

# Android Tabanlı E-Ticaret Uygulaması (BiDaha)

Ad Soyad: Mehmet Eren Yenioğlu  
Öğrenci Numarası: 241307023  
Üniversite: Kocaeli Üniversitesi  
Bölümü: Bilişim Sistemleri  
Mühendisliği  
Ders: Mobil Uygulama Geliştirme

**Özet** — Bu çalışmada, Android platformu üzerinde çalışan bir mobil e-ticaret uygulaması geliştirilmiştir. Proje, Kotlin programlama dili ve Android Studio geliştirme ortamı kullanılarak hayata geçirilmiştir. Veri depolama ve kimlik doğrulama işlemleri için Firebase Realtime Database ve Firebase Authentication teknolojileri entegre edilmiştir. Uygulama, "Yönetici" (Admin) ve "Kullanıcı" (User) olmak üzere iki farklı rol tabanlı erişim kontrolüne sahiptir. Yöneticiler ürün ve kullanıcı yönetimi yapabilirken, kullanıcılar kendi ürünlerini ekleyebilir, ürünleri inceleyip sipariş verebilmektedir.

## I. GİRİŞ

Günümüzde mobil cihazların yaygınlaşmasıyla birlikte e-ticaret masaüstü platformlardan mobil platformlara kaymıştır. Bu projede, küçük veya orta ölçekli işletmelerin veya bireysel satıcıların ürünlerini kolayca ekleyebileceği, yönetebileceği, müşterilerin ise hızlıca sipariş verebileceği dinamik bir Android uygulaması geliştirilmesi amaçlanmıştır.

Uygulama, kullanıcıların güvenli bir şekilde kayıt olup giriş yapmasını, ürünlerini satışa sunmasını, başkalarının ürünlerini arayüzde görüntülemesini kendi ürünlerini düzenlemelerini ve sepet mantığıyla sipariş oluşturmalarını sağlamaktadır. Aynı zamanda yönetici yetkisine sahip kullanıcılar, panel üzerinden ürün silme, güncelleme işlemlerini, kullanıcıların hesaplarını dondurabilme veya silme işlemini gerçekleştirebilmekte ve gelen siparişlerin durumunu (Onaylandı/Reddedildi) değiştirebilmektedir.

## II. YÖNTEM

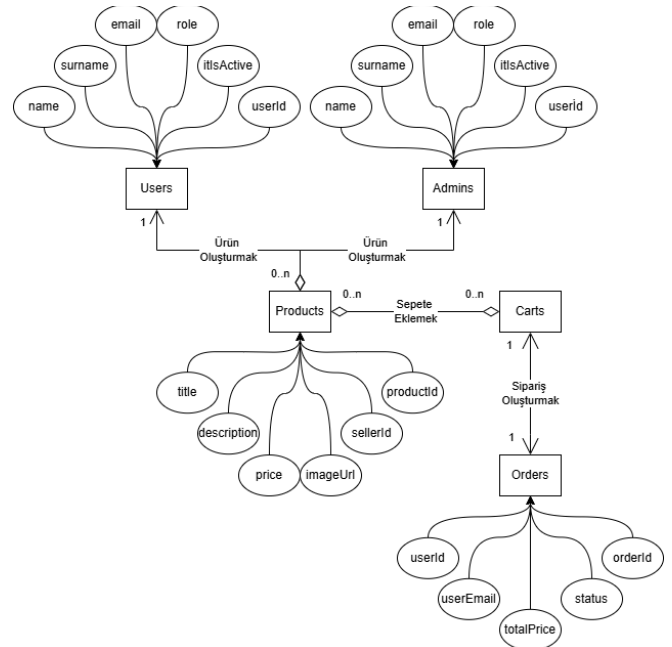
### A. Sistem Mimarisi ve Araçlar

Uygulama geliştirme sürecinde Android Studio IDE ve Kotlin programlama dili tercih edilmiştir. Arayüz tasarımı XML kullanılmıştır. Veri tabanı olarak NoSQL yapısına sahip Firebase Realtime Database kullanılmıştır. Bu seçim, verilerin anlık olarak (real-time) senkronize edilmesini sağlamıştır. Kullanıcıların kimliklerini doğrulaması için Firebase Authentication

kullanılmıştır. Bu sayede kullanıcıların verileri bulut sistemi ile güvenli bir şekilde korunmuş, kullanıcıların hesap oluşturmaları ve hesaplarına giriş yapmaları sağlanmıştır.

### B. Veritabanı Tasarımı

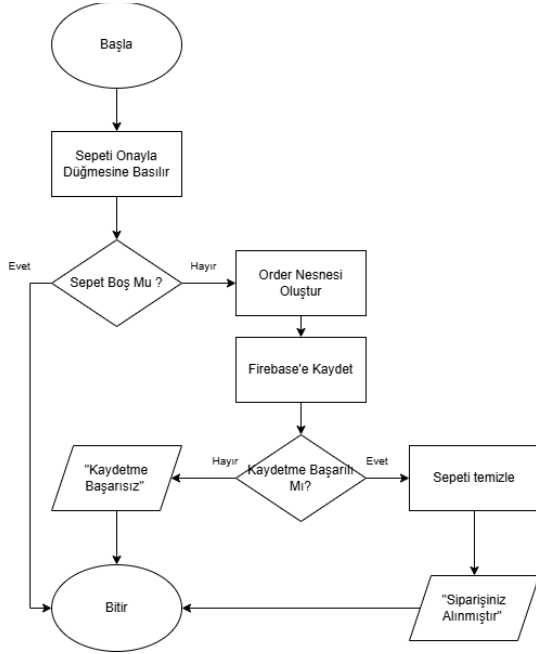
Projede ilişkisel veri bütünlüğünü sağlamak adına "Users", "Admins", "Products", "Carts" ve "Orders" olmak üzere beş ana düğüm tasarlanmıştır. "Users" ve "Admins" kullanıcıların isim, soyisim, email, kullanıcı id, hesabın aktif olup olmadığı ve rol bilgilerini, "Products" ürünlerin açıklaması, resmi, satıcı id, ürün id ve fiyat detaylarını, "Carts" hangi kullanıcıların hangi ürünleri sepetine eklediğini, "Orders" ise sipariş veren kullanıcıların eposta ve idsi, siparişlerin toplam fiyatını ve siparişlerin durumunu tutmaktadır.



Şekil 1: Veritabanı ERD Diyagramı

### C. Algoritma ve İş Akışı

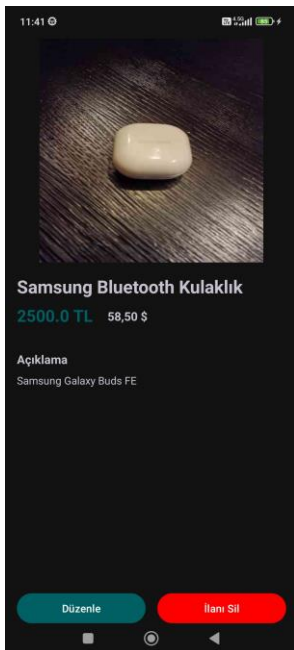
Kullanıcı "Sepeti Onayla" butonuna bastığında tetiklenen süreç Şekil 2'de gösterilmiştir.



Şekil 2: Sipariş Oluşturma Akış Şeması

### D. API Entegrasyonu

Uygulamaya Retrofit 2 kütüphanesi ile, anlık kur bilgilerini ücretsiz ve açık kaynaklı bir servis olan *ExchangeRate-API* sayesinde anlık döviz kuru takibi özelliği entegre edilmiştir. Bu özellik sayesinde, kullanıcı ürün detay sayfasına girdiğinde, uygulama API'ye istek atar. Gelen yanıtta "1 TL = X USD" dönüşüm oranı çekilir ve matematiksel işlem yapılarak ürünün Dolar cinsinden fiyatı dinamik olarak TextView üzerinde gösterilir.



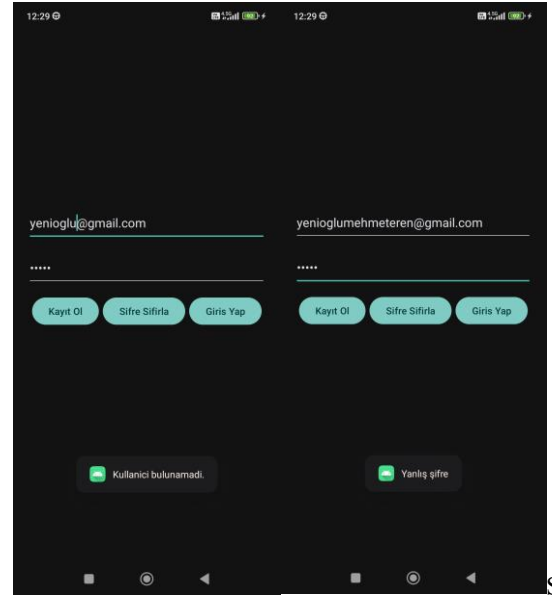
Şekil 3: Ürün Detay Ekranı

### III. DENEYSEL SONUÇLAR

Geliştirilen uygulama Android emülatörlerde (Medium Phone, Pixel 9 Pro) ve fiziksel cihazlarda (Xiaomi) test edilmiştir.

#### A. Kimlik Doğrulama

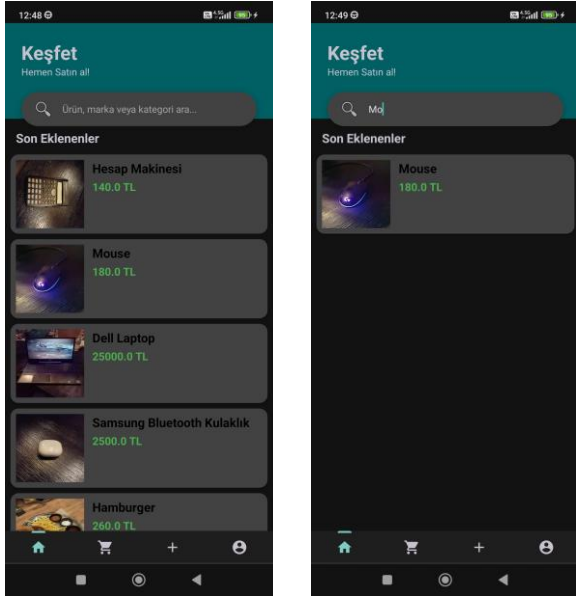
Uygulamanın giriş modülü, Firebase Authentication servisi ile sağlanmıştır. Yapılan testlerde, hatalı e-posta formatı veya eksik şifre girilmesi durumunda sistemin kullanıcıyı uyardığı ve girişi engellediği doğrulanmıştır.



Şekil 3 ve 4: Giriş Ekranı ve Hatalı Girilen Bilgiler

#### B. Ürün Listeleme

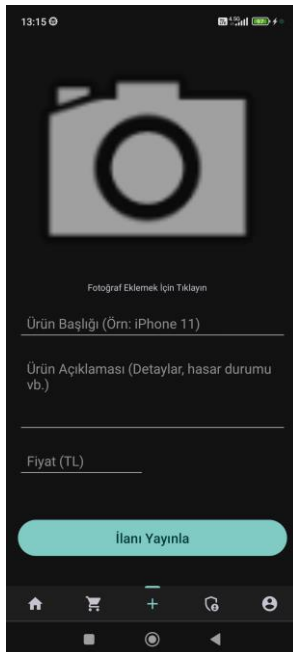
Kullanıcılar tarafından giriş yapıldığında, anasayfada ürünlerin RecyclerView bileşeni kullanılarak listelendiği "Keşfet" ekranı görüntülenmektedir. Arama çubuğu kullanılarak ürünlerin isimlerine göre arama gerçekleştirilebilir. Şekil 5'de görüldüğü üzere, ürünler kart yapısında, fiyat, isim ve resim ile birlikte kullanıcıya sunulmaktadır.



Şekil 5 ve 6: Keşfet Ekranı

### C. Ürün Ekleme

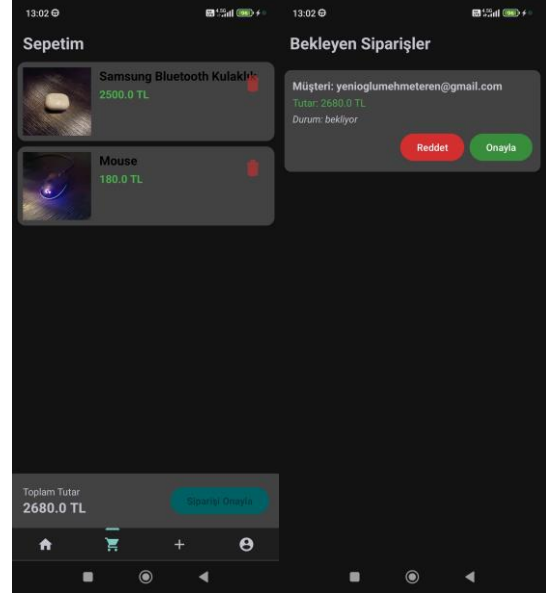
Yeni ürün oluşturma arayüzü, metinsel verilerin yanı sıra görsel medya yönetimini de destekleyecek şekilde tasarlanmıştır. Bu ekranda kullanıcı; ürün adı, fiyatı ve ürün açıklaması gibi temel verileri EditText bileşenleri aracılığıyla sisteme girmektedir. Uygulamanın en önemli yeteneklerinden biri cihazın yerel depolama birimiyle etkileşime girebilmesidir. Kullanıcı ürün görseli alanına tıkladığında telefon hafızasından dilediği fotoğrafı seçebilir, ilgili görselin URI adresi uygulama tarafından yakalanmakta ve önizleme olarak ekrana yansıtılmaktadır. "Kaydet" butonuna basıldığında, girilen tüm metinsel veriler ve seçilen görsel Firebase Realtime Database üzerindeki Products düğümüne kaydedilmektedir.



Şekil 7: Ürün Ekleme Arayüzü

### D. Sepet Yönetimi

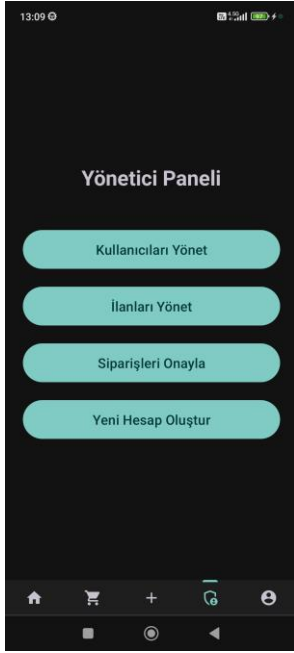
Uygulamanın e-ticaret fonksiyonlarının testinde, kullanıcıların ürünleri sepete ekleyip çıkartabildiği ve anlık olarak toplam tutarın hesaplandığı görülmüştür. "Sipariş Ver" butonuna tıkladığında sistem siparişi yöneticilerin onay vermesi için beklettiği görülmüştür.



Şekil 8 ve 9: Sepet Ekranı ve Onay Ekranı

### E. Yönetici Paneli

Admin yetkisine sahip bir hesapla giriş yapıldığında, alt navigasyon çubuğunda standart kullanıcılarda görünmeyen "Admin Paneli" sekmesi aktif hale gelmektedir. Bu panel üzerinden kullanıcı hesaplarını yönetme, ilanları yönetme, yeni kullanıcı/yönetici hesabı oluşturma ve bekleyen siparişleri onaylama/reddetme işlemleri başarıyla gerçekleştirilmiştir.



Şekil 10: Yönetici Paneli

#### IV. SONUÇ

Bu proje çalışması kapsamında, mobil yazılım geliştirme ekosisteminin temelleri atılmış; kullanıcı deneyimi tasarımı, üçüncü parti servis entegrasyonları, rol tabanlı yetkilendirme ve donanım/medya erişimi konularında kapsamlı yetkinlikler kazanılmıştır.

Geliştirme sürecinde karşılaşılan en önemli zorluk, veri depolama yöntemi seçiminde yaşanmıştır. Projenin başlangıç aşamasında, kullanıcı verilerini saklamak için Android'in yerel depolama birimi olan **SharedPreferences** yapısı kullanılmıştır. Ancak SharedPreferences'ın verileri sadece yerel cihazda (Local) tuttuğu test aşamasında fark edilmiştir. Projenin çok kullanıcı ve bulut tabanlı yapısı gereği verilerin merkezi bir sunucuda tutulması zorunluluğu doğduğundan, Firebase Realtime Database çözümüne geçiş yapılmıştır. Bu süreç zaman kaybı oluşturmuştur.

Bunun yanı sıra, Firebase entegrasyonu sırasında ağ işlemlerinden kaynaklı asenkron veri çekme sorunları yaşanmıştır. Verilerin internet hızına bağlı olarak geç gelmesi, arayüzde donmalara veya uygulamada çökmelere neden olmuştur. Bu problem, Firebase platformunun sağladığı `addOnSuccessListener` mekanizması kullanılarak çözülmüştür.

GitHub bağlantısı:

<https://github.com/YeniogluMehmetEren/BiDahaUygulaması>

GoogleDrive bağlantısı:

<https://drive.google.com/drive/folders/1kXmIbe8RWCSuAiiEbcW7-pSRn-NaReF-?usp=sharing>

#### V. KAYNAKÇA

- [1] Android Developers, "Button - Material Design 3 (Compose)," *Android Developers Documentation*, 2025. [Çevrimiçi]. Erişim: <https://developer.android.com/develop/ui/compose/components/button?hl=tr>
- [2] Android Developers, "Uygulamanızda hata ayıklama (Debug your app)," *Android Studio User Guide*, 2025. [Çevrimiçi]. Erişim: <https://developer.android.com/studio/debug?hl=tr>
- [3] Android Developers, "Uygulamanızın kullanıcı arayüzü kaynaklarını yönetme (Resource Manager)," *Android Studio User Guide*, 2025. [Çevrimiçi]. Erişim: <https://developer.android.com/studio/write/resource-manager?hl=tr#import>
- [4] Indently, "Saving data with Shared Preferences in Android Studio (Kotlin 2020)", YouTube, 2020. [Çevrimiçi]. Erişim: <https://www.youtube.com/watch?v=S5uLAGnBvUY>
- [5] Kotlin, "MutableList - Kotlin Standard Library," *Kotlin API Reference*, 2025. [Çevrimiçi]. Erişim: <https://kotlinlang.org/api/core/kotlin-stdlib/kotlin.collections/-mutable-list>
- [6] Insane Developer, "Modern Tab Layout With Fragments In Kotlin | Android UI", YouTube, 2020. [Çevrimiçi]. Erişim: <https://www.youtube.com/watch?v=LoSf9vHRnXg&t=827s>
- [7] Kotlin, "AlertDialog - Androidx.compose.material3," *Kotlin API Reference*, 2025. [Çevrimiçi]. Erişim: <https://kotlinlang.org/api/compose-multiplatform/material3/androidx.compose.material3/-alert-dialog.html>
- [8] Google Firebase, "Android'de Firebase Realtime Database'i Kullanmaya Başlama," *Firebase Documentation*, 2025. [Çevrimiçi]. Erişim: <https://firebase.google.com/docs/database/android/start?hl=tr>
- [9] Google Firebase, "Android'de Parolalarla Firebase Kimlik Doğrulaması (Password Auth)," *Firebase Documentation*, 2025. [Çevrimiçi]. Erişim: <https://firebase.google.com/docs/auth/android/password-auth?hl=tr>
- [10] Google Firebase, "Kullanıcıları Yönetme (Android) - Profili Güncelleme," *Firebase Documentation*, 2025. [Çevrimiçi]. Erişim: [https://firebase.google.com/docs/auth/android/manage-users?hl=tr#update\\_a\\_users\\_profile](https://firebase.google.com/docs/auth/android/manage-users?hl=tr#update_a_users_profile)

