

Kullanım Durumu Diyagramları (Use-case Diyagramları)

Analiz aşaması projeler için hayati önem taşır. İyi bir analizden geçmemiş projelerin başarı şansı azdır. Analiz ile birlikte kendimize “Ne?” sorusunu yöneltirken, sistemin fonksiyonel gereksinimlerini yakalamaya çalışırız. Use Case UML içerisinde yer alan ve analiz aşamalarında sıkça kullanılan bir tekniktir. Başarılı projelere göz attığımızda, bu projelerin çok iyi analiz aşamaları geçirdikleri görülür. Analiz ve tasarım genellikle birlikte kullanılan iki terim. Bu iki terimi kısaca şu iki kelime ile açıklamak mümkün. “Ne? ve Nasıl?”. Ne sorusu analizi temsil ederken, Nasıl sorusu ise tasarım etkinliğinden sorumludur. Ne sorusunun cevabı nasıl sorusunu tetiklediğinden tasarım analizin bir sonucudur denilebilir.

Bir sistemi modellemek, gereksinimleri ortaya koymak ve bunun sonucunda istenilen ürünü tam olarak gerçekleştirmek için en önemli nokta sistemde gerçekleştirilecek olan dinamik davranışları yakalamaktır. Sistemin tasarlanmasında ihtiyaçları ayrıntılı olarak açıklığa kavuşturmak için bahsedilen bu dinamik davranış, sistemin çalıştırıldığındaki davranışını ifade eder.

Dolayısıyla sadece statik davranış bir sistemi modellemek için yeterli değildir, zira dinamik davranış statik davranıştan daha önemlidir. UML'de dinamik davranışları modellemek için mevcut beş diyagram vardır ve kullanım durumu (Use-Case-- UC) diyagramı bunlardan biridir.

Use case diyagramları aktörler, kullanım örnekleri ve bunların ilişkilerinden oluşur. Diyagram, bir uygulamanın sistemini / alt sistemini modellemek için kullanılır. Tek bir kullanım durumu diyagramı, bir sistemin belirli bir işlevselliğini yakalar.

Durum diyagramının amacı, bir sistemin dinamik yönünü yakalamaktır. Fakat bu tanım, amacı tanımlamak için çok kapsamlıdır. Çünkü kullanılan diğer dört diyagram (etkinlik, sıralama, işbirliği ve Durum Çizelgesi) da aynı amacı taşımaktadır. Bu yüzden, use-case diyagramlarını diğer dört diyagramdan ayıran belirli bir amaca yoğunlaşmak gerekmektedir.

Kullanım durumu diyagramları (use-case diyagramları), iç ve dış etkileri de içeren bir sistemin gereksinimlerini toplamak için kullanılır. Bu gereksinimler çoğunlukla tasarım gereklilikleridir. Dolayısıyla bir sistem, fonksiyonlarını toplamak için analiz edildiğinde, kullanım örnekleri hazırlanır ve aktörler tanımlanır.

Kullanım Durumu Diyagramlarının Amacı (Use-Case diyagramları)

- Bir sistemin gereksinimlerini toplamak için kullanılır. Toplanan gereksinimler üzerinde anlaşmayı kolaylaştırır.
- Bir sistemin dış görünümünü elde etmek için kullanılır.
- Sistemi etkileyen dış ve iç faktörleri tanımlamak için kullanılır.
- Gereksinimler arasındaki etkileşimi aktörler bazında göstermek için kullanılır.
- Use case diyagramları yazılım ekip üyeleri arasındaki iletişimi geliştirir.
- İş süreçlerinin anlaşılmasını kolaylaştırır.

Dolayısıyla bir sistemin dinamiklerini anlamak için farklı diyagram türlerini kullanmamız gerekir. Kullanım durumu diyagramı bunlardan biridir ve özel amacı sistem gereksinimlerini ve aktörleri toplamaktır. Kullanım durumu diyagramları, bir sistemin olaylarını ve akışlarını belirtir. Kullanım öyküsü diyagramı, yalnızca kara kutunun giriş, çıkış ve işlevinin bilinmekte olduğu bir kara kutu olarak düşünülebilir.

Kullanım diyagramlarının kullanıldığı yerler:

- Gereksinim analizi ve üst düzey tasarım.
- Bir sistem bağlamını modelleyin.
- Tersine mühendislik.
- İleri mühendislik.

Use-Case (UC) Ogeleri:

UC temelde çok basit bir tekniktir, esas zorluk gereksinimleri hedef odaklı en etkin biçimde ifade edebilmektir. Sistem, yani üzerinde çalıştığımız uygulama ile uygulamadan etkilenen dış kullanıcılar (aktör) arasında geçen iş hedeflerine uygun bir dizi etkileşime Use Case denir. Belki bu aşamada bu tanım biraz karışık gelmiş olabilir, şimdi isterseniz tanımda geçen terimleri biraz açalım. Tanımlara geçmeden temel bir UC diyagramını gösterelim.

Senaryo: Sistem ve aktör (kullanıcı) arasında geçen etkileşimi anlatan bir dizi adım'a (operasyon) senaryo adı verilir. Basit bir senaryo örneği verecek olursak, online bir mağazadan ürün satın alma işlemini düşünebiliriz. Ürün satın alma senaryosu için su adımları sırasıyla örnek verebiliriz;

1. Müşteri ürün kataloğunu gezer,
2. Müşteri ürün detaylarını inceler,
3. Müşteri ürünü sepete ekler,
4. Müşteri sipariş işlemine geçer,
5. Teslimat ve fatura bilgileri girilir,
6. Ödeme bilgileri girilir,
7. Sistem tarafından sipariş onayı alınır,
8. Teslimat oluşturulur ve kargo şirketine iletilir,
9. Müşteri ürünü teslimat adresinden teslim alır

gibi bir dizi işlem başarılı bir ürün satın alma senaryosunu oluşturur.

Aktör : Aktör belirlenen sistem üzerinde bir rol oynayan ve etkileşime gecen her türlü ögedir. Burada aktör sistemi kullanan bir site ziyaretçisi olabileceği gibi sistemden etkilenen dış sistemler yani kargo şirketi de bir aktör olarak yerini alır. Yukarıdaki senaryoda müşteri Use Case'i tetikleyen, kargo şirketi ise tetiklenen yani UC'den etkilenen aktörlerdir. Aktör kelimesi bazen yanlış anlaşılmalara neden olabilir, bu yüzden aktör kelimesini açmak için; sistem üzerinde belli bir role sahip kullanıcı, açıklamasını getirmek yerinde olur.

Sistem: Use Case'lerin yer aldığı sınırları ve kapsamı belli, aktörlerin üzerinde çalıştığı yer sistemdir. Sistem projelerinizin kapsamını belirtmesi açısından önemlidir.

Use Case: Ortak bir kullanıcı hedefi etrafında oluşturulmuş bir takım senaryoların bütünleşmesinden meydana gelen yapıdır. Use caseler temelde senaryolardan oluşur, fakat her senaryo yukarıda verilen örnek gibi başarılı ve tek düze yürümez. Senaryoların bazı istisnai durumları ve alt senaryoları içermesi gerekmektedir. Bu durumda Use Case verilen senaryoların

ötesinde bazı artı özellikler de taşır. Örnek olarak müşterinin ödeme işlemini havale ya da kredi kartı olarak yapabilmesi, kredi kartı onayında hata alınması ve teslimat seçenekleri gibi istisnai durum ve alternatif yaklaşımların başarılı senaryo üzerinde toparlanması işlemi UClerin sorumluluğu altındadır.

İlişki: UCler arasında kurulan ilişkiler, bu ilişkiler ileriki bölümlerde daha detaylı anlatılacaktır. İlişkilerin yardımı ile UCler üzerinde genelleme, kapsama (inclusion) ve genişleme (extension) gibi gösterimler yapılabilir.

Bir örnek: ATM uygulaması



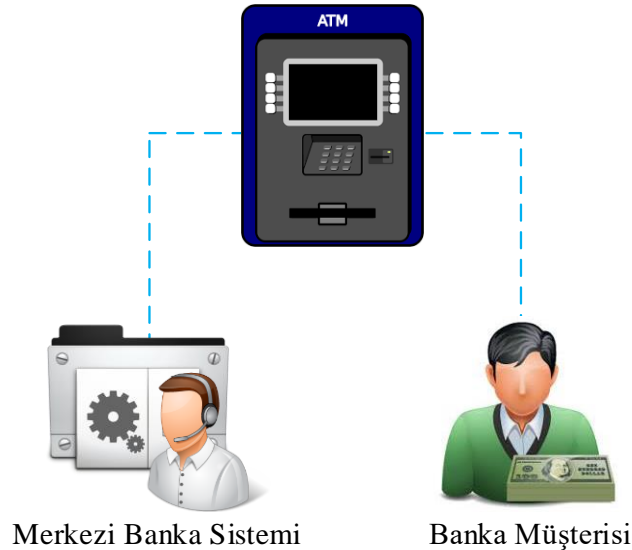
Bir bankanın ATM cihazı için yazılım geliştirilecektir. ATM, banka kartı olan müşterilerin hesaplarından para çekmelerine, hesaplarına para yatırmalarına ve hesapları arasında para transferi yapmalarına olanak sağlayacaktır. ATM, banka müşterisi ve hesapları ile ilgili bilgileri, gerektiğinde merkezi banka sisteminden alacaktır.

ATM uygulama yazılımının kullanıcıları:

Aktörler

- Banka müşterisi
- Merkezi Banka Sistemi

ATM Yazılımı Kullanıcıları



Belirlenen aktörler ATM'den ne istiyorlar?

- **Aktör: Banka müşterisi**

- ✓ Para çekme
- ✓ Para yatırma
- ✓ Para transferi

- **Aktör: Merkezi Banka Sistemi**

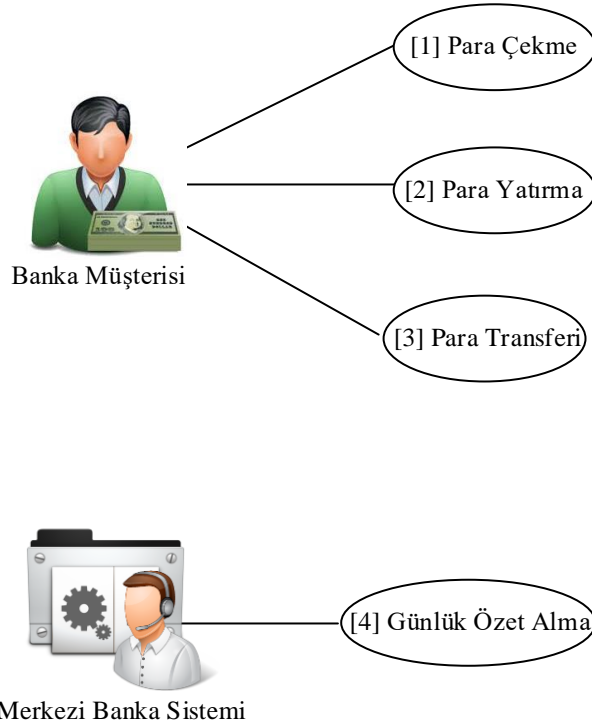
- ✓ Günlük özet alma

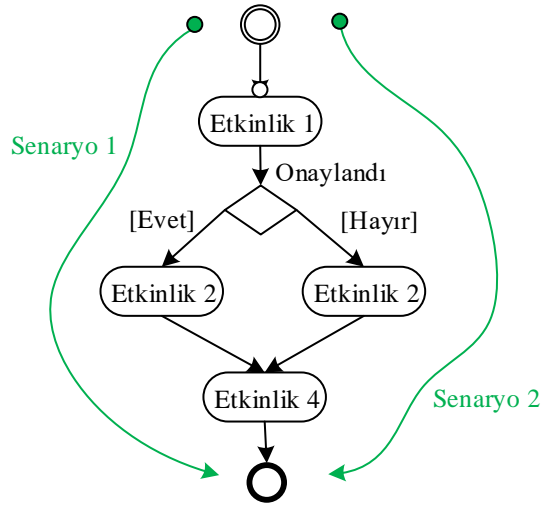
- **Aktör: Banka müşterisi**

- Bankada hesabı ve banka kartı olan, ATM'den işlem yapma hakkı olan kişidir.

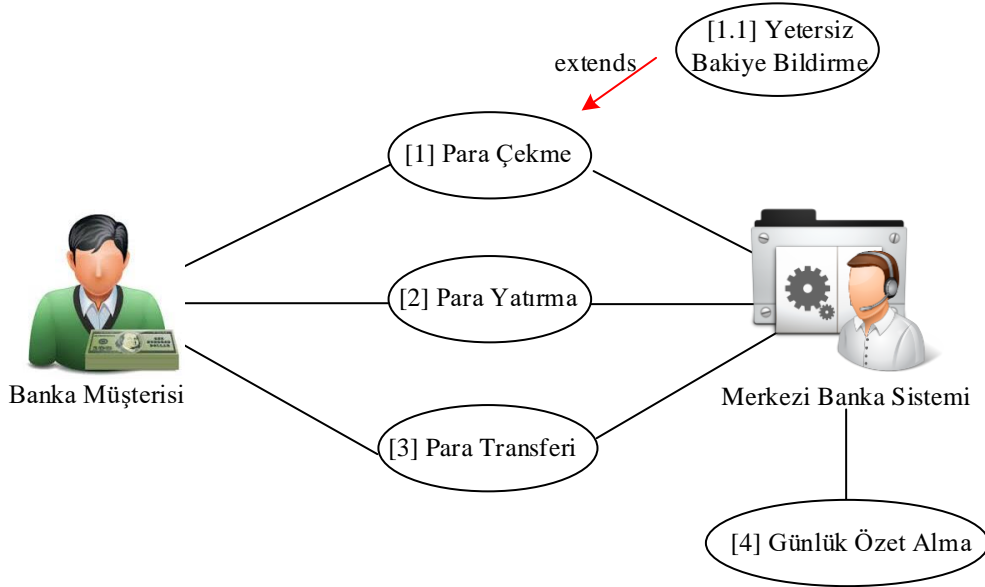
Kullanım Durumu: Para çekme

- Banka müşterisinin nasıl para çekeceğini tanımlar. Para çekme işlemi sırasında banka müşterisinin istediği tutarı belirtmesi ve hesabında bu tutarın mevcut olması gerekir.





Activity Diagram



Use-case Diagram

Kaynaklar:

www.kirbas.com/index.php?id=183