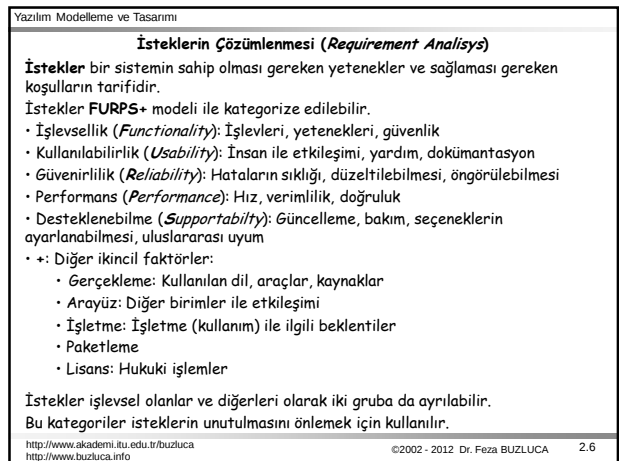
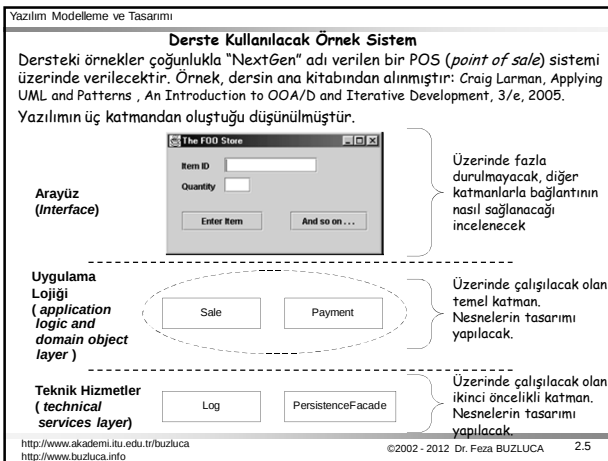
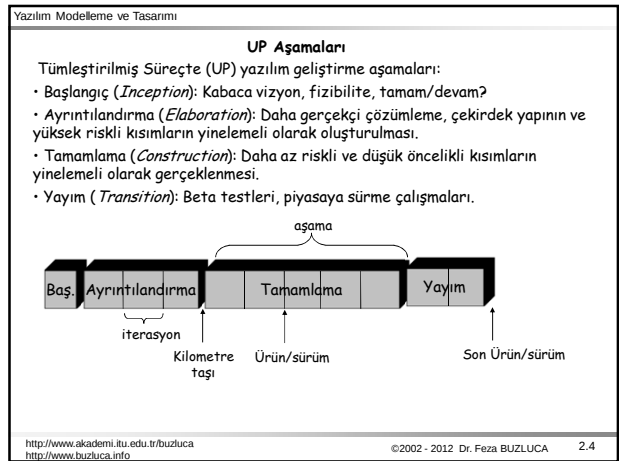
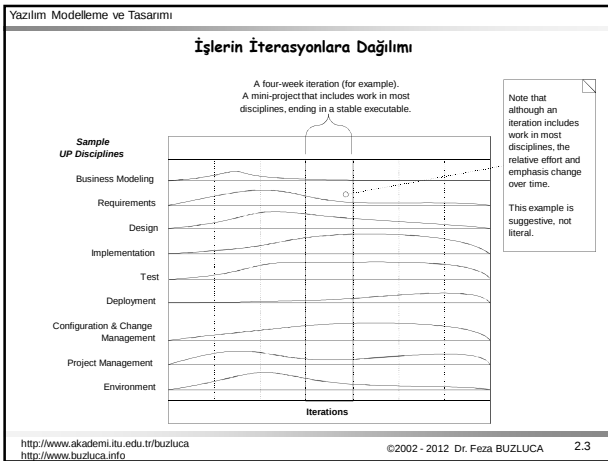
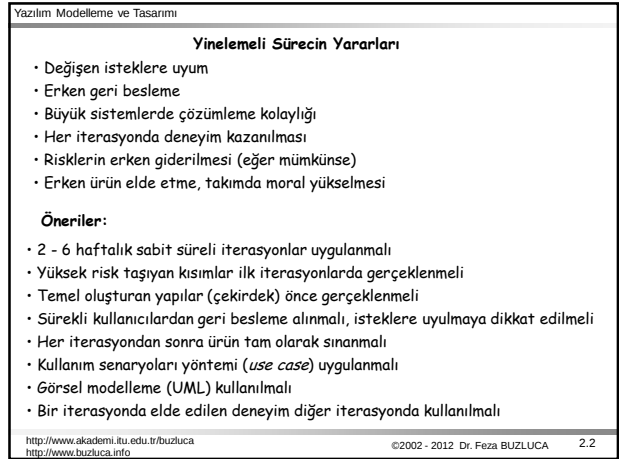
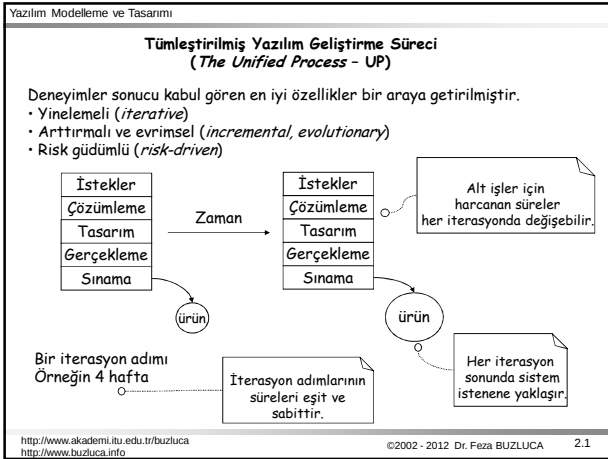




Yazılım Modelleme ve Tasarımı

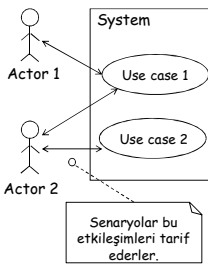
Kullanım Senaryoları (Use-Case Model)

İsteklerin anlaşılmasını ve ifade edilmesini sağlayan bir yöntemdir. Özellikle işlevsel isteklerin ifade edilmesinde kullanılır. Senaryolar sadece bir doküman değildir. Senaryolar olmadan sistemin ne yapması gerektiği ne olarak belirlenemez.

Ivar Jacobson, İsveçli müh. (Ericsson), sonra Rational'de, şimdi kendi firmasında.
http://www.ivarjacobson.com

Tanım:

- "A use case specifies a sequence of actions, including variants, that a system performs and that yields an observable result of value to a particular actor." (Three amigos: Jacobson, Booch, Rumbaugh 1999)
- "A use case is a collection of possible sequences of interactions between the system under discussion and its external actors, related to a particular goal." (Cockburn 2000)



http://www.akademi.itu.edu.tr/buzluca
http://www.buzluca.info

©2002 - 2012 Dr. Feza BUZLUCA 2.7

Yazılım Modelleme ve Tasarımı

Senaryo : Anlamlı bir sonuca (amaca) ulaşmak için aktör ile sistem arasında gerçekleşen olayların belli bir zinciridir. Bir sistemin çalışması sırasında birden fazla senaryo gerçekleştirilebilir. Olası tüm senaryolar kullanım senaryolarını (*use case*) oluştururlar. Örnek:

Bir otomatik para çekme makinesinde (ATM) müşteri ile sistem arasında gerçekleştirilecek olan olayların oluşturduğu senaryolar şunlar olabilir.

1. Müşteri kartını makineye takar.
2. Sistem şifreyi sorar.
3. Müşteri şifreyi girer.
4. Sistem şifreyi onaylar.
5. Müşteri para çekme işlemini seçer.
6. Müşteri çekeceği para miktarını seçer.
7. Sistem parayı, makbuzu ve kartı verir.

Yukarıdaki akış bu sistemdeki olası senaryolardan sadece biridir. Aynı sistemdeki başka bir senaryo da müşterinin bakiyesinin yeterli olmaması durumuyla ilgilidir.

http://www.akademi.itu.edu.tr/buzluca
http://www.buzluca.info

©2002 - 2012 Dr. Feza BUZLUCA 2.8

Yazılım Modelleme ve Tasarımı

Aktör: Sistemin kullanıcılarını tanımlamak için kullanılan mekanizmadır.

- Aktör tasarlanmakta olan sistemin kullanıcısı ya da o sistemden etkilenen diğer birimlerdir; insan, başka bir sistem, bir cihaz olabilir.
- Aktörler tasarlanacak olan sistemin dışında kalan birimlerdir.
- Aktör sistemden hizmet istğinde bulunabilir, sisteme hizmet verebilir. Farklı gruplara ayrılırlar:
- Birincil Aktör (*Primary Actor*): Sistemden asıl faydayı sağlayan, işlemleri başlatan kullanıcı.
- Destek Aktörü: Sisteme bilgi (destek) sağlayan aktör. Genellikle bir bilgisayar sistemidir.
- Diğer Aktörler: Bu aktörler sistemi doğrudan kullanmazlar ve sisteme bilgi desteği vermezler ancak o senaryoda gerçekleşen olaylarla ilgilendirilir ve bu olaylardan etkilenirler. Aktörlere ilişkin örnekler derslerin ilerleyen bölümlerinde verilecektir.

Birincil Aktör ve Sistemin Sınırları:

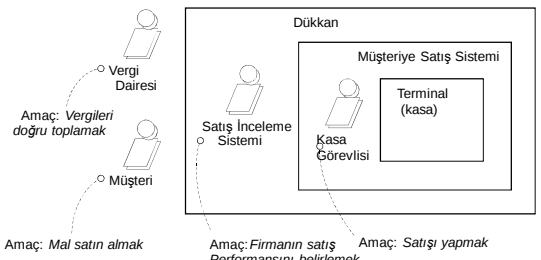
Üzerinde çalıştığımız sistemi hangi düzeyde incelediğimize ve sınırlarını ne şekilde çizdiğimize bağlı olarak birincil aktörler değişiklik gösterir. Kullanım senaryolarını yazarken sistemin sınırlarını doğru olarak belirlemek, nelerin dışında nelerin içeride olacağına doğru karar vermek gerekir.

http://www.akademi.itu.edu.tr/buzluca
http://www.buzluca.info

©2002 - 2012 Dr. Feza BUZLUCA 2.9

Yazılım Modelleme ve Tasarımı

Birincil Aktör ve Sistemin Sınırları



Şekilde görüldüğü gibi o anda tasarlamakta olduğumuz sistem sadece terminal programı ise bu durumda sistemin birincil aktörü (kullanıcısı) kasa görevlisidir. Ancak dükkan sistemini bir bütün olarak inceliyorsak kasa görevlisi bu sistemin içinde bir parçadır ve aktör değildir. Bu durumda birincil aktör müşteridir. Vergi dairesi ise bu sistemden etkilenen diğer aktördür.

http://www.akademi.itu.edu.tr/buzluca
http://www.buzluca.info

©2002 - 2012 Dr. Feza BUZLUCA 2.10

Yazılım Modelleme ve Tasarımı

Kullanım Senaryolarının Yazılması

Kullanım senaryolarının ifade edilmesi:

- İhtiyaçların ve istenen özelliklerin listelenmesi şeklinde **DEĞİL**.
- Sistem **kara kutu** olarak ele alınır. Sistemin iç yapısı görülmez, sistemin dışıyaya (aktörlere) karşı sorumlulukları ifade edilir.
- Aktörler ile sistem arasındaki etkileşim **etken** cümleler ile ifade edilir.
- "Ne yapar?" sorusu cevaplanır, "Nasıl yapar?" **değil**. Sistemin sorumluluklarını nasıl yerine getireceği daha sonra gelinecek olan tasarım aşamasında ele alınacak problemidir. Kullanım senaryolarını yazdığımız şimdiki aşamada ise sadece istekler anlaşılmalı çalışılıyor.
- Sistemin bitmiş hali hayal edilerek bu sistem çalıştığında oluşabilecek senaryolar yazılır.

Kullanım senaryolarında yer alan bölümler:

Her kullanım senaryoları grubunun (*use case*) bir adı ve numarası vardır. İsmden sonra aşağıdaki bölümler gelir.

a) Önsöz (Preface) Bölümü

Aşağıdaki alt bölümlerden oluşur:

- Birincil Aktör (*Primary Actor*): Sistemden asıl faydayı sağlayan, işlemleri başlatan kullanıcı.

http://www.akademi.itu.edu.tr/buzluca
http://www.buzluca.info

©2002 - 2012 Dr. Feza BUZLUCA 2.11

Yazılım Modelleme ve Tasarımı

İlgililer ve Beklentileri (Stake holders and interests): Sistemin çalışmasından etkilenen ve bu sistemden beklentileri olan unsurlar (diğer aktörler).

Birincil aktör, destek aktörü ve diğer aktörlerin belirlenmesi sistemin sınırlarını çizer.

Kullanım senaryoları ilgililerin (aktörlerin) tüm beklentilerini karşılayan tüm olayları ve sadece onları içerir.

Tüm ilgililerin ve beklentilerin ilk başta belirlenmesi önemlidir. Aksi durumda senaryolarda bazı durumlar unutulabilir ve bu eksiklik ancak ileriki aşamalarda anlaşılabilir.

- **Ön koşullar (Preconditions):** Belli bir senaryo grubunu (*use case*) oluşturan olayların başlaması için sağlanması gereken koşullar.

Bu koşullar senaryo içinde test edilmez, doğru oldukları varsayılır.

- **Son koşullar (Postconditions, Success Guarantees):** Senaryolar tamamlandığında sistemin ulaşacağı durumlardır.

Son koşullar ilgililerin beklentilerine (amaçlarına) denk düşer.

http://www.akademi.itu.edu.tr/buzluca
http://www.buzluca.info

©2002 - 2012 Dr. Feza BUZLUCA 2.12

Yazılım Modelleme ve Tasarımı
b) Ana Başarılı Senaryo (Temel Akış) Bölümü (<i>Main Success Scenario or Basic Flow</i>) Sistemin en doğal çalışma şekli adım adım yazılır. Her adım numaralanır. Koşullar ve dallanmalar içermez. Etken cümleler kullanılır; "kim ne yapar" açıktır. Adımlar üç farklı gruba ayrılır: 1. Kullanıcılar ile sistem arasında etkileşim, tetikleme. 2. Onaylama (çoğunlukla sistem tarafından) 3. Sistemde durum değişikliği, bir bilginin kayıt edilmesi. Örnek: 1. Müşteri şifresini girer. 2. Sistem ekrana müşterinin adını çıkarır. 3. Belirsiz ve edilgen cümleler kullanılmaz. Örnek: "Toplam belirlenir." Bu uygun bir senaryo cümlesi değildir. Kim belirleyecek? Sistem mi? Aktörlerden biri mi?
http://www.akademi.itu.edu.tr/buzluca http://www.buzluca.info
©2002 - 2012 Dr. Feza BUZLUCA 2.13

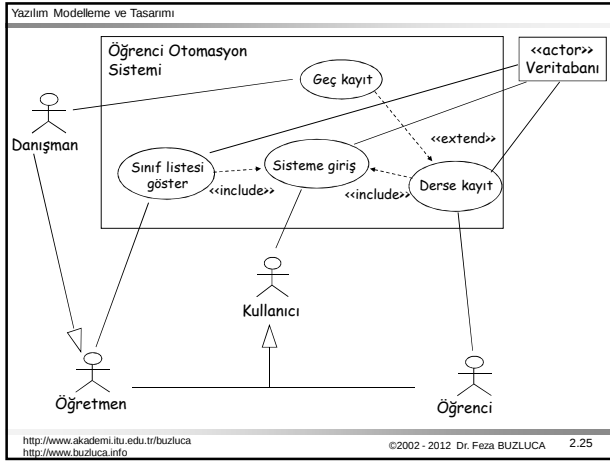
Yazılım Modelleme ve Tasarımı
c) Uzanlılar (Alternatif Akışlar) Bölümü (<i>Extensions or Alternate Flows</i>) Ana senaryonun dışında kalan başarılı/başarısız sonuçlara götüren tüm senaryolar sıralanır. Ana senaryodan (temel akış) dallanmalar şeklinde yazılırlar. Ana senaryoda hangi adımdan buraya gelinecekse o adının numarası kullanılır. Alternatif akışa (dallanma) neden olan koşullar aktörler ya da sistem tarafından fark edilecek şekilde yazılmalı. Alternatif senaryolar ile aktörlerin tüm amaçları sağlanmış olmalı. Örnek: Ana senaryoda "2. Müşteri şifresini girer" satırı varsa, temel akışta şifrenin doğru olduğu durum ele alınır. Şifrenin yanlış girilmesi durumu ise aşağıda gösterildiği gibi uzantılarda incelenir. Uzanlılar: 2a. Müşteri şifresini yanlış girmiştir. 1. Sistem hata mesajı verir ve şifreyi yeniden ister. d) Sıra Dışı Durumlar Bölümü (<i>Exceptions</i>) Sistemde hatalar oluştuğunda yapılacaklar sıralanır. Bazı tasarımcılar bu bölümdeki olayları da uzantılar bölümünde ele alırlar.
http://www.akademi.itu.edu.tr/buzluca http://www.buzluca.info
©2002 - 2012 Dr. Feza BUZLUCA 2.14

Yazılım Modelleme ve Tasarımı
e) Özel İstekler Bölümü (<i>Special Requirements</i>) İşlevler ile ilgili olmayan istekler bu bölümde belirtilir. Bu istekler genellikle hız, güvenilirlik, rahat kullanım gibi kalite kriterlerine yöneliktir. f) Teknolojik Beklentiler Bölümü Kullanıcıların ön gördükleri donanım özellikleri burada sıralanır. Örneğin giriş/çıkış işlemlerinin hangi cihazlar ile yapılması istendiği bu bölüme yazılır. Örnek: Metin (text) tipindeki bir kullanım senaryoları grubuna örnek olarak bir marketteki satış noktası (POS) uygulaması verilmiştir. Bir sistemde bir çok senaryo grubu (<i>use case</i>) bulunabilir. Örneğin market sisteminde de satış işlemleri bir senaryolar grubu, ürün iadesi de başka bir senaryolar grubu olabilir. Bu örnekte satış işlemleri (<i>Process Sale</i>) senaryo grubu gösterilmiştir.
http://www.akademi.itu.edu.tr/buzluca http://www.buzluca.info
©2002 - 2012 Dr. Feza BUZLUCA 2.15

Yazılım Modelleme ve Tasarımı
Senaryo Grubu (Use Case) SG1: Satış İşlemleri: Konu: NextGen POS Market Sistemi Birincil Aktör: Kasa Görevlisi İlgililer (Aktörler) ve Beklentileri (Stakeholders and Interests): • Kasa Görevlisi: Bilgilerin doğru ve hızlı girilmesi, toplamın doğru hesaplanması, para üstünün doğru hesaplanması • Satış Elemanı: Komisyonun doğru hesaplanması ve kayıt edilmesi • Müdür: Yetkili işlemleri (kasa görevlisinin yapamadığı) kolaylıkla yapabilmek • Vergi Dairesi: Vergilerin doğru hesaplanabilmesi ve toplanabilmesi • Kredi Kartı Asıllama Merkezi: Ödeme bilgilerinin doğru formatta gelmesi ve asıllama bilgilerinin kayıt edilmesi Ön Koşullar (Preconditions): Kasa görevlisi sisteme giriş yapmıştır. Son Koşullar (Postconditions): Satış bilgileri kayıt edilmiştir. Vergi doğru olarak hesaplanmıştır. Muhasebe ve envanter kayıtları güncellenmiştir. Komisyon kayıt edilmiştir. Fatura oluşturulmuştur. Kredi kartı onayı kayıt edilmiştir.
http://www.akademi.itu.edu.tr/buzluca http://www.buzluca.info
©2002 - 2012 Dr. Feza BUZLUCA 2.16

Yazılım Modelleme ve Tasarımı
Ana Başarılı Senaryo (Doğal Akış) (Main Success Scenario or Basic Flow) : 1. Müşteri ödeme noktasına almak istediği ürün ve hizmetler ile gelir. 2. Kasa görevlisi yeni bir satış başlatır. 3. Kasa görevlisi ürün kodunu sisteme girer. 4. Sistem satış kalemini (maddesini) kayıt eder ve ürünün tanıtıcı bilgisini, fiyatını ve o anda kadar oluşan toplamı gösterir. Kasa görevlisi 3ncü ve 4ncü maddeleri ürün kalmayınca kadar tekrar eder. 5. Sistem toplamı vergilerle birlikte gösterir. 6. Kasa görevlisi müşteriye toplamı söyler ve ödeme yapmasını ister. 7. Müşteri ödeme yapar ve sistem ödeme bilgisini alır. 8. Sistem tamamlanan satış bilgisini kayıt eder; satış ve ödeme ile ilgili bilgileri muhasebe ve envanter sistemlerine (bunlar dış sistemlerdir) gönderir. 9. Sistem faturayı oluşturur. 10. Müşteri ürün ve hizmetler ile ayrılır. Uzanlılar (Alternatif Akışlar) (Extensions or Alternate Flows): *a. Herhangi bir anda müdür yetkili bir işlem yapmak ister ve şifresini girer: 1. Sistem müdür-yetkisi konumuna geçer. 2. Müdür yetkili bir işlem gerçekleştirir. Örneğin satışı iptal eder, bir ürünün fiyatını indirir vs. 3. Müdür sistemden çıkar. 4. Sistem normal konuma (kasa görevlisi yetkisi) geçer.
http://www.akademi.itu.edu.tr/buzluca http://www.buzluca.info
©2002 - 2012 Dr. Feza BUZLUCA 2.17

Yazılım Modelleme ve Tasarımı
Uzanlılar (Alternatif Akışlar) (Extensions or Alternate Flows) Devamı: *b. Herhangi bir anda sistemde bir hata oluşur: Bu durumlarda bilgilerin kayıt edilmesi ve sistemin kaldığı yerden devam edebilmesi istenir. 1. Kasa görevlisi sistemi yeniden başlatır, sisteme giriş yapar ve sistemin önceki durumdan devam etmesini ister. 2. Sistem önceki durumu oluşturur. 2a. Sistem önceki durumu oluştururken anormallik sezer. 1. Sistem hata uyarısı verir, hatayı kayıt eder ve temiz (başlangıç) duruma geçer. 2. Kasa görevlisi yeni bir satış başlatır. 3a. Geçersiz bir ürün kodu (Sistemde bulunmadı): 1. Sistem hata uyarısı verir, ürünü reddeder. 2. Kasa görevlisi hataya tepki verir: 2a. Ürünün üstünde okunabilir bir kod vardır: 1. Kasa görevlisi kodu sisteme elle (<i>manual</i>) girer. 2. Sistem ürünün tanıtıcı bilgisini ve fiyatını gösterir. 2b. Ürünün üstünde kod yoktur, ama fiyatı yazılıdır: 1. Kasa görevlisi müdürden yetkili bir işlem yapmasını ister. 2. Müdür şifresini girer. 3. Kasa görevlisi fiyatı elle girer.
http://www.akademi.itu.edu.tr/buzluca http://www.buzluca.info
©2002 - 2012 Dr. Feza BUZLUCA 2.18



Yazılım Modelleme ve Tasarımı

Kullanım Senaryolarının Yazılması

Kullanım senaryoları, yazılımın müşterisi ile görüşülerek yazılır. Müşteri bazen yazılımı yapan firmanın kendisi de olabilir. Tasarıma geçmeden önce bütün isteklerin belirlenmesi gerekli (ve olası) değildir. İstatistiklere göre proje süresinde isteklerin %25'i değişebilmektedir. Müşteri ile görüşmelerde sistemin nasıl çalışacağı (iş akışları) açıkça ortaya konmalıdır. Yazılımı doğrudan kullanacak olan kişilerle de görüşmeler yapılmalıdır. Bu görüşmelerde aşağıdaki sorular sorularak senaryoların yazılmasında gerekli olan bilgilere ulaşılabilir.

Aktörlerin belirlenmesi için sorulabilecek sorular:

- Sistemin temel işlevlerini kim kullanacak?
- Günlük işlerini yapmak üzere kim sistemin desteğine gerek duyar?
- Sistemin bakımını ve işletmesini kim yapacak?
- Sistem hangi cihazları kullanacak?
- Hangi diğer sistemler ile etkileşimde bulunacak?
- Bu sistemin sonuçları kimi ilgilendirir?

http://www.akademi.itu.edu.tr/buzluca
http://www.buzluca.info

©2002 - 2012 Dr. Feza BUZLUCA 2.26

Yazılım Modelleme ve Tasarımı

Aktörlerden yararlanarak sistem davranışının belirlenmesi için sorulabilecek sorular:

- Aktörlerin temel işleri nedir?
- Aktör sistem bilgilerine erişmeli mi? Erişim tipi?
- Aktör dış durumlardaki değişiklikleri bildirecek mi?
- Durum değişiklikleri (hangileri?) aktöre bildirilecek mi?
- Aktör hangi işlevlere gerek duyar?

Diğer Sorular: Bazı davranışlar aktörlerden yola çıkarak belirlenemeyebilir. Bu durumda aşağıdaki soruları da sormakta yarar vardır:

- Sistemin gerek duyduğu girişler ve çıkışlar nelerdir?
- Sistem hangi dış olaylardan etkilenir?
- Şu andaki sistemin (eğer firmada aynı iş için kullanılan eski bir sistem varsa) eksikleri ve problemleri nelerdir?
- Periyodik olarak gerçekleştirilen işler var mı?

Senaryolar ve UP Yaklaşımı:

- UP'de, özellikle çevik yaklaşımda senaryoların tamamının projenin başında yazılması hedeflenmez (yinelemeli, evrimsel yaklaşım).
- Bir senaryonun gerçekleşmesi birden daha fazla yineleme adımı sürebilir. Daha küçük senaryolar tek bir adımda da gerçekleştirilebilir.

http://www.akademi.itu.edu.tr/buzluca
http://www.buzluca.info

©2002 - 2012 Dr. Feza BUZLUCA 2.27

Yazılım Modelleme ve Tasarımı

Kullanım Senaryoları Yönteminin Yararları

İsteklerin doğru ve eksiksiz olarak belirlenebilmesi yazılımın kalitesi açısından önemlidir. Kullanım senaryoları yöntemi bu noktada aşağıdaki yararları sağlar:

- Kolay anlaşılır. Müşteri (yazılımın kullanıcısı) ile yazılımı hazırlayacak grup arasında iletişimi kolaylaştırır.
- Sistemde gerekli olan unsurların belirlenmesini kolaylaştırır, unutulmalarını önler.
- Sınama (verification) olanağı sağlar. Gerçeklenen sistem senaryolar ile sınanabilir. Tamamlanmış olan yazılıma senaryolar uygulandığında eğer sistem her adımda senaryoda yazılmış olanları yerine getiriyorsa yazılımın sağlanması yapılmış olur.
- Kullanım senaryoları nesneye dayalı değildir. Bu yöntem gerekirse başka programlama yöntemleri için de kullanılabilir.
- Diğer taraftan kullanım senaryoları nesneye dayalı modelleme için uygun bir başlangıç noktası oluştururlar.

Bundan sonraki bölümlerde çözümleme ve tasarlama konuları anlatılırken kullanım senaryolarının bu yararı da gösterilecektir.

<http://alistair.cockburn.us/usecases/usecases.html>
<http://www.pols.co.uk/use-case-zone/>

http://www.akademi.itu.edu.tr/buzluca
http://www.buzluca.info

©2002 - 2012 Dr. Feza BUZLUCA 2.28