

# ARABA GALERİSİ OTOMASYON SİSTEMİ

## FİNAL RAPORU (FINAL REPORT)

Yazılım İnşaası Dersi

### 1. GİRİŞ

Bu rapor, Java programlama dili kullanılarak geliştirilen **Araba Galeri Otomasyon Sistemi** yazılımının final aşamasını kapsamaktadır. Proje, Yazılım İnşaası dersi kapsamında hazırlanan Analiz Raporu ve oluşturulan Tasarım Raporu temel alınarak gerçekleştirılmıştır.

Final raporunun temel amacı; sistemin gerçekleştirim (implementation) sürecini ayrıntılı biçimde belgelemek, analiz ve tasarım aşamalarında belirlenen hedeflerle ne ölçüde uyum sağlandığını değerlendirmek, yapılan geliştirmeleri ve iyileştirmeleri açıklamak, kullanıcıya sistemin nasıl kullanılacağını göstermek ve mevcut durumu bütüncül bir bakış açısıyla ortaya koymaktır.

Bu kapsamda raporda aşağıdaki konular detaylı olarak ele alınmıştır:

- Gerçekleştirim süreci ve kullanılan araçlar
- Analiz–tasarım uyumunun değerlendirilmesi
- Gerçekleştirilen fonksiyonel özellikler
- Kullanıcı arayüzü ve işlem akışları
- Tasarım kararları ve sapmalar
- Karşılaşılan zorluklar ve çözümleri
- Sistemin eksik yönleri ve gelecekteki geliştirme önerileri

Proje boyunca nesne yönelimli yazılım mühendisliği prensipleri (özellikle **SOLID**) titizlikle uygulanmış; modülerlik, okunabilirlik ve genişletilebilirlik ön planda tutulmuştur.

## 2. GERÇEKLEŞTİRİM SÜRECİ

Araba Galerisi Otomasyon Sistemi, **Java** programlama dili kullanılarak geliştirilmiştir. Uygulama, analiz raporunda belirtilen sistem kısıtlarına (veritabanı kullanılmaması, konsol tabanlı arayüz) tam uyumlu olacak şekilde **CLI (Command Line Interface)** tabanlı olarak kodlanmıştır.

Gerçekleştirim süreci aşağıdaki adımlar izlenerek yürütülmüştür:

### 2.1 Analiz ve Tasarım İncelemesi

Analiz Raporu'nda tanımlanan fonksiyonel ve fonksiyonel olmayan gereksinimler detaylı biçimde incelenmiş; Tasarım Raporu'nda belirlenen **katmanlı mimari (Presentation – Business – Data)** temel alınarak kodlama süreci başlatılmıştır.

### 2.2 Proje Kurulumu ve Versiyon Kontrolü

Proje, GitHub üzerinde oluşturulan ortak bir depo üzerinden yürütülmüştür. Grup üyeleri geliştirme işlemlerini yerel ortamlarında (IntelliJ IDEA) gerçekleştirmiştir; düzenli commit ve push işlemleri ile kod bütünlüğü korunmuştur.

### 2.3 Katmanlı Kodlama Süreci

- Sunum Katmanı:**  
Kullanıcı ile etkileşim, menü gösterimi ve girdi alma işlemleri **KonsolArayuz** sınıfı üzerinden gerçekleştirilmiştir.
- İş Mantığı Katmanı:**  
Araba ekleme, satış işlemi, stok kontrolü ve doğrulamalar **GaleriService** ve kullanıcı girişi için **AuthService** sınıflarında merkezleştirilmiştir.
- Veri Katmanı:**  
Dosya tabanlı veri saklama işlemleri **DosyaService** sınıfı ile yönetilmiştir. Bu sınıf, TXT dosyalarına erişimi merkezi hale getirerek veri bütünlüğünü sağlamaktadır.
- Model Sınıfları:**  
**Araba** ve **Kullanıcı** sınıfları, analiz raporunda tanımlanan alan modeline birebir karşılık gelecek şekilde tasarılmıştır.

## 2.4 Modülerlik ve SRP Uygulaması

Her sınıf, **Single Responsibility Principle (SRP)** doğrultusunda yalnızca tek bir sorumluluğa sahip olacak şekilde kodlanmıştır. Ortak işlemler tek noktada toplanarak kod tekrarından kaçınılmıştır.

## 2.5 Test ve Doğrulama

Her fonksiyon geliştirildikten sonra manuel testler uygulanmıştır. Özellikle araba satış işlemleri, stok güncellemeleri, kullanıcı giriş kontrolleri ve hatalı giriş senaryoları detaylı şekilde test edilmiştir.

## 2.6 Son Düzenlemeler

Konsol çıktıları sade ve anlaşılır hale getirilmiş, kullanıcı mesajları Türkçe ve açıklayıcı şekilde düzenlenmiştir. Menü yapısı kullanıcıyı adım adım yönlendirecek biçimde son haline getirilmiştir.

# 3. ANALİZ VE TASARIM RAPORLARI İLE UYUM

Geliştirilen sistem, analiz ve tasarım raporlarında tanımlanan tüm gereksinimleri eksiksiz şekilde karşılamaktadır.

## 3.1 Fonksiyonel Gereksinimlerin Karşılanması

Gereksinim	Karşılanma Durumu	Açıklama
FR1 – Araba Yönetimi	Tamamen karşılandı	Araba ekleme, listeleme ve stok takibi
FR2 – Kullanıcı Yönetimi	Tamamen karşılandı	Kullanıcı ekleme ve giriş doğrulama
FR3 – Satış İşlemleri	Tamamen karşılandı	Stok kontrolü, satış kaydı oluşturma

FR4 – Satış Kayıtları	Tamamen karşılandı	Satılan arabaların ayrı dosyada tutulması
FR5 – Sistem Kontrolü	Tamamen karşılandı	Menü sistemi, hatalı giriş uyarıları

### 3.2 Tasarım Uyumu

- KonsolArayuz yalnızca kullanıcı etkileşiminden sorumludur.
- GaleriService tüm iş kurallarını merkezileştirir.
- DosyaService veri bütünlüğünü tek noktadan yönetir.
- Model sınıfları analizdeki kavramsal modele birebir uymaktadır.
- Katmanlar arası bağımlılık yönü doğru şekilde kurulmuştur.

## 4. GERÇEKLEŞTİRİLEN SİSTEM ÖZELLİKLERİ

### 4.1 Araba Yönetimi

- Yeni araba ekleme
- Araba listeleme
- Satış sonrası stok güncelleme

### 4.2 Kullanıcı Girişи

- Kullanıcı adı ve şifre doğrulaması
- Yetkisiz girişlerin engellenmesi

### 4.3 Satış İşlemleri

- Stok kontrolü
- Satış kaydı oluşturma
- Satılan araçların ayrı dosyada tutulması

## 5. KULLANICI KILAVUZU

### 5.1 Programın Başlatılması

- Proje klasörüne gidilir
- java Main komutu ile program çalıştırılır

### 5.2 Ana Menü

Ana menüde araba ekleme, listeleme, satış yapma ve çıkış seçenekleri yer almaktadır.

## 6. TASARIMDAN SAPMALAR VE ALINAN KARARLAR

- Veritabanı yerine TXT dosyaları kullanılmıştır
- Grafik arayüz yerine CLI tercih edilmiştir
- Çoklu kullanıcı ve eşzamanlı erişim desteklenmemektedir

## 7. KARŞILAŞILAN ZORLUKLAR VE ÇÖZÜMLER

- Dosya tutarlılığı:** Merkezi DosyaService ile çözüldü
- Stok senkronizasyonu:** Tüm işlemler GaleriService üzerinden yürütüldü
- ID benzersizliği:** Dosya bazlı otomatik ID üretimi uygulandı

Kişi	Sorumlu Olduğu Sınıflar	Katman
Ferhat Demir	GaleriService, AuthService	İş Mantığı
İdris Şeyhun	DosyaService, Araba, Kullanıcı	Veri & Model
Sabri Kaçar	KonsolArayuz, Main	Sunum

