



Yıldız Teknik Üniversitesi
BLM2012 Nesneye Yönelik Programlama
2025-2026 Güz Projesi
Dr. Öğr. Üyesi Furkan ÇAKMAK

Ayşe Sude Cami
22011064
sude.cami@std.yildiz.edu.tr

Zeynep Feryat Gözüngül
22011011
feryat.gozungul@std.yildiz.edu.tr

İÇİNDEKİLER

UML Diyagramı

Class ve Methodlar

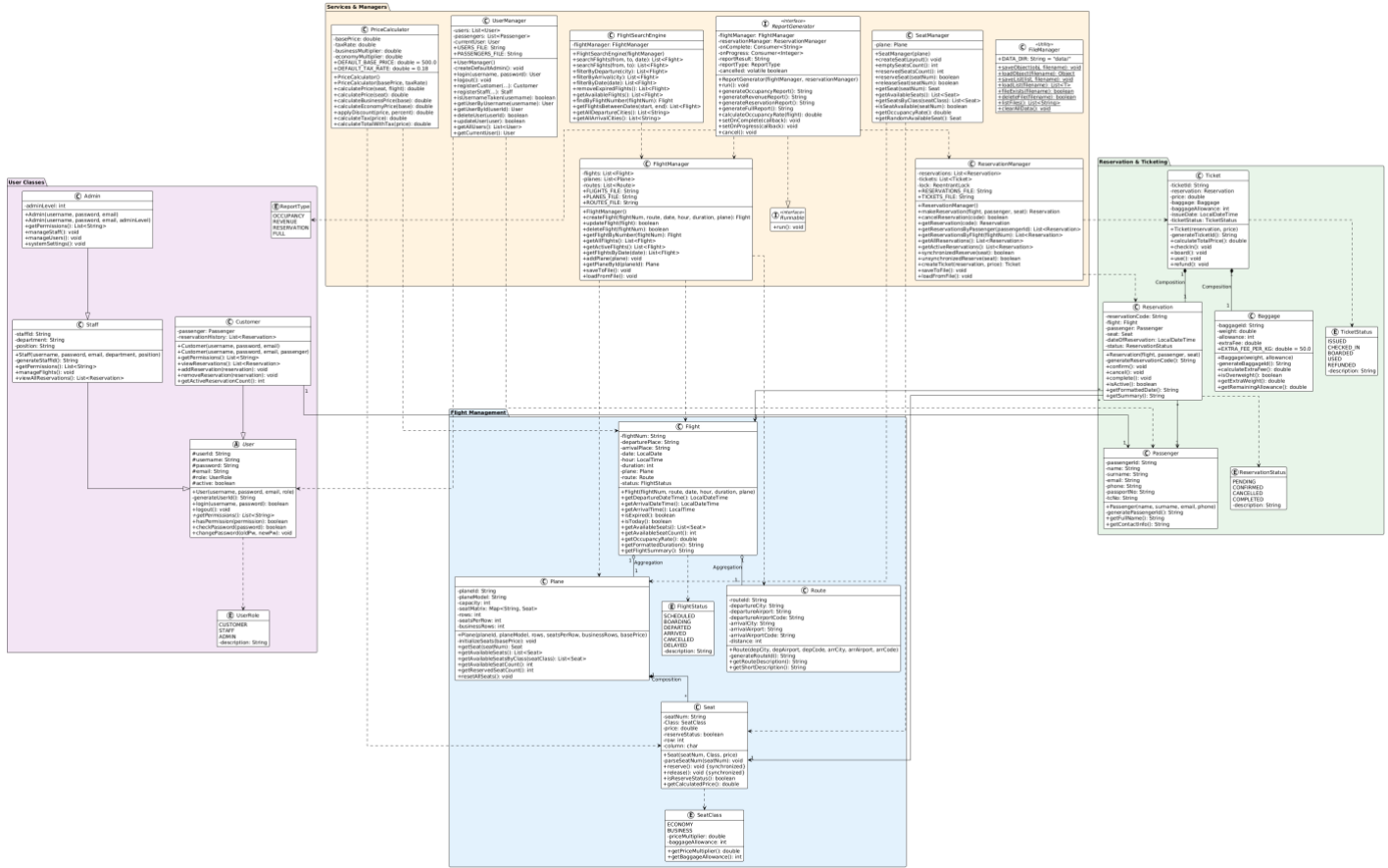
OOP Prensiplerinin Projede Kullanımı

Arayüz Ekranları

Terminalde çalıştırma

UML DİYAGRAMI:

Airline Reservation System - UML Class Diagram



CLASS VE METHODLAR

Module 1: Flight Management

Flight

ATTRIBUTES

- flightNum: String
- departurePlace: String
- arrivalPlace: String
- date: LocalDate
- hour: LocalTime
- duration: int
- plane: Plane
- route: Route
- status: FlightStatus

METHODS

- + getDepartureDateTime(): LocalDateTime
- + getArrivalDateTime(): LocalDateTime
- + isExpired(): boolean
- + isToday(): boolean
- + getAvailableSeats(): List<Seat>
- + getOccupancyRate(): double
- + getDepartureWithAirport(): String
- + getArrivalWithAirport(): String

Plane

ATTRIBUTES

- planeId: String
- planeModel: String
- capacity: int
- seatMatrix: Map<String,Seat>
- rows: int
- businessRows: int

METHODS

- initializeSeats(basePrice): void
- + getSeat(seatNum): Seat
- + getAvailableSeats(): List<Seat>
- + getAvailableSeatsByClass(seatClass List<Seat>
- + getAvailableSeatCount(): int
- + resetAllSeats(): void

Seat

ATTRIBUTES

- seatNum: String
- Class: SeatClass
- price: double
- reserveStatus: boolean
- row: int
- column: char

METHODS

- + reserve(): void [synchronized]
- + release(): void [synchronized]
- + isReserveStatus(): boolean
- + getCalculatedPrice(): double

Route

ATTRIBUTES

- routeId: String
- departureCity: String
- departureAirportCode: String
- arrivalCity: String
- arrivalAirportCode: String
- distance: int

METHODS

- + getRouteDescription(): String
- + getShortDescription(): String

«enum» SeatClass

VALUES

- ECONOMY
- BUSINESS

ATTRIBUTES

- priceMultiplier: double
- baggageAllowance: int
- displayName: String

«enum» FlightStatus

VALUES

- SCHEDULED
- BOARDING
- DEPARTED
- ARRIVED
- CANCELLED
- DELAYED

«enum» AdditionalService

VALUES

- FREE_CANCELLATION
- IN_FLIGHT_ENTERTAINMENT
- MEAL_SELECTION
- EXTRA_BAGGAGE

ATTRIBUTES

- displayName: String
- icon: String
- price: double

Module 2: Reservation & Ticketing

Reservation

ATTRIBUTES

- reservationCode: String
- flight: Flight
- passenger: Passenger
- seat: Seat
- dateOfReservation: LocalDateTime
- status: ReservationStatus

METHODS

- + confirm(): void
- + cancel(): void
- + complete(): void
- + isActive(): boolean
- + getSummary(): String

Ticket

ATTRIBUTES

- ticketId: String
- reservation: Reservation
- price: double
- baggage: Baggage
- issueDate: LocalDateTime
- ticketStatus: TicketStatus

METHODS

- + calculateTotalPrice(): double
- + checkIn(): void
- + board(): void
- + refund(): void
- + getTicketSummary(): String

Passenger

ATTRIBUTES

- passengerId: String
- name: String
- surname: String
- email: String
- phone: String
- passportNo: String

METHODS

- + getFullName(): String
- + getContactInfo(): String

Baggage

ATTRIBUTES

- baggageId: String
- weight: double
- allowance: int
- + EXTRA_FEE_PER_KG: double

METHODS

- + getExtraWeight(): double
- + getExtraFee(): double
- + isOverweight(): boolean

«enum» ReservationStatus

VALUES

- PENDING
- CONFIRMED
- CANCELLED
- COMPLETED

«enum» TicketStatus

VALUES

- ISSUED
- CHECKED_IN
- BOARDED
- USED
- REFUNDED

Module 3: Services & Managers

FlightManager

ATTRIBUTES

```
- flights: List<Flight>
- planes: List<Plane>
- routes: List<Route>
```

METHODS

```
+ createFlight(...): Flight
+ deleteFlight(flightNum): boolean
+ getFlightByNumber(num): Flight
+ getAllFlights(): List<Flight>
+ saveToFile(): void
+ loadFromFile(): void
```

ReservationManager

ATTRIBUTES

```
- reservations: List<Reservation>
- tickets: List<Ticket>
- reservationLock: ReentrantLock
```

METHODS

```
+ makeReservation(flight,passenger,s
Reservation
+ synchronizedReserve(seat):
boolean
+ unsynchronizedReserve(seat):
boolean
+ cancelReservation(code):
boolean
+ createTicket(res,price): Ticket
+ getAllTickets(): List<Ticket>
```

UserManager

ATTRIBUTES

```
- users: List<User>
- passengers: List<Passenger>
- currentUser: User
```

METHODS

```
+ login(username,password): User
+ logout(): void
+ registerCustomer(...): Customer
+ registerStaff(...): Staff
+ isCurrentUserAdmin(): boolean
```

SeatManager

```
ATTRIBUTES
- plane: Plane
```

METHODS

```
+ reserveSeat(seatNum): boolean
+ releaseSeat(seatNum): boolean
+ isSeatAvailable(seatNum):
boolean
+ getOccupancyRate(): double
+ getRandomAvailableSeat(): Seat
```

PriceCalculator

ATTRIBUTES

```
- basePrice: double
- taxRate: double
- businessMultiplier: double
- serviceFee: double
```

METHODS

```
+ calculatePrice(seat,flight):
double
+ calculateBusinessPrice(base):
double
+ applyDiscount(price,percent):
double
+ calculateServicesTotal(services):
double
+ calculatePriceWithServices(...):
double
```

FlightSearchEngine

ATTRIBUTES

```
- flightManager: FlightManager
```

METHODS

```
+ searchFlights(from,to,date):
List<Flight>
+ filterByDeparture(city):
List<Flight>
+ getAvailableFlights():
List<Flight>
+ getAllDepartureCities():
List<String>
```

ReportGenerator Implements Runnable

ATTRIBUTES

```
- flightManager: FlightManager
- reservationManager:
ReservationManager
- reportType: ReportType
- cancelled: volatile boolean
```

METHODS

```
+ run(): void
+ generateOccupancyReport(): String
+ generateRevenueReport(): String
+ generateReservationReport():
String
+ cancel(): void
```

FileManager (Utility)

METHODS (STATIC)

```
+ saveObject(obj,filename): void
+ loadObject(filename): Object
+ saveList(list,filename): void
+ loadList(filename): List<T>
```

Module 4: User Classes

«abstract» User

ATTRIBUTES

```
# userId: String
# username: String
# password: String
# email: String
# role: UserRole
# active: boolean
```

METHODS

```
+ login(username,password):
boolean
+ logout(): void
+ abstract getPermissions():
List<String>
+ hasPermission(perm): boolean
+ checkPassword(pwd): boolean
```

Customer extends User

ATTRIBUTES

```
- passenger: Passenger
- reservationHistory:
List<Reservation>
```

METHODS

```
+ getPermissions(): List<String>
+ viewReservations():
List<Reservation>
+ addReservation(res): void
+ getActiveReservationCount():
int
```

Staff extends User

ATTRIBUTES

```
- staffId: String
- department: String
- position: String
```

METHODS

```
+ getPermissions(): List<String>
+ manageFlights(): void
+ viewAllReservations():
List<Reservation>
```

Admin extends Staff

ATTRIBUTES

```
- adminLevel: int
```

METHODS

```
+ getPermissions(): List<String>
+ manageStaff(): void
+ manageUsers(): void
+ isSuperAdmin(): boolean
```

«enum» UserRole

VALUES

```
CUSTOMER STAFF ADMIN
```

«enum» ReportType

VALUES

```
OCCUPANCY REVENUE RESERVATION
FULL
```

Module 5: Exceptions

AirlineException extends RuntimeException

ATTRIBUTES

```
- errorCode: String
```

METHODS

```
+ getErrorCode(): String
```

SeatNotAvailableException

ATTRIBUTES

```
- seatNumber: String
```

FlightNotFoundException

ATTRIBUTES

```
- flightNum: String
```

SeatNotFoundException

ATTRIBUTES

```
- seatNum: String
```

DuplicateEntryException

ATTRIBUTES

```
- entryType: String
- entryValue: String
```

Modül 1: Uçuş Yönetimi

Bu modül, uçuş operasyonlarının temelini oluşturan sınıfları içermektedir.

1.1 Flight (Uçuş) Sınıfı

Flight sınıfı, bir uçuşu temsil eden ana varlık sınıfıdır.

Attributes:

- flightNum**: Uçuş numarası (örn: "TK101")
- departurePlace** ve **arrivalPlace**: Kalkış ve varış şehirleri
- date** ve **hour**: Uçuş tarihi ve saati (LocalDate, LocalTime)
- duration**: Uçuş süresi (dakika cinsinden)
- plane**: Uçuşa atanan uçak nesnesi (Plane referansı)
- route**: Uçuş rotası (Route referansı)
- status**: Uçuş durumu (FlightStatus enum)

Temel Metotları:

- getDepartureDateTime()** ve **getArrivalDateTime()**: Kalkış ve varış zamanlarını hesaplar
- isExpired()**: Uçuşun geçip geçmediğini kontrol eder
- getAvailableSeats()**: Boş koltukları döndürür
- getOccupancyRate()**: Doluluk oranını hesaplar

1.2 Plane Sınıfı

Plane sınıfı, fiziksel uçağı ve koltuk düzenini modellemektedir.

Attributes:

- planeId**: Uçak kimlik numarası (örn: "TC-101")
- planeModel**: Uçak modeli (örn: "Boeing 737-800")
- capacity**: Toplam koltuk kapasitesi
- seatMatrix**: Koltuk numarası ile Seat nesnelerini eşleştiren Map yapısı
- rows** ve **seatsPerRow**: Sıra sayısı ve sıra başına koltuk sayısı
- businessRows**: Business class için ayrılan sıra sayısı

Önemli Metotları:

- initializeSeats()**: Koltuk düzenini oluşturur (private metot)
- getSeat(seatNum)**: Belirtilen koltuğu döndürür
- getAvailableSeatsByClass()**: Sınıfa göre boş koltukları filtreler

1.3 Seat Sınıfı

Seat sınıfı, uçaktaki her bir koltuğu temsil eder.

Attributes:

- seatNum**: Koltuk numarası (örn: "15A")
- Class**: Koltuk sınıfı (SeatClass enum)
- price**: Koltuk fiyatı
- reserveStatus**: Rezervasyon durumu (boolean)
- row** ve **column**: Sıra ve kolon bilgisi

Thread-Safe Metotları:

- reserve()** ve **release()**: synchronized anahtar kelimesiyle tanımlanmış, eşzamanlı erişimde güvenli rezervasyon işlemleri sağlar

1.4 Route Sınıfı

Route sınıfı, kalkış ve varış noktalarını detaylı olarak tanımlar.

Attributes:

- routeId**: Benzersiz rota kimliği
- departureCity** ve **arrivalCity**: Kalkış ve varış şehirleri
- departureAirportCode** ve **arrivalAirportCode**: Havalimanı kodları (örn: "IST", "ESB")
- distance**: Mesafe (km)

1.5 Enum Tipleri

- SeatClass**: ECONOMY, BUSINESS - Koltuk sınıflarını tanımlar; fiyat çarpanı ve bagaj hakkı bilgilerini içerir
- FlightStatus**: SCHEDULED, BOARDING, DEPARTED, ARRIVED, CANCELLED, DELAYED - Uçuş durumlarını tanımlar
- AdditionalService**: FREE_CANCELLATION, IN_FLIGHT_ENTERTAINMENT, MEAL_SELECTION, EXTRA_BAGGAGE - Ek hizmetleri tanımlar.

Modül 2: Rezervasyon ve Biletleme

Bu modül, rezervasyon sürecini ve bilet yönetimini kapsamaktadır.

2.1 Reservation Sınıfı

Reservation sınıfı, bir yolcunun uçuş rezervasyonunu temsil eder.

Attributes:

- **reservationCode**: 6 karakterlik benzersiz rezervasyon kodu (otomatik üretilir)
- **flight**: İlişkili uçuş nesnesi
- **passenger**: Yolcu bilgisi
- **seat**: Rezerve edilen koltuk
- **dateOfReservation**: Rezervasyon tarihi
- **status**: Rezervasyon durumu

Temel Metotları:

- **confirm()**: Rezervasyonu onaylar ve koltuğu rezerve eder
- **cancel()**: Rezervasyonu iptal eder ve koltuğu serbest bırakır
- **isActive()**: Rezervasyonun aktif olup olmadığını kontrol eder

2.2 Ticket Sınıfı

Ticket sınıfı, onaylanmış rezervasyondan üretilen bileti temsil eder.

Attributes:

- **ticketId**: Benzersiz bilet numarası
- **reservation**: İlişkili rezervasyon
- **price**: Bilet fiyatı
- **baggage**: Bagaj bilgisi
- **issueDate**: Düzenleme tarihi
- **ticketStatus**: Bilet durumu (TicketStatus enum)

Temel Metotları:

- **calculateTotalPrice()**: Bilet ve bagaj ek ücretini hesaplar
- **checkIn(), board(), use()**: Bilet durumunu günceller
- **refund()**: İade işlemi yapar ve rezervasyonu iptal eder

2.3 Passenger Sınıfı

Passenger sınıfı, yolcu bilgilerini içeren varlık sınıfıdır.

Attributes:

- **passengerId**: Benzersiz yolcu kimliği
- **name** ve **surname**: Ad ve soyad
- **email** ve **phone**: İletişim bilgileri
- **passportNo** ve **tcNo**: Kimlik bilgileri

2.4 Baggage Sınıfı

Baggage sınıfı, bagaj ağırlığı ve ek ücret hesaplamalarını yönetir.

Attributes:

- **baggageId**: Benzersiz bagaj kimliği
- **weight**: Gerçek ağırlık (kg)
- **allowance**: İzin verilen ağırlık (kg)
- **EXTRA_FEE_PER_KG**: Kg başına ek ücret sabiti

Kademeli Ücretlendirme: `calculateExtraFee()` metodu kademeli tarife uygular: İlk 10 kg fazlalık için standart ücret, sonrası için %40 zamlı ücret hesaplanır.

2.5 Enum Tipleri

- **ReservationStatus**: PENDING, CONFIRMED, CANCELLED, COMPLETED
- **TicketStatus**: ISSUED, CHECKED_IN, BOARDED, USED, REFUNDED

Modül 3: Servisler ve Yönetici Sınıflar (Services & Managers)

Bu modül, iş mantığını uygulayan yönetici sınıfları içermektedir.

3.1 FlightManager

Uçuş CRUD (Create, Read, Update, Delete) işlemlerini yönetir.

Attributes:

- `flights`: Uçuş listesi
- `planes`: Uçak listesi
- `routes`: Rota listesi

Temel Metotları:

- `createFlight()`: Yeni uçuş oluşturur
- `deleteFlight()`: Uçuş siler
- `getFlightByNumber()`: Uçuş numarasına göre arama
- `saveToFile()` ve `loadFromFile()`: Dosya persistence işlemleri

3.2 ReservationManager

Rezervasyon işlemlerini thread-safe olarak yönetir.

Önemli Attributes:

- `reservations`: Rezervasyon listesi
- `tickets`: Bilet listesi
- `reservationLock`: ReentrantLock nesnesi (eşzamanlılık kontrolü için)

Multithreading Metotları:

- `makeReservation()`: Lock kullanarak thread-safe rezervasyon yapar
- `synchronizedReserve()`: Senkronize koltuk rezervasyonu
- `unsynchronizedReserve()`: Race condition gösterimi için senkronizasyonsuz metot

3.3 UserManager

Kullanıcı kimlik doğrulama ve yetkilendirme işlemlerini yönetir.

Attributes:

- `users`: Kullanıcı listesi
- `passengers`: Yolcu listesi
- `currentUser`: Oturum açmış kullanıcı

Temel Metotları:

- `login()` ve `logout()`: Oturum yönetimi
- `registerCustomer()` ve `registerStaff()`: Kullanıcı kaydı
- `isCurrentUserAdmin()`: Yetki kontrolü

3.4 SeatManager

Belirli bir uçağın koltuk işlemlerini yönetir.

Temel Metotları:

- `reserveSeat()` ve `releaseSeat()`: Koltuk rezervasyonu
- `isSeatAvailable()`: Müsaitlik kontrolü
- `getOccupancyRate()`: Doluluk oranı hesaplama
- `getRandomAvailableSeat()`: Rastgele boş koltuk seçimi

3.5 PriceCalculator

Bilet fiyatlandırma mantığını içerir.

Attributes:

- `basePrice`: Temel fiyat
- `taxRate`: Vergi oranı
- `businessMultiplier`: Business class çarpanı
- `serviceFee`: Hizmet ücreti

3.6 FlightSearchEngine

Uçuş arama ve filtreleme işlevlerini sağlar.

3.7 ReportGenerator

Runnable arayüzünü implement eder, arka planda rapor üretimi yapar.

Önemli Nitelikleri:

- `reportType`: Rapor tipi (ReportType enum)
- `cancelled`: volatile boolean - thread'in iptal edilip edilmediğini kontrol eder

3.8 FileManager

Statik metotlarla dosya okuma/yazma işlemlerini gerçekleştirir (Utility sınıfı).

Modül 4: Kullanıcı Sınıfları (User Classes)

Bu modül, kullanıcı hiyerarşisini ve yetkilendirme sistemini içermektedir.

4.1 User - Abstract Sınıf

Tüm kullanıcı tiplerinin temel sınıfıdır.

Attributes (protected erişim):

- **userId**: Benzersiz kullanıcı kimliği
- **username** ve **password**: Kimlik bilgileri
- **email**: E-posta adresi
- **role**: Kullanıcı rolü (UserRole enum)
- **active**: Hesap durumu

Abstract Metot:

- **getPermissions()**: Alt sınıflarda override edilmesi zorunlu, kullanıcı yetkilerini döndürür

4.2 Customer (Müşteri) - User'dan Türetilmiş

Normal müşteri kullanıcıyı temsil eder.

Ek Attributes:

- **passenger**: İlişkili yolcu bilgisi
- **reservationHistory**: Rezervasyon geçmişi listesi

Yetkileri: SEARCH_FLIGHTS, VIEW_FLIGHT_DETAILS, MAKE_RESERVATION, VIEW_OWN_RESERVATIONS, CANCEL_OWN_RESERVATION, VIEW_TICKET, UPDATE_PROFILE

4.3 Staff - User'dan Türetilmiş

Havayolu personelini temsil eder.

Ek Attributes:

- **staffId**: Personel kimlik numarası
- **department**: Departman
- **position**: Pozisyon

Ek Yetkileri: Müşteri yetkilerine ek olarak: VIEW_ALL_RESERVATIONS, MANAGE_FLIGHTS, ADD_FLIGHT, EDIT_FLIGHT, VIEW_REPORTS, CHECK_IN_PASSENGER

4.4 Admin - Staff'tan Türetilmiş

Sistem yöneticisini temsil eder (çok seviyeli kalıtım örneği).

Ek Attributes:

- **adminLevel**: Yönetici seviyesi (1 veya 2)

Ek Yetkileri: Personel yetkilerine ek olarak:

MANAGE_USERS, CREATE_USER, DELETE_USER, MANAGE_STAFF, SYSTEM_SETTINGS, VIEW_ALL_REPORTS, DELETE_FLIGHT, CANCEL_ANY_RESERVATION

Level 2 adminler için ekstra yetkiler:

MANAGE_ADMINS, SYSTEM_BACKUP, SYSTEM_RESTORE

4.5 Enum Tipleri

- **UserRole**: CUSTOMER, STAFF, ADMIN
- **ReportType**: OCCUPANCY, REVENUE, RESERVATION, FULL

Modül 5: Exceptions

Özel hata durumlarını yönetmek için tasarlanmış exception sınıfları.

5.1 AirlineException

Temel exception sınıfı, RuntimeException'dan türetilmiştir.

- **errorCode**: Hata kodu niteliği

5.2 Alt Exception Sınıfları

- **SeatNotAvailableException**: Koltuk müsait olmadığında fırlatılır
- **FlightNotFoundException**: Uçuş bulunamadığında fırlatılır
- **SeatNotFoundException**: Koltuk bulunamadığında fırlatılır
- **DuplicateEntryException**: Mükerrer kayıt durumunda fırlatılır

OOP PRENSİPLERİNİN PROJEDE KULLANIMI

1. ENCAPSULATION (Kapsülleme)

Kapsülleme, verilerin private olarak saklanması ve public getter/setter metodları ile erişimin kontrol altına alınmasıdır.

Örnek kullanım alanları:

Sınıf	Private Alanlar	Kontrollü Erişim
Seat.java	reserveStatus, price	reserve(), release() metodları
Flight.java	route, plane, status	Getter/Setter ile
Reservation.java	status, seat	confirm(), cancel() metodları
PriceCalculator.java	basePrice, taxRate	Setter'da validasyon

2. INHERITANCE (Kalıtım)

Kalıtım, bir sınıfın başka bir sınıftan özelliklerini ve davranışlarını miras almasıdır.

Örnek kullanım:

```
public class Customer extends User {
    private static final long serialVersionUID = 1L;

    private Passenger passenger;
    private List<Reservation> reservationHistory;

    public Customer(String username, String password, String email) {
        super(username, password, email, UserRole.CUSTOMER);
        this.reservationHistory = new ArrayList<>();
    }
}

public class Admin extends Staff {
    private static final long serialVersionUID = 1L;

    private int adminLevel;

    public Admin(String username, String password, String email) {
        super(username, password, email, "Yönetim", "Sistem Yöneticisi");
        this.setRole(UserRole.ADMIN);
        this.adminLevel = 1;
    }
}
```

3. POLYMORPHISM (Çok Biçimlilik)

Aynı metodun farklı sınıflarda farklı davranışlar sergilemesidir.

Örnek kullanım (getPermissions() metodunun her sınıfta farklı implementasyonu):

```
@Override
public List<String> getPermissions() {
    return Arrays.asList(
        "SEARCH_FLIGHTS",
        "VIEW_FLIGHT_DETAILS",
        "MAKE_RESERVATION",
        "VIEW_OWN_RESERVATIONS",
        "CANCEL_OWN_RESERVATION",
        "VIEW_TICKET",
        "UPDATE_PROFILE"
    );
}

@Override
public List<String> getPermissions() {
    List<String> permissions = new ArrayList<>(super.getPermissions());
    permissions.addAll(Arrays.asList(
        "MANAGE_USERS",
        "CREATE_USER",
        "DELETE_USER",
        "MANAGE_STAFF",
        "CREATE_STAFF",
        "DELETE_STAFF",
        "SYSTEM_SETTINGS",
        "VIEW_ALL_REPORTS",
        "DELETE_FLIGHT",
        "CANCEL_ANY_RESERVATION",
        "VIEW_SYSTEM_LOGS",
        "MANAGE_ROUTES",
        "MANAGE_PLANS"
    ));

    if (adminLevel >= 2) {
        permissions.addAll(Arrays.asList(
            "MANAGE_AGENCIES",
            "SYSTEM_BACKUP",
            "SYSTEM_RESTORE"
        ));
    }

    return permissions;
}
```

Başka örnek kullanım (toString() metodunun her sınıfta farklı implementasyonu):

```
@Override
public String toString() {
    return String.format("User{id='%s', username='%s', role='%s', active=%s}",
        userId, username, role, active);
}

@Override
public String toString() {
    return String.format("Customer{id='%s', username='%s', passenger='%s', reservations=%d}",
        getUserId(), getUsername(),
        passenger != null ? passenger.getFullName() : "N/A",
        reservationHistory.size());
}
```

4. ABSTRACTION (Soyutlama)

Karmaşık implementasyonun gizlenerek basit bir arayüz sunulmasıdır.

Örnek kullanım:

```
public abstract class User implements Serializable {

    public abstract List<String> getPermissions();

    public boolean hasPermission(String permission) {
        return getPermissions().contains(permission);
    }
}
```

Interface Implementasyonu:

```
public class ReportGenerator implements Runnable {

    @Override
    public void run() {
        try {
            switch (reportType) {
                case OCCUPANCY:
                    reportResult = generateOccupancyReport();
                    break;
                case REVENUE:
                    reportResult = generateRevenueReport();
                    break;
                case RESERVATION:
                    reportResult = generateReservationReport();
                    break;
                case FULL:
                    reportResult = generateFullReport();
                    break;
            }

            if (onComplete != null && !cancelled) {
                onComplete.accept(reportResult);
            }
        } catch (Exception e) {
            reportResult = "Rapor oluşturulurken hata: " + e.getMessage();
            if (onComplete != null) {
                onComplete.accept(reportResult);
            }
        }
    }
}
```



→ Havayolu Rezervasyon Sistemi

Hoş Geldiniz

Kullanıcı Adı

Şifre

Giriş Yap

Yeni Hesap Oluştur

1. KULLANICI SAYFASI

Havayolu Rezervasyon Sistemi

Hoş geldiniz, Ayşe Sude Cami

Çıkış

Uçuş Ara

Rezervasyonlarım

Gidiş Dönüş

Kalkış:

Ülke, şehir veya hava:

Nereye

Ülke, şehir veya hava:

Kalkış

Tarih ekleyin

Dönüş

Tarih ekleyin

Yolcular ve kabin sınıfı

I yetişkin

Ekonomi

Ara

Tüm Uçuşları Göster

Uçuş Sonuçları

Uçuş No	Kalkış	Varış	Tarih	Saat	Süre	Boş Koltuk	Durum
TK101	İstanbul (IST)	Ankara (ESB)	2026-01-09	23:59	1 sa 0 dk	177	Planlandı
TK201	İstanbul (IST)	İzmir (ADB)	2026-01-09	23:59	1 sa 15 dk	177	Planlandı
TK301	Ankara (ESB)	Antalya (AYT)	2026-01-10	09:00	1 sa 30 dk	177	Planlandı
TK102	İstanbul (IST)	Ankara (ESB)	2026-01-10	14:00	1 sa 0 dk	180	Planlandı

Seçili Uçuşu Rezerve Et

Gidiş -Dönüş ve Tek yön şeklinde arama yapabiliriz. Eğer seçili tarihte dönüş yoksa sadece gidiş gösterilir.

İstenen uçuşa basıldıktan sonra rezerve et tuşuna basılıp koltuk seçilebilir. Business veya Ekonomi seçenekleri var. Koltuk seçiminden sonra ek hizmet seçenekleri çıkar.

Koltuk Seçimi

TK101: İstanbul → Ankara (2026-01-09 23:59)

Sınıf Seçin:

Ekonomi

Koltuk Seçin:

- Koltuk 6A (Ekonomi)
- Koltuk 6B (Ekonomi)
- Koltuk 6D (Ekonomi)
- Koltuk 6E (Ekonomi)
- Koltuk 6F (Ekonomi)
- Koltuk 7A (Ekonomi)
- Koltuk 7B (Ekonomi)
- Koltuk 7C (Ekonomi)
- Koltuk 7D (Ekonomi)

Fiyat: -

Cancel Rezerve Et

Ek Hizmetler

UÇUŞUNA EK HİZMET EKLE

Ücretsiz İptal
+75 TL

Uçak İçi Eğlence
+50 TL

Yemek Seçimi
+45 TL

Ek Bagaj Hakkı
+120 TL

Bilet Fiyatı: 633,15 TL
Toplam: 633,15 TL

Cancel Devam Et

Başarılı

Rezervasyon tamamlandı!
Rezervasyon Kodu: 45JXMV
Toplam Fiyat: 798,15 TL

Seçilen Ek Hizmetler:

- Ek Bagaj Hakkı
- Yemek Seçimi

OK

Kullanıcı kendisine ait rezervasyonları “Rezervasyonlarım” sayfasından görüntüleyebilir ve iptal edebilir.

[→ Havayolu Rezervasyon Sistemi](#)

Hoş geldiniz, Ayşe Sude Cami

Çıkış

Uçuş Ara

Rezervasyonlarım

Rezervasyonlarım

Rezervasyon Kodu	Uçuş	Rota	Tarih	Koltuk	Durum
S8FH8G	TK101	İstanbul (IST) → Ankara (ESB)	2026-01-09	6C	Onaylandı
45JXNV	TK101	İstanbul (IST) → Ankara (ESB)	2026-01-09	6A	Onaylandı

Yenile

Seçili Rezervasyonu İptal Et

Admin sayfasına girebilmek için kullanıcı adı: admin, şifre: admin123

[illegible]

Yeni Uçuş Ekle

Uçuş bilgilerini girin

Uçuş No:

TK001

Kalkış:

İstanbul

Variş:

Ankara

Tarih:

10.01.2026



Saat (HH:mm):

10:30

Süre (dk):

60

Uçak ID:

TC-001

Uçak Model:

Boeing 737

Cancel


Ekle

Uçuş Düzenle

Uçuş: TK101

Tarih:

9.01.2026



Saat:

23:59

Süre (dk):

60

Durum:

SCHEDULED

SCHEDULED

BOARDING

DEPARTED

ARRIVED

CANCELLED

DELAYED

Kaydet

“Rezervasyonlar” sayfasından yapılan tüm rezervasyonlar görüntülenebilir:

[illegible]

“Raporlar” sayfasından Doluluk Raporu, Gelir Raporu ve Rezervasyon Raporu ayrı ayrı görüntülenebilir:

→ Havayolu Yönetim Paneli

admin (Yönetici)

Çıkış

→ Uçuş Yönetimi

Rezervasyonlar

Raporlar

Simülasyon

Raporlar (Asenkron)

Doluluk Raporu

Gelir Raporu

Rezervasyon Raporu

Tam Rapor

Rapor hazır!

UÇUŞ DOLULUK ORANI RAPORU

Oluşturulma: 09.01.2026 13:04:33

TK101		İstanbul	→	Ankara		Doluluk: 2,22%
TK102		İstanbul	→	Ankara		Doluluk: 0,00%
TK201		İstanbul	→	İzmir		Doluluk: 2,22%
TK301		Ankara	→	Antalya		Doluluk: 2,22%

ORTALAMA DOLULUK ORANI: 1,67%

Toplam Uçuş Sayısı: 4

→ Havalolu Yönetim Paneli

admin (Yönetici) Çıkış

→ Uçuş Yönetimi

Rezervasyonlar

Raporlar

Simülasyon

Raporlar (Asenkron)

Doluluk Raporu

Gelir Raporu

Rezervasyon Raporu

Tam Rapor

Rapor hazır!

GELİR RAPORU

Oluşturulma: 09.01.2026 13:05:37

Onaylanan Rezervasyon : 4
İptal Edilen : 3
Toplam Rezervasyon : 7

TOPLAM GELİR : 3096.30 TL
(Ek hizmetler dahil)

Havayolu Yönetim Paneli

admin (Yönetici) Çıkış

→ Uçuş YönetimiRezervasyonlarRaporlarSimülasyon

Raporlar (Asenkron)

Doluluk RaporuGelir RaporuRezervasyon RaporuTam Rapor

Rapor hazır!

REZERVASYON RAPORU

Oluşturulma: 09.01.2026 13:05:48

Kod: A8NXVV | Ayşe Sude Cami | İstanbul → Antalya | Koltuk: 6A | İptal Edildi
Kod: 6J5ZW2 | Ayşe Sude Cami | Ankara → Antalya | Koltuk: 9B | İptal Edildi
Kod: PCBTRT | Ayşe Sude Cami | İstanbul → İzmir | Koltuk: 6C | İptal Edildi
Kod: N9V2IE | Ayşe Sude Cami | İstanbul → Ankara | Koltuk: 1A | Onaylandı
Kod: WTP9CO | zeyn gul | İstanbul → İzmir | Koltuk: 24B | Onaylandı
Kod: S8FH8G | Ayşe Sude Cami | İstanbul → Ankara | Koltuk: 6C | Onaylandı
Kod: 45JXMY | Ayşe Sude Cami | İstanbul → Ankara | Koltuk: 6A | Onaylandı

Toplam Rezervasyon: 7

“Simülasyon” sayfasında da 1. senaryoda istenen asenkron ve senkron seçenekli 90 yolcuya ait simülasyon gösterilir:

Havayolu Yönetim Paneli

admin (Yönetici) Çıkış

→ Uçuş YönetimiRezervasyonlarRaporlarSimülasyon

Eşzamanlı Koltuk Rezervasyonu Simülasyonu

Bu simülasyon, 90 yolcunun aynı anda 180 koltuğa

☒ Synchronized (Thread-Safe) Simülasyonu Başlat

	A	B	C	D	E	F
1	■	■	■	■	■	■
2	■	■	■	■	■	■
3	■	■	■	■	■	■
4	■	■	■	■	■	■
5	■	■	■	■	■	■
6	■	■	■	■	■	■
7	■	■	■	■	■	■
8	■	■	■	■	■	■
9	■	■	■	■	■	■
10	■	■	■	■	■	■
11	■	■	■	■	■	■
12	■	■	■	■	■	■
13	■	■	■	■	■	■

Tamamlandı (Synchronized) - Doğru sonuç! Dolu: 90 Boş: 90

Eşzamanlı Koltuk Rezervasyonu Simülasyonu

Bu simülasyon, 90 yolcunun aynı anda 180 koltuğa

☐ Synchronized (Thread-Safe) Simülasyonu Başlat

	A	B	C	D	E	F
1	■	■	■	■	■	■
2	■	■	■	■	■	■
3	■	■	■	■	■	■
4	■	■	■	■	■	■
5	■	■	■	■	■	■
6	■	■	■	■	■	■
7	■	■	■	■	■	■
8	■	■	■	■	■	■
9	■	■	■	■	■	■
10	■	■	■	■	■	■
11	■	■	■	■	■	■
12	■	■	■	■	■	■
13	■	■	■	■	■	■

Tamamlandı (Unsynchronized) - HATALI! Race condition tespit edildi! Dolu: 68 (Çakışma: 22) Boş: 112

Projeyi istersek gui olmadan terminalde de

çalıştırabiliriz:

```
ayse@ayse@ayse:~/ayse$ ./run.sh -c
Terminal modunda başlatılıyor...

HAYATOLU REZERVASYON SİSTEMİ - Terminal Modu

KİMLİK DOĞRULAMA EKRANI

1. Giriş Yap
2. Yeni Hesap Oluştur
0. Çıkış

Seçiminiz: 1
```

Giriş yapma (Kullanıcı ekranı):

```
Seçiminiz: 1

===== GİRİŞ =====
Kullanıcı adı: asude
Şifre: asude123

✓ Hoş geldiniz, asude!

===== ANA MENÜ =====
1. Uçuş Ara
2. Tüm Uçuşları Listele
3. Rezervasyon Yap
4. Rezervasyonlarımı Görüntüle
5. Rezervasyon İptal Et
6. Uçuş Detayı Görüntüle
0. Çıkış

Seçiminiz: 1
```

Uçuş arama:

```
Seçiminiz: 1

===== UÇUŞ ARA =====
Kalkış şehri (boş bırakılabilir): İstanbul
Varış şehri (boş bırakılabilir): Ankara
Tarih (YYYY-MM-DD, boş bırakılabilir):

2 uçuş bulundu:

Uçuş No | Kalkış | Varış | Tarih/Saat | Fiyat | Boş Kol.
-----|-----|-----|-----|-----|-----
TK101 | İstanbul Havalim.. | Ankara Havalimanı | 09.01.2026 23:59 | 500,00 | 176
TK102 | İstanbul Havalim.. | Ankara Havalimanı | 10.01.2026 14:00 | 450,00 | 180
```

Tüm uçuşları görüntüleme:

```
Seçiminiz: 2

===== TÜM UÇUŞLAR =====

Uçuş No | Kalkış | Varış | Tarih/Saat | Fiyat | Boş Kol.
-----|-----|-----|-----|-----|-----
TK101 | İstanbul Havalim.. | Ankara Havalimanı | 09.01.2026 23:59 | 500,00 | 176
TK102 | İstanbul Havalim.. | Ankara Havalimanı | 10.01.2026 14:00 | 450,00 | 180
TK201 | İstanbul Havalim.. | İzmir Havalimanı | 09.01.2026 23:59 | 500,00 | 176
TK301 | Ankara Havalimanı | Antalya Havalimanı | 10.01.2026 09:00 | 500,00 | 176
```

Rezervasyon yapma:

```
Seçiminiz: 3

===== REZERVASYON YAP =====
Uçuş numarası: TK101

Müsait koltuklar:
1B (Business) 1C (Business) 1D (Business) 1E (Business) 1F (Business) 2A (Business) 2B (Business) 2C (Business)
2D (Business) 2E (Business) 2F (Business) 3A (Business) 3B (Business) 3C (Business) 3D (Business) 3E (Business)
3F (Business) 4A (Business) 4B (Business) 4C (Business) 4D (Business) 4E (Business) 4F (Business) 5A (Business)
5B (Business) 5C (Business) 5D (Business) 5E (Business) 5F (Business) 6B (Ekonomi) 6D (Ekonomi) 6E (Ekonomi)
6F (Ekonomi) 7A (Ekonomi) 7B (Ekonomi) 7C (Ekonomi) 7D (Ekonomi) 7E (Ekonomi) 7F (Ekonomi) 8A (Ekonomi)
8B (Ekonomi) 8C (Ekonomi) 8D (Ekonomi) 8E (Ekonomi) 8F (Ekonomi) 9A (Ekonomi) 9B (Ekonomi) 9C (Ekonomi)
9D (Ekonomi) 9E (Ekonomi) 9F (Ekonomi) 10A (Ekonomi) 10B (Ekonomi) 10C (Ekonomi) 10D (Ekonomi) 10E (Ekonomi)
10F (Ekonomi) 11A (Ekonomi) 11B (Ekonomi) 11C (Ekonomi) 11D (Ekonomi) 11E (Ekonomi) 11F (Ekonomi) 12A (Ekonomi)
12B (Ekonomi) 12C (Ekonomi) 12D (Ekonomi) 12E (Ekonomi) 12F (Ekonomi) 13A (Ekonomi) 13B (Ekonomi) 13C (Ekonomi)
13D (Ekonomi) 13E (Ekonomi) 13F (Ekonomi) 14A (Ekonomi) 14B (Ekonomi) 14C (Ekonomi) 14D (Ekonomi) 14E (Ekonomi)
14F (Ekonomi) 15A (Ekonomi) 15B (Ekonomi) 15C (Ekonomi) 15D (Ekonomi) 15E (Ekonomi) 15F (Ekonomi) 16A (Ekonomi)
16B (Ekonomi) 16C (Ekonomi) 16D (Ekonomi) 16E (Ekonomi) 16F (Ekonomi) 17A (Ekonomi) 17B (Ekonomi) 17C (Ekonomi)
17D (Ekonomi) 17E (Ekonomi) 17F (Ekonomi) 18A (Ekonomi) 18B (Ekonomi) 18C (Ekonomi) 18D (Ekonomi) 18E (Ekonomi)
18F (Ekonomi) 19A (Ekonomi) 19B (Ekonomi) 19C (Ekonomi) 19D (Ekonomi) 19E (Ekonomi) 19F (Ekonomi) 20A (Ekonomi)
20B (Ekonomi) 20C (Ekonomi) 20D (Ekonomi) 20E (Ekonomi) 20F (Ekonomi) 21A (Ekonomi) 21B (Ekonomi) 21C (Ekonomi)
21D (Ekonomi) 21E (Ekonomi) 21F (Ekonomi) 22A (Ekonomi) 22B (Ekonomi) 22C (Ekonomi) 22D (Ekonomi) 22E (Ekonomi)
22F (Ekonomi) 23A (Ekonomi) 23B (Ekonomi) 23C (Ekonomi) 23D (Ekonomi) 23E (Ekonomi) 23F (Ekonomi) 24A (Ekonomi)
24C (Ekonomi) 24D (Ekonomi) 24E (Ekonomi) 24F (Ekonomi) 25A (Ekonomi) 25B (Ekonomi) 25C (Ekonomi) 25D (Ekonomi)
25E (Ekonomi) 25F (Ekonomi) 26A (Ekonomi) 26B (Ekonomi) 26C (Ekonomi) 26D (Ekonomi) 26E (Ekonomi) 26F (Ekonomi)
27A (Ekonomi) 27B (Ekonomi) 27C (Ekonomi) 27D (Ekonomi) 27E (Ekonomi) 27F (Ekonomi) 28A (Ekonomi) 28B (Ekonomi)
28C (Ekonomi) 28D (Ekonomi) 28E (Ekonomi) 28F (Ekonomi) 29A (Ekonomi) 29B (Ekonomi) 29C (Ekonomi) 29D (Ekonomi)
29E (Ekonomi) 29F (Ekonomi) 30A (Ekonomi) 30B (Ekonomi) 30C (Ekonomi) 30D (Ekonomi) 30E (Ekonomi) 30F (Ekonomi)

Koltuk numarası: 27A
Yolcu adı: Ayşe Sude
Yolcu soyadı: Cami
Email: ayse@ayse@gmail.com
Telefon: 5558905895

✓ Rezervasyon başarıyla oluşturuldu!
Rezervasyon Kodu: J8K7RR
Uçuş: TK101
Koltuk: 27A
Yolcu: Ayşe Sude Cami
```

Rezervasyonları görüntüleme:

Seçiminiz: 4					
REZERVASYONLARIM					
Rez. Kodu	Uçuş No	Kalkış	Varış	Koltuk	Durum
ABNXVV	TK103	İstanbul Havalim..	Antalya Havalimanı	6A	CANCELLED
6J5ZW2	TK202	Esenboğa Havalim..	Antalya Havalimanı	9B	CANCELLED
PCBTRT	TK203	İstanbul Havalim..	Adnan Menderes H..	6C	CANCELLED
NSV21E	TK101	İstanbul Havalim..	Esenboğa Havalim..	1A	CONFIRMED
WTP9G0	TK201	İstanbul Havalim..	İzmir Havalimanı	24B	CONFIRMED
50FH8G	TK101	İstanbul Havalim..	Ankara Havalimanı	6C	CONFIRMED
45JXMV	TK101	İstanbul Havalim..	Ankara Havalimanı	6A	CONFIRMED
J8K7RR	TK101	İstanbul Havalim..	Ankara Havalimanı	27A	CONFIRMED

Rezervasyon iptal etme:

```
Seçiminiz: 5

===== REZERVASYON İPTAL =====
Rezervasyon kodu: 45JXMV
İptal etmek istediğinizden emin misiniz? (E/H): E
✓ Rezervasyon başarıyla iptal edildi.
```

Uçuş detayı görüntüleme:

```
Seçiminiz: 6

===== UÇUŞ DETAYI =====
Uçuş numarası: TK101

UÇUŞ BİLGİLERİ

Uçuş No : TK101
Kalkış : İstanbul Havalimanı
Varış : Ankara Havalimanı
Tarih : 09.01.2026 23:59
Uçak : Boeing 737
Kapasite : 180
Boş Koltuk : 176
Fiyat : 500,00
Durum : SCHEDULED
```

Admin girişi yapma:

```
Seçiminiz: 1

===== GİRİŞ =====
Kullanıcı adı: admin
Şifre: admin123

✓ Hoş geldiniz, admin!

===== ANA MENÜ =====
1. Uçuş Ara
2. Tüm Uçuşları Listele
3. Rezervasyon Yap
4. Rezervasyonlarımı Görüntüle
5. Rezervasyon İptal Et
6. Uçuş Detayı Görüntüle
7. [Admin] Tüm Kullanıcıları Listele
8. [Admin] Tüm Rezervasyonları Listele
0. Çıkış
```

Tüm kullanıcıları listeleme:

Seçiminiz: 7		
TÜM KULLANICILAR		
Kullanıcı Adı	Email	Rol
admin	admin@airline.com	ADMIN
asude	asudecmi@gmail.com	CUSTOMER
feryat	std	CUSTOMER