

Örnek-1

Kütüphanemizin abone olduğu süreli yayınların yönetimi ile ilgili bir yazılıma ihtiyaç vardır. Aşağıda verilen gereksinimler belgesi ışığında (a) kütüphaneye yeni gelen süreli yayınlar ile ilişkili kullanım senaryosunu yazarak (b) alan modelini oluşturunuz. Ayrıca (c) UML kullanım şeması da çiziniz

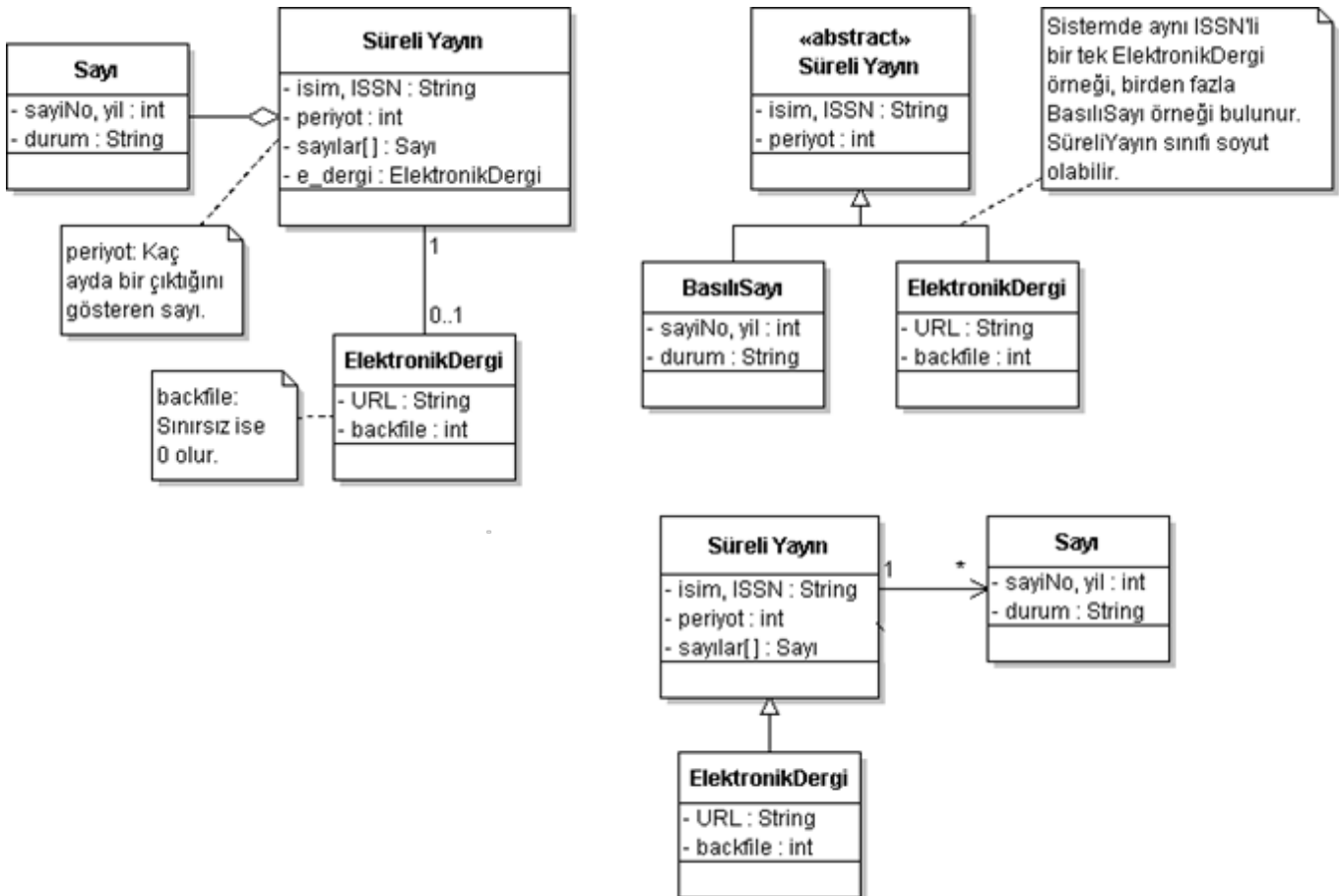
Süreli Yayın Yönetim Yazılımı

Kütüphanemiz için bir süreli yayın yönetimi yazılımına ihtiyacımız vardır. Süreli yayın; yeni sayılarının aylık – üç aylık – 4 aylık gibi periyotlarda geldiği ve bir yayın ismine sahip dergilerdir. Her süreli yayın bir isme ve kendisini diğerlerinden ayırt eden 8 karakterlik ISSN numarasına sahiptir. ISSN numarasının ilk 7 karakteri 0-9 arası sayı, son karakteri 0-9 arası sayı veya X karakteridir. Örneğin “Nature” adlı derginin ISSN’i 09534563’tür. Daha doğru bir terminoloji ile Nature bir aylık süreli yayındır. Kütüphanemiz de bu süreli yayına 2014 Mart sayısından itibaren abonedir. Ancak arada kütüphanemize ulaşmamış sayılar olabilir.

Abone olunan süreli yayınlara bazen sadece basılı, bazen sadece elektronik, bazen de hem elektronik hem de basılı olarak abone olunmaktadır. Elektronik yayınlar için, derginin bütün sayılarına erişim için kullanılan tek bir URL ve bu derginin geriye dönük kaç yıllık sayılarına erişimimizin olduğu (backfile) bilgisi saklanmalıdır. Bazı elektronik yayınlarda ise backfile sınırı bulunmamaktadır. Basılı yayınlar için, derginin kütüphanemizde bulunan her sayısı için yıl ve sayı bilgileri saklanır. Kütüphaneci yeni gelen basılı dergi sayısını sisteme gireceği zaman, derginin mevcut sayılarının her biri bir kutucuk içerisinde yıl, sayı ve durum bilgilerine göre gösterilir. Kütüphaneci kutulardan birine tıklayarak derginin o sayısının durumunu değiştirebilir. Olası durumlar şunlardır: Geldi ve bekleniyor. O anki tarihe ve derginin periyoduna göre, sistem gelecek sayıların durumunu ‘bekleniyor’ olarak belirler. Kütüphaneci ise bu bilgiyi ‘geldi’ olarak değiştirebilir.

Kütüphanemize gelen bir süreli yayın sayısı olduğunda, eğer sistemde yayının kaydı yoksa önce kütüphaneci bu kaydı oluşturur. Bu amaçla kütüphaneci derginin ISSN ve periyot bilgilerini girer. Adından aboneliğin basılı ve/veya elektronik olmasına göre gerekli bilgiler sisteme girilir. Eğer yayının kaydı zaten mevcutsa, kütüphaneci yukarıda belirtilen şekilde sayıyı sisteme ‘geldi’ olarak işler.

(b) Olası Alan modelleri (Metotsuz UML sınıf şemaları):



(a) Olası kullanım senaryosu

KULLANIM SENARYOSU: Yeni Sayı Gelişi (check-in)

Birincil Aktör: Kütüphaneci.

Diğer İlgililer: Yok.

İlgi Alanları: Kütüphaneci: Yeni ve eski her dergi sayısının bilgilerini doğru olarak işleyebilmeyi ister.

Ön Koşullar: Kütüphaneci sisteme girmiştir.

Son Koşullar: Süreli yayın ve sayı kayıtları sisteme girilmiş ve/veya güncellenmiştir.

Ana Senaryo:

1. Kütüphaneci kütüphaneye yeni gelen derginin ISSN'ini girer.
2. Derginin basılı aboneliği varsa; ad, ISSN ve periyot bilgileri gösterilir.
3. Derginin basılı aboneliği varsa, bu yıla ait sayıları kutucuklar şeklinde gösterilir. Mevcut sayıların durumu sistemden elde edilir, gelecek sayıların durumu 'bekleniyor' olarak gösterilir.
4. Derginin elektronik aboneliği varsa, URL ve backfile bilgileri de gösterilir.
5. Kütüphaneci ekranda sunulan tüm bilgileri gereken şekilde günceller ve onay düğmesine basar.

Alternatif Senaryolar:

1a. Yeni abonelik.

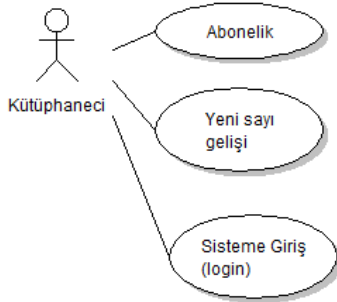
1. Sistem derginin ISSN'ini bulamamıştır, kütüphaneciye yeni abonelik oluşturmayı önerir.
2. Öneri onaylanırsa sistem kütüphaneciden derginin adını, ISSN'ini ve periyodunu alır.
3. Derginin elektronik aboneliği varsa URL ve backfile bilgileri kütüphaneciden alınır.
4. Derginin basılı aboneliği varsa yıl ve sayı bilgileri kütüphaneciden alınır.
5. Sistem girilen sayının durumunu 'Geldi' olarak atar.

3a. Kütüphaneci önceki yıla ait bir sayının işlemini yapmak ister.

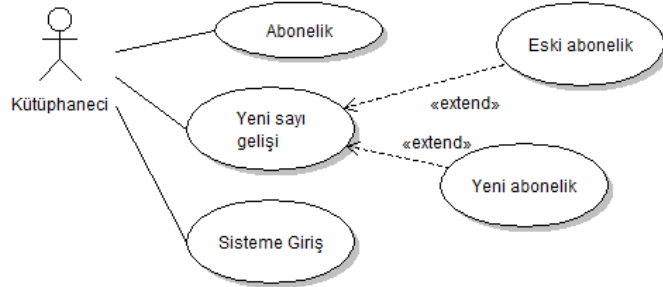
1. Kütüphaneci görmek istediği yılı belirtir.
2. Sistem girilen yılın kutucuklarını çizerek sayıların durumlarını veritabanından alır.

(c) Olası UML kullanım şemaları

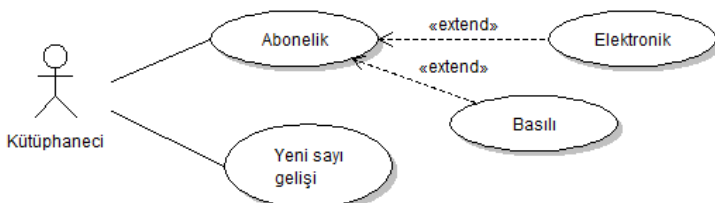
Uygun şemalar



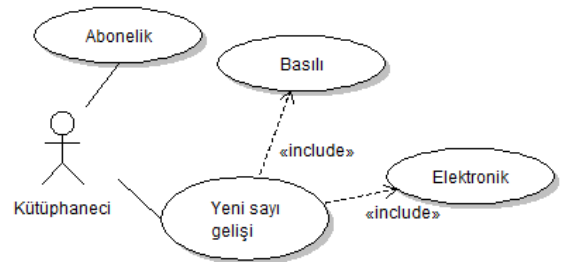
Şema uygun ancak doğru bir include veya extend ilişkisi göstermeniz yararlıdır.



Uygun şema



Kütüphanecilik bilgisine göre uygun şema: Bir dergiye sadece basılı, sadece elektronik veya basılı+elektronik olarak 3 şekilde abone olunabilir.



Bu şema metne bakınca doğru gibi görünüyor ama kütüphanecilik bilgisine göre yanlış. "basılı aboneliği varSA, elektronik aboneliği varSA" ifadelerine dikkat!

Ayrıca: Gereksinimler belgesine son bir paragraf olarak şunu eklersek ne olur?

“Kütüphanecinin içerisine tıklayarak değiştirebileceği, okuyucuların ise değiştiremeden sadece içeriğini görebilme yetkisine sahip olduğu yukarıda anlatılan ekrana derginin check-in kartı denilir.”



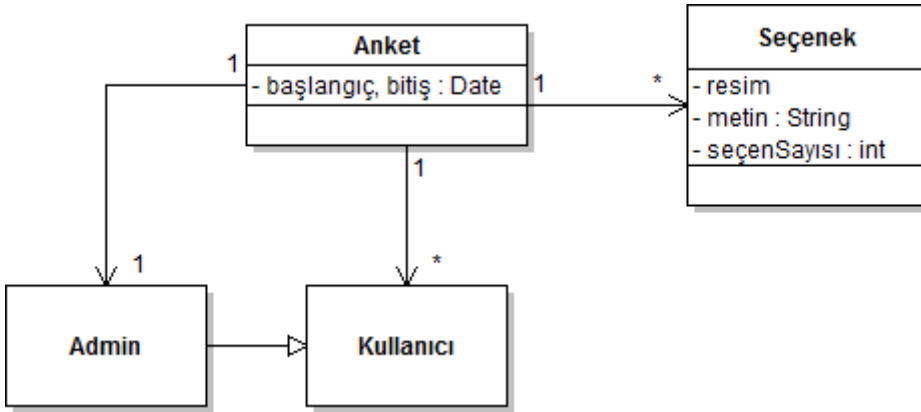
Örnek-2

Aşağıda verilen gereksinimler belgesi ışığında, bir anket oluşturma sürecini anlatan kullanım senaryosunu yazarak alan modelini UML sınıf şeması çizerek oluşturunuz.

Online Anket Yönetimi Bilgi Sistemi

İnternet üzerinde anketler yapmamıza yarayacak ve bize bu anketler ile ilgili raporlar verebilecek bir bilgi sistemine sahip olmak istiyoruz. Her anketin başlangıç ve bitiş süreleri olmalıdır. Bir anketin seçenekleri sadece metinden oluşmamalı, birer resim de içerebilmelidir. Anketlere ancak kayıtlı kullanıcılar oy verebilmelidir. Bir anket ancak o anketin yöneticisi olarak atanmış bir kullanıcı tarafından oluşturulabilmelidir.

Örnek sınıf şeması:



Örnek kullanım senaryosu:

Kullanım Senaryosu: Anket Oluşturma

Birincil Aktör: Anket yöneticisi

İlgililer ve İlgili Alanları:

- Anket yöneticisi (Admin): Atandığı anketi başkasının değiştirmesini istemez
- Kayıtlı kullanıcılar (Kullanıcı): Çeşitli anketlerde oy kullanmak ve bütün anketlerin sonuçlarını görebilmek ister.

Ön koşullar:

- Anket yöneticisi sisteme giriş yapmıştır

Ana Senaryo

- Anket yöneticisi yeni bir anket oluşturur.
- Anket yöneticisi anketin sorularını oluşturur ve anket ile ilişkilendirir.
- Anket yöneticisi anketi yayınlar.

Alternatif akışlar:

2. Anket sorusu resim içeriyordur.

- a. Anket yöneticisi bu soru için sisteme bir resim dosyası yükler

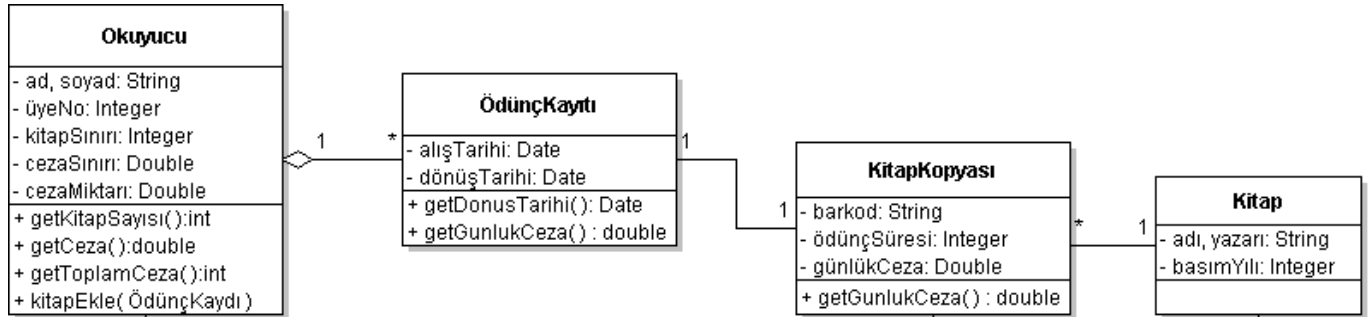
Örnek-3

Bir kütüphane otomasyon yazılımının ödünç verme sistemi ile ilgili **alan şeması** ve okuyucuya kitap ödünç verme işleminin ayrıntıları aşağıda verilmiştir. Bu bilgilere göre **tasarım modelini** oluşturarak ödünç alma işleminin **etkileşim şemasını** ve **okuyucunun kitap alıp alamaması üzerine odaklanan durum şemasını** çiziniz.

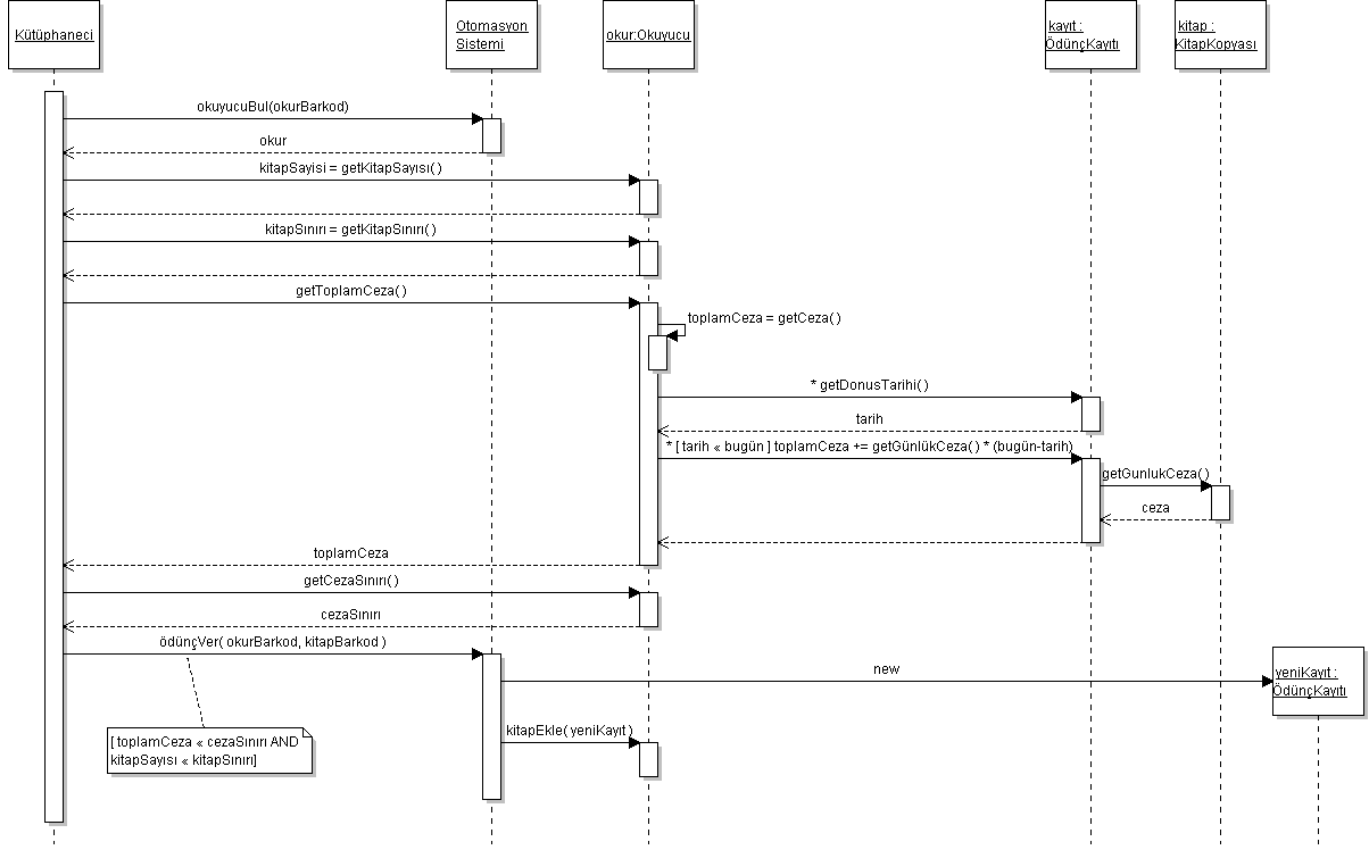
Kitap ödünç verilirken izlenen yol:

- Okuyucunun üzerinde kaç kitap olduğuna bakılır. Eğer okuyucu kitap alma sınırını doldurmuşsa yeni kitap verilmez.
- Okuyucunun üzerindeki TOPLAM para cezasının, okuyucu için önceden tanımlanmış ceza sınırından (Okuyucu.cezaSınırı) yüksek olup olmadığına bakılır. (Toplam ceza = Okuyucu.cezaMiktarı + bu sırada okuyucuda bulunan iade tarihi geçmiş kitaplardan kaynaklanan cezaların miktarı).
- Ceza sınırı aşılmışsa okuyucuya yeni kitap verilmez. Aksi halde bir ödünç kaydı oluşturularak okuyucu kaydı ile ilişkilendirilir. Ödünç kaydının alış tarihi bugünün tarihine, dönüş tarihi ise istenen kitabın kaç günlüğüne ödünç verileceğine bakılarak belirlenir.

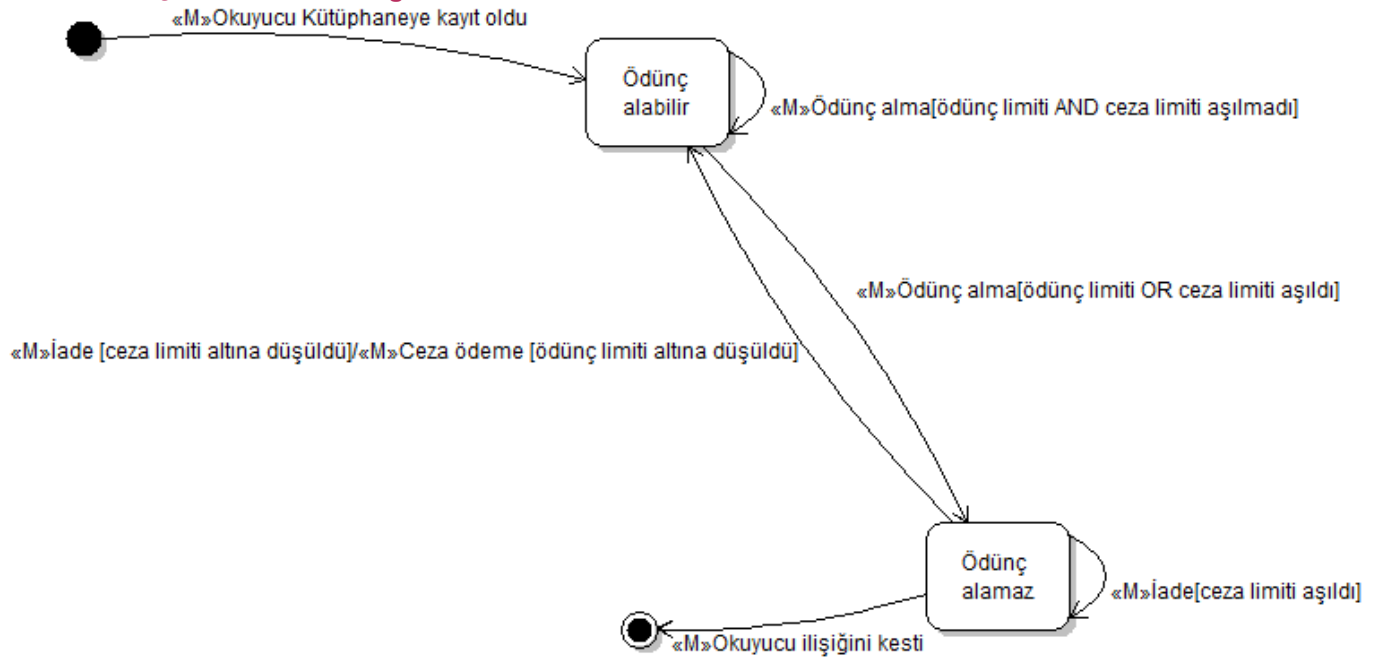
Alan Şeması



Etkileşim şeması olarak -> Sequence Diagram



Durum Şeması - State Diagram



Örnek- 4 : Ders sunumlarındaki örnek.

NextGenPos satış işlemi kullanım senaryosunu hatırlayalım ve bir etkileşim şeması ile bir durum de şeması çizelim. Durum şeması kasa ile ilgili

Kullanım Senaryosu: Satış İşlemi

Birincil Aktör: Kasiyer

İlgililer ve İlgili Alanları:

- Kasiyer: Doğru ve hızlı giriş ister, kasa açığı maaşından kesildiğinden ödeme hataları istemez
- Satıcı: Satış komisyonlarının güncellenmesini ister
- Müşteri: En az çaba ile hızlı hizmet ister. Ürün iadesinde kullanmak üzere fiş ister.

Ön koşullar:

- Kasiyerin kimliği doğrulanır.

Son koşullar:

- Ödeme tahsil edilir. Satış kaydedilir. Fiş yazılır.

Ana Senaryo

1. Müşteri kasaya alacağı ürünlerle gelir.
2. Kasiyer yeni bir satış işlemi başlatır.
3. Kasiyer ürünün barkodunu girer.
4. Sistem bir satış kanalı maddesi oluşturur. Bu maddede ürün tanımı, fiyatı ve toplam bedel (aynı maldan birden fazla alınmış olabilir) yer alır.
5. Kasiyer 3. ve 4. adımları müşterinin alacağı tüm ürünler için tekrarlar.
6. Sistem toplam bedeli vergi iadesi ile birlikte hesaplar.
7. Kasiyer müşteriye toplamı bildirir ve ödeme ister.
8. Müşteri ödemeyi yapar ve sistem ödemeyi tahsil eder.
9. Sistem tamamlanan işlemin kaydını tutmayı tamamlar ve harici envanter ile mali sistemlere gerekli bilgileri gönderir.
10. Sistem makbuz verir.
11. Müşteri ürünlerle birlikte ayrılır

Alternatif akışlar:

3a. Geçersiz barkod

1. Sistem uyarı mesajı verir ve kayıt girişini reddeder.

3-7a. Müşteri bir kalem malı alışverişten çıkartmak ister.

1. Kasiyer satıştan çıkarmak üzere ürünün barkodunu okutur.
2. Sistem güncel toplamı bildirir.

