



T.C

MARMARA ÜNİVERSİTESİ

TEKNİK BİLİMLER

BİLGİSAYAR PROGRAMCILIĞI

VERİTABANI YÖNETİM SİSTEMLERİ

FİNAL PROJESİ ÖDEVİ

EMİR KARA

361522987

Önsöz

Veritabanı Yönetim Sistemleri dersinden edindiğimiz bilgileri bir uygulama üzerinde pratiğe döküebilmek ve pekiştirebilmek için verilen final projesinde kullanmayı hedefledim. Proje konusu olarak, Öğrenci Bilgi Otomasyon Sistemini seçtim. Bu raporda, otomasyon sistemi için gerekli senaryoyu daha sonra uygulama tasarımı için veritabanı ve arayüz işlemlerine açıklık getirerek sistemin çalışır durumu ile ilgili bilgiler verilecektir.

1. Giriş

Üniversiteler, öğrencilerin kayıt, ders seçimi, not girişi ve akademik takip gibi bir dizi işlemi yönetmek zorundadır. Bu süreçlerin etkin bir şekilde gerçekleştirilebilmesi için Öğrenci Bilgi Sistemi kullanımı önemli hale gelmiştir. Öğrenci Bilgi Sistemi, modern teknoloji ve bilişim altyapısının üniversite yönetimiyle birleşmesini sağlayarak, öğrenci verilerinin güvenli ve kolay bir şekilde yönetilmesini amaçlamaktadır.

Bu sistem, öğrenci bilgilerinin dijital olarak saklanmasını ve erişilebilir olmasını sağlar. Öğrenci işleri birimi, sisteme dönem, bölüm, akademisyen, ders ve öğrenci gibi verileri ekleyebilir ve güncelleyebilir. Bu sayede öğrencilerin kayıtları, ders seçimleri ve akademik performansları merkezi bir veritabanında tutulur. Akademisyenler, not girişi yapabilir ve öğrencilerin akademik ilerlemelerini takip edebilir. Öğrenciler ise kendi bilgilerini görüntüleyebilir ve ders seçimleriyle ilgili işlemleri gerçekleştirebilir.

Öğrenci Bilgi Sistemi, veri güvenliği ve yetkilendirme önlemleriyle desteklenir. Kullanıcıların giriş yapabilmesi için kullanıcı adı ve şifre gereklidir. Sistem, güçlü şifreleme algoritmalarıyla kullanıcı bilgilerinin korunmasını sağlar. Ayrıca, kullanıcı rolleri ve yetkilendirmeleri sayesinde, sadece ilgili kişilerin belirli işlemleri gerçekleştirmesine izin verilir. Öğrenci Bilgi Sistemi, üniversitelerin veri yönetimi süreçlerini otomatize eder, zaman ve iş gücü tasarrufu sağlar ve öğrenci bilgilerinin doğruluğunu ve güncelliğini korur.

2. Genel Bakış

Üniversiteler, öğrencilerin kayıt, akademik takip, ders seçimi ve not girişi gibi bir dizi önemli işlemi yönetmek zorundadır. Bu süreçlerin etkin ve verimli bir şekilde yürütülmesi için Öğrenci Bilgi Sistemi kullanımı kaçınılmaz hale gelmiştir. Öğrenci Bilgi Sistemi, üniversite yönetiminin bilişim teknolojileriyle entegre olarak çalışmasını sağlayan bir platformdur. Öğrencilerin akademik verileri merkezi bir veritabanında saklanır ve güncel bilgilere kolayca erişebilirler.

Öğrenci Bilgi Sistemi, üniversite personeli, akademisyenler ve öğrenciler arasında etkili iletişim ve veri paylaşımını sağlar. Sistem, öğrenci işleri birimine kayıt, bölüm, ders ve öğrenci bilgileri gibi verilerin girişini ve güncellemesini kolaylaştırır. Akademisyenler, öğrencilerin ders performanslarını takip ederken not girişi yapabilirler. Öğrenciler ise ders seçimleri, transkript bilgileri ve kişisel bilgileri gibi kendi verilerini görüntüleyebilir ve yönetebilirler.

Öğrenci Bilgi Sistemi, öğrenci verilerinin güvenliğini ve gizliliğini sağlamak için gelişmiş güvenlik önlemleriyle donatılmıştır. Kullanıcıların sisteme erişimi için kimlik doğrulama mekanizmaları kullanılır ve veriler şifreleme teknolojileriyle korunur. Ayrıca, kullanıcı rolleri ve yetkilendirmeleri belirlenerek, sadece ilgili kişilerin belirli işlemleri gerçekleştirmesine izin verilir. Öğrenci Bilgi Sistemi, üniversitelerin veri yönetimi süreçlerini otomatikleştirir, hataları azaltır, verimliliği artırır ve daha iyi kararlar alınmasına yardımcı olur.

3. Uygulama Tasarımı

Gereksinimler

- MSSQL: Öncelikle MSSQL veritabanı yönetim sistemini kurmanız gerekmektedir.
- .NET 3.1: Sistemin çalışması için .NET 3.1'in yüklü olması gerekmektedir.

Kurulum Adımları

Eğer Git üzerinde bir repoya yüklenirse;

- Github Repository: Sistemi kullanmak için ilgili projeyi Github'dan clone edin. Önerilen branch, güncellemeleri içeren branch olmalıdır.
- Database Ayarları: Proje dosyası içerisindeki Database.cs dosyasında yer alan database adını düzenleyin.
- Package Manager Console: Visual Studio'da Package Manager Console'a girerek "add-migration firstMig" komutunu çalıştırın. Ardından "Update-Database" komutunu çalıştırarak modellerin oluşturulmasını sağlayın.

Proje Çalıştırma

- Projeyi çalıştırın: Projeyi çalıştırmak için geliştirme ortamında uygun bir web sunucusu seçin ve projeyi başlatın.
- Yönetici Kaydı: <https://localhost:44357/Admin/Register> adresine giderek "super" rolünde "öğrenci işleri" kullanıcısını oluşturun. Kullanıcı adı: super, şifre: abC.1234!
- Akademisyen Kaydı: <https://localhost:44357/Admin/Akademisyen> adresine giderek "admin" rolünde bir akademisyen kullanıcısı oluşturun. Kullanıcı adı: admin, şifre: abC.1234!
- Öğrenci Kaydı: <https://localhost:44357/Admin/Ogrenci> adresine giderek "öğrenci" rolünde bir öğrenci kullanıcısı oluşturun. Kullanıcı adı: öğrenci, şifre: abC.1234!

Sistem Kullanımı

Öğrenci İşleri Girişi: "Öğrenci İşleri" kullanıcısı olarak giriş yaptıktan sonra sırasıyla dönem, bölüm, akademisyen, ders ve öğrenci eklemelerini yapabilirsiniz.

Akademisyen Yetkileri: "Akademisyen" kullanıcısı, tüm verileri görüntüleyebilir ve not ekleyebilir.

Öğrenci Yetkileri: "Öğrenci" kullanıcısı sadece verileri görüntüleyebilir.

4. Veri Tabanı Tasarımı

