



Ders Adı : Yazılım Mühendisliği

Ders Grubu : Grup 1

Yürütücü İsmi : Prof. Dr. Oya Kalıpsız

Proje Teslim Tarihi : 04.05.2020

Öğrenci Numara, İsim ve Soy İsimleri :

14052085 – İrem Ay

15011702 – Özgür Kan

16011128 – Semih Durmaz

16011702 – Mustafa Aydın

1. Proje Tanımı

Bir dil kursu şirketinin kayıt, ödeme ve yönetim işlerinin yapılmasına yardımcı olacak bir otomasyon uygulama yazılımı.

Modüller

- **Öğrenci Kayıt Modülü**

Bu kısımda, kayıt elemanları gelen öğrencileri herhangi bir şubede açılan derslere kayıt edebilirler. Derslik kapasitesinin aşılmaması kontrol edilir.

Girdiler: Öğrenci bilgileri (ad, soy ad), kurs tercihi, ödeme bilgisi

Çıktılar: Öğrenci kursa kaydedilir.

- **Bilgi İşlem Modülü**

Bu modül, üç alt modülden oluşmaktadır:

Ders Oluşturma Modülü

Sistem yöneticisi belli bir dil için yeni bir ders açabilir. Bu dersi açabilmek için uygun öğretmen ve dersliğin seçimi yapılır.

Girdiler: Ders dilinin seçilmesi, öğretmenin seçilmesi, dersliğin seçilmesi

Çıktılar: Ders kaydı oluşturulur.

Öğretmen Kayıt Modülü

Sistem yöneticisi ya da şube elemanı sisteme yeni öğretmen kaydı yapabilir.

Girdiler: Öğretmen bilgileri (öğretmen adı soyadı, öğretebileceği diller, çalışabileceği günler ve saatler)

Çıktılar: Öğretmen bilgilerinin sisteme kayıt edilmesi

Şube Ekleme Modülü

Sistem yöneticisi sisteme yeni şubeler ekleyebilir. Bunun yanında şubelere yeni derslikler de ekleyebilir.

Girdiler: Şube bilgileri (şube adı, adresi vb.)

Çıktılar: Bilgilerin sisteme kaydedilmesi

- **Ödeme Modülü**

Öğrenci tercihinin göre kurs ücreti peşin veya taksitli olarak ödenebilir.

Girdiler: Öğrencinin ödeme bilgileri

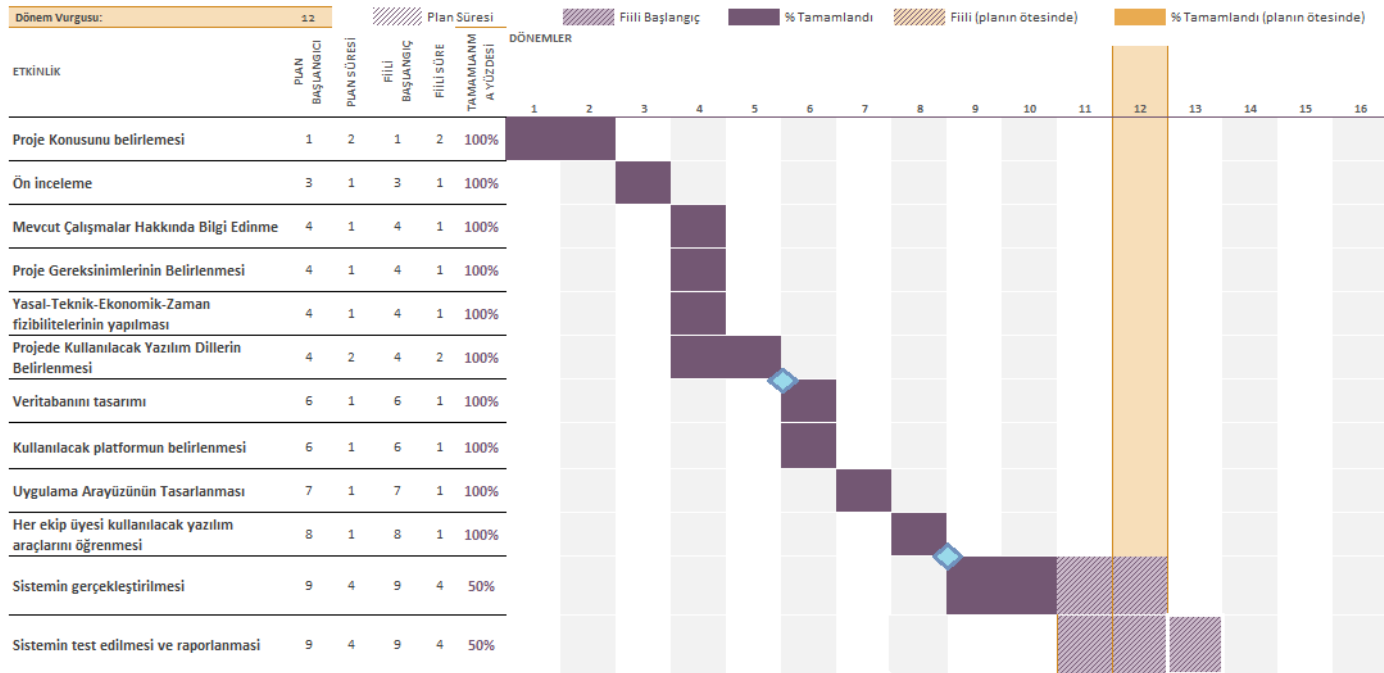
Çıktılar: Fatura oluşturulması

2. Ön İnceleme ve Fizibilite (Olabilirlik) Analizi

2.1 Fizibilite Matrisi

Fizibilite	Ağırlık	Açıklama
-> Teknik Fizibilite	%40	Kullanılan teknolojiler, piyasada da kullanıldığı için herhangi bir risk taşımamaktadırlar. Uygulama yazılımı için Visual Studio 2019 ortamında C#; veri tabanı için ise PostgreSql kullanılmıştır. Puan : 90
-> Zaman Fizibilitesi	%20	2 ay Puan : 90
-> Operasyonel Fizibilite	%40	Sadece muhasebe ve kayıt görevlilerinin ve sistem yöneticisinin ihtiyaçlarını karşılamaktadır. Puan : 80
Verimlilik puanı :		86

2.2 Zaman Fizibilitesi



Kilometre taşları (Milestones) :

- Yazılım dillerini belirleme ve veri tabanı tasarımı arası (Tasarım aşaması öncesi)
- Ekibin, yazılım araçlarını öğrenmesi ve sistem gerçekleştirimi arası (Gerçekleştirim öncesi)
-

Bolluklara (Slacks) sahip etkinlikler :

- Mevcut çalışmalar hakkında bilgi edinme
- Proje gereksinimlerinin belirlenmesi
- Veri tabanı tasarımı

3. Ekip Organizasyon Dağılımı

Proje Yöneticisi	: Mustafa Aydın
Sistem Analisti	: İrem Ay
Programcı	: Özgür Kan
Bilgi İşlem Destek Personeli	: Semih Durmaz

4. Proje Çözümleme

4.1 Gereksinimler (SRS Belgesi)

Yazılım Gereksinimleri Belirtimi Belgesi Dil Kursu Otomasyon Uygulama Yazılımı

Başlangıç :

Bir dil kursunun yönetim, kayıt ve muhasebe işlemlerini yönetmesi için bir uygulama yazılımı gereklidir.

Bilgi Toplama :

Gereksinimler :

- Gerekli olan ders için kurs oluşturabilme
- Seçilecek öğretmeni atayabilme
- Şube işlemlerini yönetebilme
- Muhasebe işlemlerini gerçekleştirebilme
- Dil kursuna kayıt olacak öğrenciler için öğrenci ders kaydı oluşturma
- Tüm bu işlemleri kayıt altına alabilmek için bir veri tabanı oluşturma

İşleme : Kullanıcı senaryoları hazırlanmış ve ilgili yazılım raporunda gösterilmiştir.

Tanımlama : Kullanım şemaları hazırlanmış ve ilgili yazılım raporunda gösterilmiştir.

Doğrulama :

- Gerekli olan ders için kurs oluşturuluyor mu? (evet)
- Seçilecek öğretmeni atanabiliyor mu? (evet)
- Şube işlemlerini yönetilebiliyor mu? (evet)
- Muhasebe işlemlerini gerçekleştiriliyor mu? (evet)
- Dil kursuna kayıt olacak öğrenciler için öğrenci ders kaydı oluşturulabiliyor mu? (evet)
- Tüm bu işlemleri kayıt altına alabilmek için bir veri tabanı oluşturuldu mu? (evet)

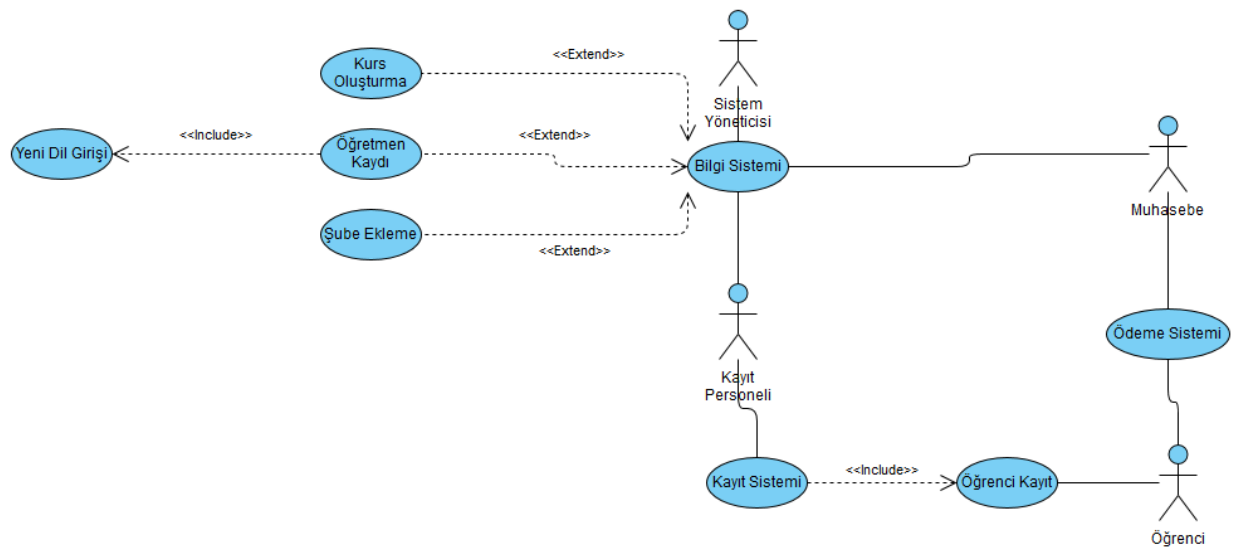
Paydaşlar :

- İrem Ay (Proje Sistem Analisti)
- Semih Durmaz (Proje Bilgi İşlem Destek Personeli)
- Oya Kalıpsız (Müşteri Temsilcisi)
- Mustafa Aydın (Gözlemci ve Oturum Yöneticisi)

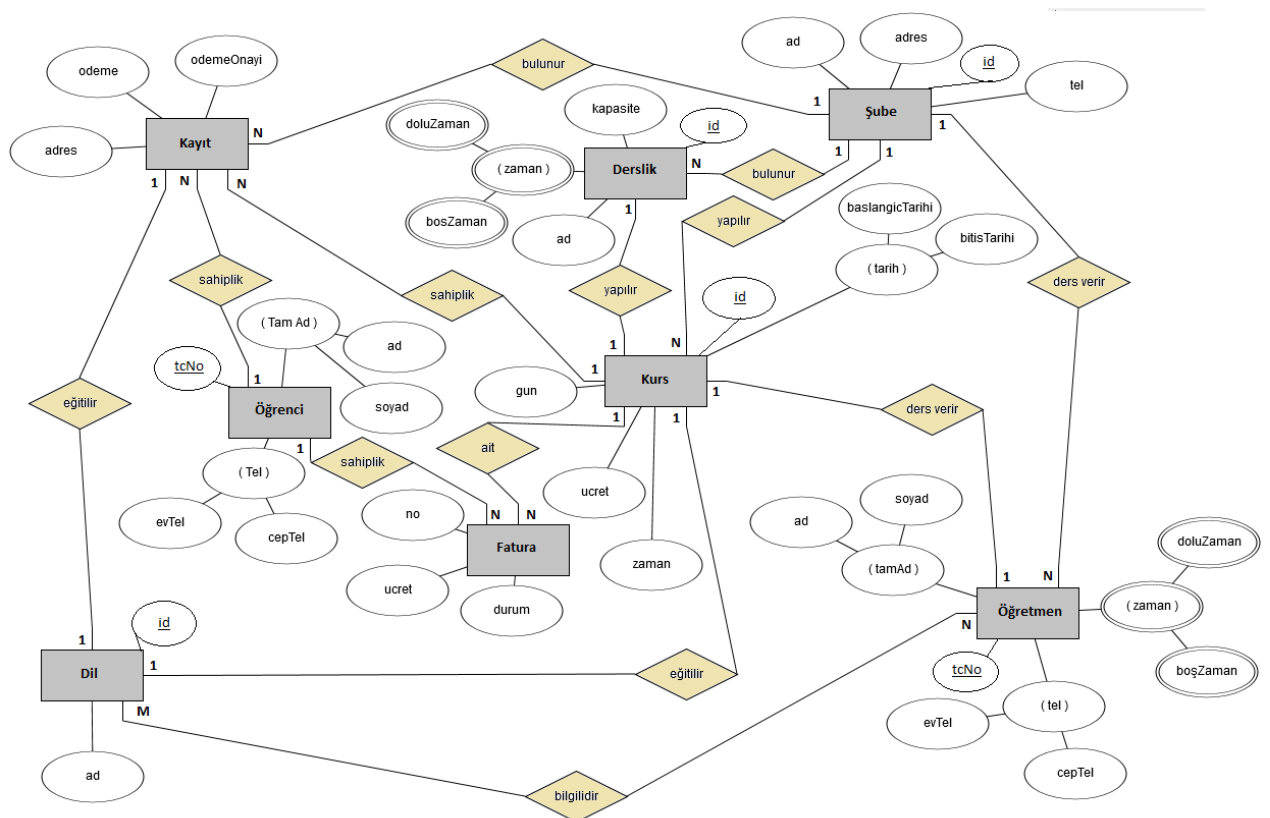
4.2 Kullanım Senaryosu

Senaryo:	Dil Kursu Otomasyon Uygulama Yazılımı
Birincil Aktör:	<ul style="list-style-type: none">• Sistem yöneticisi• Kayıt Personeli• Muhasebe
İlgililer:	<ul style="list-style-type: none">• Öğrenci• Öğretmen
Ön Koşul:	Kullanıcının sisteme giriş yapması
Son Koşul:	Derslerin oluşturulması
Ana Akış:	<ol style="list-style-type: none">1. Sistem yöneticisi sisteme veri girer.2. Belirli bir dile ait kurs bilgilerini oluşturur.3. Öğrencinin istekleri doğrultusunda bilgileri sisteme girilir.4. Kurs ödemeleri gerçekleştirilir.5. Dersler oluşturulur.
Alternatif Akış:	<ol style="list-style-type: none">2a. Şubede oluşturulacak ders saatinde boş derslik bulunmaması.<ol style="list-style-type: none">1. Ders saati değiştirilir.3a. Öğrenci tarafında istenilen şubede istenilen dil derslerinin olmaması.<ol style="list-style-type: none">1. Başka bir şubeye yönlendirilir veya öğrenci kaydı yapılamaz.

4.3 UML Kullanım Şeması (Use Case)

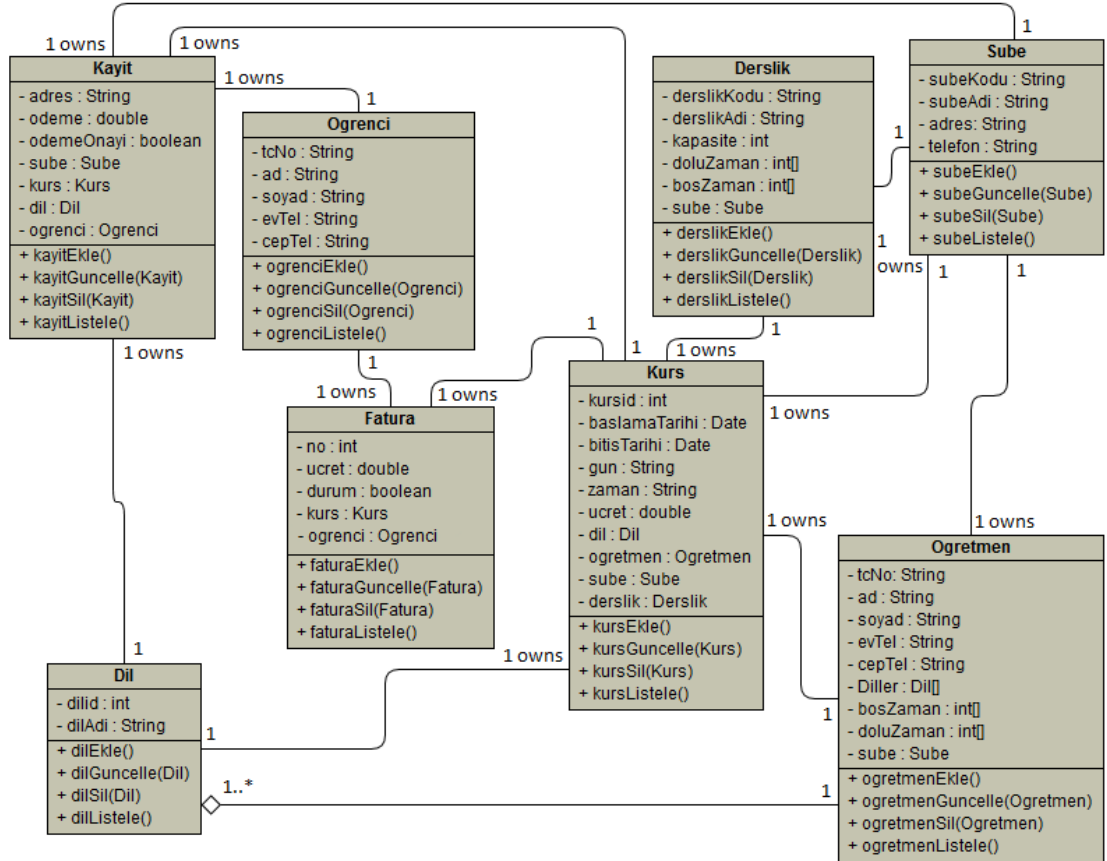


4.4 ER Diyagramı



5. Proje Tasarımı

5.1 UML Sınıf Şeması



5.2 Örnek Sözleşmeler

Her tablo için 4 ayrı sözleşme olduğundan hepsinin yerine 3 örnek verilmesi yeterli görülmüştür.

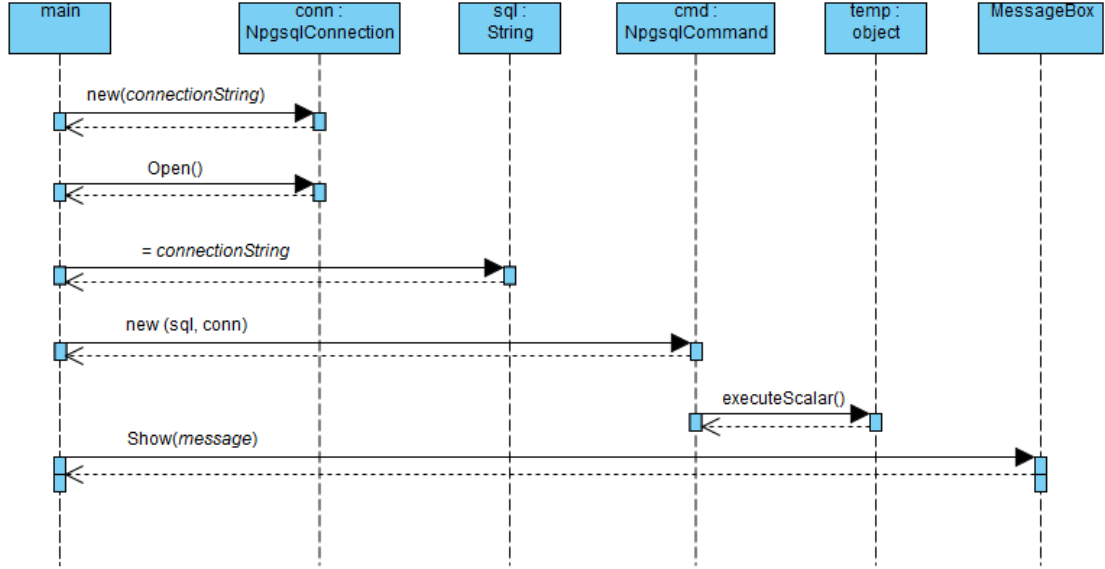
İşlem	: ogrenciEkle()
Çapraz başvurular	: Kayıt sistemi senaryosu
Ön koşullar	: Öğrenci kaydının bulunmaması
Son koşullar	: - Yeni öğrenci kaydı oluşturulur. - Kayıt için öğrenci bilgileri hazırdır. - Fatura için öğrenci bilgileri hazırdır.

İşlem	: ogretmenGuncelle(Ogretmen)
Çapraz başvurular	: Bilgi sistemi senaryosu
Ön koşullar	: Öğretmen bilgisi değişikli talebi
Son koşullar	: - İlgili öğretmenin bilgisi güncellenmiştir. - Şube değişikliği ya da dilbilgisi değişikliği durumlarına göre işlemler yapılacaktır.

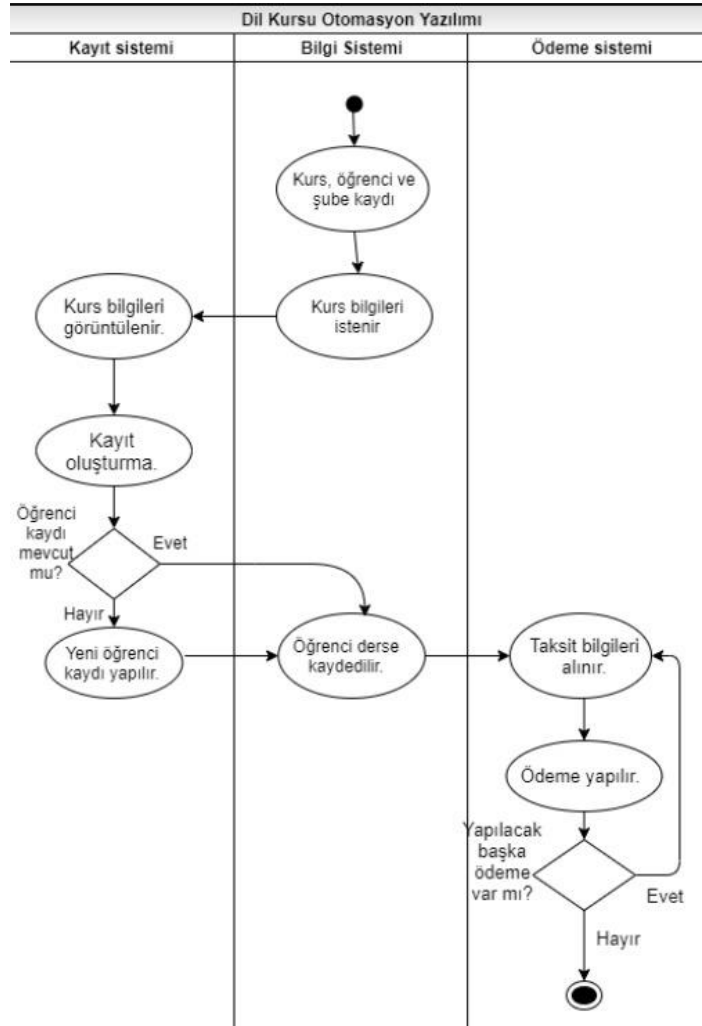
İşlem	: faturaListele()
Çapraz başvurular	: Ödeme sistemi senaryosu
Ön koşullar	: Ödeme için öğrenci talebi
Son koşullar	: - İlgili kursun ilgili faturaları görüntülenir.

5.3 Sıralama Şeması

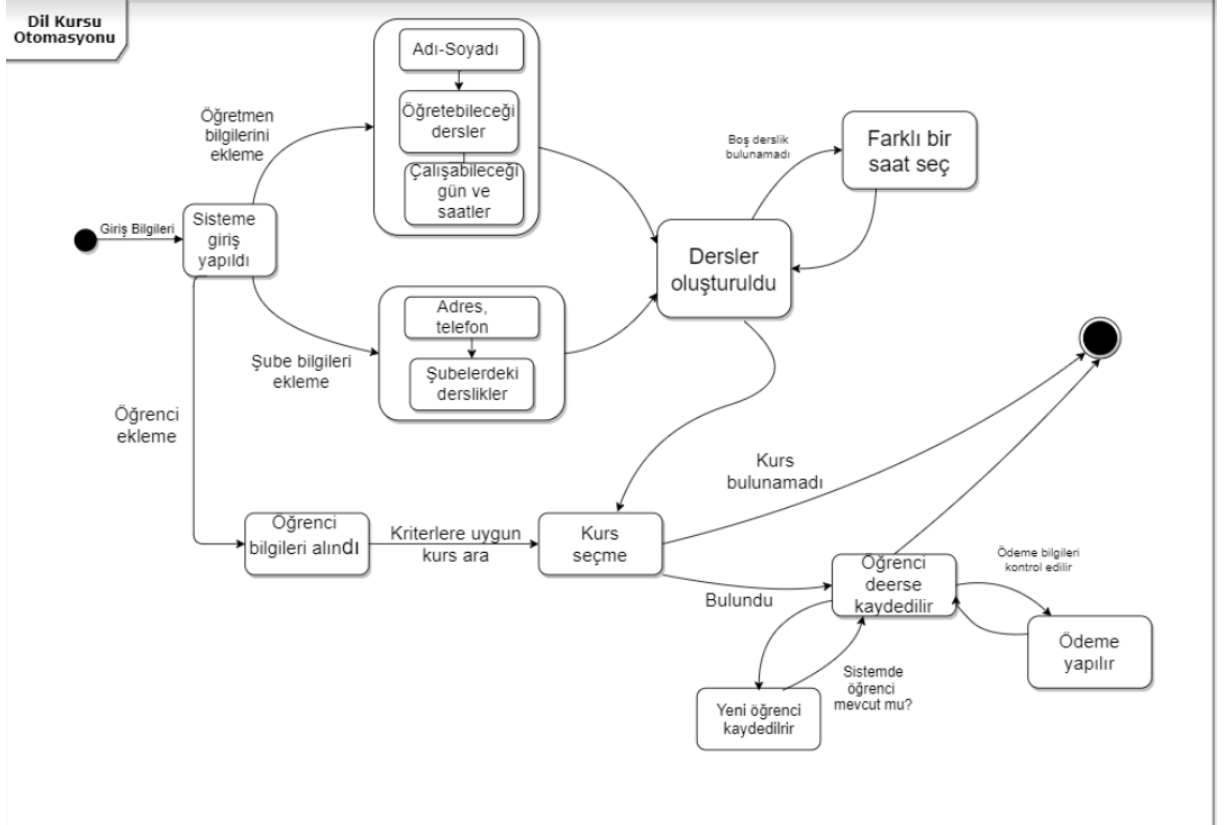
Uygulamadaki metodların her birinin genel çalışma şekli aynıdır :



5.4 UML Etkinlik Şeması



5.5 UML Durum Diyagramı



6. Risk Analizi

Risk #01 : Donanımsal Aksaklık

Olasılık : Düşük **Etki :** Yüksek **Türü :** Teknik

Açıklama : Kullanıcıların donanımları ya da veri tabanı sunucularının arızalanması.

Risk #02 : Tadilat

Olasılık : Düşük **Etki :** Orta **Türü :** Teknik

Açıklama : Dersliklerin tadilat ve restorasyon nedeniyle uygun olmaması.

Önlem : Tüm derslikleri tadilata almamak.

Risk #03 : Saat Çakışması

Olasılık : Düşük **Etki :** Orta **Türü :** Proje

Açıklama : Öğretmen ya da dersliklerin zamanlarında çakışma olma durumu.

Önlem : Tasarımı, bunu göz önüne alarak planlamak.

Risk #04 : Kayıt Gerçekleştirememe

Olasılık : Düşük **Etki :** Orta **Türü :** İş

Açıklama : Sistemsel arıza, bilgi eksikliği ya da kursun açılmaması durumunda yaşanan problem.

Önlem : - Düzenli sistem bakımı

- İyi hazırlanmış proje tasarımı

- Kayıt personelinin konuya hakimiyeti

- Saatlerin çakışmaması

Risk #05 : Ödeme

Olasılık : Orta **Etki :** Orta **Türü :** İş

Açıklama : Öğrenci tarafından ödeme yapılmaması

Risk #06 : Yazılımsal Aksaklık

Olasılık : Orta **Etki :** Yüksek **Türü :** Teknik

Açıklama : Yazılımın istenilen sonuçları vermemesi.

Önlem : - Düzenli sistem bakımı.

- Kullanıcıları, yazılım hakkında bilgilendirme.

7. Kullanılan Kaynaklar

- **Kullanılan diller :** C#, PostgreSQL

- **Kullanılan ortamlar :** Visual Studio 2019, PostgreSQL pgAdmin4

- **Yararlanılan siteler :**

<https://www.visual-paradigm.com/>

<https://vertabelo.com/>