

# TÜMLEŞİK MODELLEME DİLİ

UML

(Unified Modeling Language)

# Sınıf Diyagramları

- Sınıf Diyagramları UML 'in en sık kullanılan diyagram türüdür.
- Sınıflar nesne tabanlı programlama mantığından yola çıkarak tasarlanmıştır.
- Sınıf diyagramları bir sistem içerisindeki nesne tiplerini ve birbirleri ile olan ilişkileri tanımlamak için kullanılırlar.

# Sınıf Diyagramları

- Sınıf Adı (Name)
- Nitelikler (Property)
- İşlevler (Methods)
- Bunlara ek olarak
  - “notes”
  - “Constraints”



# Sınıf Diyagramları

- Örnek:

SınıfAdı
Özellik 1 : tür 1 Özellik 2 : yaş="19" ...
İşlev 1() İşlev 2(parametreler) İşlev 3():geri dönen değer tipi ...

Private: -

Protected: #

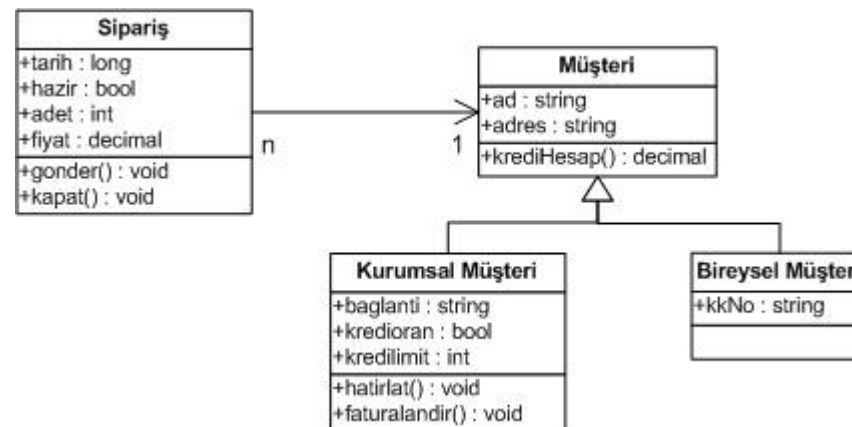
Public: +

Internal: ~

Müşteri
İsim Kod
SiparisVer() SiparisAl()

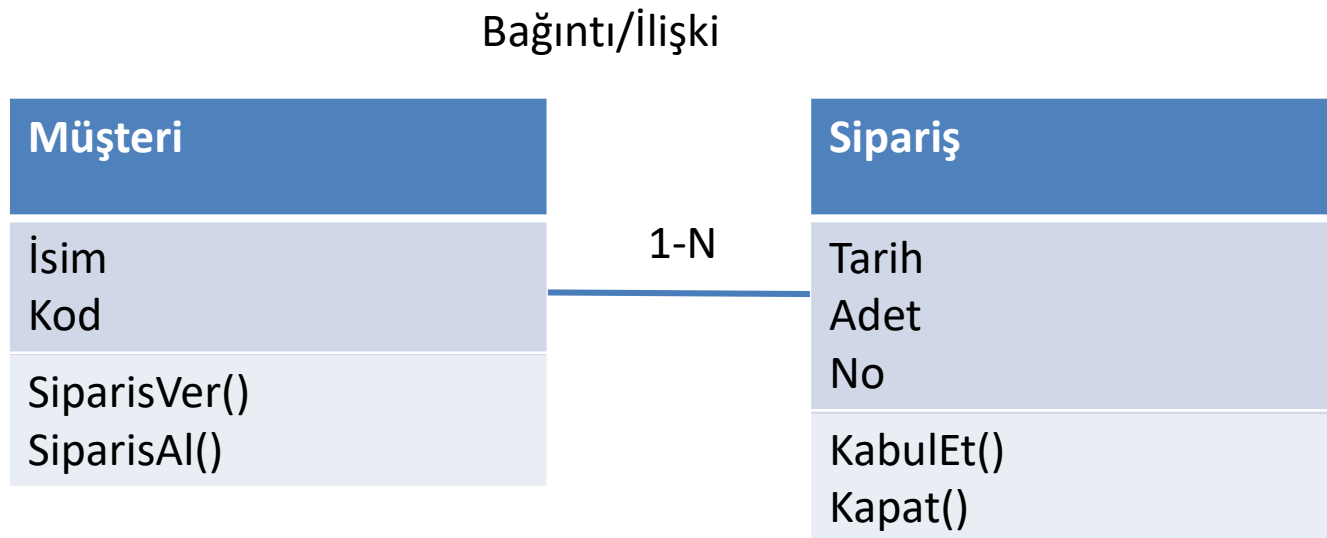


Genelleştirme.  
Kalıtım gibidir.  
İki sınıfın  
birbirine benzer  
olup da sadece  
küçük farklarla  
ayrıldığı  
durumlarda



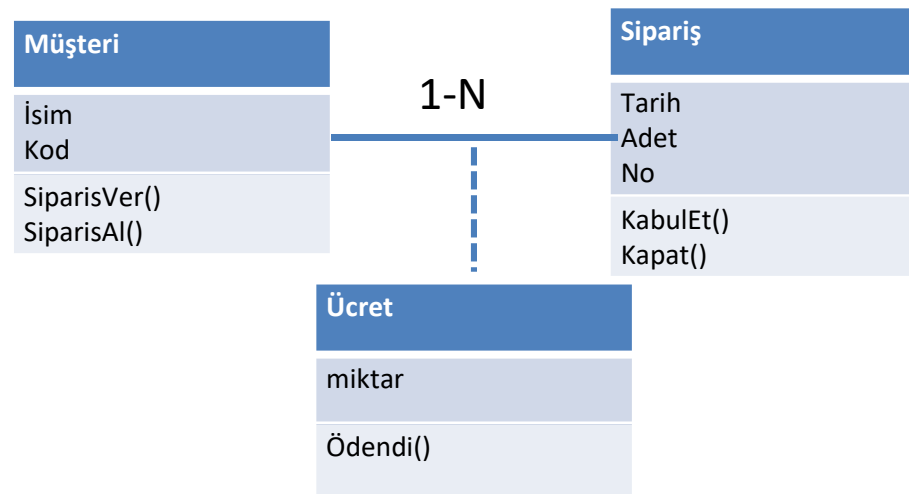
# Sınıf Diyagramları

- Sınıflar arasındaki ilişkiyi göstermek için iki sınıf arasına düz bir çizgi çekilir.
- İlişkiyi gösteren çizginin üzerine ilişkinin türü yazılır.



# Sınıf Diyagramları

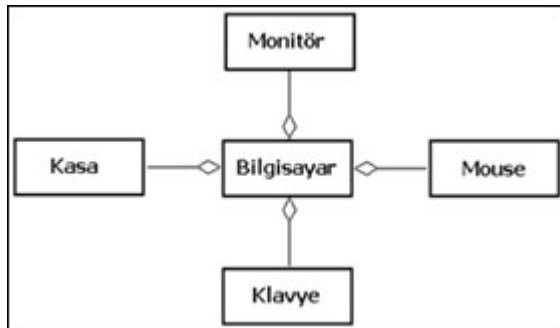
- Müşteri ile Sipariş sınıfı arasında ilişki vardır. Fakat müşteri satın alırken Ücret ödemek zorundadır
- Bu ilişkiyi göstermek için Ücret sınıfı ilişki ile kesikli çizgi ile birleştirilir.



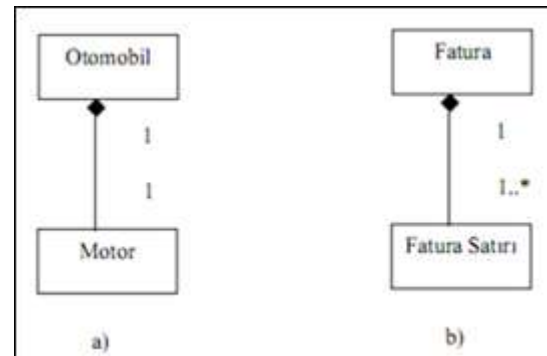
# Sınıf Diyagramları

- Aggregations İlişkisi
  - Birden fazla parçadan oluşan sınıflar arasındaki ilişki
- Composition İlişkisi
  - Bütün ve parça ilişkisi gibidir. Aggregations 'a benzerdir. Ama bu ilişkide:
  - Parçalar bütün ile anlamlıdır.
  - Tek başına kullanmak yerine bir sınıfın parçası olarak var olur ve sonlanırlar.
  - Bir parçayı birden fazla bütün kullanmaz.

Aggregations

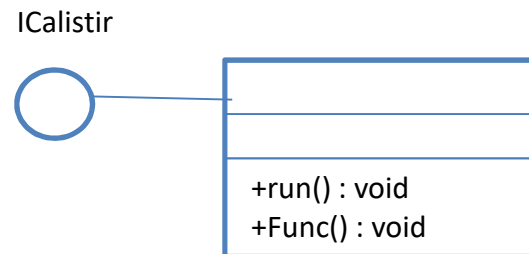
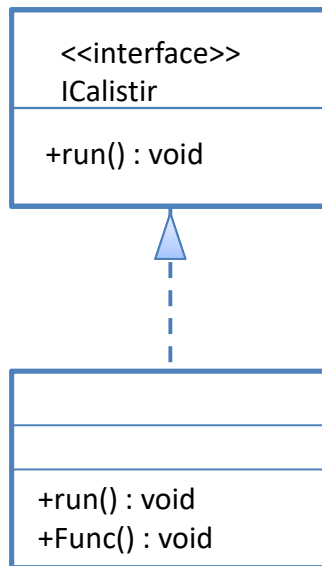


Composition



# Sınıf Diyagramları

- Realization İlişkisi
  - Interface modellemede kullanılır. Interface metodlarının gerçekleştirilmesidir.



```
public interface A {  
    ...  
} // interface A
```

```
public class B implements A {  
    ...  
} // class B
```



# Sınıf Diyagramları

Dependency (uses):

Bağımlılığı gösterir.

(uses) ile bir sınıfın başka bir sınıfın metoduna parametre olarak verildiğini ifade eder.

```
public class A {  
    public void doSomething(B b) {
```

Dependency  
(uses)  
(Bağımlılık)

```
public class A {  
    private B _b;  
    public void setB(B b) { _b = b; }
```

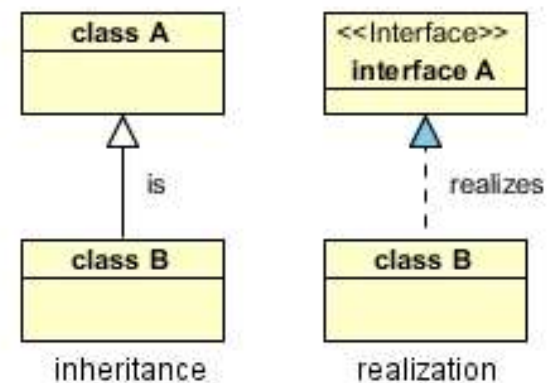
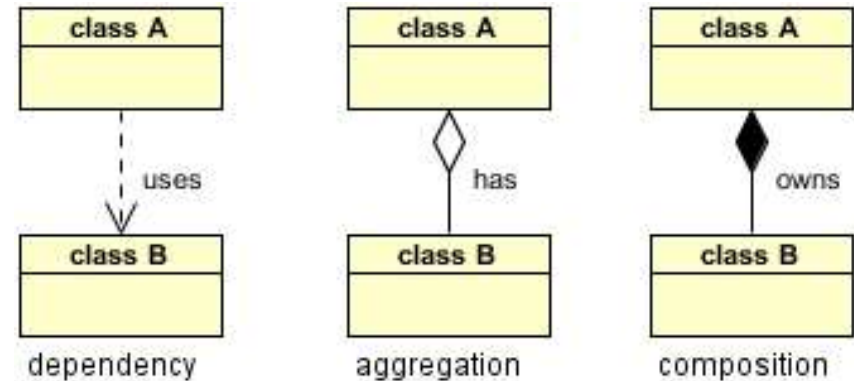
Aggregation  
(içerme)

```
public class A {  
    private B _b = new B();
```

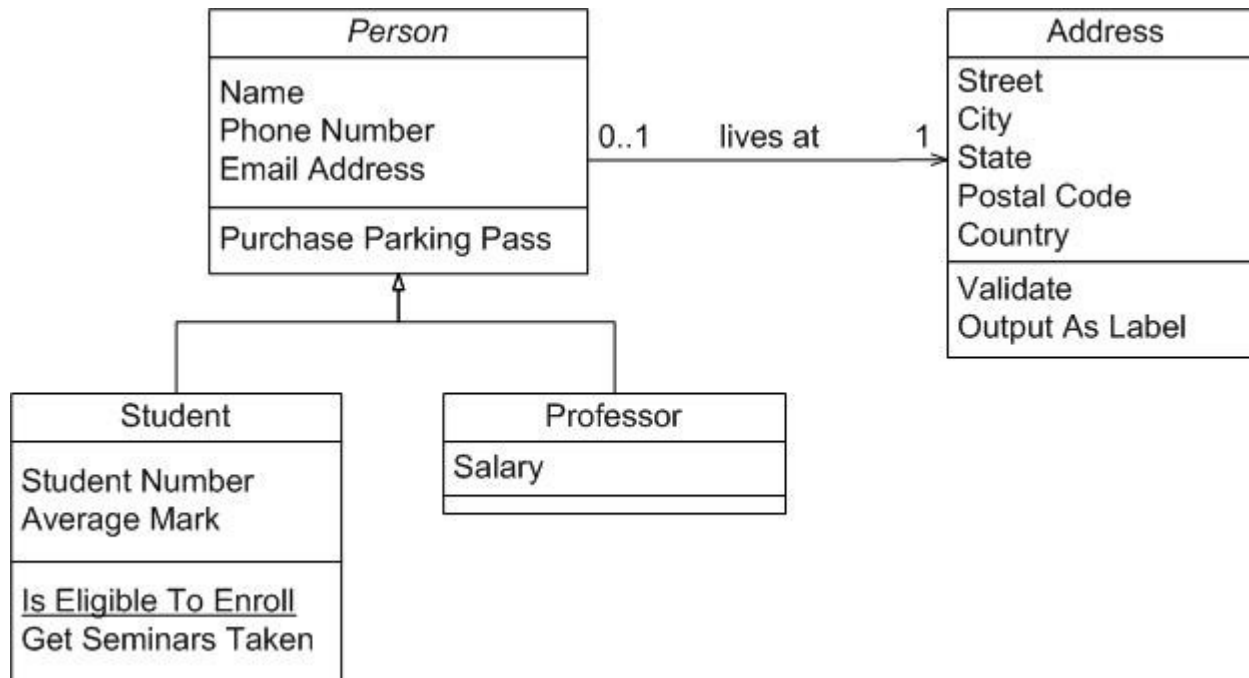
Composition  
(Oluşum)

```
public class A {  
    ...  
} // class A  
public class B extends A {  
    ....  
} // class B
```

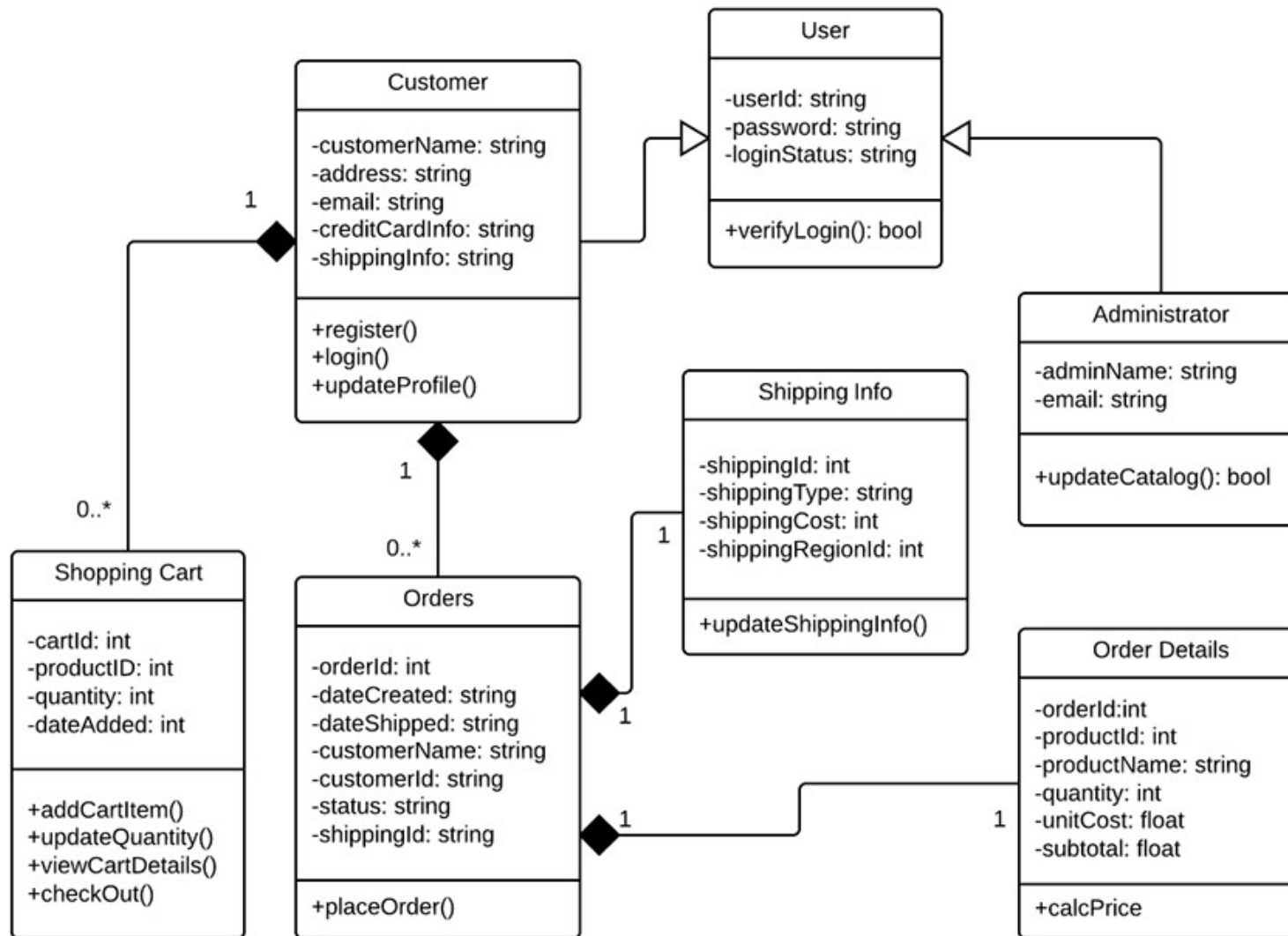
Inheritance /  
Generalization  
(Kalıtım /  
Genelleştirme)



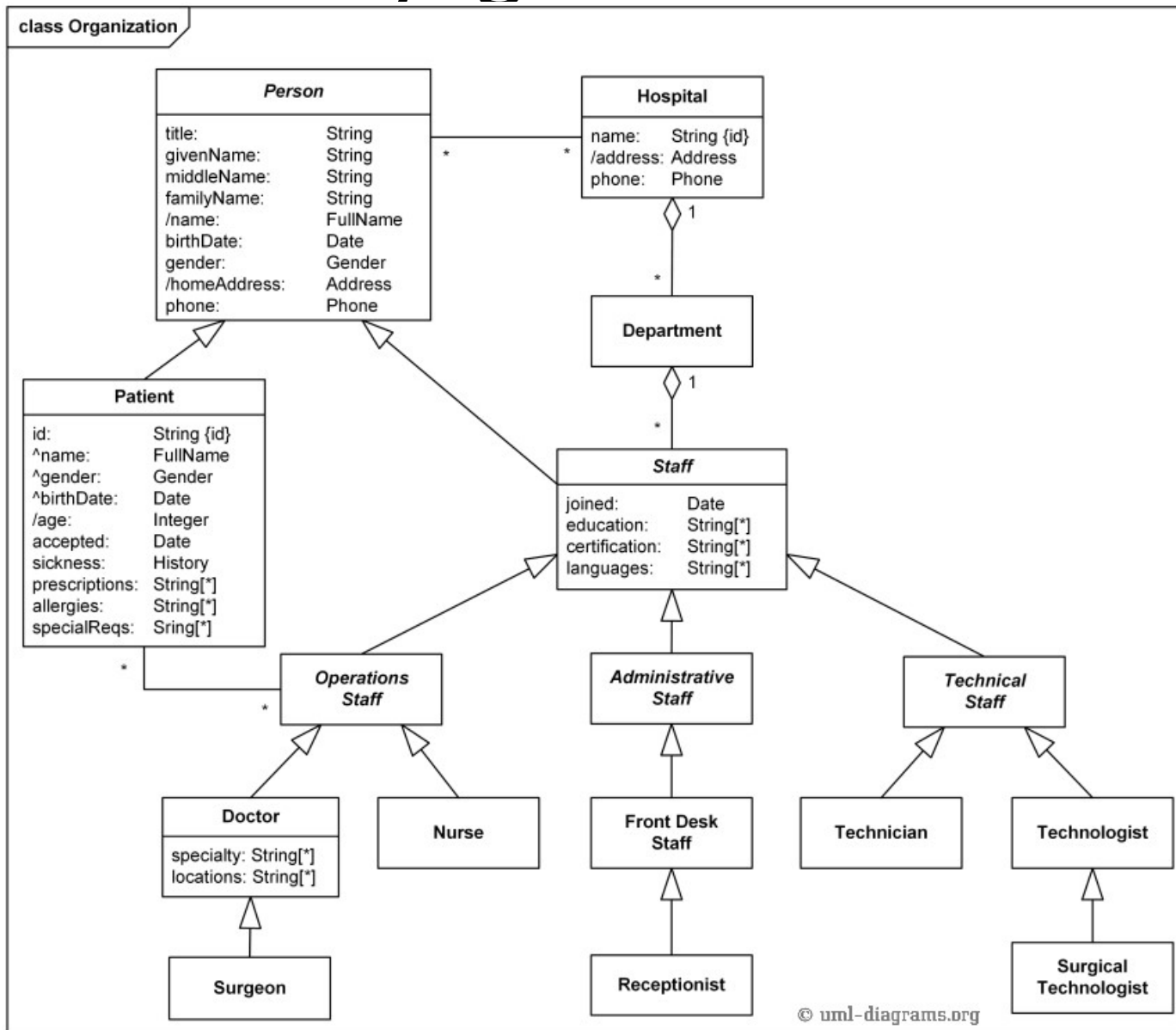
# Sınıf diyagramları Örnek



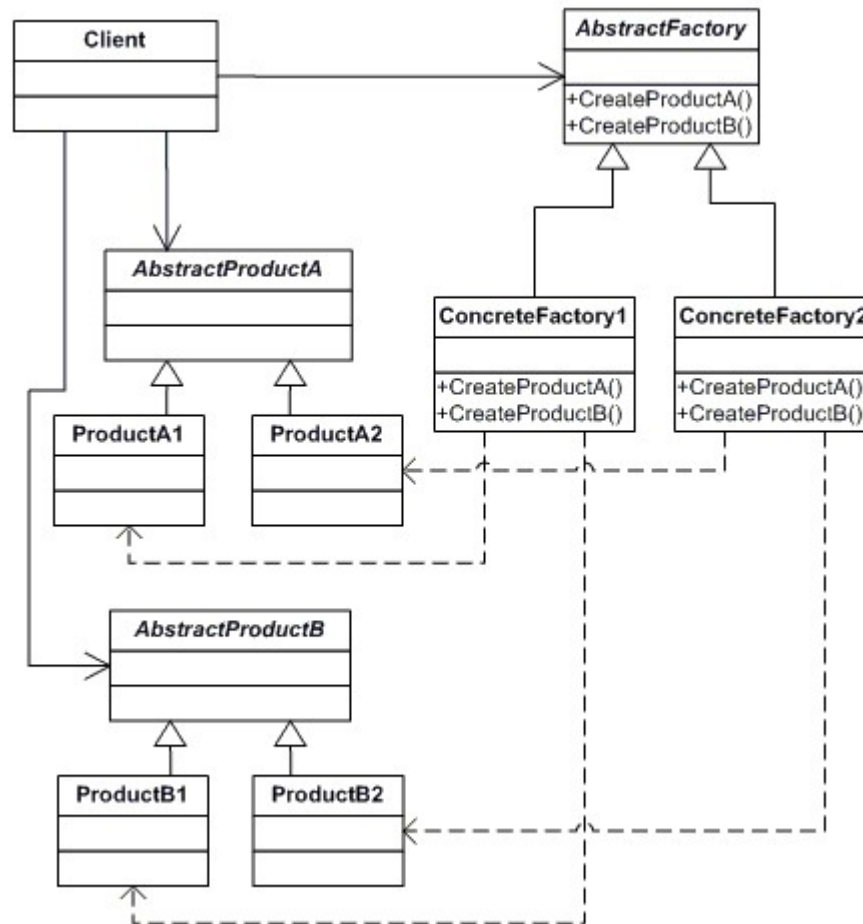
# Sınıf diyagramları Örnek



# Sınıf diyagramları Örnek



# Sınıf diyagramları Örnek



# Soru

- Üniversite kütüphane otomasyonu için class diyagramı çıkarınız.
- KTÜ kütüphanesi özelliklerini ve kullanıcılarını düşünerek tasarım yapınız.

# Kullanım Senaryosu Diyagramları

- Aktivite diyagramları ile analiz aşamasında kullanılırlar.
- Sistemin kullanıcılarına neleri sunduğunu ortaya çıkarmakta kullanılır.
- Kullanıcının taleplerinin ile sistemin sundukları ile örtüşüp örtüşmediği ya da bu taleplerin ne ölçüde karşılandığı gösterilir.
- Fonksiyonel gereksinimlerinin ortaya çıkarılması için kullanılır.
- Sistemin kullanıcısının bakış açısıyla modellenmesi amacıyla kullanılır.
- Sistemin kabaca ne yaptığı ile ilgilenir, kesinlikle nasıl ve neden yapıldığını incelemez.

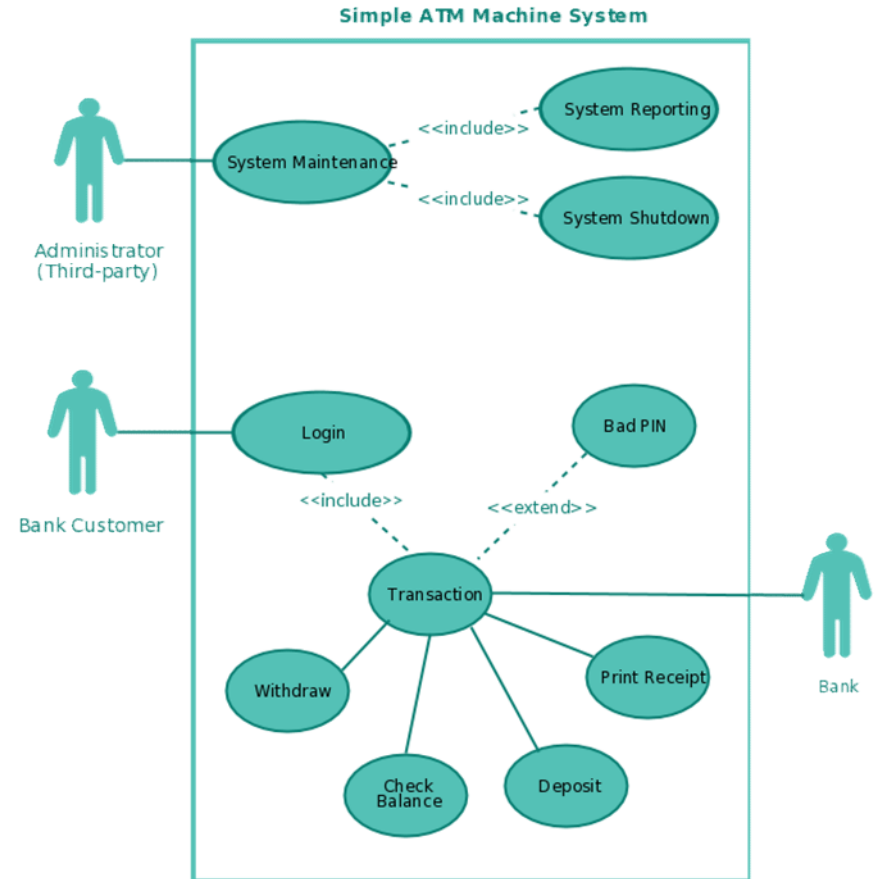
# Kullanım Senaryosu Diyagramları

- Sistemde yapılan bir işin ifade edilmesidir.
- **Örneğin:**
- Veritabanına kullanıcı eklenmesi, satış raporlarının alınması.
- **Sor!**
- *Sistemde gerekli fonksiyonlar nelerdir?*
- **Not:** Use case diyagramları anlatımla desteklenir.
- **Ödev:** Use case anlatımında kullanılan temel kavramlar nelerdir? Bu kavramları açıklayınız ve örnek bir use case ile bu kavramları kullanarak anlatım yapınız. Örnek: Assumption, Pre-condition, ...



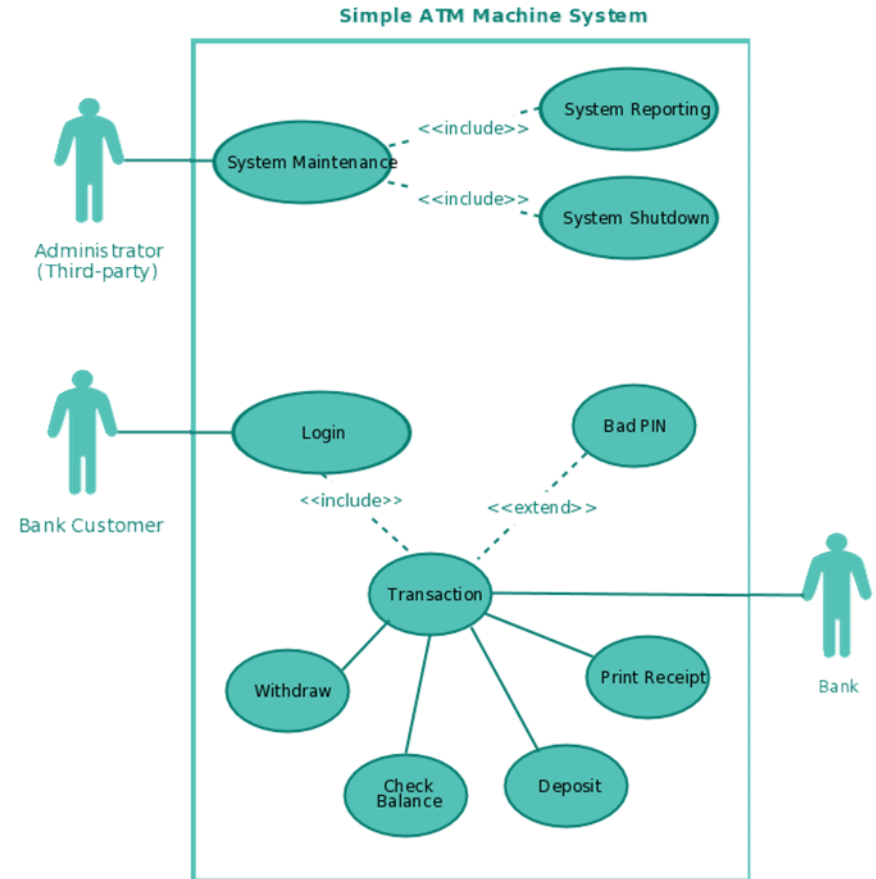
# Kullanım Senaryosu Diyagramları

- Kullanım senaryosu
  - **Aktörler:** çeşitli yetkilere sahip gerçek kişi ya da sistemler. (kullanıcılar, veritabanı, harici sistemler ya da donanımlar.)
  - **Sor!**
  - *Sistemi kimler kullanacak?*
  - Aktörler sistemin dışında olmalıdır. Sistemin içindekiler aktör değildir.

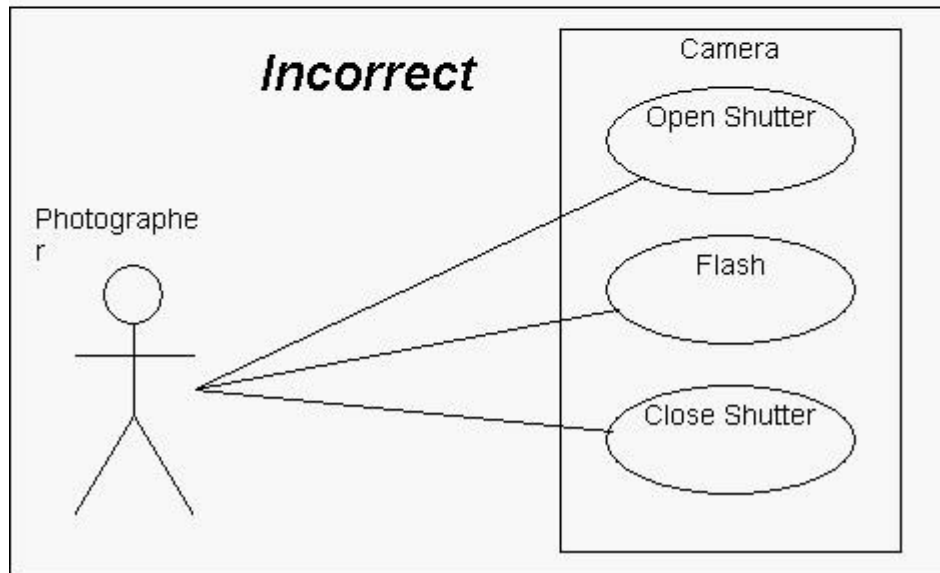
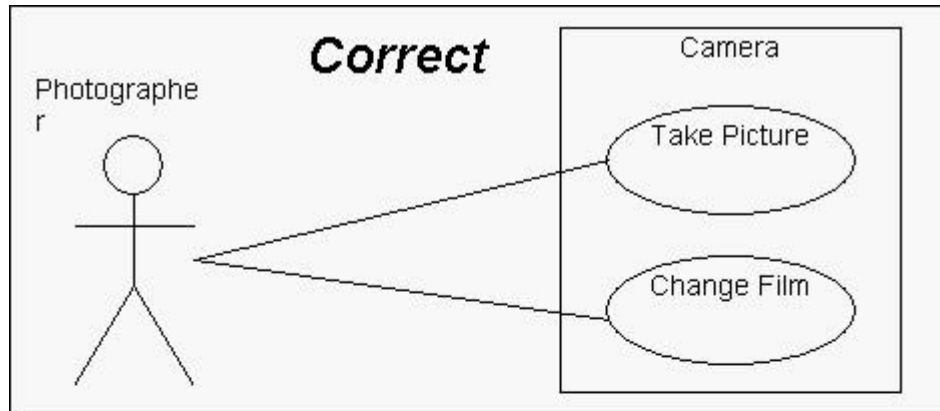


# Kullanım Senaryosu Diyagramları

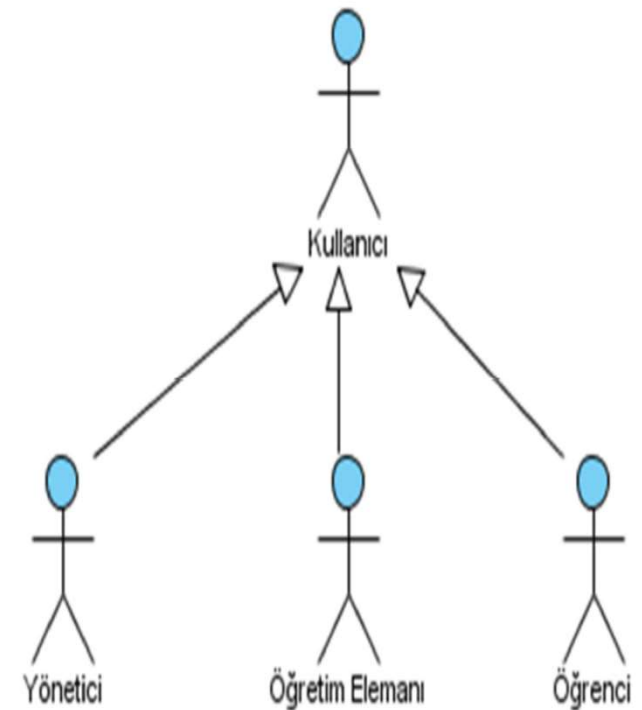
- Kullanım senaryosu
  - **<<extends>>**: Genişletme ilişkisi. Bir senaryonun özelliklerini derinleştiren senaryolar (ek özellik/durum) ya da senaryoyu ana akıştan saptıran durumları ifade etmekte kullanılır.
  - **<<include>>**: İçerme ilişkisidir. Bir senaryonun içerdiği başka bir senaryo olarak (bir alt programa dallanma) gibi düşünülebilir.



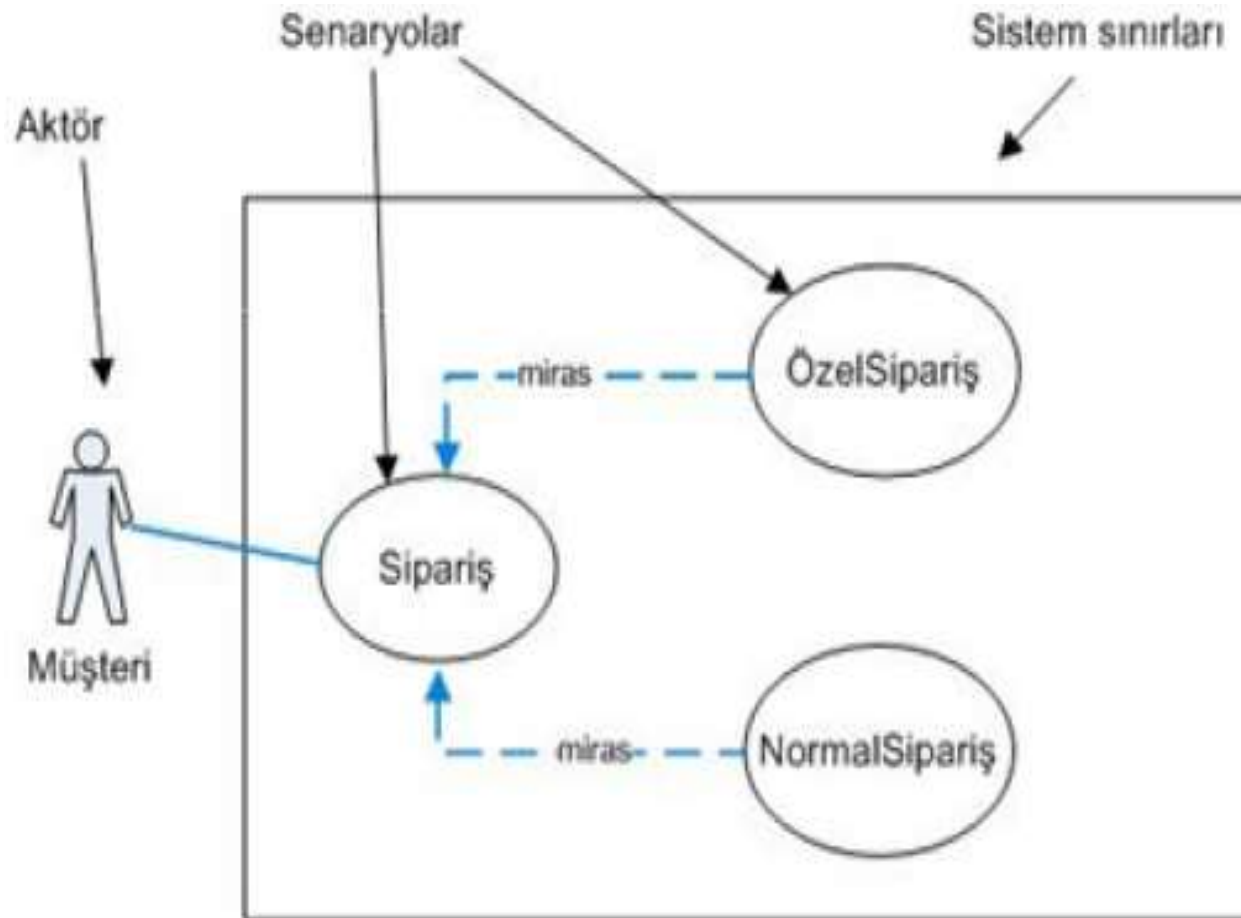
# Örnek Kullanım Senaryosu Diyagramı



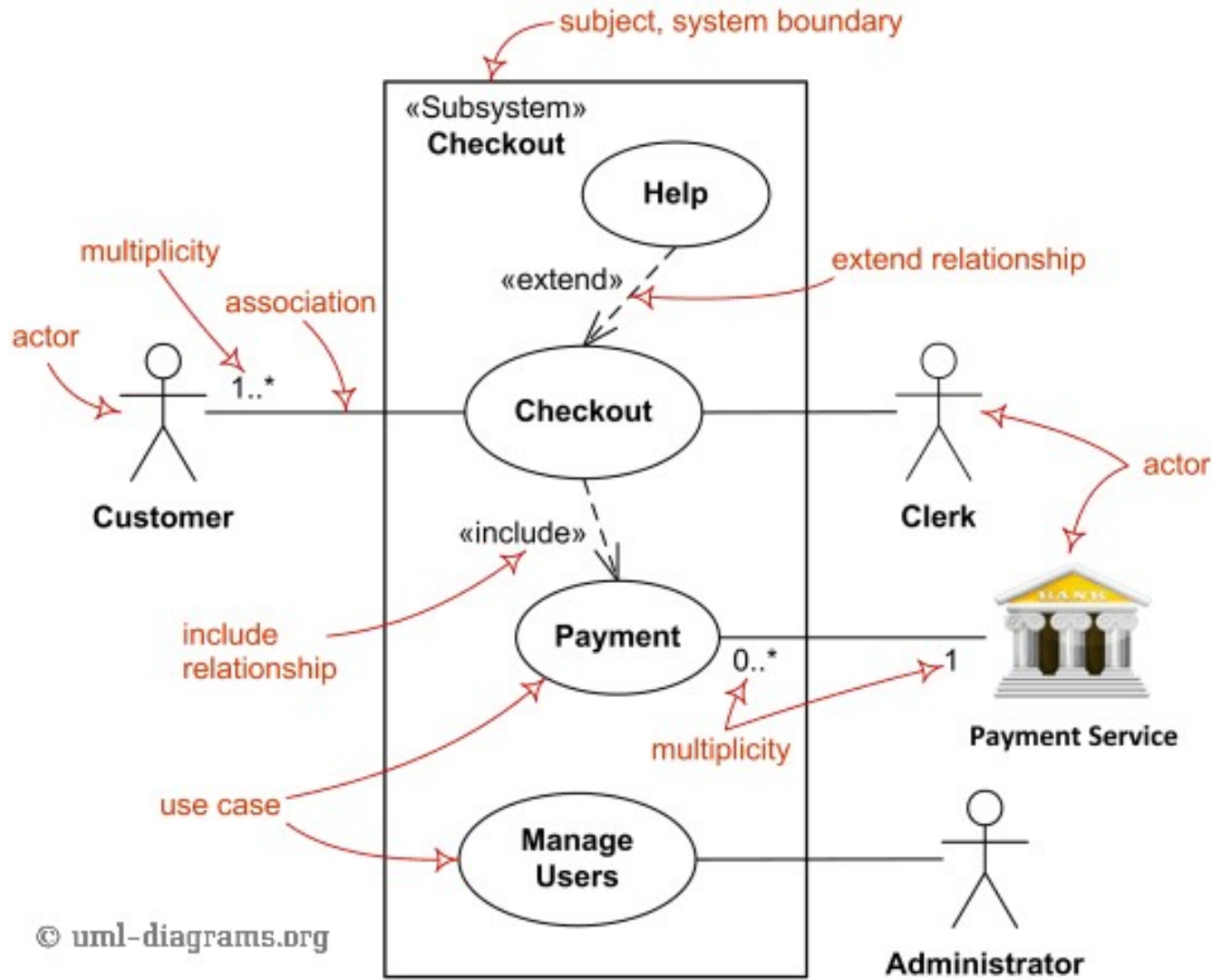
Genelleme ilişkisi  
miras almaya  
benzer.



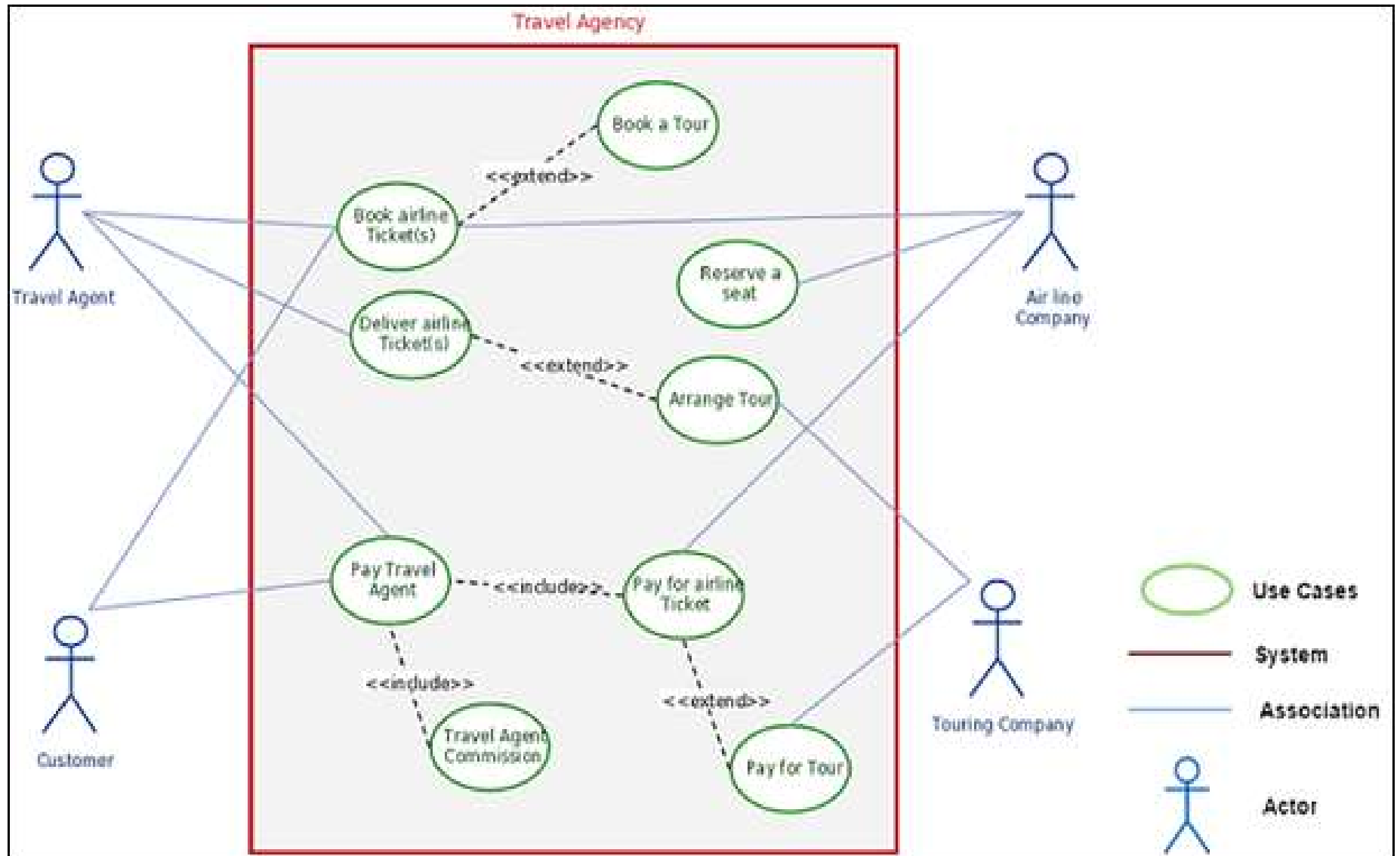
# Örnek Kullanım Senaryosu Diyagramı



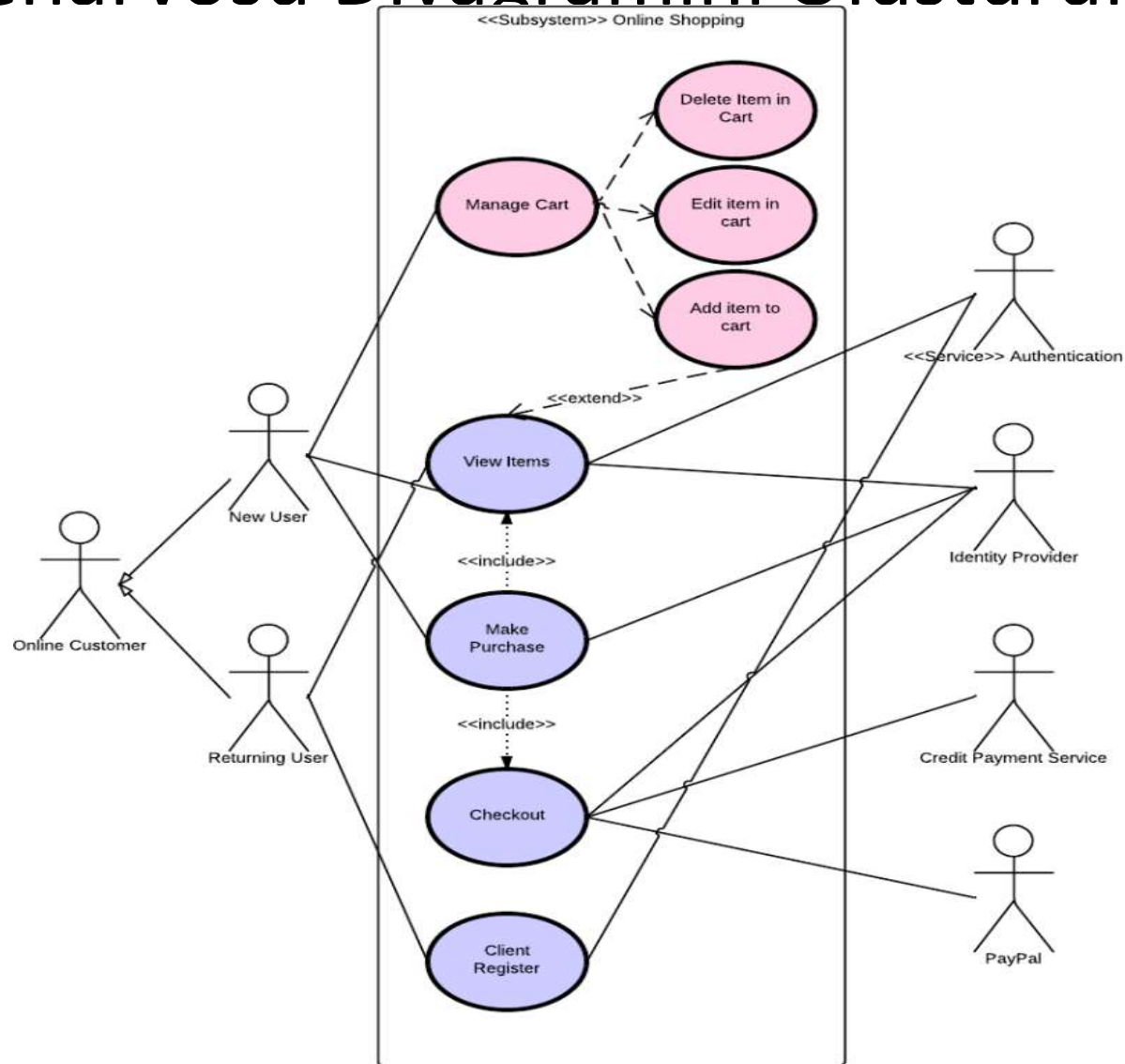
# Örnek Kullanım Senaryosu Diyagramı



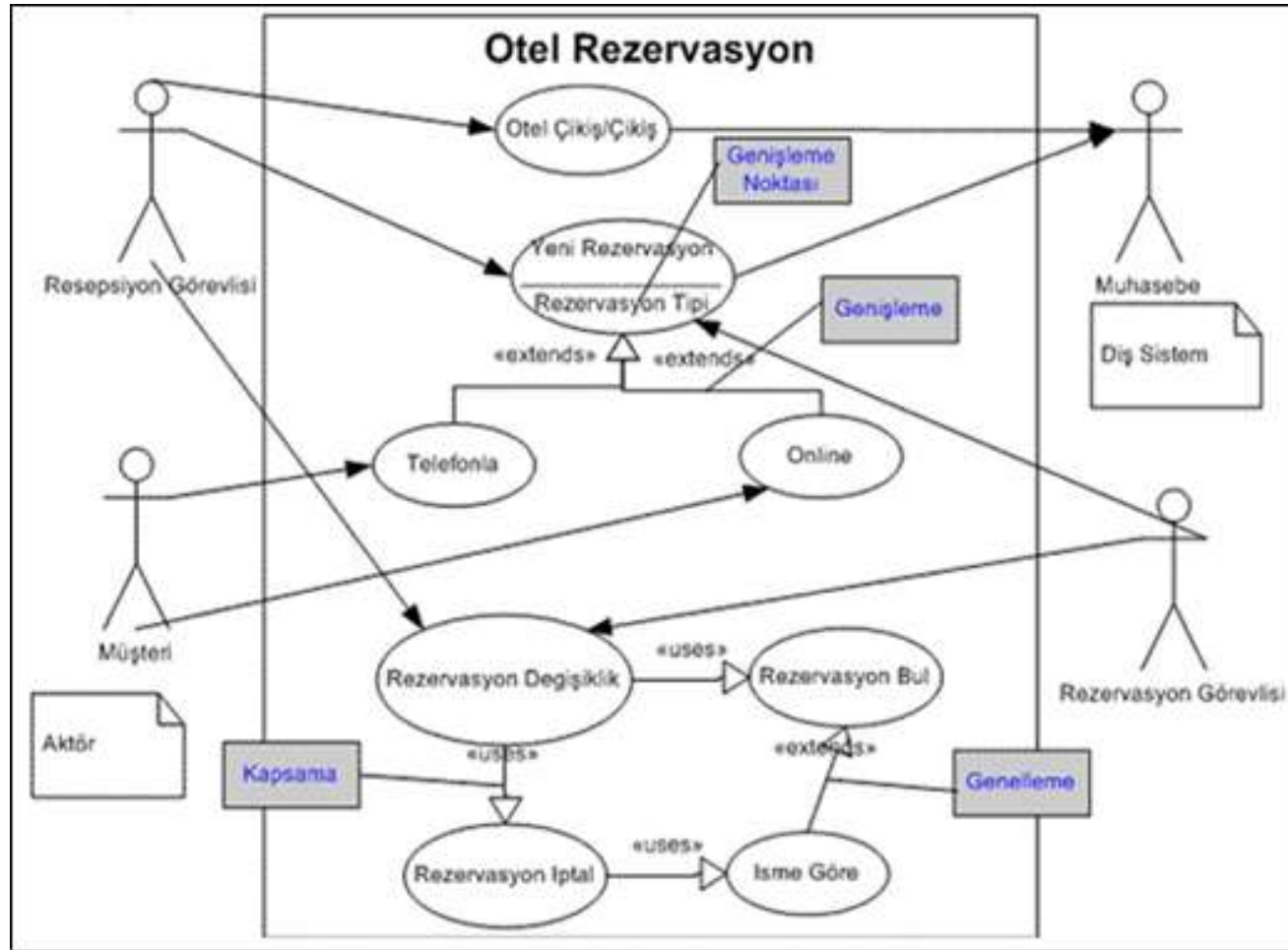
# Soru: Seyahat Acentası için Kullanım Senaryosu Diyagramı?



# Soru: Online Alışveriş için Temel Kullanım Senaryosu Diyagramını Oluşturunuz



Soru: Otel rezervasyonu için kullanım senaryosu diyagramını oluřturunuz.





# Soru

- Üniversite kütüphane otomasyonu için use case diyagramı çıkarınız.
- KTÜ kütüphanesi özelliklerini ve kullanıcılarını düşünerek tasarım yapınız.

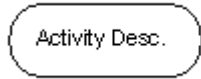
# Etkinlik Diyagramları

- Sistemdeki nesnelerin faaliyetlerini göstermek amacıyla kullanılırlar
- Durum diyagramlarının bir alt kümesi olarak değerlendirilen etkinlik diyagramları, iş akışlarının grafiksel gösteriminde kullanılırlar.

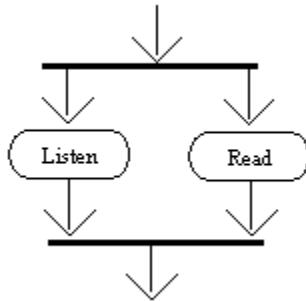
# Etkinlik Diyagramları



Başlangıç  
noktası



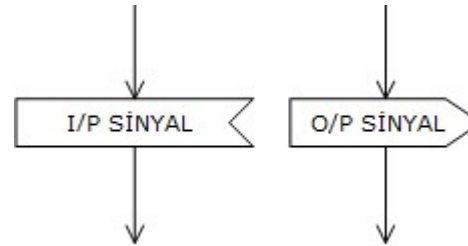
Aktivite.  
Sistem  
işleyişinin  
bir parçası



Eş  
zamanlı  
aktivite



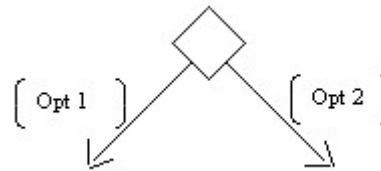
Aktivite  
bitişi



Sinyal(Signal): Aktivite mesaj  
gönderiminde yada mesaj alma işlemi  
sinyal olarak adlandırılır. Modellemede  
iki çeşit sinyal tipi bulunmaktadır.

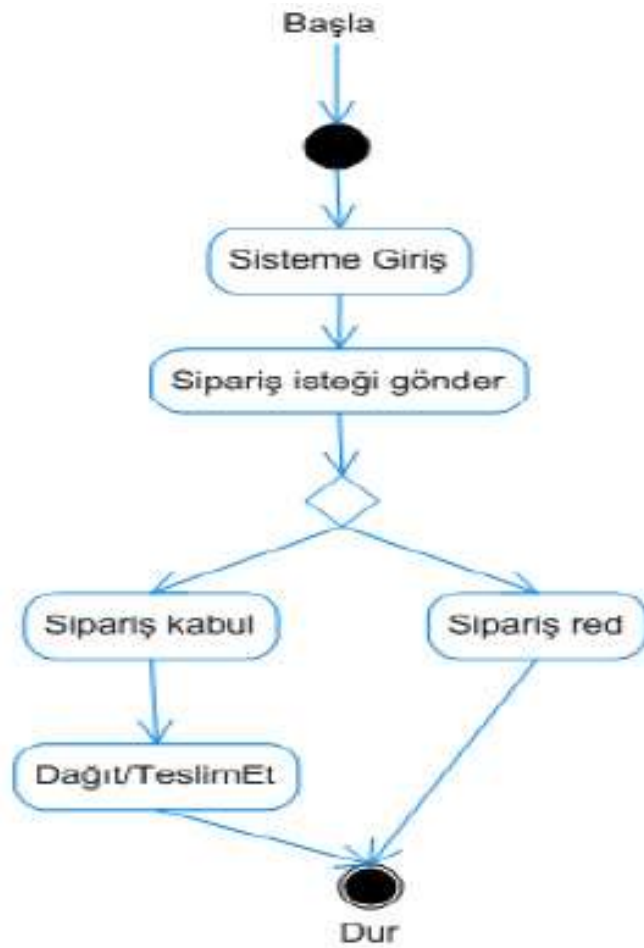
Giriş Sinyali(Input Signal): Mesajın  
aktivite alması

Çıkış Sinyali(Output Signal): Mesajın  
aktivite göndermesi.



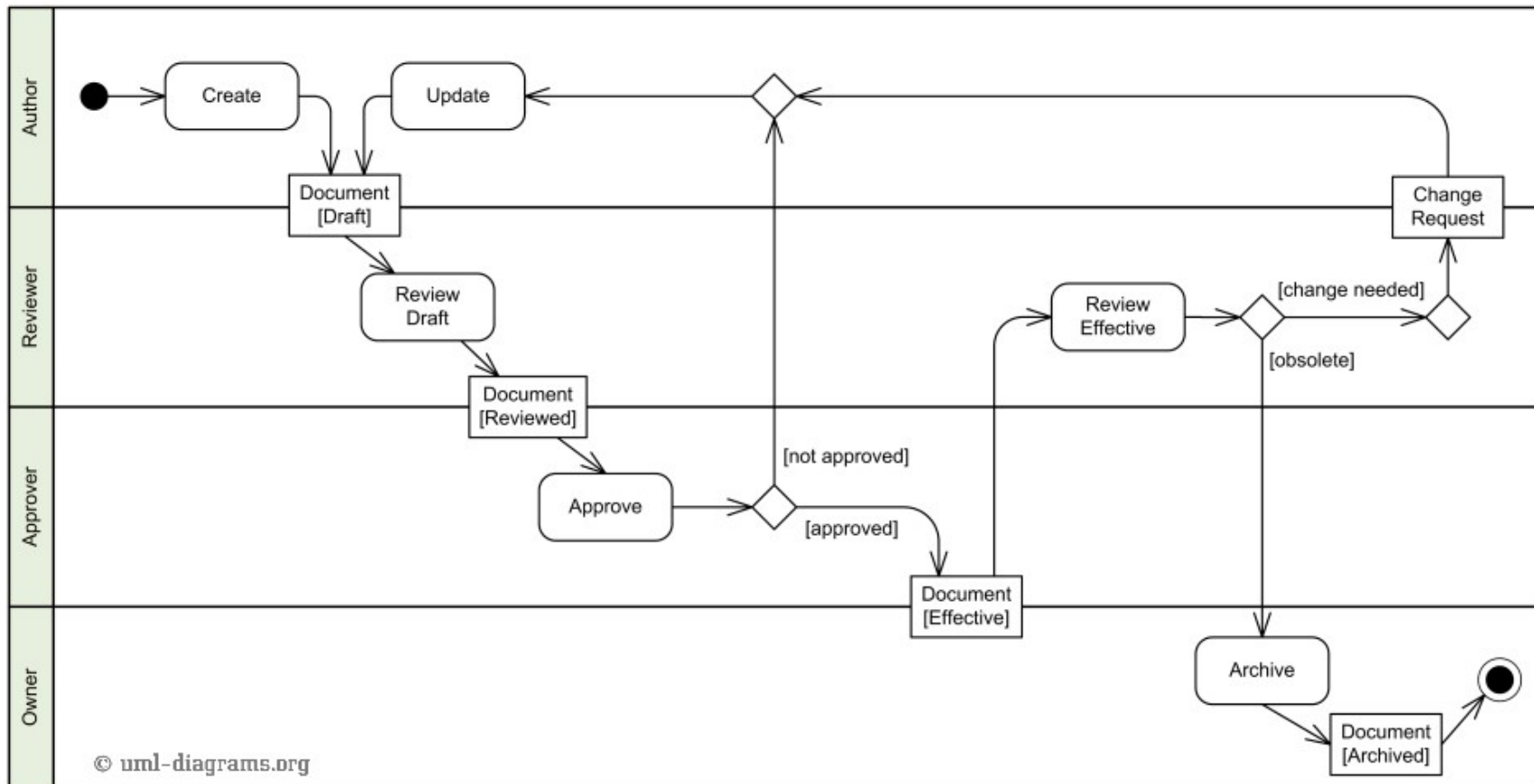
Seçim gösterimi.  
Duruma göre  
farklı noktalara  
ilerler.

# Etkinlik Diyagramları

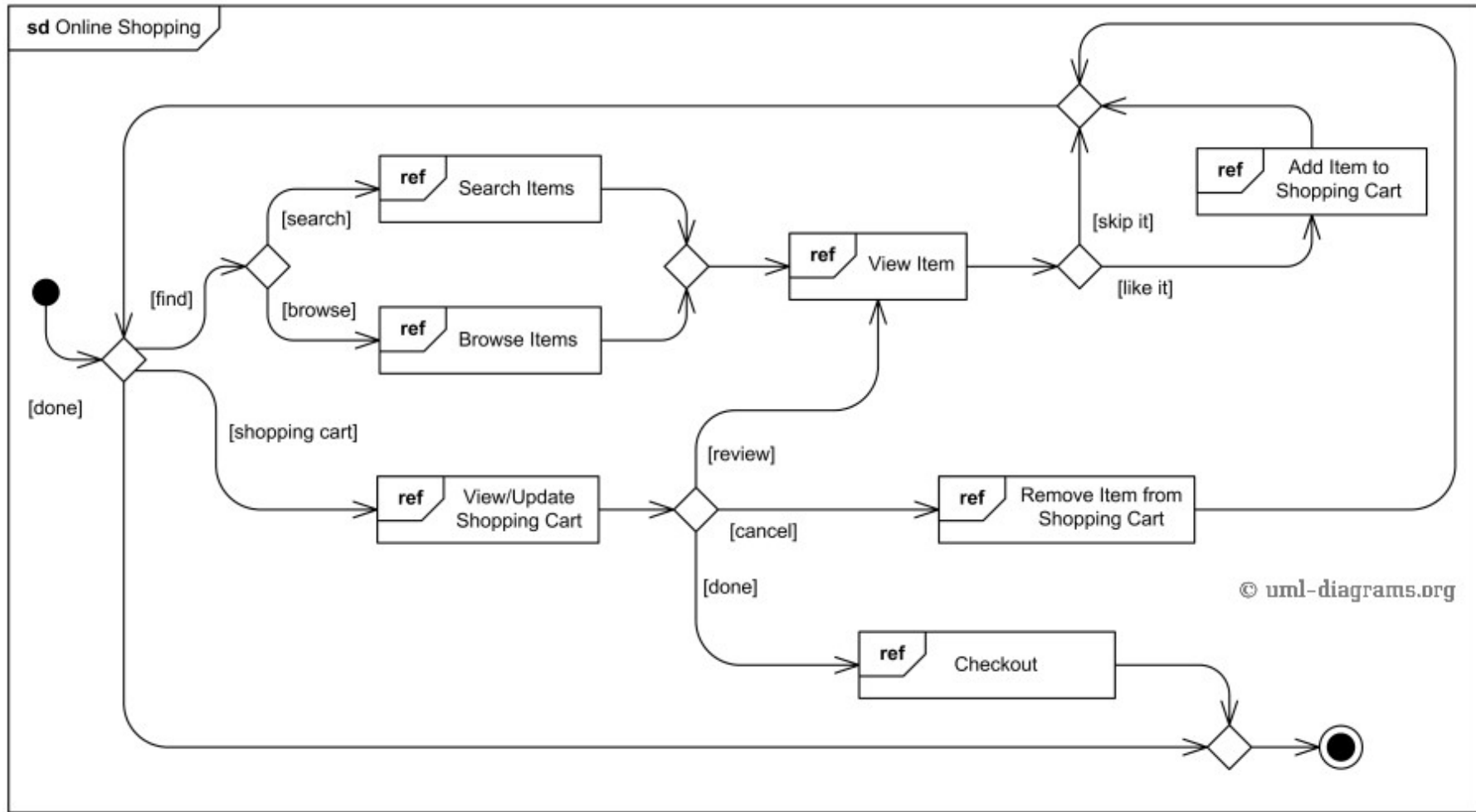


# Soru: Doküman Yönetim Sistemi

## Etkinlik Diyagramı Hazırlayınız



# Soru: Online Alışveriş için Etkinlik Diyagramı Çıkarınız



Soru: Kütüphane Otomasyonu için  
Etkinlik Diyagramı Çıkarınız