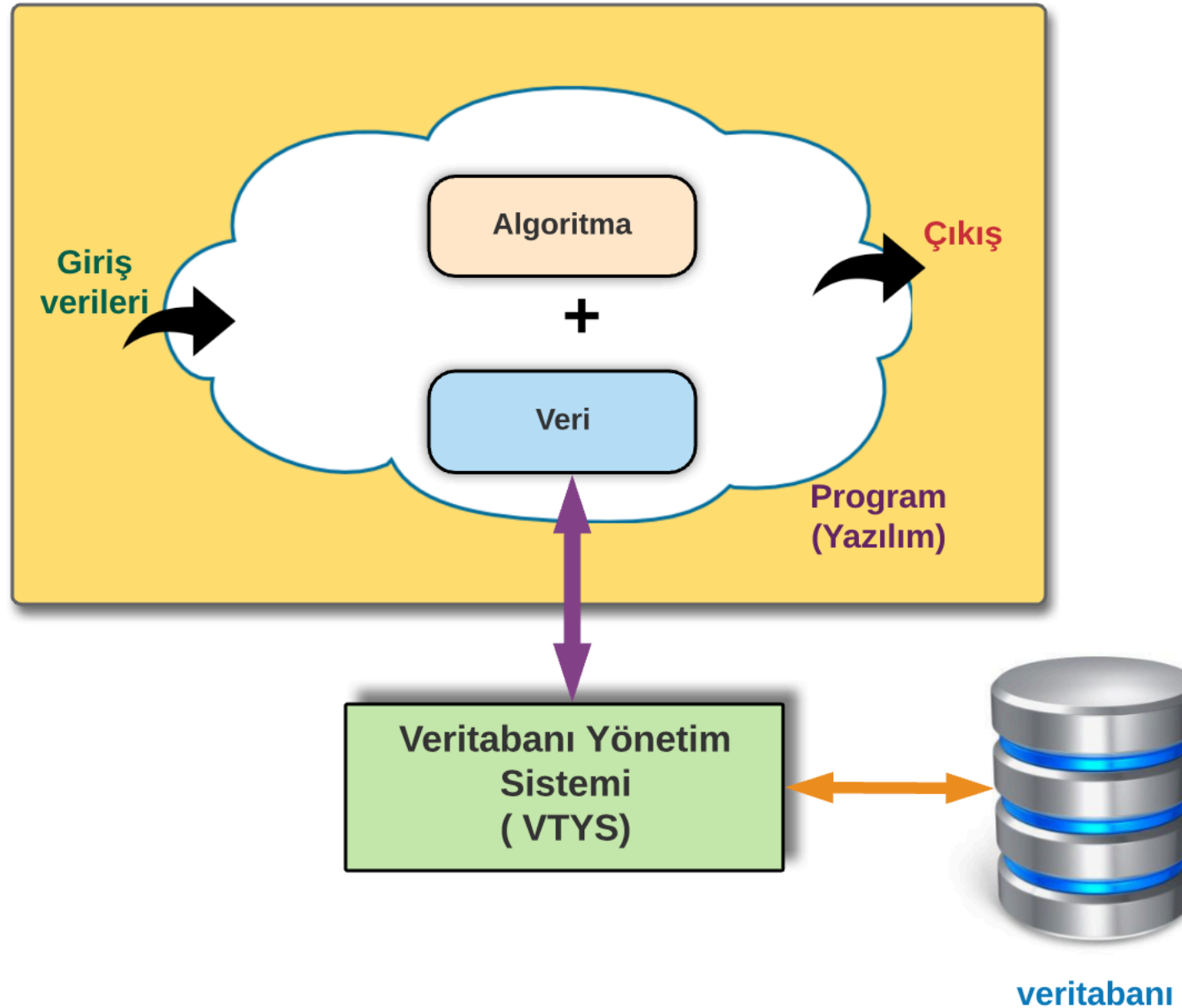


# Büyük Resim

Gerçek Dünya  
Problemi





# İyi Yazılım?

iyi bir yazılım;

- \* kullanıcı ya da müşteri açısından bakıldığında,
  - \* beklendiği gibi çalışan, istenen fonksiyonları yerine getiren, hatalı sonuçlar vermeyen yazılım
- \* yazılım geliştiriciler açısından,
  - \* sonradan değişikliği kolaylaştıran
  - \* kod tekrar kullanımının fazla olduğu
  - \* yeni özellik kazandırmanın kolay olduğu
  - \* temiz kodlanmış, anlaşılabilir, karmaşıklığı azaltılmış yazılımlardır
  - \* bunu sağlamanın yolu ise tasarım ilkelerinin ve desenlerinin uygun kullanımına bağlıdır.



# Yazılım Geliştirme Yaşam Döngüsü

YGVD içerisinde 5 temel evre bulunmaktadır.

Günümüzde farklı yazılım geliştirme yaşam döngüsü modelleri (software processes) bulunmaktadır: waterfall, spiral, iterative/ incremental, agile, unified vb...

bu modellerin tamamı, bir şekilde bu evreleri içerir.

## Nasıl yapmalı?

Tasarım ilkeleri ve Tasarım Desenleri kullanarak modüllerin bir araya getirilmesi  
UML ile Modelleme yapılması

## Test et

Fonksiyon, başarımlar ve güvenlik testleri

1

Analiz (Analysis)

2

Tasarım (Design)

3

Gerçekleme  
(Implementation)

4

Test (Test)

5

Bakım (Maintenance)

Gereksinimlerin Belirlenmesi  
Kısıtların Belirlenmesi  
Nesneler ve aralarındaki bağıntılar belirlenmesi  
UML ile Modelleme yapılması

Geliştirilen modellerin uygun bir programlama dili ile kodlanması

Yazılımın hedef sisteme konuşlandırılması (deployment)  
Sonradan değişiklik

Yap

Kullan

Ne yapmalı?

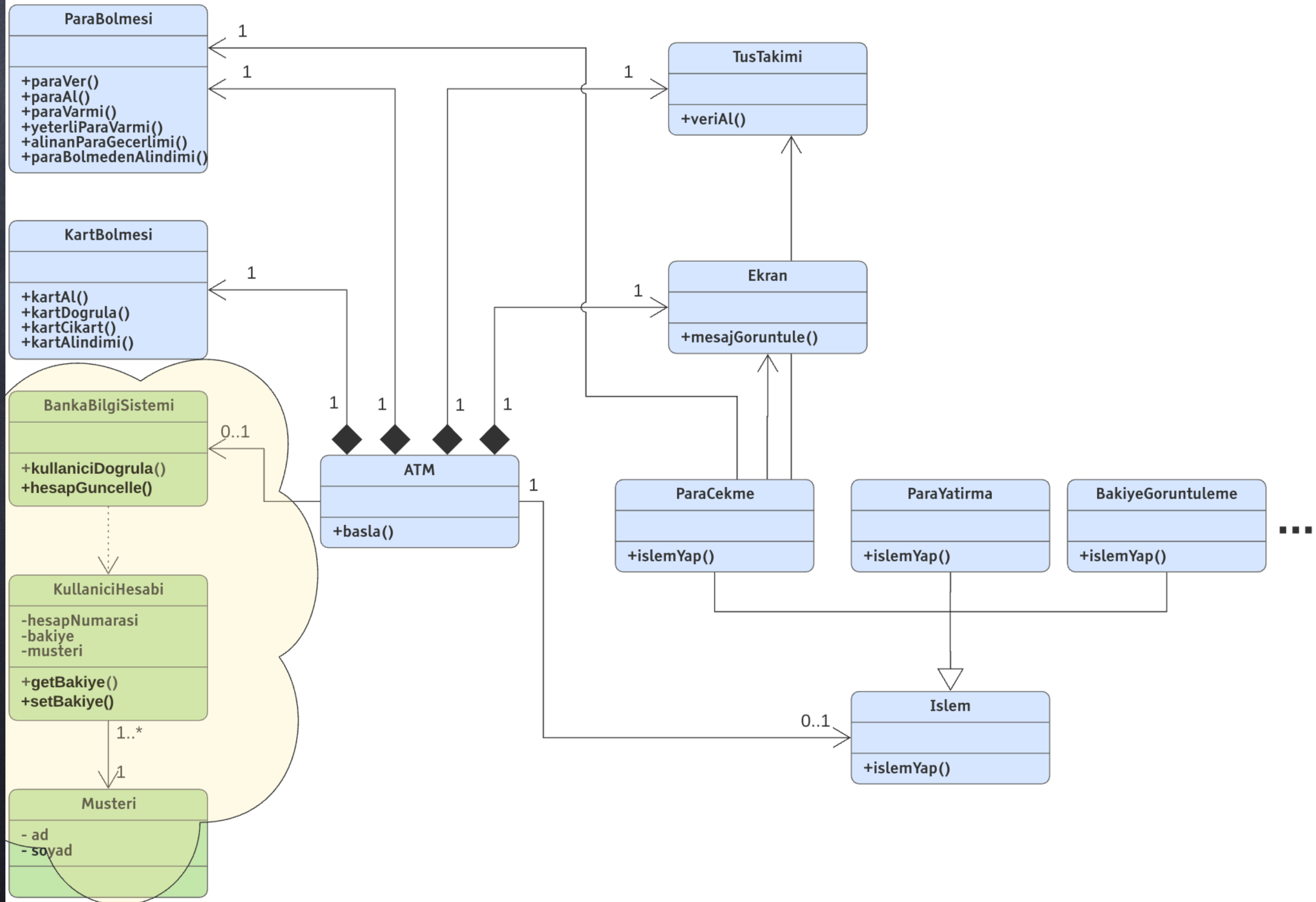


# ATM Sistemindeki Nesnelerin Belirlenmesi

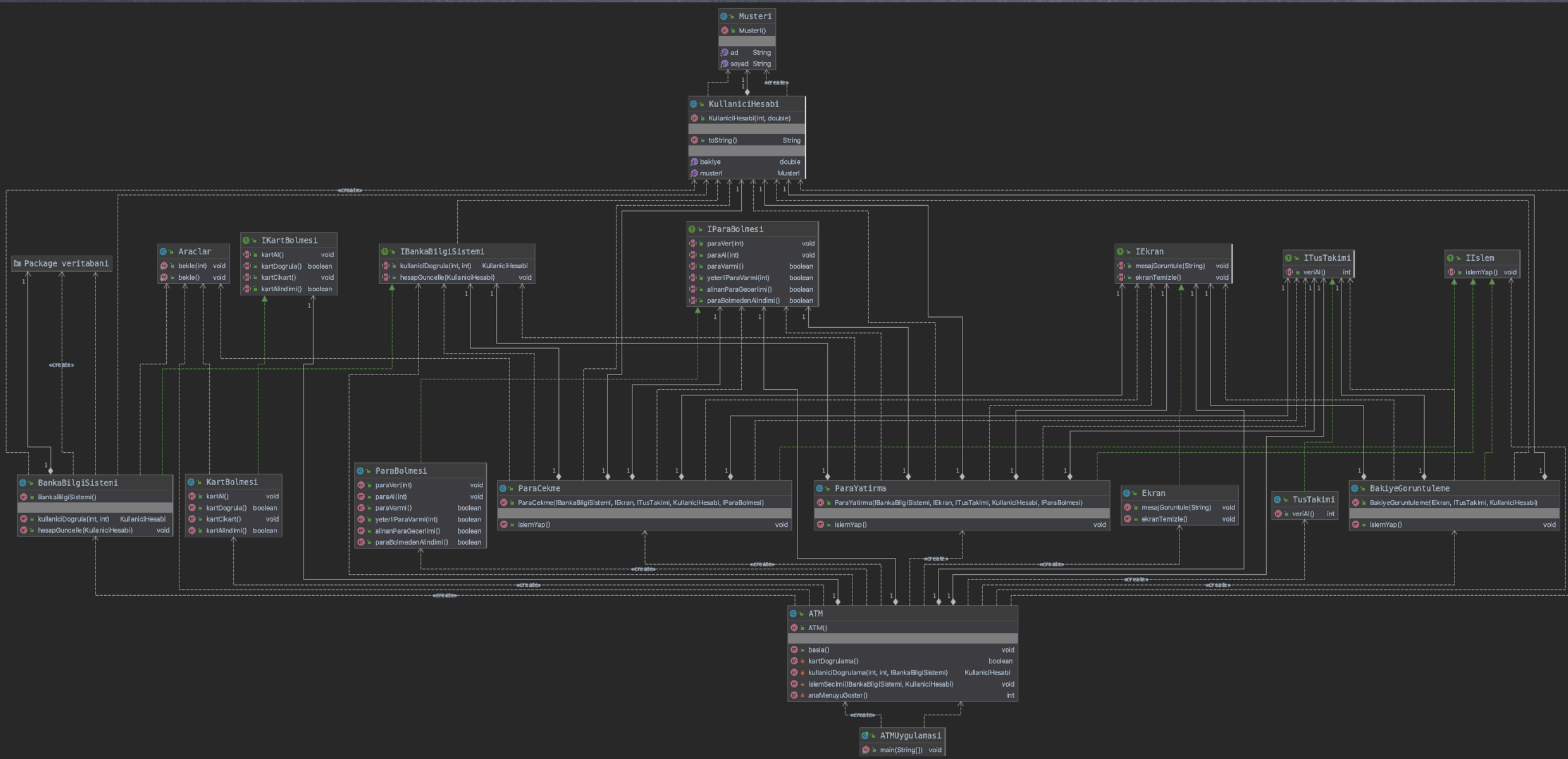
- \* ATM sistemi aşağıdaki donanım bileşenlerinden oluşur:
  - \* Tuş takımı; ile veri girişi yapılır.
  - \* Ekran; kullanıcıya mesaj görüntüler
  - \* Para bölmesi; para alma ve para verme işleminden sorumludur.
  - \* Kart bölmesi; banka kartını alır ve doğrular.
- \* Kullanıcılar banka kartı ile sistemden; para çekme, para yatırma, bakiye görüntüleme işlemlerini yapabilmelidirler.
- \* Banka görevlileri ATM sisteminin bakımını (para doldurma, sıkışan/ yutulan kartları alma v.b. ) yapmalıdır.
- \* Tüm işlemlerde güvenlik gereksinimleri göz önünde bulundurulmalıdır (yetki kontrolü).
- \* Sistem, Banka Bilgi Sistemiyle gevrimiği çalışmalıdır. Kullanıcılar Banka Bilgi Sisteminde yer alan hesaplara erişerek işlemlerini gerçekleştirirler.



# ATM Sisteminin Analiz Aşaması Sınıf Şeması

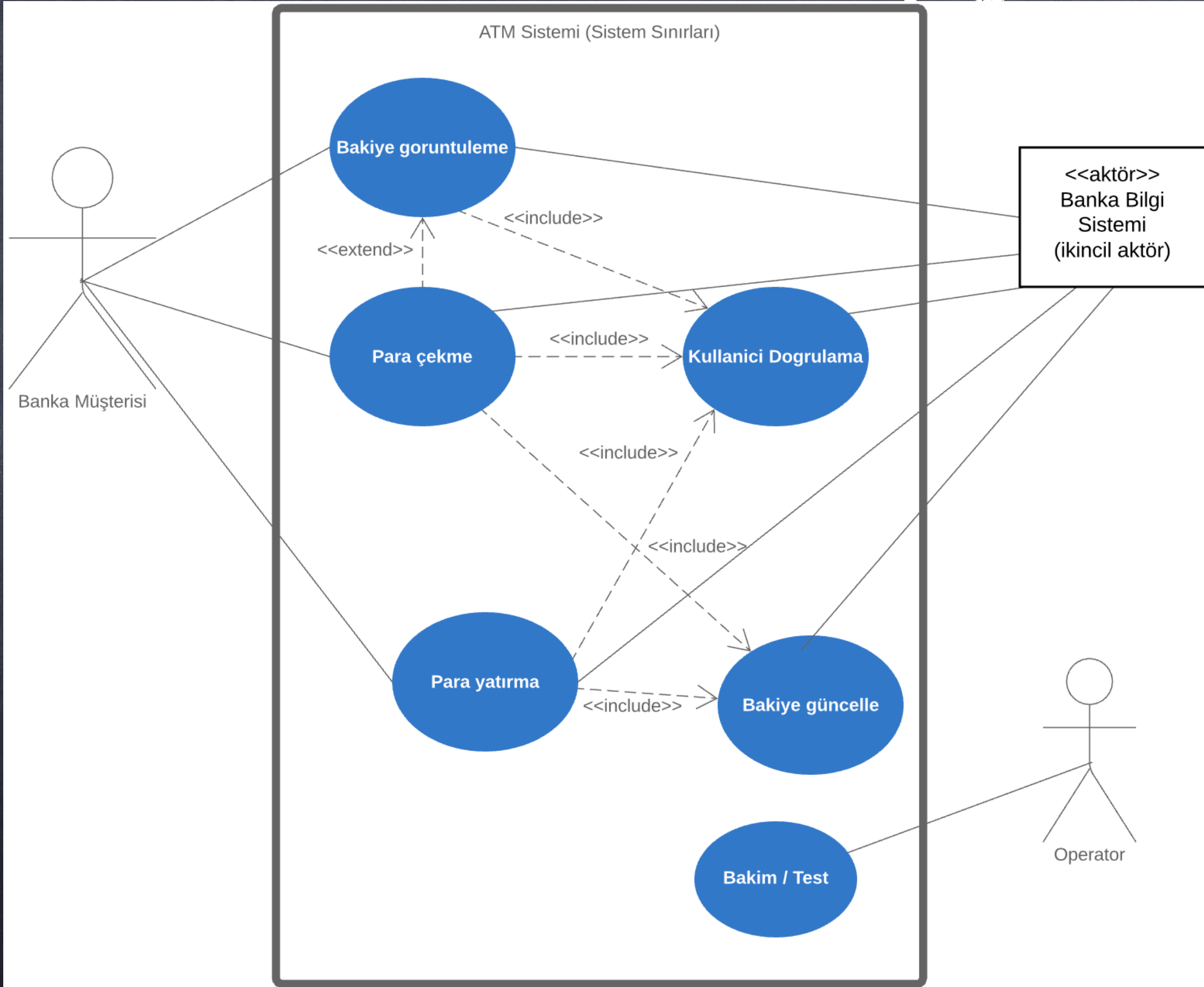


# ATM Sisteminin Analiz Aşaması Sınıf Şeması





# ATM Sistemi Durum Diyagramı





## Use Case- "Para çekme" Olay Akışı (Ana senaryo-başarılı)

1. ATM sistemi Ekrana, müşteriye "kartını kart bölmesine takmasını" isteyen mesaj yazdırır.
2. Kart sahibi kartını kart bölmesine takar.
3. ATM sistemi kart doğrulamasını yapar.
4. Ekrana şifre girilmesini isteyen mesaj gönderilir.
5. Tuş takımını kullanarak girilen şifreyi alır.
6. Kullanıcı doğrulama ve yetki kontrolü için banka bilgi sistemine istek gönderilir.
7. Banka sistemi erişim isteğini kabul eder ve gerekli bilgileri gönderir.
8. ATM sistemi ekrana çekilecek tutarları listeler.
9. Kart sahibi çekilecek tutarı girer.
10. ATM sistemi kartı çıkartır.
11. Kart sahibi kartı alır.
12. ATM sistemi parayı ve makbuzu verir.
13. Kart sahibi parayı ve makbuzu alır.

Not: Kırmızı renkli işlem basamakları Kullanıcı Doğrulama (erişim denetimi) kullanım durumunu ifade eder.



## Use Case- "Para Çekme" Olay Akışı (Alternatif akışlar)

A1. Kart doğrulanamadı (3)

4. Kart bölmesi kartı çıkartır.

5. İşlem sonlandırılır.

A2. Yanlış şifre (6)

7. 3 kez yanlış girilmiş ise kart yutulur.

8. işlem sonlandırılır.

A3. Yeterli bakiye yok (9)

10. ATM de yeterli bakiye yoksa yeni tutar girmesi istenir.

11. Hesapta yeterli bakiye yoksa yeni tutar girmesi istenir.

A4. Banka kartı alınmadı(10)

11. 30 sn. boyunca alınmadı ise kart yutulur

12. Gerekli bilgilendirme yapılır

13. İşlem sonlandırılır

A5. Para alınmadı (12)

13. 30 sn. boyunca alınmadı ise para yutulur

14. Gönderilen ve yutulan para karşılaştırılır.

15. Gerekli ise hesap güncellenmesi yapılır

12. Gerekli bilgilendirme yapılır

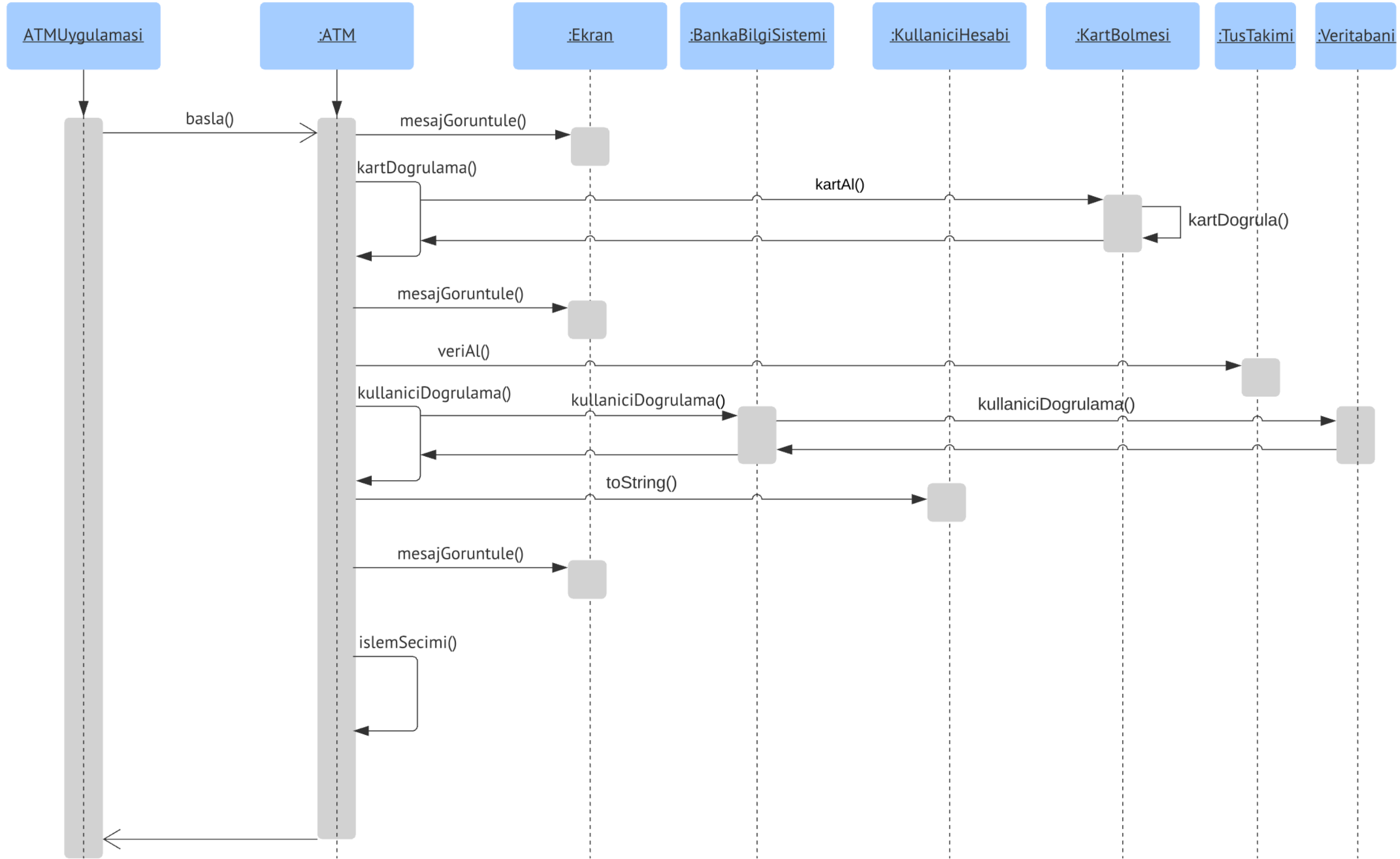
13. İşlem sonlandırılır

...



# ATM Sıralama Şeması (Kullanıcı Doğrulama)

## Kullanıcı Doğrulama İşlemi Sıralama Şeması



zaman

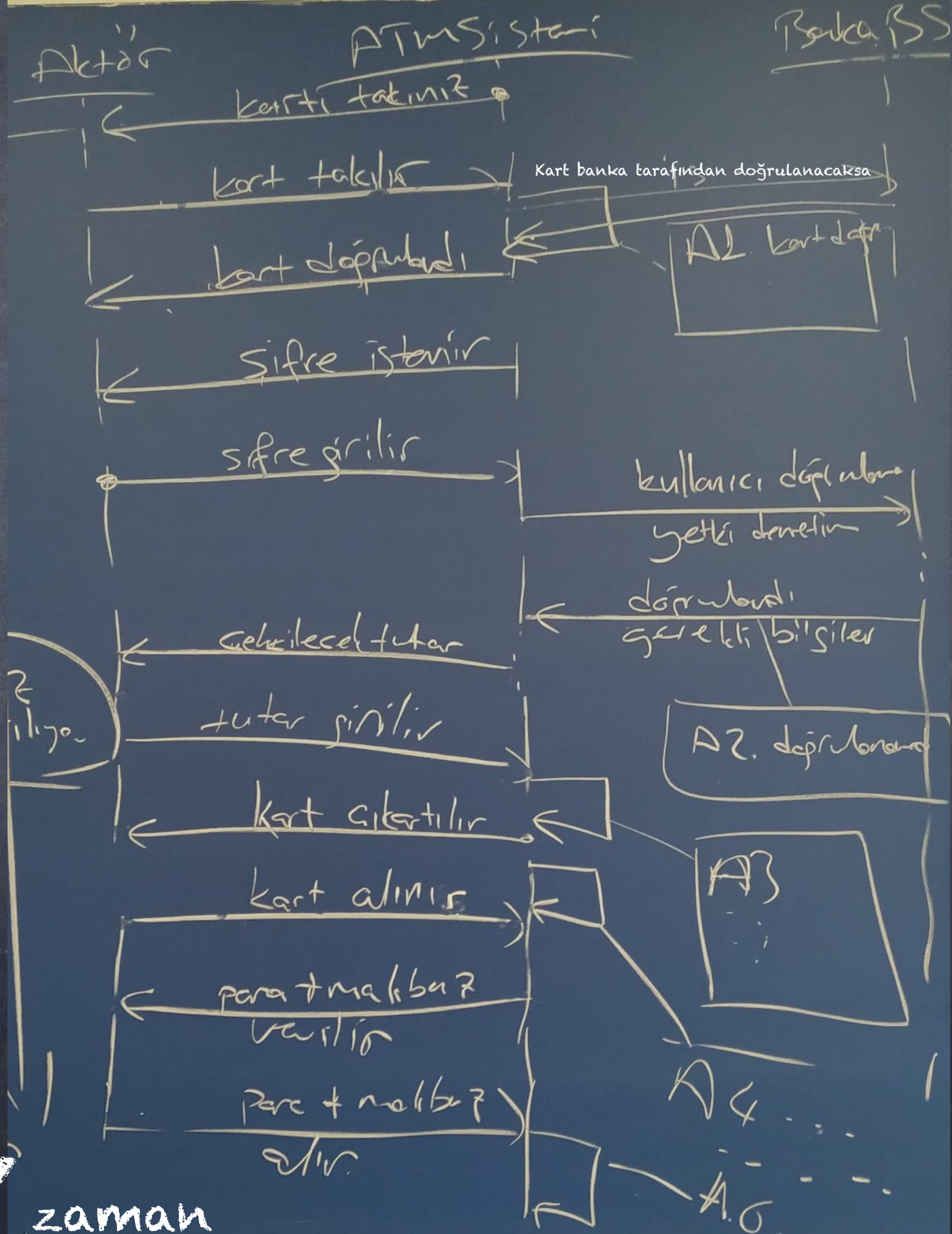


# ATM Sıralama Şeması (Kullanıcı Doğrulama)

```
public void basla() {  
    ekran.mesajGoruntule("Lütfen kartınızı bölmeye takınız...");  
    Araclar.bekle();  
    int hesapNumarasi=this.kartDogrulama();  
    if(hesapNumarasi!=0){  
        ekran.mesajGoruntule("şifreniz");  
        int sifre= tusTakimi.veriAl();  
        IBankaBilgiSistemi bankaBilgiSistemi=new BankaBilgiSistemi();  
        KullaniciHesabi kullaniciHesabi= this.kullaniciDogrulama(hesapNumarasi, sifre, bankaBilgiSistemi);  
        if (kullaniciHesabi!=null){  
            ekran.mesajGoruntule("Kullanıcı doğrulama işlemi başarılı...:...kullaniciHesabi...");  
            islemSecimi(bankaBilgiSistemi, kullaniciHesabi);  
        } else{  
            ekran.mesajGoruntule("hesabınız doğrulanamadı");  
            kartBolmesi.kartCikart();  
        }  
    }  
    } else{  
        ekran.mesajGoruntule("Kartınız doğrulanamadı");  
        kartBolmesi.kartCikart();  
    }  
}
```

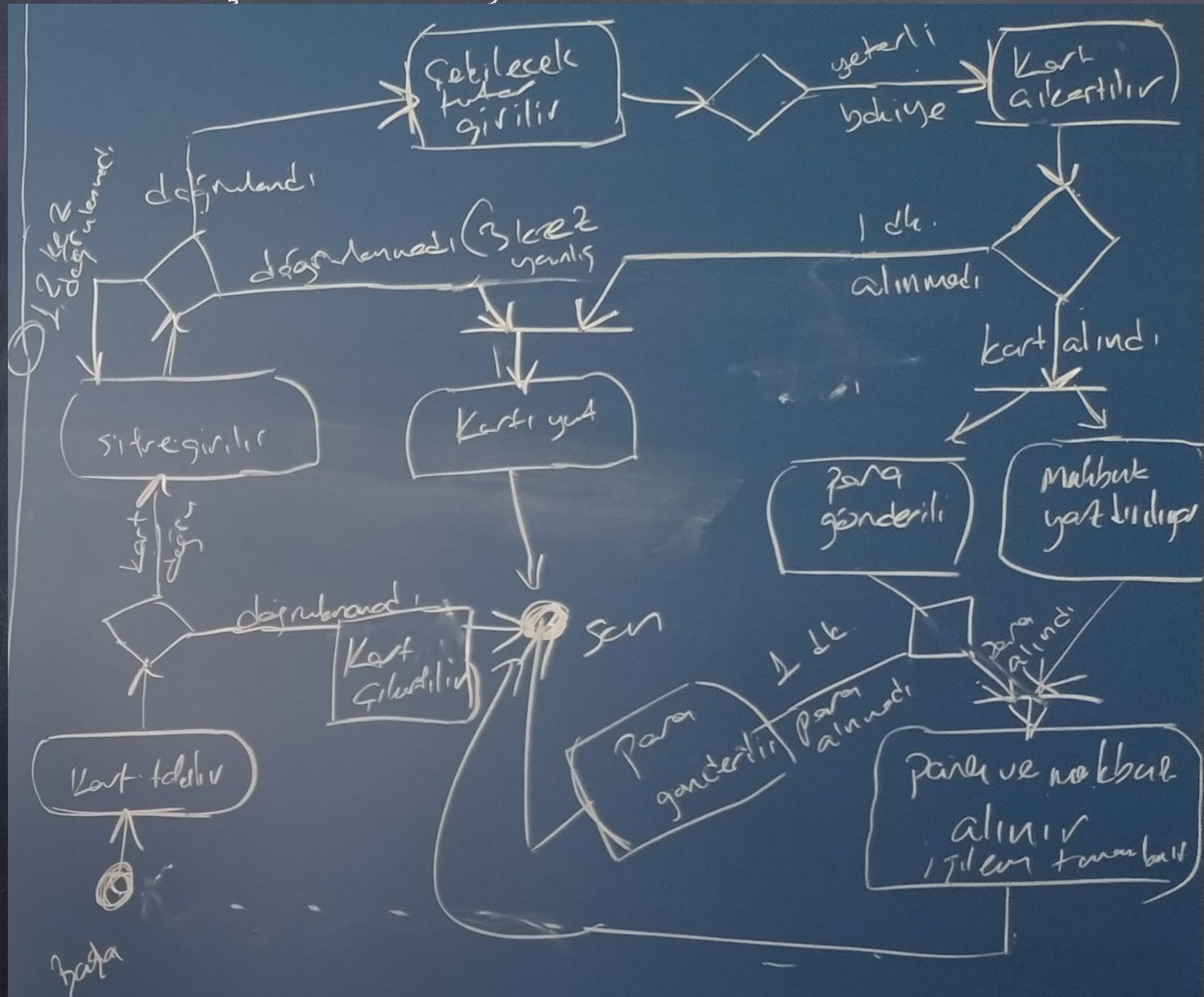


# ATM Sıralama Şeması (Para Çekme)



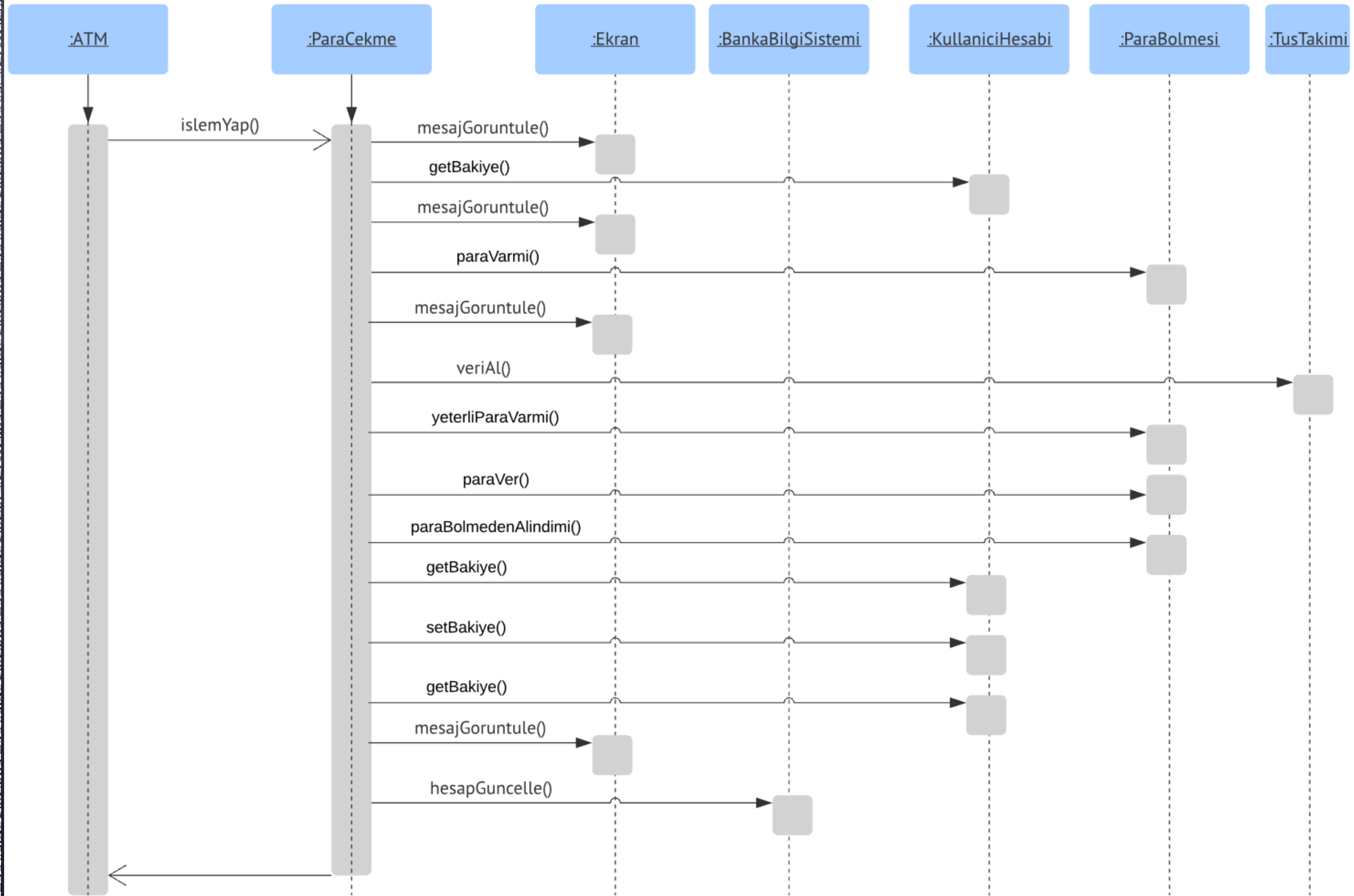


# ATM Etkinlik Şeması (Para Çekme)





# ATM Sıralama Şeması (Para Çekme)



zaman



# ATM Sıralama Şemasından (Para Çekme) Koda Dönüşüm

```
public void islemYap() {  
    ekran.mesajGoruntule("para çekme işlemini seçtiniz");  
    ekran.mesajGoruntule("Bakiyeniz:...kullaniciHesabi.getBakiye()...");  
    if(paraBolmesi.paraVarmi()) {  
        ekran.mesajGoruntule("Tutarı giriniz");  
        int cekilecekTutar = tusTakimi.veriAl();  
        if (paraBolmesi.yeterliParaVarmi(cekilecekTutar)) {  
            paraBolmesi.paraVer(cekilecekTutar);  
            if(paraBolmesi.paraBolmedenAlindimi()) {  
                kullaniciHesabi.setBakiye(kullaniciHesabi.getBakiye() - cekilecekTutar);  
                ekran.mesajGoruntule("son bakiyeniz:...kullaniciHesabi.getBakiye()...");  
                bankaBilgiSistemi.hesapGuncelle(kullaniciHesabi);  
            }else{  
                ekran.mesajGoruntule("Para bölmeden alınmadı");  
                paraBolmesi.paraAl(cekilecekTutar);  
            }  
        } else  
            ekran.mesajGoruntule("Bankamatikte yeterli para yok...");  
    } else{  
        ekran.mesajGoruntule("Bankamatikte para yok...");  
        Araclar.bekle(2000);  
    }  
}
```