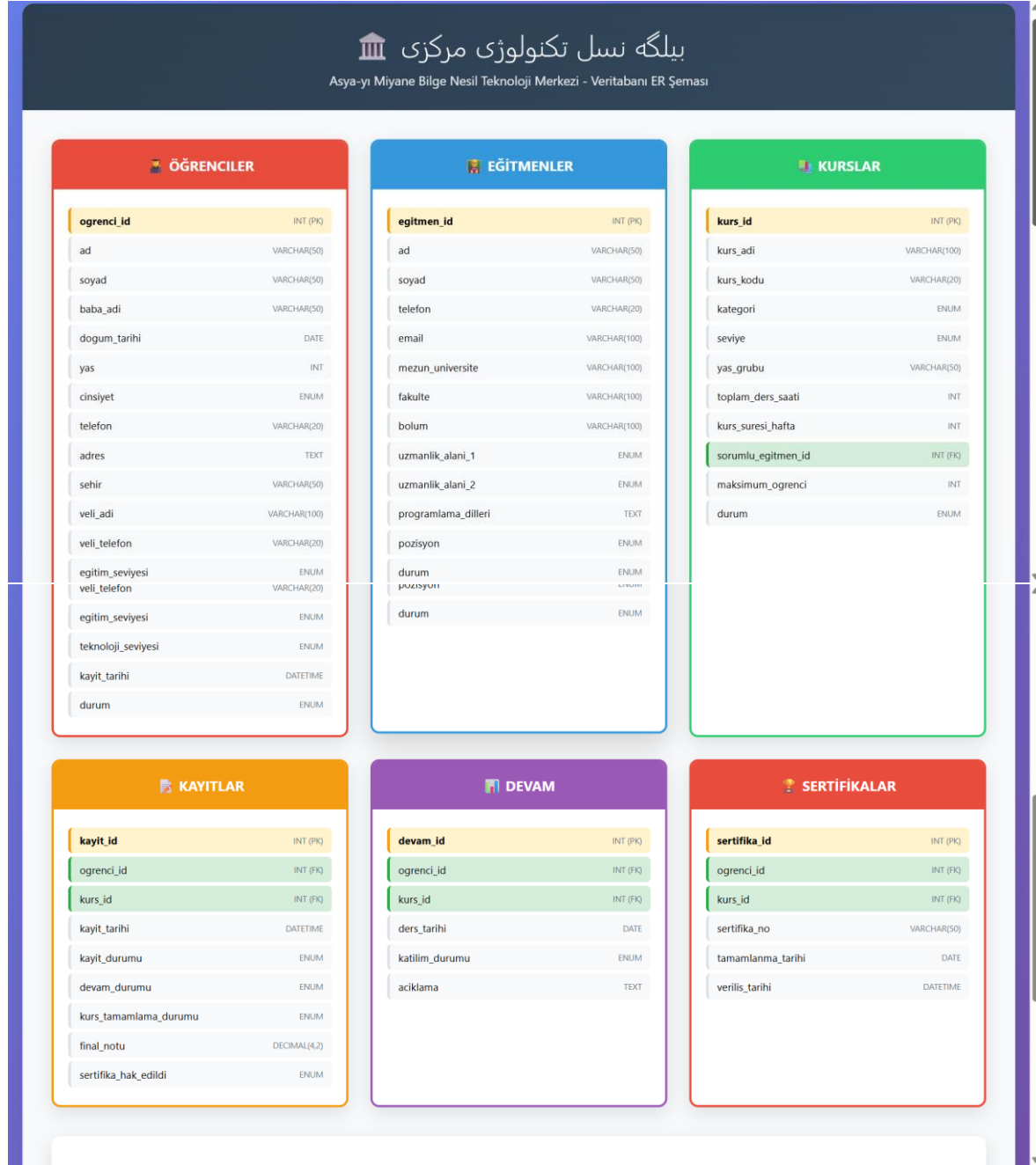
- **Öğrenciler - Kurslar**

- Bir öğrenci birden fazla kursa, bir kurs birden fazla öğrenciye sahip olabilir.
- **N:M** (Çoktan-Çoğa) - *Kayıtlar tablosu ile çözülür.*

- **Kullanıcılar - Eğitimciler / Öğrenciler**

- Bir kullanıcı kaydı bir eğitmen veya öğrenci profiliyle ilişkili olduğu için;
- Bir eğitmen veya öğrenci profili bir kullanıcı kaydına sahip olabilir.
- **1:1** veya **1:0..1** (Bire-Bire veya Bire-Sıfır-veya-Bir)

## ADIM-4: Er-Şeması



### Tablo İlişkileri ↗



## ADIM-5: İlişkisel (Mantıksal) Şema

- Eğitimciler = { egitmen\_id (PK), Ad, Soyad, E-posta, Telefon, Uzmanlık\_Alanı }
- Kurslar = { kurs\_id (PK), Kurs\_Kodu, Kurs\_Adı, Seviye, Toplam\_Ders\_Saati, Haftalık\_Ders\_Saati, Kurs\_Süresi\_Hafta, +sorumlu\_egitmen\_id (FK), Durum, Başlangıç\_Tarihi }
- Öğrenciler = { ogrenci\_id (PK), Ad, Soyad, E-posta, Telefon, Doğum\_Tarihi, Kayıt\_Tarihi }
- Kayıtlar = { kayıt\_id (PK), +ogrenci\_id (FK), +kurs\_id (FK), Kayıt\_Tarihi, Tamamlanma\_Durumu }
- Kullanıcılar = { user\_id (PK), Kullanıcı\_Adı, Parola, Rol, ilişkili\_id (FK) }

## ADIM-6: Fiziksel Yapı

### SQL Kodları (Veri Tabanı Oluşturma)

```
CREATE DATABASE IF NOT EXISTS teknoloji_merkezi;  
use teknoloji_merkezi;
```

### SQL Kodları (Tabloları Oluşturma)

```
CREATE TABLE ogrenciler(  
    ogrenci_id INT PRIMARY KEY AUTO_INCREMENT,  
    ad VARCHAR(50) NOT NULL,  
    soyad VARCHAR(50) NOT NULL,  
    baba_adi VARCHAR(50),  
    dogum_tarihi DATE,  
    yas INT,  
    cinsiyet ENUM('Erkek', 'Kız') NOT NULL,  
    telefon VARCHAR(20),  
    adres TEXT,  
    sehir VARCHAR(50) DEFAULT 'Herat',  
    ilce VARCHAR(50),  
    veli_adi VARCHAR(100),  
    veli_telefon VARCHAR(20),  
    egitim_seviyesi ENUM('İlkokul', 'Ortaokul', 'Lise', 'Mezun') DEFAULT 'İlkokul',  
    teknoloji_seviyesi ENUM('Hiç Bilmiyor', 'Temel', 'Orta', 'İleri') DEFAULT 'Hiç Bilmiyor',  
    kayıt_tarihi DATETIME DEFAULT CURRENT_TIMESTAMP,  
    durum ENUM('Aktif', 'Pasif', 'Mezun', 'Ayrılmış') DEFAULT 'Aktif',  
    notlar TEXT,  
    olusturan_admin VARCHAR(50),  
    guncelleme_tarihi TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP  
);
```

```
CREATE TABLE egitmenler  
(  
    egitmen_id INT PRIMARY KEY AUTO_INCREMENT,  
    ad VARCHAR(50) NOT NULL,  
    soyad VARCHAR(50) NOT NULL,  
    baba_adi VARCHAR(50),  
    dogum_tarihi DATE,  
    yas INT,  
    cinsiyet ENUM('Erkek', 'Kadın') NOT NULL,  
    telefon VARCHAR(20) NOT NULL,  
    email VARCHAR(100) UNIQUE,  
    adres TEXT,
```

```

-- Eğitim Bilgileri
mezun_universite VARCHAR(100),
fakulte VARCHAR(100),
bolum VARCHAR(100),
mezuniyet_yili YEAR,
lisans_notu DECIMAL(3,2), -- 0.00-4.00 arası
yuksek_lisans ENUM('Var', 'Yok', 'Devam Ediyor') DEFAULT 'Yok',
yuksek_lisans_bolum VARCHAR(100),

-- Uzmanlık Alanları
uzmanlik_alani_1 ENUM('Yazılım Geliştirme', 'Web Tasarım', 'Robotik', 'Elektronik', 'Veri
Analizi', 'Siber Güvenlik', 'Mobil Uygulama', 'Oyun Geliştirme') NOT NULL,
uzmanlik_alani_2 ENUM('Yazılım Geliştirme', 'Web Tasarım', 'Robotik', 'Elektronik', 'Veri
Analizi', 'Siber Güvenlik', 'Mobil Uygulama', 'Oyun Geliştirme'),
programlama_dilleri TEXT, -- Python, Java, C++ vs.
sertifikalar TEXT,

-- Deneyim Bilgileri
is_deneyimi_yil INT DEFAULT 0,
onceki_is_yerleri TEXT,
egitmenlik_deneyimi ENUM('Hiç Yok', '0-1 Yıl', '1-3 Yıl', '3+ Yıl') DEFAULT 'Hiç Yok',

-- Merkez Bilgileri
baslama_tarihi DATE NOT NULL,
maas DECIMAL(8,2),
pozisyon ENUM('Gönüllü', 'Yarı Zamanlı', 'Tam Zamanlı') DEFAULT 'Gönüllü',
durum ENUM('Aktif', 'İzinli', 'Ayrılmış') DEFAULT 'Aktif',

-- Kişisel Özellikler
dil_bilgisi VARCHAR(200) DEFAULT 'Farsça, Türkçe, İngilizce',
ozel_yetenekler TEXT,
motivasyon_notu TEXT,

-- Sistem Bilgileri
kayit_tarihi DATETIME DEFAULT CURRENT_TIMESTAMP,
son_giris_tarihi DATETIME,
olusturan_admin VARCHAR(50),
guncelleme_tarihi TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP
);

```

```

CREATE TABLE kayitlar
(
kurs_id INT PRIMARY KEY AUTO_INCREMENT,
kurs_adi VARCHAR(100) NOT NULL,
kurs_kodu VARCHAR(20) UNIQUE NOT NULL,
aciklama TEXT,
kategori ENUM('Temel Bilgisayar', 'Yazılım Geliştirme', 'Yapay Zeka', 'Siber Güvenlik')
NOT NULL,

-- Seviye ve Süre Bilgileri
seviye ENUM('Başlangıç', 'Orta', 'İleri') NOT NULL,
yas_grubu VARCHAR(50), -- örn: "12-15 yaş", "16+ yaş"
toplam_ders_saati INT NOT NULL,
haftalik_ders_saati INT NOT NULL,
kurs_suresi_hafta INT NOT NULL,

-- Ön Koşullar ve Hedefler
on_kosul TEXT,
hedef_beceriler TEXT,
kullanilacak_araclari TEXT,

-- Öğitmen ve Kapasite
sorumlu_ogitmen_id INT,
yardimci_ogitmen_id INT,
maksimum_ogrenci INT DEFAULT 12,
minimum_ogrenci INT DEFAULT 5,

```



```

-- Tarih ve Durum Bilgileri
baslama_tarihi DATE,
bitis_tarihi DATE,
kayit_baslama DATE,
kayit_bitis DATE,
durum ENUM('Planlanıyor', 'Kayıt Açık', 'Başladı', 'Tamamlandı', 'İptal') DEFAULT
'Planlanıyor',

-- Mali Bilgiler
kurs_ucreti DECIMAL(8,2) DEFAULT 0.00, -- Ücretsiz kurslar için
sertifika_ucreti DECIMAL(6,2) DEFAULT 50.00,

-- Sistem Bilgileri
olusturma_tarihi DATETIME DEFAULT CURRENT_TIMESTAMP,
olusturan_admin VARCHAR(50),
guncelleme_tarihi TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,

-- Foreign Key Constraints
FOREIGN KEY (sorumlu_egitmen_id) REFERENCES egitmenler(egitmen_id) ON DELETE SET NULL,
FOREIGN KEY (yardimci_egitmen_id) REFERENCES egitmenler(egitmen_id) ON DELETE SET NULL
);

```

```

CREATE TABLE devam
(devam_id INT PRIMARY KEY AUTO_INCREMENT,
ogrenci_id INT NOT NULL,
kurs_id INT NOT NULL,
ders_tarihi DATE NOT NULL,
katilim_durumu ENUM('Katıldı', 'Gelmedi', 'Geç Geldi', 'İzinli') DEFAULT 'Katıldı',
aciklama TEXT, -- Devamsızlık veya geç kalma nedeni gibi açıklamalar
kayit_tarihi DATETIME DEFAULT CURRENT_TIMESTAMP,
kayit_yapan_admin VARCHAR(50), -- Bu kaydı kimin girdiğini tutmak için

-- Bir öğrencinin belirli bir kursta, belirli bir tarihte sadece bir devam kaydı olabilir
UNIQUE KEY unique_devam_kaydi (ogrenci_id, kurs_id, ders_tarihi),

-- Foreign Key Constraints
FOREIGN KEY (ogrenci_id) REFERENCES ogrenciler(ogrenci_id) ON DELETE CASCADE,
FOREIGN KEY (kurs_id) REFERENCES kurslar(kurs_id) ON DELETE CASCADE
);

```

```

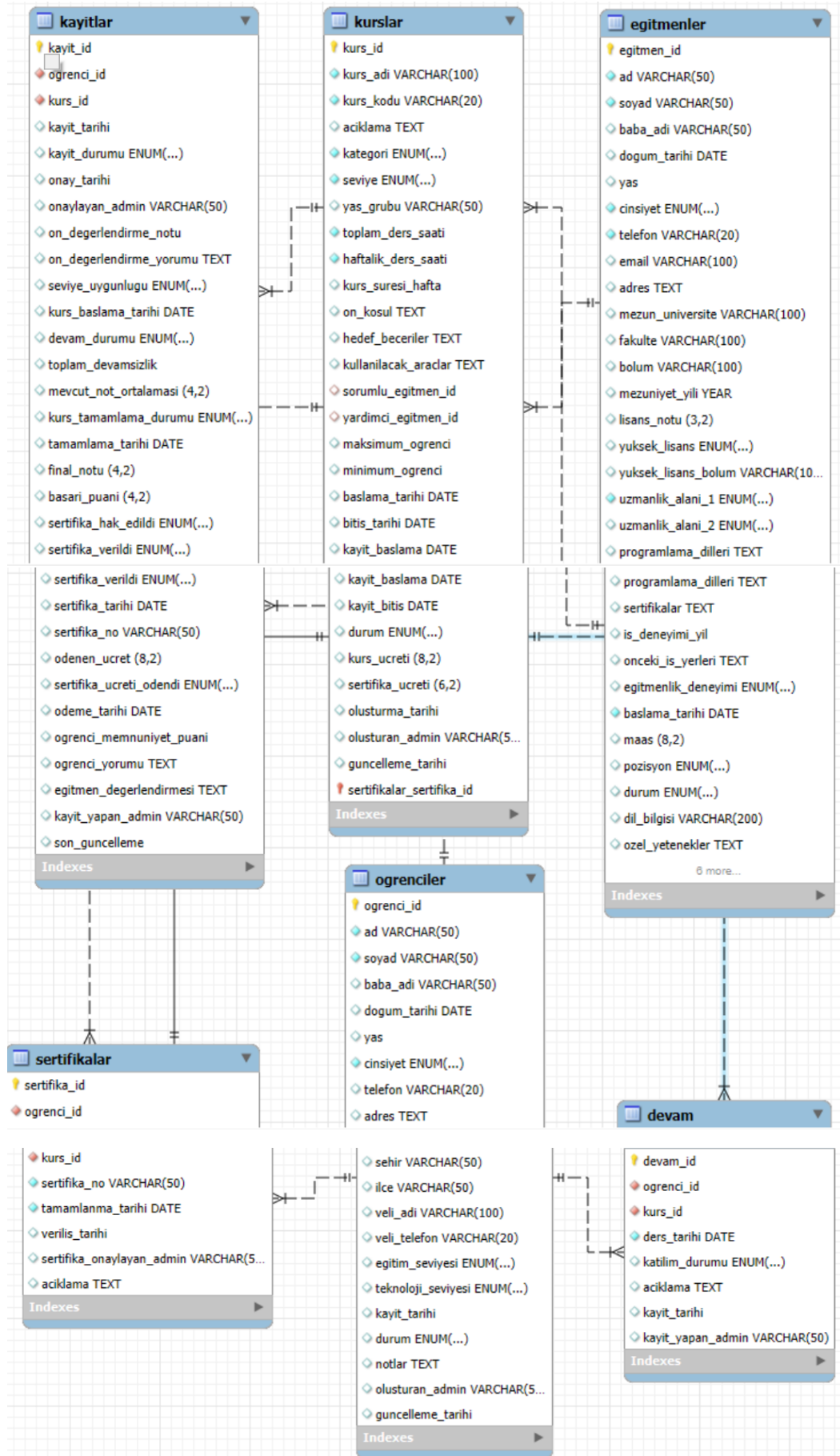
CREATE TABLE sertifikalar
(sertifika_id INT PRIMARY KEY AUTO_INCREMENT,
ogrenci_id INT NOT NULL,
kurs_id INT NOT NULL,
sertifika_no VARCHAR(50) UNIQUE NOT NULL, -- Her sertifikanın benzersiz bir numarası
olmalı
tamamlanma_tarihi DATE NOT NULL,
verilis_tarihi DATETIME DEFAULT CURRENT_TIMESTAMP,
sertifika_onaylayan_admin VARCHAR(50),
aciklama TEXT, -- Özel başarılar veya notlar gibi

-- Bir öğrenci belirli bir kurs için sadece bir sertifika alabilir
UNIQUE KEY unique_sertifika_ogrenci_kurs (ogrenci_id, kurs_id),

-- Foreign Key Constraints
FOREIGN KEY (ogrenci_id) REFERENCES ogrenciler(ogrenci_id) ON DELETE CASCADE,
FOREIGN KEY (kurs_id) REFERENCES kurslar(kurs_id) ON DELETE CASCADE
);

```

## İlişki Şeması



## Stored Procedures

```
DELIMITER //
CREATE PROCEDURE sp_OgrencilerHepsi ()
BEGIN
    SELECT
        ogrenci_id AS ID,
        CONCAT(ad, ' ', soyad) AS AdiSoyadi,
        yas AS Yas,
        cinsiyet AS Cinsiyet,
        telefon AS Telefon,
        sehir AS Sehir
    FROM ogrenciler
    WHERE durum = 'Aktif';
END //

CREATE PROCEDURE sp_OgrenciEkle (
    IN p_ad VARCHAR(50),
    IN p_soyad VARCHAR(50),
    IN p_yas INT,
    IN p_cinsiyet ENUM('Erkek', 'Kız'),
    IN p_telefon VARCHAR(20),
    IN p_sehir VARCHAR(50),
    IN p_olusturan_admin VARCHAR(50)
)
BEGIN
    DECLARE EXIT HANDLER FOR SQLEXCEPTION
    BEGIN
        ROLLBACK;
        RESIGNAL;
    END;
    START TRANSACTION;
    INSERT INTO ogrenciler (ad, soyad, yas, cinsiyet, telefon, sehir, olusturan_admin)
    VALUES (p_ad, p_soyad, p_yas, p_cinsiyet, p_telefon, p_sehir, p_olusturan_admin);
    COMMIT;
END //

CREATE PROCEDURE sp_OgrenciGuncelle (
    IN p_ogrenci_id INT,
    IN p_ad VARCHAR(50),
    IN p_soyad VARCHAR(50),
    IN p_yas INT,
    IN p_cinsiyet ENUM('Erkek', 'Kız'),
    IN p_telefon VARCHAR(20),
    IN p_sehir VARCHAR(50),
    IN p_olusturan_admin VARCHAR(50)
)
BEGIN
    DECLARE EXIT HANDLER FOR SQLEXCEPTION
    BEGIN
        ROLLBACK;
        RESIGNAL;
    END;
    START TRANSACTION;
    UPDATE ogrenciler
    SET ad = p_ad, soyad = p_soyad, yas = p_yas, cinsiyet = p_cinsiyet,
        telefon = p_telefon, sehir = p_sehir, olusturan_admin = p_olusturan_admin
    WHERE ogrenci_id = p_ogrenci_id;
```

```

        COMMIT;
    END //

CREATE PROCEDURE sp_OgrenciSil (
    IN p_ogrenci_id INT
)
BEGIN
    DECLARE EXIT HANDLER FOR SQLEXCEPTION
    BEGIN
        ROLLBACK;
        RESIGNAL;
    END;
    START TRANSACTION;
    DELETE FROM ogrenciler WHERE ogrenci_id = p_ogrenci_id;
    COMMIT;
END //

DELIMITER ;

```

```

DELIMITER //
CREATE PROCEDURE sp_EgitmenlerHepsi ()
CREATE PROCEDURE
BEGIN
    SELECT
        egitmen_id AS ID,
        CONCAT(ad, ' ', soyad) AS AdiSoyadi,
        telefon AS Telefon,
        email AS Email,
        pozisyon AS Pozisyon
    FROM egitmenler
    WHERE durum = 'Aktif';
END //

CREATE PROCEDURE sp_EgitmenEkle (
    IN p_ad VARCHAR(50),
    IN p_soyad VARCHAR(50),
    IN p_telefon VARCHAR(20),
    IN p_email VARCHAR(100),
    IN p_pozisyon ENUM('Gönüllü', 'Yarı Zamanlı', 'Tam Zamanlı'),
    IN p_olusturan_admin VARCHAR(50)
)
BEGIN
    DECLARE EXIT HANDLER FOR SQLEXCEPTION
    BEGIN
        ROLLBACK;
        RESIGNAL;
    END;
    START TRANSACTION;
    INSERT INTO egitmenler (ad, soyad, telefon, email, pozisyon, olusturan_admin)
    VALUES (p_ad, p_soyad, p_telefon, p_email, p_pozisyon, p_olusturan_admin);
    COMMIT;
END //

CREATE PROCEDURE sp_EgitmenGuncelle (
    IN p_egitmen_id INT,
    IN p_ad VARCHAR(50),
    IN p_soyad VARCHAR(50),
    IN p_telefon VARCHAR(20),
    IN p_email VARCHAR(100),

```

```

        IN p_pozisyon ENUM('Gönüllü', 'Yarı Zamanlı', 'Tam Zamanlı'),
        IN p_olusturan_admin VARCHAR(50)
    )
BEGIN
    DECLARE EXIT HANDLER FOR SQLEXCEPTION
    BEGIN
        ROLLBACK;
        RESIGNAL;
    END;
    START TRANSACTION;
    UPDATE egitmenler
    SET ad = p_ad, soyad = p_soyad, telefon = p_telefon, email = p_email,
        pozisyon = p_pozisyon, olusturan_admin = p_olusturan_admin
    WHERE egitmen_id = p_egitmen_id;
    COMMIT;
END //

CREATE PROCEDURE sp_EgitmenSil (
    IN p_egitmen_id INT
)
BEGIN
    DECLARE EXIT HANDLER FOR SQLEXCEPTION
    BEGIN
        ROLLBACK;
        RESIGNAL;
    END;
    START TRANSACTION;
    DELETE FROM egitmenler WHERE egitmen_id = p_egitmen_id;
    COMMIT;
END //
DELIMITER ;

```

```

DELIMITER //
CREATE PROCEDURE sp_KurslarHepsi ()
BEGIN
    SELECT
        kurs_id AS ID,
        kurs_adi AS KursAdi,
        kategori AS Kategori,
        seviye AS Seviye,
        durum AS Durum
    FROM kurslar;
END //

CREATE PROCEDURE sp_KurseKle (
    IN p_kurs_adi VARCHAR(100),
    IN p_kurs_kodu VARCHAR(20),
    IN p_kategori ENUM('Temel Bilgisayar', 'Yazılım Geliştirme', 'Yapay Zeka', 'Siber Güvenlik'),
    IN p_seviye ENUM('Başlangıç', 'Orta', 'İleri'),
    IN p_toplam_ders_saati INT,
    IN p_olusturan_admin VARCHAR(50)
)
BEGIN
    DECLARE EXIT HANDLER FOR SQLEXCEPTION
    BEGIN
        ROLLBACK;
        RESIGNAL;
    END;
    START TRANSACTION;

```

```

        INSERT INTO kurslar (kurs_adi, kurs_kodu, kategori, seviye, toplam_ders_saati,
olusturan_admin)
        VALUES (p_kurs_adi, p_kurs_kodu, p_kategori, p_seviye, p_toplam_ders_saati,
p_olusturan_admin);
        COMMIT;
END //

CREATE PROCEDURE sp_KursGuncelle (
    IN p_kurs_id INT,
    IN p_kurs_adi VARCHAR(100),
    IN p_kurs_kodu VARCHAR(20),
    IN p_kategori ENUM('Temel Bilgisayar', 'Yazılım Geliştirme', 'Yapay Zeka', 'Siber Güvenlik'),
    IN p_seviye ENUM('Başlangıç', 'Orta', 'İleri'),
    IN p_toplam_ders_saati INT,
    IN p_olusturan_admin VARCHAR(50)
)
BEGIN
    DECLARE EXIT HANDLER FOR SQLEXCEPTION
    BEGIN
        ROLLBACK;
        RESIGNAL;
    END;
    START TRANSACTION;
    UPDATE kurslar
    SET kurs_adi = p_kurs_adi, kurs_kodu = p_kurs_kodu, kategori = p_kategori,
        seviye = p_seviye, toplam_ders_saati = p_toplam_ders_saati, olusturan_admin =
p_olusturan_admin
    WHERE kurs_id = p_kurs_id;
    COMMIT;
END //

CREATE PROCEDURE sp_KursSil (
    IN p_kurs_id INT
)
BEGIN
    DECLARE EXIT HANDLER FOR SQLEXCEPTION
    BEGIN
        ROLLBACK;
        RESIGNAL;
    END;
    START TRANSACTION;
    DELETE FROM kurslar WHERE kurs_id = p_kurs_id;
    COMMIT;
END //
DELIMITER ;

```

```

DELIMITER //
CREATE PROCEDURE sp_KayitlarHepsi ()
BEGIN
    SELECT
        kayit_id AS ID,
        CONCAT(o.ad, ' ', o.soyad) AS OgrenciAdi,
        k.kurs_adi AS KursAdi,
        kayit_durumu AS KayitDurumu
    FROM kayitlar k
    JOIN ogrenciler o ON k.ogrenci_id = o.ogrenci_id
    JOIN kurslar k2 ON k.kurs_id = k2.kurs_id;
END //

CREATE PROCEDURE sp_KayitEkle (
    IN p_ogrenci_id INT,
    IN p_kurs_id INT,
    IN p_kayit_durumu ENUM('Beklemede', 'Onaylandı', 'Reddedildi', 'İptal'),
    IN p_onaylayan_admin VARCHAR(50),
    IN p_kayit_yapan_admin VARCHAR(50)
)
BEGIN
    DECLARE EXIT HANDLER FOR SQLEXCEPTION
    BEGIN
        ROLLBACK;
        RESIGNAL;
    END;
    START TRANSACTION;
    INSERT INTO kayitlar (ogrenci_id, kurs_id, kayit_durumu, onaylayan_admin,
kayit_yapan_admin)
        VALUES (p_ogrenci_id, p_kurs_id, p_kayit_durumu, p_onaylayan_admin, p_kayit_yapan_admin);
    COMMIT;
END //

CREATE PROCEDURE sp_KayitGuncelle (
    IN p_kayit_id INT,
    IN p_kayit_durumu ENUM('Beklemede', 'Onaylandı', 'Reddedildi', 'İptal'),
    IN p_onaylayan_admin VARCHAR(50),
    IN p_kayit_yapan_admin VARCHAR(50)
)
BEGIN
    DECLARE EXIT HANDLER FOR SQLEXCEPTION
    BEGIN
        ROLLBACK;
        RESIGNAL;
    END;
    START TRANSACTION;
    UPDATE kayitlar
        SET kayit_durumu = p_kayit_durumu, onaylayan_admin = p_onaylayan_admin, kayit_yapan_admin
= p_kayit_yapan_admin
        WHERE kayit_id = p_kayit_id;
    COMMIT;
END //

CREATE PROCEDURE sp_KayitSil (
    IN p_kayit_id INT
)
BEGIN
    DECLARE EXIT HANDLER FOR SQLEXCEPTION
    BEGIN

```

```

        ROLLBACK;
        RESIGNAL;
    END;
    START TRANSACTION;
        DELETE FROM kayitlar WHERE kayit_id = p_kayit_id;
    COMMIT;
END //
DELIMITER ;

```

```

DELIMITER //
CREATE PROCEDURE sp_DevamHepsi()
BEGIN
    SELECT
        devam_id AS ID,
        CONCAT(o.ad, ' ', o.soyad) AS OgrenciAdi,
        k.kurs_adi AS KursAdi,
        ders_tarihi AS DersTarihi,
        katilim_durumu AS KatilimDurumu
    FROM devam d
    JOIN ogrenciler o ON d.ogrenci_id = o.ogrenci_id
    JOIN kurslar k ON d.kurs_id = k.kurs_id;
END //

CREATE PROCEDURE sp_DevamEkle (
    IN p_ogrenci_id INT,
    IN p_kurs_id INT,
    IN p_ders_tarihi DATE,
    IN p_katilim_durumu ENUM('Katıldı', 'Gelmedi', 'Geç Geldi', 'İzinli'),
    IN p_kayit_yapan_admin VARCHAR(50)
)
BEGIN
    DECLARE EXIT HANDLER FOR SQLEXCEPTION
    BEGIN
        ROLLBACK;
        RESIGNAL;
    END;
    START TRANSACTION;
        INSERT INTO devam (ogrenci_id, kurs_id, ders_tarihi, katilim_durumu, kayit_yapan_admin)
        VALUES (p_ogrenci_id, p_kurs_id, p_ders_tarihi, p_katilim_durumu, p_kayit_yapan_admin);
    COMMIT;
END //

CREATE PROCEDURE sp_DevamGuncelle (
    IN p_devam_id INT,
    IN p_katilim_durumu ENUM('Katıldı', 'Gelmedi', 'Geç Geldi', 'İzinli'),
    IN p_kayit_yapan_admin VARCHAR(50)
)
BEGIN
    DECLARE EXIT HANDLER FOR SQLEXCEPTION
    BEGIN
        ROLLBACK;
        RESIGNAL;
    END;
    START TRANSACTION;
        UPDATE devam
        SET katilim_durumu = p_katilim_durumu, kayit_yapan_admin = p_kayit_yapan_admin
        WHERE devam_id = p_devam_id;
    COMMIT;
END //

```



```

CREATE PROCEDURE sp_DevamSil (
    IN p_devam_id INT
)
BEGIN
    DECLARE EXIT HANDLER FOR SQLEXCEPTION
    BEGIN
        ROLLBACK;
        RESIGNAL;
    END;
    START TRANSACTION;
    DELETE FROM devam WHERE devam_id = p_devam_id;
    COMMIT;
END //
DELIMITER ;

```

```

DELIMITER //
CREATE PROCEDURE sp_SertifikalarHepsi ()
BEGIN
    SELECT
        sertifika_id AS ID,
        CONCAT(o.ad, ' ', o.soyad) AS OgrenciAdi,
        k.kurs_adi AS KursAdi,
        sertifika_no AS SertifikaNo,
        verilis_tarihi AS VerilisTarihi
    FROM sertifikalar s
    JOIN ogrenciler o ON s.ogrenci_id = o.ogrenci_id
    JOIN kurslar k ON s.kurs_id = k.kurs_id;
END //

```

```

CREATE PROCEDURE sp_SertifikaEkle (
    IN p_ogrenci_id INT,
    IN p_kurs_id INT,
    IN p_sertifika_no VARCHAR(50),
    IN p_tamamlama_tarihi DATE,
    IN p_sertifika_onaylayan_admin VARCHAR(50)
)
BEGIN
    DECLARE EXIT HANDLER FOR SQLEXCEPTION
    BEGIN
        ROLLBACK;
        RESIGNAL;
    END;
    START TRANSACTION;
    INSERT INTO sertifikalar (ogrenci_id, kurs_id, sertifika_no, tamamlanma_tarihi,
sertifika_onaylayan_admin)
        VALUES (p_ogrenci_id, p_kurs_id, p_sertifika_no, p_tamamlama_tarihi,
p_sertifika_onaylayan_admin);
    COMMIT;
END //

```

```

CREATE PROCEDURE sp_SertifikaGuncelle (
    IN p_sertifika_id INT,
    IN p_sertifika_no VARCHAR(50),
    IN p_tamamlama_tarihi DATE,
    IN p_sertifika_onaylayan_admin VARCHAR(50)
)
BEGIN
    DECLARE EXIT HANDLER FOR SQLEXCEPTION

```

```
BEGIN
    ROLLBACK;
    RESIGNAL;
END;
START TRANSACTION;
    UPDATE sertifikalar
    SET sertifika_no = p_sertifika_no, tamamlanma_tarihi = p_tamamlama_tarihi,
        sertifika_onaylayan_admin = p_sertifika_onaylayan_admin
    WHERE sertifika_id = p_sertifika_id;
COMMIT;
END //
```

```
CREATE PROCEDURE sp_SertifikaSil (
    IN p_sertifika_id INT
)
BEGIN
    DECLARE EXIT HANDLER FOR SQLEXCEPTION
    BEGIN
        ROLLBACK;
        RESIGNAL;
    END;
    START TRANSACTION;
        DELETE FROM sertifikalar WHERE sertifika_id = p_sertifika_id;
    COMMIT;
END //
```

```
DELIMITER ;
```

## Trigger

```
-- Trigger 1: Güncelleme Takibi
-- Her tablo güncellemesinde bir log kaydı oluşturur.

DELIMITER //
CREATE TABLE log_kayitlari (
    log_id INT AUTO_INCREMENT PRIMARY KEY,
    tablo_adi VARCHAR(50),
    kayit_id INT,
    islem_tarihi DATETIME DEFAULT CURRENT_TIMESTAMP,
    yapan_admin VARCHAR(50)
);

CREATE TRIGGER trg_GuncellemeTakip
AFTER UPDATE ON ogrenciler
FOR EACH ROW
BEGIN
    INSERT INTO log_kayitlari (tablo_adi, kayit_id, yapan_admin)
    VALUES ('ogrenciler', NEW.ogrenci_id, NEW.olusturan_admin);
END //
DELIMITER ;
```

```
-- Trigger 2: Durum Değişim Kontrolü
-- Öğrenci "Ayrılmış" olursa, kayıtları etkiler

DELIMITER //
CREATE TRIGGER trg_DurumKontrol
BEFORE UPDATE ON ogrenciler
FOR EACH ROW
BEGIN
    IF NEW.durum = 'Ayrılmış' AND OLD.durum != 'Ayrılmış' THEN
        UPDATE kayitlar
        SET kurs_tamamlama_durumu = 'Yarıda Bıraktı'
        WHERE ogrenci_id = NEW.ogrenci_id AND kurs_tamamlama_durumu = 'Devam Ediyor';
    END IF;
END //
DELIMITER ;
```

## ADIM-7 ARAYÜZ TASARIMI ve UYGULAMA GELİŞTİRME

(Web-masaüstü-mobil istediğiniz ortamda hazırlayabilirsiniz.)

PHP-MySql KOD (GITHUB)

[https://github.com/Azizfazil236/My\\_Project](https://github.com/Azizfazil236/My_Project)

Web Sayfası fotoğrafları:

```
DBMS_Final_Project/
├── PHP_Doc/
│   ├── admin.php
│   ├── background4.jpg
│   ├── db.php
│   ├── devam.php
│   ├── egitmenler.php
│   ├── kayitlar.php
│   ├── kurslar.php
│   ├── login.php
│   ├── Logo2.png
│   ├── logout.php
│   ├── ogrenciler.php
│   ├── php kodları/
│   ├── sertifikalar.php
│   └── signup.php
└── SQL_Doc/
    ├── er_diagram.html
    ├── teknoloji_merkezi_1.0.sql
    ├── Workbench_ER_Schema_Final_Odevi.mwb
    └── Workbench_ER_Schema_Final_Odevi.mwb.bak
```

localhost/teknoloji\_merkezi/signup.php

### Eğitmen Kayıt

Lütfen tüm zorunlu alanları doldurun.

Ad:

Soyad:

E-posta:


Şifre:

Telefon (İsteğe Bağlı):

[Kayıt Ol](#)

[Zaten Hesabım Var, Giriş Yap](#)

localhost/teknoloji\_merkezi/login.php



### Bilge Nesil Teknoloji Merkezi

Asya-ya Milyane'deki gençler için bir gelecek vizyonu!

#### Admin Girişi

E-posta:

Şifre:

[Giriş Yap](#)

[Kayıt Ol](#) | [Şifremi Unuttum](#)

← → ↻ 🔍 localhost/teknoloji\_merkezi/admin.php

Kayıtlar Yönetimi Yönetimine Git

📅 Devam Takibi

Öğrenci	Kurs	Tarih	Durum
Henüz kayıt bulunmamaktadır.			

Devam Takibi Yönetimine Git

📄 Sertifikalar

Sertifika ID	Öğrenci	Kurs	Sertifika No	Tamamlanma Tarihi
Henüz kayıt bulunmamaktadır.				

Sertifikalar Yönetimine Git

← → ↻ 🔍 localhost/teknoloji\_merkezi/admin.php

Bilge Nesil Teknoloji Merkezi Admin Paneli

Hoş Geldiniz, Admin.

Çıkış Yap

Veritabanı bağlantısında veya sorgularında bir hata oluştu: SQLSTATE[42S22]: Column not found: 1054 Unknown column 'baslangic\_tarihi' in 'order clause'

👤 Öğrenci Yönetimi

ID	Ad Soyad	Yaş	Teknoloji Seviyesi
Henüz kayıt bulunmamaktadır.			

Öğrenci Yönetimi Yönetimine Git

👤 Eğitimci Yönetimi

ID	Ad Soyad	Uzmanlık Alanı	Pozisyon
Henüz kayıt bulunmamaktadır.			

← → ↻ 🔍 localhost/teknoloji\_merkezi/egitmenler.php?success=Egitmen+başarıyla+eklendi%21

2	Fatima Karimi	+93702222222	fatima.karimi@outlook.com	Robotik	Yarı Zamanlı	2024-02-01	Aktif	<div>Düzenle</div> <div>Sil</div>
1	Mohammad Reza Ahmadi	+93701111111	reza.ahmadi@gmail.com	Yazılım Geliştirme	Tam Zamanlı	2024-01-15	Aktif	<div>Düzenle</div> <div>Sil</div>
3	Hassan Nazari	+93703333333	hassan.nazari@yahoo.com	Siber Güvenlik	Tam Zamanlı	2023-12-01	Aktif	<div>Düzenle</div> <div>Sil</div>

Yeni Eğitimci Ekle

Ad:

Soyad:

Cinsiyet:

Seçiniz

Telefon:

localhost/teknoloji\_merkezi/admin.php

Henüz kayıt bulunmamaktadır.

Eğitmen Yönetimi Yönetimine Git

### Kurs Yönetimi

Kurs Kodu	Kurs Adı	Kategori	Seviye	Durum
Henüz kayıt bulunmamaktadır.				

Kurs Yönetimi Yönetimine Git

### Kayıtlar Yönetimi

Kayıt ID	Öğrenci	Kurs	Kayıt Tarihi	Durum
Henüz kayıt bulunmamaktadır.				

Kayıtlar Yönetimi Yönetimine Git

### Devam Takibi

localhost/teknoloji\_merkezi/egitmenler.php?success=Egitmen+başarıyla+eklendi%21

Telefon:

E-posta:

Şifre (Boş bırakılırsa değişmez):

Şifreyi değiştir

Uzmanlık Alanı:

Seçiniz

Pozisyon:

Seçiniz

Başlama Tarihi:

13.06.2025

Ekle

Admin Paneline Geri Dön

localhost/teknoloji\_merkezi/egitmenler.php?success=Egitmen+başarıyla+eklendi%21

Admin Paneline Geri Dön

### Eğitmen Yönetimi

Eğitmen başarıyla eklendi!

ID	Ad Soyad	Telefon	Email	Uzmanlık Alanı	Pozisyon	Başlama Tarihi	Durum	İşlem
4	Fazıl Ahmed Azizi	05056341775	fazilahmedazizi@gmail.com	Yazılım Geliştirme	Tam Zamanlı	2025-06-12	Aktif	<a href="#">Düzenle</a> <a href="#">Sil</a>
5	Maysam Wazin	05365494855	abc@gmail.com	Siber Güvenlik	Tam Zamanlı	2025-06-12	Aktif	<a href="#">Düzenle</a> <a href="#">Sil</a>
6	Deneme DenemeSoy		abcd@gmail.com	Yazılım Geliştirme	Gönüllü	2025-06-12	Aktif	<a href="#">Düzenle</a> <a href="#">Sil</a>
2	Fatima Karimi	+93702222222	fatima.karimi@outlook.com	Robotik	Yarı Zamanlı	2024-02-01	Aktif	<a href="#">Düzenle</a> <a href="#">Sil</a>

[Admin Paneline Geri Dön](#)

Öğrenci Yönetimi

ID	Ad	Soyad	Yaş	Cinsiyet	Telefon	Durum	Kayıt Tarihi	İşlem
1	Ahmad	Ahmadi	16	Erkek	+93701234567	Pasif	2025-06-11 07:47:43	<a href="#">Düzenle</a> <a href="#">Sil</a>
2	Fatima	Karimi	15	Kız	+93702345678	Aktif	2025-06-11 07:47:43	<a href="#">Düzenle</a> <a href="#">Sil</a>
3	Hassan	Nazari	17	Erkek	+93703456789	Aktif	2025-06-11 07:47:43	<a href="#">Düzenle</a> <a href="#">Sil</a>
4	Zainab	Hosseini	14	Kız	+93704567890	Aktif	2025-06-11 07:47:43	<a href="#">Düzenle</a> <a href="#">Sil</a>
5	Tayyip	Azizi	11	Erkek	+93707270023	Aktif	2025-06-11 07:47:43	<a href="#">Düzenle</a> <a href="#">Sil</a>

Yeni Öğrenci Ekle

Ad:

[Admin Paneline Geri Dön](#)

1	Temel Bilgisayar Kullanımı - Başlangıç	BK-001	Başlangıç	40	4	Mohammad Reza Ahmadi	<a href="#">Düzenle</a> <a href="#">Sil</a>
---	--	--------	-----------	----	---	----------------------	---

Yeni Kurs Ekle

Kurs Adı:

Seviye:

Seçiniz

Toplam Ders Saati:

Haftalık Ders Saati:

Eğitmen:

Seçiniz

Ekle

[Admin Paneline Geri Dön](#)

Yeni Öğrenci Ekle

Ad:

Soyad:

Yaş:

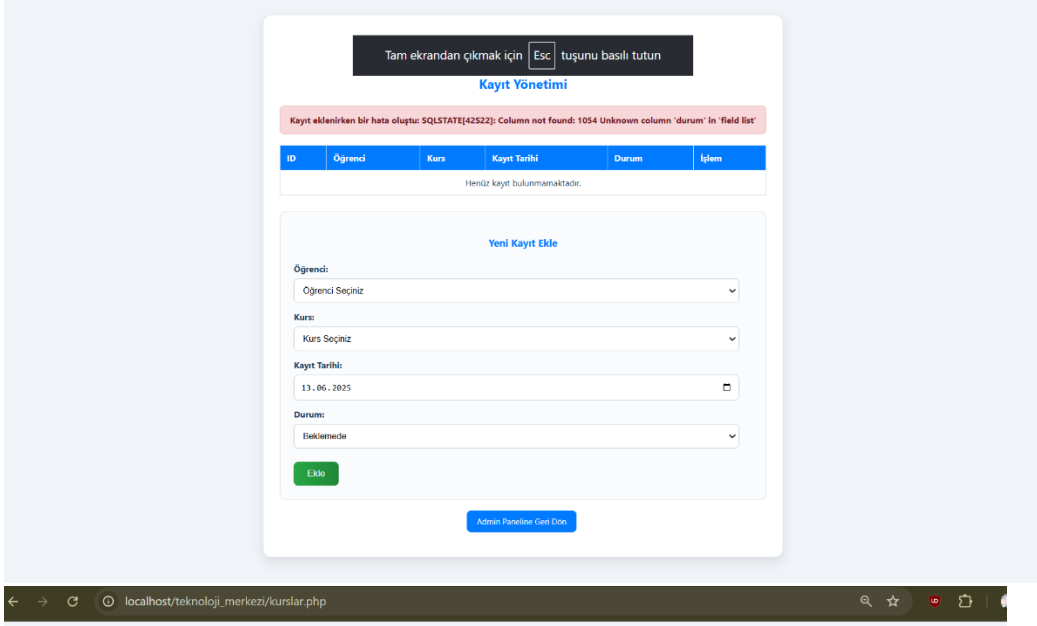
Cinsiyet:

Seçiniz

Telefon:

Ekle





[Admin Paneline Geri Don](#)

### Kurs Yönetimi

ID	Kurs Adı	Kurs Kodu	Seviye	Toplam Ders Saati	Haftalık Ders Saati	Eğitmen	İşlem
10	Java	K-681ae1ceae6d78	Başlangıç	60	5	Maysam Wazin	<a href="#">Düzenle</a> <a href="#">Sil</a>
9	Etik Hacker Eğitimi - Temel	SG-002	İleri	80	8	Hassan Nazari	<a href="#">Düzenle</a> <a href="#">Sil</a>
8	Dijital Güvenlik ve Farkındalık	SG-001	Başlangıç	20	2	Hassan Nazari	<a href="#">Düzenle</a> <a href="#">Sil</a>
7	Python ile Makine Öğrenmesine Giriş	YZ-002	İleri	48	4	Fatima Karimi	<a href="#">Düzenle</a> <a href="#">Sil</a>
6	Yapay Zeka Dünyasına Giriş	YZ-001	Başlangıç	24	2	Fatima Karimi	<a href="#">Düzenle</a> <a href="#">Sil</a>
5	Web Geliştirme - Temel	KY-003	İleri	72	6	Mohammad	<a href="#">Düzenle</a> <a href="#">Sil</a>

Tam ekrandan çıkmak için **Esc** tuşunu basılı tutun

### Devam Takibi

Devam listesi çekilirken bir hata oluştu: SQLSTATE[42522]: Column not found: 1054 Unknown column 'd.tarih' in 'field list'

ID	Tarih	Öğrenci	Kurs	Katılım Durumu	İşlem
Henüz devam kaydı bulunmamaktadır.					

#### Yeni Devam Kaydı Ekle

Tarih:

Öğrenci:

Kurs:

Katılım Durumu:

**Ekle**

[Admin Paneline Geri Don](#)

ADIM-8: Uygulamanızı kendiniz anlatın (max 5 dk video)

**Proje Linki ve Tanıtım Videosu:**

- **GitHub Proje Kodları:**

[https://github.com/Azizfazil236/My\\_Project](https://github.com/Azizfazil236/My_Project)

- 

- ~~**YouTube Tanıtım Videosu:**~~

~~[video5270780804.mp4](#) (kaldırıldı-hatalı sonuç)~~

- **OneDrive Tanıtım Videosu:** [video5270780804.mp4](#) (GÜNCEL LİNK)