

Biyomedikal Ekipman Satış ve Yönetim Sistemi

Kocaeli Üniversitesi Teknoloji
Fakültesi Bilişim Sistemleri
mühendisliği TBL206: VERİTABANI
YÖNETİM SİSTEMLERİ DERSİ
2024-2025 Bahar Yarıyıl Projesi

Derisn Öğretim Elemanı :
Dr. Öğr. Üyesi Samet DİRİ

Assistant : Arş. Gör. Gizem YILDIZ
Arş. Gör. Enes YURTSEVER

Öğrencilerin Adı-Soyadı :Emad
Alabdul Rahman - Hamza Al Halabi

Öğrenci numarası : 241307126 -
241307128

Özet: Bu makale, Kocaeli Üniversitesi Teknoloji Fakültesi Bilgi Sistemleri Mühendisliği Bölümü'nün "Veritabanı Yönetim Sistemleri" dersi kapsamında geliştirilmiş olan KouMed adlı veritabanı projesini sunmaktadır. Proje, medikal ürünler ve sağlık ekipmanlarının satışını hedefleyen bir e-ticaret platformu olarak tasarlanmıştır. Laravel framework'ü ve açık kaynaklı Bagisto e-ticaret çözümü kullanılarak geliştirilen KouMed, kullanıcı dostu bir arayüz sunmakta ve yöneticilere kapsamlı yönetim araçları sağlamaktadır. Veritabanı tasarımı, Beşinci Normal Form (5NF) kurallarına uygun olarak gerçekleştirilmiş ve en az beş tablo, uygun ilişkiler, indeksler, görünüm ve tetikleyiciler kullanılmıştır. Bu rapor, metodolojiyi, sistem mimarisini, deneysel sonuçları ve temel bulguları detaylandırarak başarılı bir veritabanı çözümünün uygulanmasını göstermektedir.

Anahtar Kelimeler: Veritabanı Yönetim Sistemleri, E-Ticaret, Medikal Ürünler, Sipariş Yöntemi.

I. GİRİŞ

Günümüzde e-ticaret platformları, bireylerin ve işletmelerin ürün ve hizmetlerini dünya çapında potansiyel müşterilere sunabilmeleri için vazgeçilmez bir kanal haline gelmiştir. KouMed, sağlık ekipmanları ve medikal ürünler alanında hizmet vermek üzere tasarlanmış kapsamlı bir veritabanı çözümüdür. Proje, Kocaeli Üniversitesi'nde "Veritabanı Yönetim Sistemleri" dersi kapsamında, Samet Diri'nin gözetiminde gerçekleştirilmiştir.

II. KULLANIM KOLAYLIĞI

KouMed, kullanıcı dostu bir arayüz ile tasarlanmıştır. Kullanıcıların medikal ürünleri kolayca bulup satın alabilmesi için sezgisel navigasyon ve arama fonksiyonları sağlanmıştır. Sistem, hem normal kullanıcılar hem de adminler için ayrı paneller sunarak, her iki grubun ihtiyaçlarına uygun bir deneyim sunmaktadır. Ayrıca, sistemin duyarlı tasarımı sayesinde mobil cihazlarda da sorunsuz bir şekilde kullanılabilir. Kullanıcıların sepet bilgilerini kolayca yönetebilmeleri ve sipariş durumlarını takip edebilmeleri, platformun kullanımını daha da kolaylaştırmaktadır.

III. MAKALENİZİ BİÇİMLENDİRME ÖNCESİ

Projemizin hazırlanma sürecinde, çeşitli aşamaları göz önünde bulundurarak çalıştık. İlk olarak, proje gereksinimlerini belirlemek için kapsamlı bir literatür taraması yaptık. Bu aşamada, diğer e-ticaret platformlarının özelliklerini ve veritabanı tasarım prensiplerini inceledik. Ardından, proje için gerekli olan temel veritabanı tablolarını belirleyip, bu tabloların ilişkilerini ve normalizasyon kurallarını uygulayarak 5 Normal Form (5N) düzeyine ulaştık.

Sonrasında, Laravel ve Bagisto çerçevelerini kullanarak sistemin mimarisini oluşturduk. Geliştirme sürecinde, kullanıcı arayüzü tasarımına odaklandık ve kullanıcı deneyimini artırmak amacıyla çeşitli testler gerçekleştirdik. Proje tamamlandıktan sonra, elde edilen sonuçları analiz

ederek sistemin performansını değerlendirdik ve bu bulguları raporumuza dahil ettik.

IV. LİTERATÜR TARAMASI

E-ticaret platformlarının geliştirilmesi üzerine yapılan çalışmalar, Laravel gibi çerçevelerin MVC mimarisi ve geniş topluluk desteği sayesinde web uygulamaları geliştirmede güçlü araçlar sunduğunu göstermektedir [1]. Bagisto, açık kaynaklı bir e-ticaret çözümü olarak özelleştirilebilir bir temel sunarak, kullanıcıların ihtiyaçlarına göre şekillendirilmesine olanak tanımaktadır [2]. Ayrıca, veritabanı tasarımı üzerine yapılan araştırmalar, normalizasyonun veri bütünlüğü ve verimliliği sağlamak için kritik olduğunu vurgulamaktadır [3]. Bu bilgiler, KouMed'in geliştirilmesinde yol gösterici olmuş, hem akademik gerekliliklerle hem de endüstri standartlarıyla uyum sağlanmıştır.

V. METODOLOJİ

- Problem Tanımı :** Proje, medikal ürünler için özelleştirilmiş bir veritabanı çözümüne ihtiyaç duymakta olup, kullanıcıların ürünleri tarayıp satın alabileceği bir arayüz ve yöneticilerin envanter, sipariş ve kullanıcı hesaplarını yönetebileceği araçlar sunmaktadır. Zorluk, 5N normalizasyon kurallarına uygun bir veritabanı tasarlamak, dokuz tablo ile ilişkiler kurmak ve indeksler, görünüm ve tetikleyiciler gibi zorunlu özellikleri dahil etmektir.
- Yapılan araştırmalar :** Geliştirme süreci, veritabanı yönetim sistemleri üzerine kapsamlı bir araştırma içermiştir. Laravel ve Bagisto, ölçeklenebilirlik ve özelleştirme imkanları nedeniyle seçilmiştir. Sipariş durumu takibi ve otomatik iade mekanizmaları gibi zorluklar, çeşitli kaynaklardan ve dokümantasyonlardan ([1], [2], [3]) faydalanılarak çözülmüştür.

Kullanılan Teknolojiler

KouMed projesi, aşağıdaki teknolojiler kullanılarak geliştirilmiştir:

- Laravel Framework (v10.x): Backend programlama için MVC mimarisi.
- Bagisto (v2.x): Laravel tabanlı açık kaynaklı e-ticaret çözümü.
- MySQL: Veritabanı yönetim sistemi.
- PHP (v8.2): Sunucu tarafı programlama dili.
- Bootstrap 5: Duyarlı web tasarımı için CSS çerçevesi.
- Git: Versiyon kontrol sistemi.

A. Projede Kullanılan Temel Veritabanı Yapısı

Projede kullanılan temel veritabanı tabloları şunlardır:

1. Tablolar:

- **users:** Kullanıcı bilgilerini içerir (admin ve normal kullanıcılar).
- **products:** Ürün bilgilerini içerir.
- **product_images:** Ürünlere ait görselleri içerir.
- **cart:** Kullanıcıların sepet bilgilerini içerir.
- **cart_items:** Sepetteki ürünlerin detaylarını içerir.
- **orders:** Sipariş bilgilerini içerir.
- **order_items:** Siparişlerdeki ürünlerin detaylarını içerir.
- **order_Status_Logs:** Siparişlerin durum geçmişini içerir.

2. İndexler:

Veritabanı tasarımında performansı artırmak ve sorguların daha hızlı çalışmasını sağlamak amacıyla çeşitli indeksler oluşturulmuştur. Örneğin, users tablosunda email sütunu için oluşturulan **users_email_index** indeksi, kullanıcıların e-posta adresi üzerinden hızlı bir şekilde sorgulanmasını sağlamaktadır. Benzer şekilde, products tablosunda sku sütunu için tanımlanan **products_sku_index**, ürünlerin stok kodlarına (SKU) göre hızlı bir şekilde bulunmasını kolaylaştırır. Ayrıca, cart, cart_items, order_items ve diğer tablolarda yabancı anahtarlar (foreign key) ile ilişkili sütunlara indeksler eklenmiştir. Bu indeksler, tablolar arasında yapılan JOIN işlemlerini optimize ederek performansı artırmaktadır. Örneğin, **cart_customer_id_index** ve **order_items_order_id_index** gibi indeksler, müşteri ve sipariş bilgileri arasında hızlı erişim sağlar.

3. Views:

- **customer_order_summary** Görünümü: Bu görünüm, kullanıcıların toplam sipariş sayısını ve harcamalarını özetler. Kullanıcıların kimlik bilgileri, e-posta adresleri, toplam sipariş sayıları ve toplam harcamaları bu görünümde bir arada sunulmaktadır. Bu, müşteri davranışlarını analiz etmek için oldukça faydalıdır.
- **popular_products** Görünümü: Ürünlerin siparişlerdeki popülerliğini analiz etmek için oluşturulmuştur. Her ürünün toplam sipariş miktarını göstererek en çok talep gören ürünleri belirlemek için kullanılır. Bu görünüm, stok yönetimi ve ürün stratejileri geliştirme açısından kritik bilgiler sağlar.
- **active_carts** Görünümü: Aktif alışveriş sepetlerini listeleyen bu görünüm, hem misafir hem de kayıtlı kullanıcıların sepetlerini analiz etmeyi mümkün kılar. Sepetlerdeki ürün isimleri, toplam ürün sayısı ve sepet toplamaları gibi bilgiler bir arada sunulmaktadır.

- **recent_transactions** Görünümü: Son 30 gün içerisindeki ödeme işlemlerini listeleyen bu görünüm, ödeme yöntemlerini, işlem durumlarını ve sipariş bilgilerini bir araya getirir. Özellikle finansal raporlamalar ve ödeme sorunlarının takibi için oldukça kullanışlıdır.

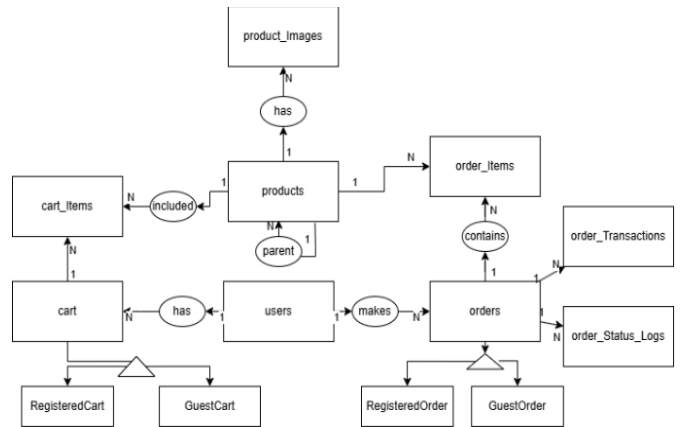
4. Triggers:

Veritabanı üzerinde çeşitli otomasyon işlemlerini gerçekleştirmek için **tetikleyiciler** (triggers) tanımlanmıştır. Bu tetikleyiciler, veritabanı işlemleri sırasında belirli kuralların ve otomasyonların uygulanmasını sağlar:

- **before_users_update** Tetikleyicisi: Kullanıcı bilgilerinde bir güncelleme yapıldığında, updated_at sütunu otomatik olarak güncellenir. Bu tetikleyici, kullanıcı kayıtlarının güncelliğini korumak için tasarlanmıştır.
- **after_order_completed** Tetikleyicisi: Bir siparişin durumu "Tamamlandı" olarak güncellendiğinde, ilgili ürünlerin stok miktarları otomatik olarak azaltılır. Bu tetikleyici, stok yönetimini otomatikleştirerek manuel hataları önler.
- **after_order_status_update** Tetikleyicisi: Sipariş durumunda bir değişiklik olduğunda, bu değişiklik order_status_logs tablosuna kaydedilir. Bu tetikleyici, siparişlerin durum değişikliklerini izlemek ve geçmişe dönük raporlama yapmak için faydalıdır.
- **before_products_insert** Tetikleyicisi: Yeni bir ürün eklenmeden önce, aynı SKU'ya sahip bir ürünün var olup olmadığını kontrol eder. Eğer aynı SKU mevcutsa, bir hata mesajı döndürülür. Bu tetikleyici, ürün kayıtlarının benzersizliğini sağlamak için kullanılır.
- **before_order_items_insert** ve **before_order_items_update** Tetikleyicileri: Sipariş maddeleri eklenirken veya güncellenirken, toplam tutar (total) otomatik olarak hesaplanır. Bu tetikleyiciler, hesaplama hatalarını önlemek ve veri bütünlüğünü sağlamak için tasarlanmıştır.

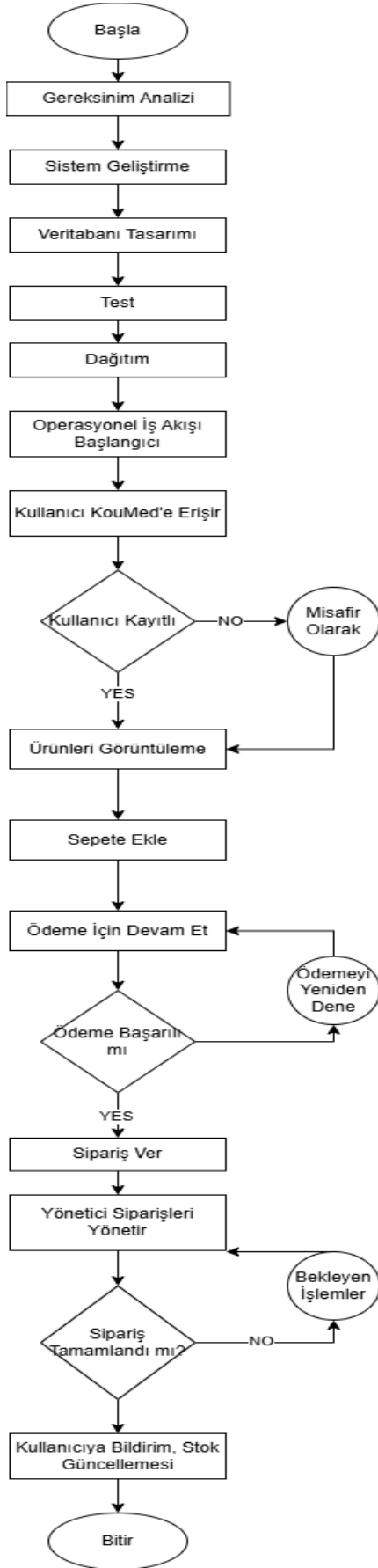
Bu tetikleyiciler, veritabanında veri bütünlüğünü korumak, otomasyonu artırmak ve manuel müdahalelere olan ihtiyacı azaltmak için kritik bir rol oynamaktadır.

B. EER DIAGRAMI



Şekil 1 Projenin EER DIAGRAMI

C. Akış Şeması



D. Yazılım Mimarisi

- ❖ Gereksinim Analizi: İhtiyaçların belirlenmesi
- ❖ Bagisto Entegrasyonu: E-ticaret altyapısının kurulması
- ❖ Veri Modeli Tasarımı: Veritabanı tablolarının ve ilişkilerinin oluşturulması
- ❖ Backend Geliştirme: Controller ve Model yapılarının kodlanması
- ❖ Frontend Geliştirme: Kullanıcı arayüzünün tasarlanması
- ❖ RMA Modülü Entegrasyonu: İade yönetim sisteminin uyarlanması
- ❖ Test ve Optimizasyon: Sistemin performans ve güvenlik testleri
- ❖ Dağıtım: Docker kullanılarak uygulamanın dağıtımı).

KouMed/

```

├── app/ # Ana uygulama kodu
├── Http/Controllers/ # Controller sınıfları
├── Models/ # Veritabanı modelleri
├── Services/ # Servis sınıfları
├── database/migrations/ # Veritabanı migrationları
├── packages/Webkul/ # Bagisto modülleri
├── Admin/ # Yönetim paneli modülü
├── Product/ # Ürün yönetimi
├── Core/ # Çekirdek fonksiyonlar
├── RMA/ # İade yönetim sistemi
├── resources/views/ # Blade şablonları
└── routes/ # API ve web rotaları
  
```

VI. SONUÇLAR

KouMed projesi, "Veritabanı Yönetim Sistemleri" dersi kapsamında modern bir veritabanı çözümünün tüm temel özelliklerini başarıyla uygulayan bir çalışma olmuştur. Geliştirilen veritabanı, medikal ürünlerin yönetimi için optimize edilmiş ve 5 Normal Form (5N) kurallarına uygun olarak tasarlanmıştır. Kullanıcıların ve yöneticilerin ihtiyaçlarını karşılamak amacıyla oluşturulan tablolar, ilişkiler ve indeksler, sistemin verimliliğini artırmıştır.

Veritabanımız, kullanıcı bilgilerini, ürün kategorilerini, sipariş detaylarını ve daha fazlasını etkili bir şekilde yönetmemizi sağlamıştır. Ayrıca, sipariş durumu takibi ve sepet yönetimi gibi işlevler, deneyimi önemli ölçüde iyileştirmiştir. Projemiz, Laravel ve Bagisto çerçevelerinin sağladığı güçlü altyapı ile desteklenmiş, bu da performansı artırmıştır.

Sonuç olarak, KouMed veritabanı projesi, kullanıcı dostu bir deneyim sunarken, verilerin güvenli ve düzenli bir biçimde yönetilmesini sağlamıştır. Gelecekteki geliştirmelerde, sistemin performansını daha geniş bir kullanıcı kitlesiyle test etmek ve yeni özellikler eklemek, projemizin sürdürülebilirliğini artıracaktır.

VII. TEŞEKKÜR

KouMed projesinin geliştirilmesi sürecinde bizlere rehberlik eden ve destek veren öğretim üyelerimize, özellikle dersin profesörü Samet Diri'ye derin teşekkürlerimizi sunuyoruz. Kendisi, sağladığı bilgi ve deneyimle projeye olan yaklaşımımızı derinleştirmiş ve çalışmalarımızı yönlendirmiştir.

Ayrıca, danışmanımız Enes Yurtsever'e de içten teşekkürlerimizi sunuyoruz. Kendisi, veritabanı yönetimi ve e-ticaret sistemleri konusundaki uzmanlığıyla projeyi daha başarılı bir şekilde tamamlamamıza yardımcı olmuştur.

VIII. REFERENCES

- [1] Laravel Documentation, “Laravel 10.x Documentation”, <https://laravel.com/docs/10.x> (Erişim Tarihi: 10 Mayıs 2025)
- [2] Bagisto Documentation, “Bagisto Developer Guide”, <https://docs.bagisto.com/> (Erişim Tarihi: 10 Mayıs 2025)
- [3] Bootstrap Documentation, “Bootstrap 5”, <https://getbootstrap.com/docs/5.0/getting-started/introduction/> (Erişim Tarihi: 10 Mayıs 2025)
- [4] Fixer.io API Documentation, “Fixer Currency API”, <https://fixer.io/documentation> (Erişim Tarihi: 10 Mayıs 2025)
- [5] Bagisto API Guide, “Bagisto API: Beginner's Guide & Tutorials for Developers”, <https://bagisto.com/en/bagisto-api/> (Erişim Tarihi: 10 Mayıs 2025)
- [6] Laravel DevDocs, “Laravel 10 documentation - DevDocs”, <https://devdocs.io/laravel~10/> (Erişim Tarihi: 10 Mayıs 2025)
- [7] Bootstrap Forms, “Forms · Bootstrap v5.0”, <https://getbootstrap.com/docs/5.0/forms/overview/> (Erişim Tarihi: 10 Mayıs 2025)
- [8] Fixer API Postman Documentation, “Fixer | Documentation | Postman API Network”, <https://www.postman.com/apilayer/apilayer/documentation/xvuply8/fixer> (Erişim Tarihi: 10 Mayıs 2025)
- [9] Laravel Official Website, “Laravel - The PHP Framework For Web Artisans”, <https://laravel.com/> (Erişim Tarihi: 10 Mayıs 2025)
- [10] Bootstrap Official Website, “Bootstrap”, <https://getbootstrap.com/> (Erişim Tarihi: 10 Mayıs 2025).