



Kocaeli Üniversitesi

Teknoloji Fakültesi

Bilişim Sistemleri Mühendisliği

Yazılım Geliştirme Laboratuvarı

Muhammet İkbāl Çakır
Bilişim Sistemleri Mühendisliği
Teknoloji Fakültesi
Kocaeli, Türkiye
mamicakir30@gmail.com

Ayşenur Elibüyük
Bilişim Sistemleri Mühendisliği
Teknoloji Fakültesi
Kocaeli, Türkiye
ayselibuyuk171@gmail.com

Mustafa Toprak
Bilişim Sistemleri Mühendisliği
Teknoloji Fakültesi
Kocaeli, Türkiye
musttoprakk@gmail.com

Özet— Bizim projemiz, öğretmenlerin ders programlarını hazırlarken karşılaştıkları zorlukları ele alarak, çizge renklendirme algoritmasını kullanarak bir optimizasyon çözümü sunar. Hocaların ders çakışmalarını önlemek, dersleri belirli gün ve saatlere atamak ve sınıf kısıtlarını göz önünde bulundurmak için geliştirilen sistem, veritabanında depolanan hoca-ders ilişkilerini ve kısıtları kullanarak çözüm üretir. Gerçekleştirilen deneyler ve sonuçlar, önerilen çözümün etkili ve ölçeklenebilir olduğunu göstermektedir. Bu çalışma, eğitim kurumları için ders programlama sürecini optimize etme konusunda yeni bir yaklaşım sunmaktadır.

Anahtar Kelimeler—Ders program, graflar, öğretmen, renklendirme algoritması, gün ve saat.

seaAmaç ve Temel Hedef— Yaptığımız proje geleneksel yöntemlerle yapılan zaman alıcı ve hata oranı yüksek olan ders programlama sürecini, çizge renklendirme algoritması kullanarak daha etkili ve daha kolay bir şekilde çözme amacını taşımaktadır. Projemiz; Öğretmenler, eğitim kurumları ve ders programları hazırlama sürecinde yer alan herkes için tasarlanmıştır. Öğretmenler, ders programı hazırlarken çakışmaları önlemek ve özel tercihleri dikkate almak isteyen herkes bu projenin hedef kitlesini oluşturmaktadır.

I. GİRİŞ

Eğitim kurumlarının düzenledikleri ders programları, öğretmenlerin, öğrencilerin ve tüm eğitim kadrosunun sorunsuz bir şekilde işlemesi için temel bir yapı taşını oluşturur. Ancak, bu programları hazırlamak, özellikle büyük ölçekli kurumlarda, karmaşık bir görev olabilir. Ders programlarını oluştururken karşılaşılan en büyük zorluklardan

biri, öğretmenlerin ders çakışmalarını ve kısıtlamalarını etkili bir şekilde yönetmektir. Geleneksel planlama yöntemleri genellikle zaman alıcıdır ve manuel hatalara açıktır.

Bu tür problemlerinin çözümünde kullanılan birçok yaklaşım bulunmaktadır. Bunlardan bazıları; Kümeleme, Genetik Algoritmalar, Ard Arda Sıralama, Mantıksal Programlama, Yerel Arama ve Graph Coloring'dir. Gerçekleştirilen bu projede bu yaklaşımlardan biri olan Graph renklendirme Algoritması kullanılmıştır.

Bu bağlamda, bu projenin amacı, öğretmenlerin ders programlarını daha etkili bir şekilde hazırlamalarına yardımcı olmak ve bu süreci optimize etmek için yeni bir yaklaşım sunmaktır. Çizge renklendirme algoritması, karmaşık programlama sorunlarını çözmek için güçlü bir araçtır ve bu proje, bu algoritmayı ders programlama sürecine başarıyla entegre ederek, öğretmenlerin ders çakışmalarını önlemelerine, özel tercihleri dikkate almalarına ve sınıf kısıtlamalarını göz önünde bulundurmalarına olanak tanımayı hedeflemektedir.

II. ÇİZGE RENKLENDİRME ALGORİTMASI

Çizge teorisi, çeşitli problemleri modelleme ve çözme yeteneğiyle birçok alanda önemli bir rol oynamaktadır. Bu alanlardan biri de ders programlamasıdır. Bu yazıda, graf renklendirme algoritmalarının ders programlama sürecindeki uygulamalarından ve optimizasyonundan bahsedeceğiz.

A. Graf Renklendirme Nedir?

Graf renklendirme, bir grafın düğümlerini ve kenarlarını belirli kurallara göre renklendirme sürecidir. Temel amaç, bir düğümün tüm komşu düğümlerinden farklı bir renk almasını sağlamaktır. Bu, özellikle bir düğümün komşularıyla çakışmadan renklendirilmesini sağlayarak belirli kısıtlamalara tabi olan problemlerde önemlidir..

B. Ders Programlama ve Graf Renklendirme

Ders programlaması, öğretmenlerin dersleri, sınıfları ve zamanları dikkate alarak etkili bir şekilde düzenlemelerini gerektiren karmaşık bir görevdir. Graf renklendirme algoritmaları, bu karmaşıklığı ele almak ve çözüm üretmek için kullanılır. Örneğin, bir öğretmenin farklı günlerdeki derslerini çakıştırmadan planlamak, sınıf sınırlamalarını göz önünde bulundurmak veya öğretmenlerin belirli gün ve saatlere ders verme isteklerini karşılamak için graf renklendirme algoritmalarından yararlanılabilir.

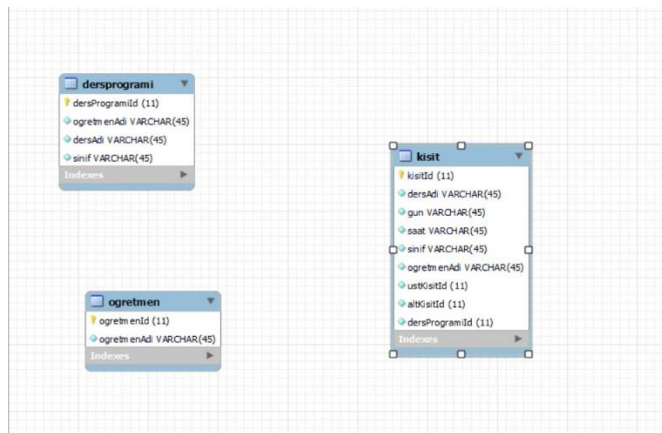
Yapılan araştırmalar, graf renklendirme algoritmalarının ders programlamasında kullanımının sadece çakışmaları önleme üzerinde değil, aynı zamanda öğretmenlerin tercihleri, sınıf sınırlamaları ve diğer kısıtlamaları ele alma konusunda da büyük potansiyel taşıdığını göstermektedir. Bu algoritmaların adaptasyonu, ders programlaması süreçlerini daha etkili ve optimize edilmiş hale getirebilir, bu da eğitim kurumlarının kaynaklarını daha verimli kullanmalarına olanak tanır.

III. PROJEDEKİ BÖLÜMLER HAKKINDA BİLGİ

Projemizi veritabanı bölümü ve projenin genel yapısı olmak üzere iki kısma ayırdık.

A. Veritabanı Bölümü

Veritabanımızda ders programı, öğretmen ve kısıt olmak üzere 3 adet tablomuz var. Ders program tablosunun içinde dersProgramId, ogretmenAdi, dersAdi ve sinif alanları mevcut. Öğretmen tablosunda; ogretmenId ve ogretmenAdi alanları var. Kısıt tablosu ise kisiId, dersAdi, gun, saat, sinif, ogretmenAdi, ustKisiId, altKisiId ve dersProgramId alanlarından oluşmaktadır. Aşağıda veritabanına ait tabloların görüntüsü verilmiştir.



Resim[1]-Veritabanı Görüntüsü

B. Projenin Genel Yapısı

Projemiz web yapısından oluşmaktadır. Web sayfasında ilk olarak bizi karşılayan navigation bar kısmı mevcut.

Navbar; Ders İşlemleri, Öğretmen İşlemleri ve Ders Programını Gör alanlarında oluşmuştur. Ders İşlemleri ve Öğretmen İşlemleri kendi içinde alanlara sahip. Bu alanlar; Ders Ekle, Ders Çıkar, Dersleri gör, Öğretmen Ekle, Öğretmen Çıkar ve Öğretmenleri Gör.

Ders Ekle ekranından öğretmen adı-soyadı, ders adı ve sınıf alanları doldurularak sisteme ders ekle işlemi yapılabilmektedir.

The screenshot shows the 'Ders Ekle' (Add Course) form. It features the Kocaeli University logo at the top left. The form has three input fields: 'Öğretmen Adı - Soyadı' (Teacher Name - Surname), 'Ders Adı' (Course Name), and 'Sınıf' (Class). Below these fields is a 'Sınıf Seçiniz' (Select Class) dropdown menu and a 'Create' button.

Resim[2]-Ders Ekle Ekranı

Dersi düzenle ekranından istenilen bilgiler değiştirildikten sonra kaydedilebilmektedir.

The screenshot shows the 'Ders Düzenle' (Edit Course) form. It features the Kocaeli University logo at the top left. The form has three input fields: 'Ders Adı' (Course Name), 'Öğretmen Adı - Soyadı' (Teacher Name - Surname), and 'Dersin Sınıfı' (Course Class). Below these fields are two dropdown menus: 'Dersin Günü' (Course Day) and 'Dersin Saati' (Course Time). At the bottom is a 'Kaydet' (Save) button.

Resim[3]-Ders Düzenle Ekranı

Ders programını gör linki üzerinden ders programına kolayca ulaşılabilir.

The screenshot shows the 'Ders Programı Tablosu' (Course Program Table). It features the Kocaeli University logo at the top left. The table has a header row with columns for 'Pazartesi', 'Salı', 'Çarşamba', 'Perşembe', and 'Cuma'. The rows represent different times of the day: 08.00, 10.00, 11.00, 12.00, 13.00, 14.00, and 15.00. The table contains course information, including course names and teacher names, with some cells containing social media icons.

Resim[4]-Ders Programı Ekranı

Son olarak Dersle Gör ve Öğretmenleri Gör Ekranından görüntüleme işlemi yapılabilir.



Ders Id	Ders Adı	Öğretmen Adı	
0	Web Tasarım	Özkan Yılmaz	✓
2	İstatistiksel Çıkarım	Kemal Kaya	✓
3	Mat	Ömerhan	✓
4	Tarih	Mehmet	✓
5	İşletim Sistemleri ve Ağlar	Yılmaz	✓

Dersler

Resim[5]-Dersleri Gör



Öğretmen Id	Öğretmen Adı	
0	Muhammed Çelik	✓
1	Yılmaz	✓
2	Özkan Yılmaz	✓
3	Yılmaz	✓

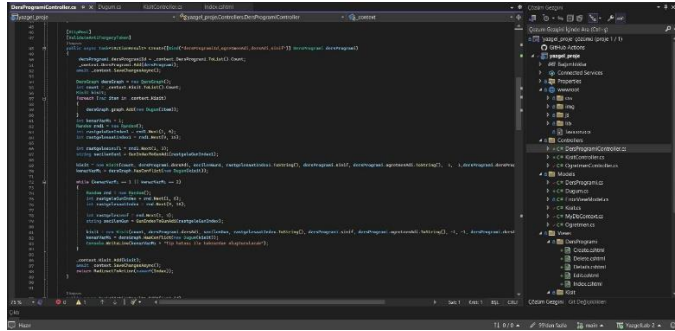
Öğretmenler

Resim[6]-Öğretmenleri Gör

IV. KOD AÇIKLAMALARI

Projenin her bir kod bloğu, projenin belirli bir yönünü ele alır. Bu kod bloklarından bahsedecek olursak;

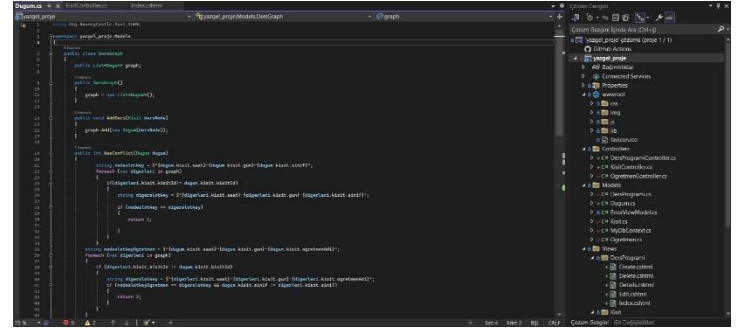
A. Ders Ekle Butonuna Basınca Çalışan Kod



Resim[7]

Yukarıdaki HTTP POST metodu, öğretmenlerin ders programlarını oluşturmasını sağlar. Kullanıcının girdiği öğretmen adı, ders adı ve sınıf bilgileriyle yeni bir ders programı oluşturulur. Ders programının benzersiz bir kimliği, mevcut ders programı sayısına bağlı olarak belirlenir ve bu program veritabanına eklenir. Ardından, oluşturulan ders programının çakışmalarını çözmek amacıyla bir graf yapısı oluşturulur. Rastgele gün, saat ve sınıf seçimleriyle çakışmalar dinamik olarak çözülür. Daha sonra, bu çakışma çözümleri veritabanındaki kısıtlar tablosuna eklenir. İlgili ders programının çakışma çözümleri başarıyla uygulandıktan sonra, kullanıcı ders programlarına ait kısıtların veritabanındaki eski kayıtlarını temizler. Bu işlemlerin ardından, ders programının başarıyla oluşturulduğu sayfaya yönlendirme yapılır. Bu kod parçası, ders programı oluşturma sürecini dinamik ve çakışmasız hale getirerek, veritabanındaki kısıtlarla uyumlu bir ders programı oluşturmaya amaçlar.

B. Graf Yapısı



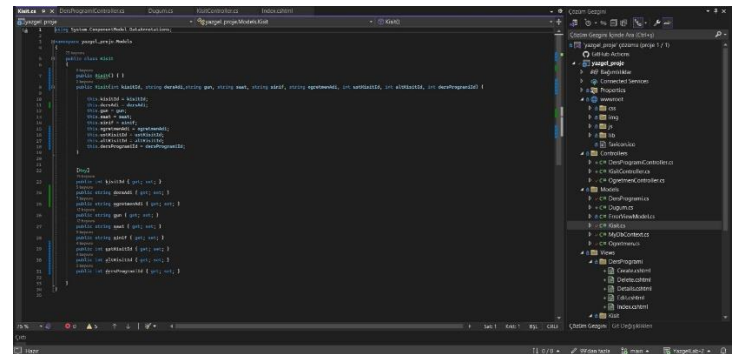
Resim[8]

'DersGraph' sınıfı, ders programının temsiliyi yapar. Bu sınıf, 'graph' adında bir liste içerir, bu liste ders programındaki her dersi temsil eden düğümleri içerir. 'AddDers' metodu, bir 'Kisit' nesnesi olarak yeni bir düğüm oluşturur ve bu düğümü 'graph' listesine ekler. 'HasConflict' metodu, bir düğümün çakışma durumlarını kontrol eder. Derslerin belirli saat, gün ve sınıf kombinasyonlarını kontrol ederek çakışma olup olmadığını belirler. Eğer çakışma varsa, farklı durumlara göre farklı değerler döndürür (1 veya 2).

'Dugum' sınıfı, bir düğümü temsil eder. Her düğüm, bir 'Kisit' nesnesini ('kisit') içerir, bu nesne o düğümdeki dersi temsil eder. Ayrıca, 'dugumlar' adında bir liste içerir, bu liste o düğüme bağlı olan diğer düğümleri temsil eder. 'kenarlar' adında bir 'Kenar' nesnesi de içerir, bu nesne düğüme bağlı olan diğer düğümlerle olan kenarları temsil eder.

'Kenar' sınıfı, iki 'Kisit' nesnesi arasındaki bağlantıyı temsil eder. Her bir kenar, bir üst ('ustKisit') ve bir alt ('altKisit') düğümü gösterir.

C. Kisit Class'ı



Resim[9]

'Kisit' sınıfı, bir dersin belirli kısıtlarını içerir. Bu kısıtlar, dersin günü ('gun'), saati ('saat'), sınıfı ('sinif'), ve dersi veren öğretmenin adı ('ogretmenAdi') şeklinde ifade edilir. Ayrıca, her bir kısıt nesnesi benzersiz bir kimliği ('kisitId') temsil eder.

Bu sınıfın yapıcı metodu, tüm bu kısıtları parametre olarak alır ve nesne oluşturulurken bu değerlere atanır. Ayrıca, sınıfın içinde 'Key' niteliği kullanılarak 'kisitId' özelliği birincil anahtar (primary key) olarak belirlenir.

