

ÖĞRENCİ OTOMASYON SİSTEMİ

1.PROJENİN TANIMI

Proje yapımında 2 farklı seçenek bulunuyor. Projeyi 1 ya da 2 kişi olarak tamamlayabilirsiniz. Eğer 1 kişi olarak devam edecekseniz projenın sadece 1. partından sorumlusunuz, 2 kişilik grup olacaksa projedeki her iki partttan da sorumlu olacaksınız. Gruplar en fazla 2 kişilik olabilir. 2. Parta dahil olan kısımlar altı çizili olarak belirtilmiştir.

Bu projede ders, öğrenci ve eğitimci bazında belirli görevleri yapan bir otomasyon sistemi yapmanız isteniyor.

Sistem şu görevleri yerine getirebilmeli:

- ++ Ekleme çıkarma ve düzenleme işlemlerine göre kendi binary dosyalarını oluşturma
- ++ Program başlangıcında var olan binary dosyalarını kontrol etme
- ++ Başka bir yerde oluşturulmuş .csv doyalarını okuma ve bunlarla ilişkili olarak eğer veritabanı dosyası yoksa oluşturma
- ++ Eğer başlangıç dosyaları yoksa boş veritabanı dosyası oluşturma (sadece 1.partı yapanlarda hata kodu verilip program sonlanacak)

Program farklı modlarda çalışabilmeli(Öğrenci, eğitimci ve yönetici modu). Programın hangi modda çalışacağı, kullanıcı sisteme giriş yaptıktan sonra program tarafından anlaşılabilmeli. Hatalı kullanıcı adı ya da şifre olursa tekrar istemeli. Giriş yapıldıktan sonra sistem şunları yerine getirmeli.

ÖĞRENCİ MODU

+Ders Ekleme: Öğrencinin alabileceği dersler listelenir ve seçim yapmasına izin verilir.Dersler şu durumlarda alınabilir

>>Ders açılmışsa ve ekleme çıkarma modu o ders için açıks

>>Öğrencinin geçemediği ön koşul dersi yoksa

+Ders Çıkarma:Öğrencinin çıkarabileceği dersler listelenir ve çıkarabilmesi için izin verilir. Sadece ekleme/çıkarma modu kapalı değilse ders çıkarma işlemi yapılabilir.

+Dersleri Listeleme: Öğrencinin aldığı dersler listelenir. Kaydetmek isteyip istemediği sorulur, kayıt isteği gelirse <öğrenciNo>_courses.csv dosyasına kaydedilir.

+Transkript Oluřturma: Öğrencinin almıř olduėu tüm dersler notlarıyla birlikte listelenir. Kaydetmek isterse <öğrenciNo> transcript.txt ye kaydedilir.

EĞİTİMCİ MODU

+Ders Açma : Eğitimci tarafından oluşturulmuş dersler listelenir. Bunlardan istediğini o öğrenime açabilir.

+Dersi Bitirme: : Eğitimci tarafından oluşturulmuş ve öğrenime açılmış dersler listelenir. Eğitimci bunlardan birini bitirebilir. Dersin bitirilebilmesi için dersi alan tüm öğrencilerin notlarının girilmiş olması gerekir. Ders kapatılınca dersi alan tüm öğrencilerinde aldığı dersler bilgisi güncellenmelidir.

+Not Verme : Eğitimcinin verdiği ve öğrenime açık olan tüm dersler listelenir. Eğitimciden bir ders seçmesi istenir. Seçilen dersi alan tüm öğrenciler listelenir ve not vereceği öğrenciyi seçmesi istenir. (Not verme işlemi sadece o ders eğitime açıksa ve ders ekleme çıkarma modu kapalı ise yapılabilir.) Notlar ondalıklı sayı biçiminde olmalıdır.

+Not Düzenleme : Eğitimcinin verdiği ve öğrenime açık olan tüm dersler listelenir. Eğitimciden bir ders seçmesi istenir. Seçilen dersi alan tüm öğrenciler listelenir ve notunun düzenleneceği öğrenciyi seçmesi istenir. Eğer daha önce öğrencinin notu hiç girilmemişse öncelikle ilk notu girmesi istenir. (Not düzeltme işlemi sadece o ders eğitime açıksa ve ders ekleme çıkarma modu kapalı ise yapılabilir.)

+Açık dersleri Listeleme : Eğitimci tarafından verilen ve açık olan tüm dersler listelenir. Kaydetme isteğı sorulur cevap evetse <name_surname>_openCourses.csv ye kaydedilir.

+ Notları listeleme : Eğitimci tarafından verilen ve açık olan tüm dersler listelenir. Eğitimciden bir ders seçmesi istenir. Dersi alan öğrenciler notlarıyla birlikte listelenir. Kaydetme isteğı sorulur, cevap evetse <dersID>_grades.csv ye kaydedilir.

SYSTEM MODE

Tek bir kiři sistem yöneticisi olabilir. Eğer başla _____ ngıçta sistem yöneticisi yoksa, sistem o anki kullanıcıdan sistem yöneticisi olmasını ister ve o kiři sistem yöneticisi olur. Programın sistem yöneticisinin silme yetkisi yoktur.

++Kullanıcı Ekleme : Yeni kullanıcı ekler, öğrenci ya da eğitimci olabilir.

++Kullanıcı Silme : İşrenilen kullanıcıyı silebilir ancak silme işleminden önce tekrardan şifre kullanıcı adı sorgusu yapılmalıdır. Kullanıcı silme işlemi 2 şekilde gerçekleşir.

++Öğrenci Silme : Öğrencininhiç bir derse kaydı yoksa silinebilir. Eğer öğrenci belirli bir ders sayısına ulaşmışsa(programı yazan sayıyı belirler) sistemden silinir ve adı soyadı mezuniyet ortalaması ile birlikte mezun öğrenciler listesine eklenir. Diğer durumlarda öğrenci sistemden silinir ve silinmiş öğrenciler listesine eklenir. Tüm verileri de silinmelidir.

++Eğitimci Silme : Eğer verdiği hiç bir ders yoksa bir eğitimci sistemden silinebilir. Sistem eğitimcinin tüm verilerini siler ve adı soyadı ile birlikte eski eğitimciler listesine kaydeder.

++ Kullanıcı Düzenleme : Kullanıcı bilgilerini düzenleyebilir.

++ Yeni Ders Açma : Yeni bir ders açabilir ve eğitimcisinin adı burada belirtilmelidir.

++ Ders Silme : Varolan tüm dersler listelenir, bunlardan birini silebilir ancak silinecek ders açık olmamalıdır.

++Ders Düzenleme : Dersler listelenir ver herhangi biri düzenlenebilir.

2. VERİ

Öğrenciler >> TC No(eşsiz ve 11 hane), ad, soyad , öğrenci no(10 hane), başlangıç yılı, aldığı dersler, önceden aldığı dersler ve notları

Eğitimciler >> TC No(eşsiz 11 hane), ad, soyad, verdiği dersler

Dersler >> ID(eşsiz), adı, eğitimcisi, önkoşulu(1de fazla olabilir), dersi oluşturan, dersin açık/kapalı olduğunu belirten bir değer, mod(ekleme/çıkarma açık ya da kapalı)

Kullanıcılar >> ID(TC No su ile aynı olmalı), tipi(öğrenci, eğitimci, yönetici), kullanıcı adı(eşsiz), şifre

3. VERİ DOSYALARI

Başlangıç Dosyaları

İsimler kesinlikle aynı olmalıdır.

>>**Students.csv** : Öğrencilerin kişisel bilgilerinin olduğu dosya

>>**Instructors.csv**: Eğitimci bilgilerinin olduğu dosya

>>**Users.csv**: Kullanıcı bilgilerinin olduğu dosya.

Veritabanı Dosyaları

Sistem kendi veritabanı dosyalarını oluşturmalıdır.

4. DİZAYN VE UYGULAMA DETAYLARI

- structure lar fonksiyonlara pointer olarak gönderilmelidir
-
- Kullanıcını girdiği tüm veriler kontrol edilmeli, yanlışsa tekrar istenmeli
- Dinamik arrayler kullanılmalı, ve program kullanılmayan belleği tutmamalı
- Sistem sadece 1 not kabul etmeli

5. BONUS

Bonus puanlar algoritmaya ve kullanım kolaylığına göre verilecektir. Gereksiz karmaşıklık yaratacak şeyler eklemeyin. Bilgileri tutabilecek yeteri sayıda structure tanımlamalısınız. Yeteri sayıda fonksiyonunuz olmalı ve kod tekrarından kaçınmalısınız. Bonus puan için işe yarar fonksiyonlar ekleyebilirsiniz.

6. PROJE SUNUMU

ANALİZ RAPORU

- Grup çalışması olup olmayacağı belirtilmeli
- Kendi kelimelerinizle sistemin işleyişini açıklayın(nasıl çalışır, ne yapar, kısıtları nelerdir..)
- Sistem girdi ve çıktılarını tanımlayın
- Yapı şemasını çıkarın (chapter 4.5 and 12.3)
- Fonksiyon prototipleri ver açıklamaları (daha sonradan bazı eklemeler yapabilirsiniz ancak ana yapı burada tanımlanmış olmalı)

UYGULAMA VE TEST RAPORU

- Programın çalışma şeklini açıklayın
- Çıkan sonuçları analiz raporundaki beklenen sonuçlarınızla karşılaştırın, farklılıkları ve bunların sebeplerini belirtin
- Sistemi test edin ve konsol ekran görüntüleriyle birlikte program çıktılarını ekleyin
- Kod satırlarını da rapora ekleyin