

MARMARA ÜNİVERSİTESİ
TEKNOLOJİ FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ
BLM2002 Veri Yapıları ve Algoritmalar
Dönem Projesi
Yabancı Öğrenci Denetim Sistemi
Üzerine Veri Yapılarının Uygulanması

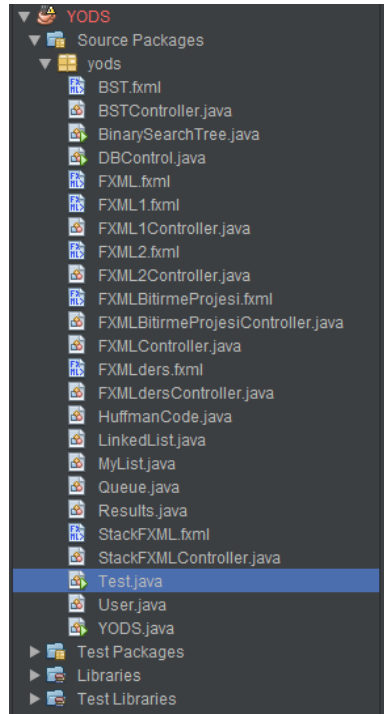
170420517 - Mehmed Emirhan AMAÇ, 170420842 - Dicle KILIÇ

[Projenin GitHub Linki](#)

1. Projenin Amacı

Yabancı öğrenci denetim sistemi, erasmus ve benzeri programlardan üniversitemize gelen öğrenciler ile üniversitede görevli ve danışmanlık yapabilecek öğretmenleri bir araya getiren bir sistemdir.

2. Çalışma Prensibi



Projeye Ait Sınıflar

Yukarıdaki şekilde projeye ait arayüz bileşenleri ve Controller sınıfları bulunmaktadır. Veritabanı işlemlerine ait metotlar “DBControl” sınıfında bulunmaktadır. “Test” sınıfında ise yazılan tüm metotlar test edilmektedir.

Öğrenci Ekranı

Öğrenci ekranında sisteme öğretmen olarak kayıt yapan kullanıcıların adları veritabanı üzerinden listelenmektedir. Öğrenci seçmek istediği öğretmeni seçebilir. Eğer boş kutucukları seçerse, birden fazla seçerse veya hiç öğretmen seçmezse öğretmenini kaydet butonuna tıkladığı zaman sağ altta hata mesajı görüntüler. Öğretmeni seçtiği zaman öğretmen giriş yaptığında onaylaması için onaylama mesajı alır. İşini tamamladıktan sonra “Giriş Ekranına Dön” butonuyla giriş ekranına dönebilir.

Hoşgeldiniz. En fazla 3 adet öğrenciye danışmanlık yapabilirsiniz.

Öğretmen İsmi: Anıl Baş

Sistemdeki Öğrenciler	Gelen Danışmanlık İstekleri	Danışmanlık Yapılan Öğrenciler
emirhan	<input type="checkbox"/> mehmet	Yok
selin	<input checked="" type="checkbox"/> Bos	Yok
tarık	<input type="checkbox"/> Bos	Yok
mehmet	<input type="checkbox"/> Bos	
	<input type="checkbox"/> Bos	
	<input type="checkbox"/> Bos	
	<input type="checkbox"/> Bos	
	<input type="checkbox"/> Bos	

İsteği Onayla

Giriş Ekranına Dön

Öğretmen Ekranı

Öğretmen ekranında öğretmen hesabıyla giriş yapan kullanıcılar kendilerine gelen danışmanlık isteklerini onaylayabilirler ve danışmanlık yaptıkları öğrencileri görebilirler. Bir öğretmen en fazla üç adet öğrenciye danışmanlık yapabilir. Herhangi bir hata durumunda sol üstte uyarı mesajını görebilmektedirler. İşini tamamladıktan sonra “Giriş Ekranına Dön” butonuyla giriş ekranına dönebilir.

2.1 Veri Yapıları Örnekleri

Açılan ana sayfadan “Veri Yapıları Örnekler” butonuna tıklandığı zaman Stack, Queue, Binary Search Tree, Huffman Kodlaması ve Linked List veri yapılarını implemente ettiğimiz GUI lere ulaşabilirsiniz. Projede yalnızca Stack veri yapısı hazır java paketlerinden kullanılarak uygulanmıştır.

Yabancı Öğrenci Denetim Sistemi

Giriş Yap

Kayıt Ol

Seç

İsim

Şifre

Giriş

Seç

İsim

Şifre

Kayıt Ol

Hoşgeldiniz..

Veri Yapıları Örnekler

Proje Ait Ekran Görüntüleri

HUFFMAN

STACK

ENCODE

Huffman kodlaması yapmak için herhangi bir metin giriniz.

Hoşgeldiniz. Yazdığınız herhangi bir şeyi Push butonu ile Stack'e ekleyip, Pop butonu ile silebilirsiniz.

Ana Ekran'a Dön

Sonraki Sayfa

POP

PUSH

HUFFMAN

STACK

ENCODE

Huffman kodlaması yapmak için herhangi bir metin giriniz.

Hoşgeldiniz. Yazdığınız herhangi bir şeyi Push butonu ile Stack'e ekleyip, Pop butonu ile silebilirsiniz.

Ana Ekran'a Dön

Sonraki Sayfa

POP

PUSH

HUFFMAN

STACK

emirhan

ENCODE

00110010111011100101
{a=100, r=111, e=00, h=011, i=010, m=110, n=101}

Çıkanıldı: 4

Ana Ekran'a Dön

Sonraki Sayfa

POP

PUSH

Binary Search Tree

85

Ekle

Sil

PostOrder

PreOrder

InOrder

Sayı Eklendi: 85

85

Root

Left

Right

Önceki Sayfa

Sonraki Sayfa

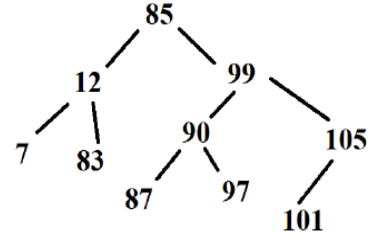
Binary Search Tree

97

7 83 12 87 97 90 101 105 99 85

85

85#12#7#99#83#105#90#87#101#97



BST 1

post : 7, 83, 12, 87, 97, 90, 101, 105, 99, 85
 pre : 85, 12, 7, 83, 99, 90, 87, 97, 105, 101
 In : 7, 12, 83, 85, 87, 90, 97, 99, 101, 105

Binary Search Tree

97

85 12 7 83 99 90 87 97 105 101

85

Binary Search Tree

97

7 12 83 85 87 90 97 99 101 105

85

Linked List

Ders Adı: Silmek istediğiniz ders kodu:

Ders Kodu:

Ders Kredisi:

Ekleme tamamlandı.

Ders Kodu	Ders Adı	Kredisi
2003	Veri Yapıları	4
2007	Mikroişlemciler	5
3004	Sistem Progra...	3

Queue

Silme Başarılı: 45

FIFO

76 689 34 15 15754 535 65467

Kuyruk başı Kuyruk Sonu

Linked List

“Ders” sınıfına ait ders adı, ders kodu, ders kredisi gibi bilgilerin girilmesiyle oluşturulan her bir ders bağılı listeye eklenmekte ve tanımlanan listeleme metodu kullanılarak arayüzde listelenmektedir. Kullanıcı silmek istediği ders kodunu ilgili text alanına girerek silme işlemini gerçekleştirebilmektedir. Silmek istediği ders koduna ait alanı boş bırakırsa veya ders ekleyeceği zaman derse ait bilgileri girerken uygun olmayan değerler kullanırsa uyarı alması sağlanmıştır.

Queue

Oluşturulan kuyruk dizisine ekleme-çıkarma metotları ve kuyruk dizisinin dolu veya boş olduğuna dair bilgiyi döndüren metotlar tanımlanmıştır. Oluşturulan arayüze ise kod entegre edilerek kullanıcının “FIFO” prensibiyle çalışan kuyruk veri yapısını uygulaması sağlanmıştır.

Binary Search Tree

Ağaca eleman ekleme ve ağaçtan eleman silme metotları tanımlandı. Daha sonra kökün konumuna göre ağaç üzerinde “Inorder”, “Postorder” ve “Preorder” gezintilerinin sonuçları yazdırıldı. Aynı zamanda arayüz tarafında ağacın kökü üzerinden yola çıkılarak içerisinde dolaşılmasını sağlandı.

Huffman Kodlaması

Girilen bir kelimenin Huffman kodlamasına göre kodlaması yapıldı. Sonuçlar gösterildi.

Stack

Stack dizisinin ve bu diziye ait eleman ekleme işleminin gerçekleştirileceği “push”, ve stack dizisinden eleman çıkarılmasını sağlayacak “pop” metotlarının tanımlanmasının ardından oluşturulan arayüzde “FILO” prensibine göre çalışan Stack veri yapısının uygulanması gösterilmiştir.