

**T.C.**

**FATİH SULTAN MEHMET VAKIF ÜNİVERSİTESİ**

**Mühendislik Fakültesi**

**Bilgisayar Mühendisliği Bölümü**

**Yazılım Proje Yönetimi**

**Vize**

**E-BANKACILIK UYGULAMASI**

İrem ERÇEL, Demet DEMİR, Furkan KARADUT

1921221038, 1921221024, 1821221043

Proje Danışmanı: Doç. Dr. Ferzat ANKA

İstanbul, [Mayıs, 2024]

**İÇİNDEKİLER**

ÖZETiii

ABSTRACTiv

ÖNSÖZv

BİRİNCİ BÖLÜM1

1. PROJENİN TANIMI1

1.1 Projenin Konusu1

1.2 Motivasyon ve Amaç 1

1.3 Misyon ve Vizyon 2

1.4 Proje Kapsamı 3

1.5 Projeksiyon 4

1.6 Yol Haritası 4

İKİNCİ BÖLÜM6

2. BENZER ÇALIŞMALAR6

2.1 Revolut6

2.2 Monzo 6

2.3 Chime 6

2.4 Simple 6

2.5 Mint 6

ÜÇÜNCÜ BÖLÜM8

3. KULLANILAN TEKNOLOJİLER8

3.1 Android Studio8

3.2 Java8

3.3 SQLite Database8

3.4 JSON8

3.5 Android Emulator 8

DÖRDÜNCÜ BÖLÜM8

4. ORGANİZASYON ŞEMASI ve GÖREV LİSTELERİ8

4.1 Organizasyon Şeması8

4.2 Görev Listeleri9

BEŞİNCİ BÖLÜM10

5. İŞLEVSEL ve KALİTE GEREKSİNİMLERİ10

5.1 İşlevsel ve Kalite Gereksinimleri10

ALTINCI BÖLÜM10

**6. SENARYOLAR**

6.1 Her bir Usecase için Oluşturulan Senaryo

**7. TABLOLAR VE DİAGRAMLAR……………………………………………………….**

7.1 Use Case Diagram

7.2 Sequence Diagram

7.3 SWOT Tablosu

7.4 Risk Tablosu

7.5 Gelir Gider Modelleri

7.6 Ms Project Management

8. KAYNAKÇA11

**ÖZET**

Android Studio kullanarak geliştireceğimiz Bankacılık uygulaması projemiz MVC yaklaşımıyla geliştirilecek olup belgelendirme, hata yönetimi, kapsamlı program yapısı ve bellek verimliliği gibi uygun programlama kurallarına uygun olacaktır. Kullanıcılar, oturum açma ekranından mevcut bir profil ile giriş yapabilecek veya yeni bir profil oluşturabileceklerdir. Oturum açtıktan sonra, kullanıcılardan ilk hesaplarını oluşturmaları istenecektir. Uygulama, kullanıcılara Kontrol Paneli, Hesap Genel Bakışı, İşlemler, Mevduatlar, Ödemeler, Transferler, Profil Ayarları ve Oturumu Kapatma gibi seçenekleri içeren bir menü sunacaktır. Bunlar sayesinde kullanıcı bankaya gitmeden işlemlerini geliştirdiğimiz bu uygulama ile kolaylıkla gerçekleştirebileceklerdir.

**ABSTRACT**

We are undertaking a project to develop a banking application for Android, utilizing Android Studio. The application will follow the Model-View-Controller (MVC) architectural pattern, ensuring a well-organized and maintainable codebase. Our development approach will prioritize adherence to best practices, including comprehensive documentation, robust error/exception handling, a clear and modular program structure, and efficient memory management techniques. The application will provide users with the ability to either log in with an existing profile or create a new one directly from the login screen. Upon successful authentication, users will be directed to a Control Panel page where they will be prompted to set up their initial account. This process will guide users through creating their first account within the banking system. Furthermore, the application will boast a dynamic menu accessible via a side navigation panel. This menu will offer a range of features and options catering to the user's banking needs. Serving as the central hub for user interaction and account management. Overall, our goal is to deliver a robust and user-friendly banking application that adheres to industry standards and offers a seamless and secure banking experience for Android users.

**ÖNSÖZ**

FSMVÜ Mühendislik Fakültesi Bilgisayar Mühendisliği Yazılım Proje Yönetimi Dersi bünyesinde kullanıcıların bankacılık deneyimlerini arttırmak amacıyla bir mobil uygulama geliştirilmesi hedeflenmiştir.

Demet DEMİR

İrem ERÇEL

Furkan KARADUT

Mayıs 2024

**BİRİNCİ BÖLÜM**

**1. PROJE TANIMI**

**1.1 Projenin Konusu**

Bu projede geliştirilen uygulama, kullanıcılar ve bankalar arasında moderatör görevi görecektir. Kullanıcılar, fiziksel olarak bankaya gitmelerine gerek kalmadan hesaplarını oluşturabilir ve yönetebilir, şifreleri ayarlayabilir veya değiştirebilir ve banka olanaklarından yararlanabilirler. İnternet üzerinden para transferi yapabilirler ve işlemleri uygulamaya kaydedilecektir.

Android Studio kullanılarak geliştirilen bu uygulamada, belgelendirme, hata/istisna yönetimi, kapsamlı program yapısı ve bellek verimliliği dahil olmak üzere uygun programlama kuralları kullanılarak bir MVC yaklaşımı kullanılarak geliştirilecektir.

**1.2 Motivasyon ve Amaç**

Bu uygulamanın geliştirilme motivasyonu, kullanıcılar ile bankalar arasındaki köprüyü oluşturarak finansal etkileşimler için kullanıcı dostu ve verimli bir platform sağlamaktır. Uygulama, kullanıcılar ile bankalar arasında bir aracı olarak hizmet ederek, kullanıcılara hesaplarını yönetme, işlemler gerçekleştirme ve bankacılık hizmetlerine erişme konusunda banka şubesine fiziksel olarak gitme ihtiyacını ortadan kaldırmayı amaçlamaktadır.

Uygulamanın temel amacı, kullanıcılar için erişilebilirlik ve kolaylık sağlamaktır. Kullanıcılar, mobil cihazlarının konforunda hesap oluşturma ve yönetme, şifre belirleme veya değiştirme ve çeşitli bankacılık özelliklerinden faydalanma yeteneği kazanırlar. İnternet tabanlı işlemleri kolaylaştırarak, uygulama finansal süreçleri hızlandırır ve coğrafi engelleri ortadan kaldırarak kullanıcıların internet bağlantısı olan herhangi bir yerden güvenli ve verimli bir şekilde işlem yapmalarını sağlar.

**1.3 Misyon ve Vizyon**

Bu projenin misyonu, kullanıcıların finansal işlemlerini kolay, güvenli ve hızlı bir şekilde gerçekleştirebilecekleri bir ortam sağlamaktır. Uygulama, gerçek para kullanımını içermese de, gerçek bir bankacılık deneyimini simüle ederek kullanıcıların çeşitli finansal işlemleri anlamalarına ve yönetmelerine olanak tanır. Bu sayede, kullanıcılar için eğitici bir araç işlevi görür ve geliştirici olarak Android platformunda uygulama geliştirme yetenekleri sergileme fırsatı sunar. Ayrıca, uygulama, programlama ve sistem tasarımı konularında derinlemesine bilgi sahibi olmak isteyenler için mükemmel bir öğrenme kaynağıdır.

Projenin vizyonu, teknoloji ve kullanıcı ihtiyaçları doğrultusunda sürekli gelişen bir uygulama sunmaktır. Uygulama, zamanla Android geliştirme alanında ortaya çıkan yeni kütüphane, araçlar ve best practices (en iyi uygulama yöntemleri) ile sürekli güncellenerek, modern ve kullanıcı dostu bir arayüz sunmayı hedefler. Bu süreçte, uygulamanın kullanıcı deneyimini artırarak, daha fonksiyonel ve etkileşimli hale getirilmesi planlanmaktadır. Ayrıca, uygulamanın daha geniş bir kullanıcı kitlesine ulaşması ve gerçek dünya senaryolarında kullanılabilirliğinin artırılması amaçlanmaktadır. Uzun vadede, uygulama üzerinde çalışmak, benim gibi geliştiricilerin kariyerlerine katkıda bulunacak deneyimler ve bilgiler edinmelerine olanak tanırken, aynı zamanda toplulukla etkileşim ve geri bildirim alma fırsatı da sunar.

Bu vizyon ve misyon, uygulamanın sadece bir proje olmaktan çıkıp, sürekli gelişen ve uyum sağlayan, gerçek kullanıcı ihtiyaçlarını karşılayan bir platforma dönüşmesine yardımcı olacaktır.

**1.4 Proje Kapsamı**

Proje kapsamında, Android Studio kullanılarak MVC (Model-View-Controller) mimari deseni uygulanacaktır. Bu yaklaşım, uygulamanın daha temiz, organize ve sürdürülebilir bir kod yapısına sahip olmasını sağlayacaktır. Model, veritabanı ile etkileşimi yönetirken; View, kullanıcı arayüzünü; Controller ise uygulama mantığını kontrol edecektir.

Uygulamanın geliştirilmesi sırasında, modern programlama prensipleri ve en iyi uygulamalar gözetilecek. Bunlar arasında:

**1.4.1 Belgelendirme**

Kodun anlaşılır ve kolay bakım yapılabilir olması için yeterli düzeyde yorum satırları ve teknik dokümantasyon sağlanacaktır.

**1.4.2 Hata ve İstisna Yönetimi**

Uygulamanın kararlılığını artırmak ve beklenmedik durumlarla başa çıkabilmek için kapsamlı hata ve istisna yönetimi yapılacaktır.

**1.4.3 Kapsamlı Program Yapısı**

MVC mimarisi kullanılarak modüller arası bağımlılıklar minimize edilecek ve her modülün sorumluluğu net bir şekilde tanımlanacaktır.

**1.4.4 Bellek Verimliliği**

Uygulamanın performansını ve kullanıcı deneyimini iyileştirmek için bellek kullanımı optimizasyonları yapılacaktır.

Projenin başarılı bir şekilde tamamlanması, kullanıcıların bankacılık işlemlerini daha hızlı ve güvenli bir şekilde gerçekleştirmelerine olanak sağlayacak ve fiziksel banka şubelerine olan ihtiyacı azaltacak. Bu, özellikle uzak bölgelerde yaşayan veya mevcut bankacılık hizmetlerine erişimi kısıtlı olan kullanıcılar için büyük bir kolaylık sağlayacaktır.

Formun Üstü

**1.5 Projeksiyon:**

* Proje Başlangıç Tarihi: 01.02.2024
* Proje Bitiş Tarihi: 30.09.2024
* Tahmini Maliyet Tutarı: 150.000 USD
* Beklenen Kullanıcı / Müşteri Sayısı:
* İlk 6 Ay: 3 Banka, 9000 Kullanıcı
* İlk 12 Ay: 8 Banka, 24000 Kullanıcı
* İlk 18 Ay: 12 Banka, 75000 Kullanıcı
* İlk 24 Ay: 18 Banka, 150000 Kullanıcı

**1.6 Yol Haritası:**

Uygulamanın gelişim süreci belirli aşamalara bölünmüştür ve her bir aşama, belirli özelliklerin geliştirilmesini ve entegrasyonunu içerir:

**1.6a Başlangıç Fazı**

Temel kullanıcı girişi ve profil yönetimi sistemlerinin kurulması.

İlk bankacılık işlevlerinin (hesap oluşturma, para transferi) geliştirilmesi.

**1.6b Gelişim Fazı**

Daha karmaşık işlevlerin (ödemeler, döviz işlemleri, kredi başvuruları) eklenmesi.

Kullanıcı arayüzünün geliştirilmesi ve kullanıcı deneyiminin iyileştirilmesi.

**1.6c Entegrasyon ve Optimizasyon Fazı**

Yapay zeka ve makine öğrenimi modellerinin entegrasyonu.

Güvenlik önlemlerinin artırılması ve performans optimizasyonlarının yapılması.

**1.6d Genişleme Fazı**

Diğer finansal servislerle entegrasyon.

Çoklu dil desteğinin eklenmesi ve uluslararası kullanıma açılma.

**1.6e Sürdürülebilirlik ve Yenilik Fazı**

Sürekli kullanıcı geri bildirimi toplama ve uygulama güncellemeleri.

Yeni teknolojik gelişmelere ve trendlere uyum sağlama.

Bu yol haritası, projenin sürdürülebilir bir şekilde gelişimini ve kullanıcı ihtiyaçlarına uyum sağlamasını hedeflemektedir. Her bir faz, projenin kapsamlı bir şekilde test edilmesini ve kullanıcıların ihtiyaçlarına cevap verecek şekilde tasarlanmasını sağlar.

**İKİNCİ BÖLÜM**

**2. BENZER ÇALIŞMALAR**

Bu bankacılık uygulaması projesi, mobil bankacılık uygulamaları alanında mevcut olan çeşitli benzer çalışmalar ve ürünlerle karşılaştırılabilir. Aşağıda, bu alandaki bazı öne çıkan ürünlerin özellikleri ve projemizin bu ürünlerden farkı listelenmiştir:

**2.1 Revolut**

Özellikleri: Birden fazla para birimi desteği, anında para transferi, harcama analizi, bütçe planlama ve kripto para alım satımı gibi özellikler sunar.

Farklılıklar: Revolut, gerçek para ile çalışır ve uluslararası bankacılık işlemlerine odaklanır.

**2.2 Monzo**

Özellikleri: Anında bildirimler, harcama kategorileri, tasarruf hedefleri ve acil durumda para kazanma gibi işlevlerle kullanıcı dostu bir arayüz sunar.

Farklılıklar: Monzo tam teşekküllü bir banka olarak faaliyet göstermektedir ve müşterilere kapsamlı bankacılık hizmetleri sunar.

**2.3 Chime**

Özellikleri: Ücret almayan bir yapı, otomatik tasarruf seçenekleri ve maaşların erken alınması gibi özellikler içerir.

Farklılıklar: Chime, özellikle ABD pazarına yöneliktir ve kullanıcıların finansal sağlığını iyileştirmeye yönelik hizmetler sunar.

**2.4 Simple**

Özellikleri: Otomatik bütçe planlama araçları, hedef tabanlı tasarruf mekanizmaları ve minimalist bir kullanıcı arayüzü.

Farklılıklar: Simple, bütçe yönetimine odaklanır ve kullanıcıların finansal hedeflerine ulaşmalarına yardımcı olacak araçlar sağlar.

**2.5 Mint**

Özellikleri: Finansal hesapların yönetimi, bütçe oluşturma, kredi skoru izleme ve fatura hatırlatıcılar gibi birçok özellik sunar.

Farklılıklar: Mint daha çok kişisel finans yönetimi ve bütçeleme üzerine odaklanmış bir uygulamadır.

**ÜÇÜNCÜ BÖLÜM**

**3. KULLANILAN TEKNOLOJİLER**

**3.1 Android Studio**

Android uygulamaları geliştirmek için kullanılan resmi entegre geliştirme ortamıdır (IDE). Android Studio, uygulama geliştirme sürecinde verimliliği ve kolaylığı artıran geniş araç setleri ve yerleşik özellikler sunar.

**3.2 Java**

Android uygulamalarını geliştirmek için yaygın olarak kullanılan nesne yönelimli programlama dilidir. Java, güçlü topluluk desteği, geniş kütüphane ekosistemi ve Android ile yüksek uyumluluk nedeniyle tercih edilir.

**3.3 SQLite Database**

Hafif, disk tabanlı bir veritabanı, Android uygulamaları için yerel veri depolama yöntemi olarak kullanılır. SQLite, mobil cihazlarda veri depolama için idealdir çünkü hızlı, güvenilir ve yapılandırma gerektirmez.

**3.4 JSON (JavaScript Object Notation)**

Hafif bir veri alışveriş formatıdır. İnsanlar ve makineler tarafından okunabilir olması, web servisleri ve API'lar aracılığıyla veri alışverişi için çok popüler hale gelmiştir. Android uygulamalarında genellikle sunucu ile mobil cihaz arasındaki veri transferi için kullanılır**.**

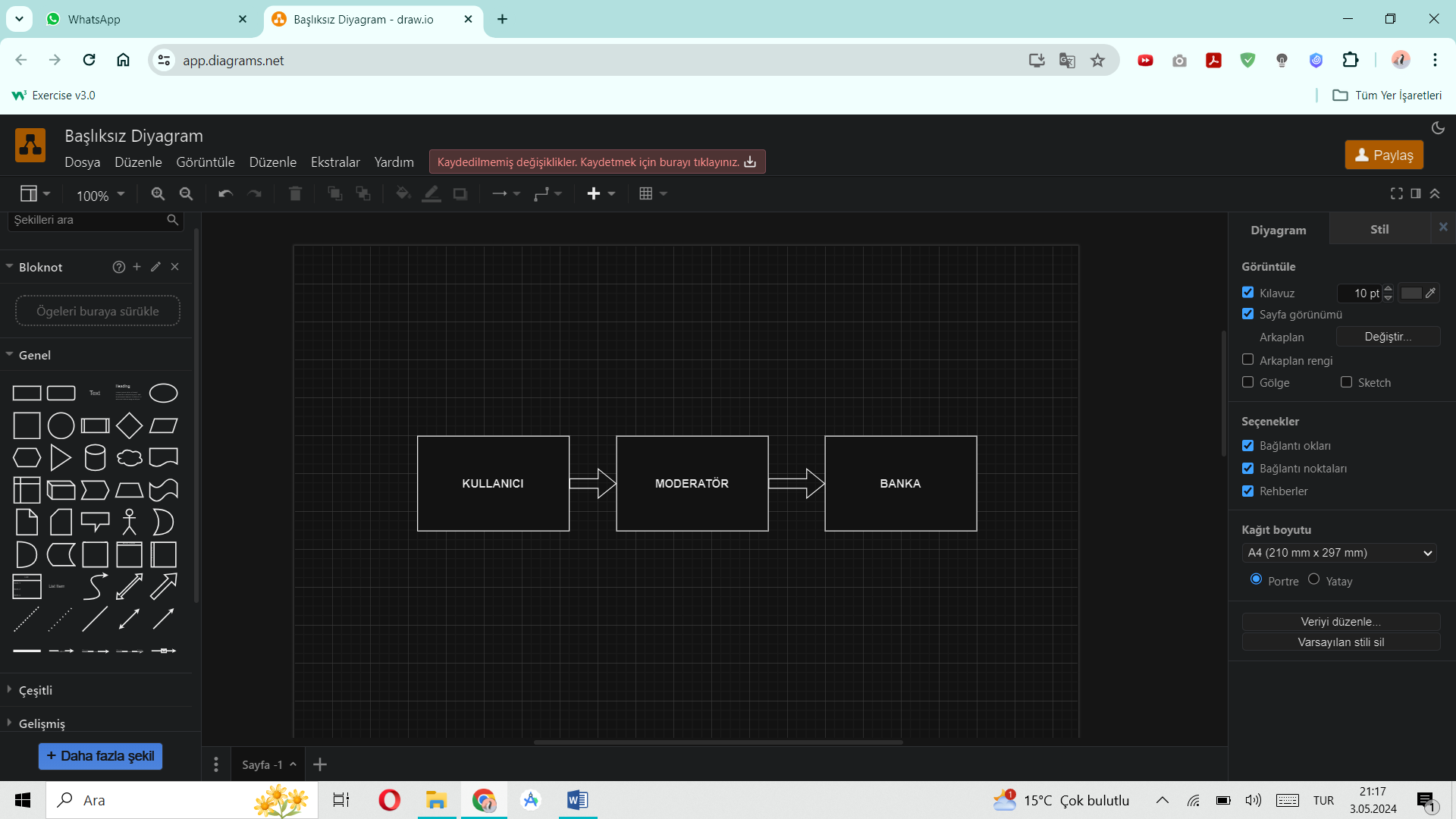
**3.5 Android Emulator**

Android Studio'nun bir parçası olan bu araç, fiziksel bir cihaz olmadan Android uygulamalarını bilgisayarlarda çalıştırmamıza olanak tanır. Bu sayede biz geliştiricilerin uygulamalarını farklı cihaz konfigürasyonları ve Android sürümleri üzerinde test etmemizi kolaylaştırır.

**DÖRDÜNCÜ BÖLÜM**

**4. ORGANİZASYON ŞEMASI ve GÖREV LİSTELERİ**

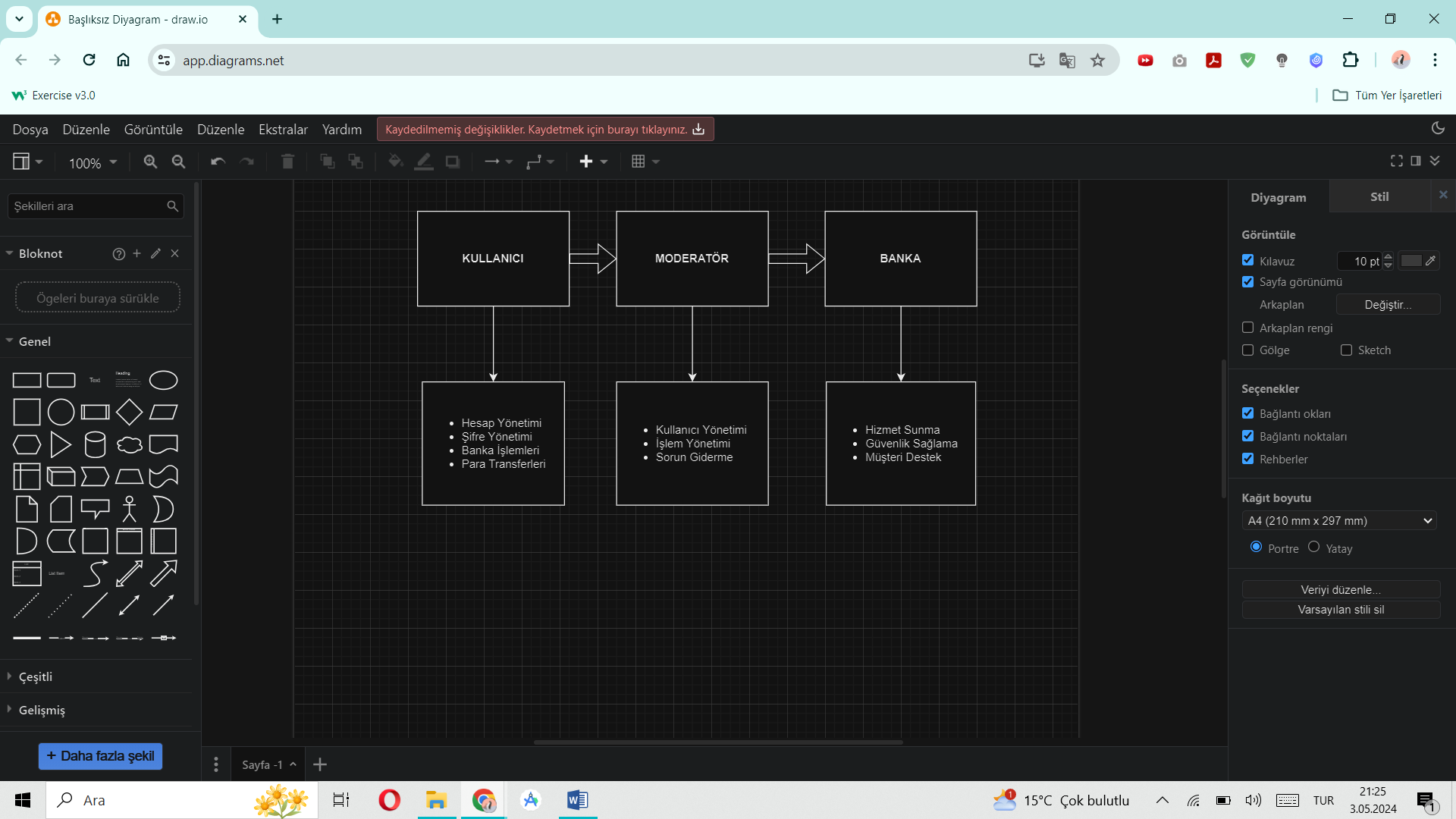
**4.1 Organizasyon Şeması**



**Şekil 4.1a:** Organizasyon Şeması

* Kullanıcılar: Uygulamayı kullanan genel kullanıcılardır.
* Moderatörler: Kullanıcılar ve bankalar arasındaki iletişimi yöneten yetkililerdir.
* Bankalar: Uygulama üzerinde hizmet veren bankalardır.

**4.2 Görev Listeleri**



**Şekil 4.2a:** Görev Listeleri

* **Kullanıcılar**

Kullanıcılar, hesaplarını oluşturabilir, güncelleyebilir ve kapatabilir.

Kullanıcılar şifrelerini ayarlayabilir veya değiştirebilir.

Kullanıcılar, banka hizmetlerinden yararlanabilir, para transferi yapabilir ve işlemleri uygulamaya kaydedebilir.

* **Moderatörler**

Kullanıcı hesaplarını doğrulama, onaylama veya kapatma gibi işlemleri yönetir.

Kullanıcılar arasındaki para transferi işlemlerini denetler ve gerektiğinde müdahale eder.

Kullanıcılar veya bankalar arasında yaşanan sorunları çözer ve destek sağlar.

* **Bankalar**

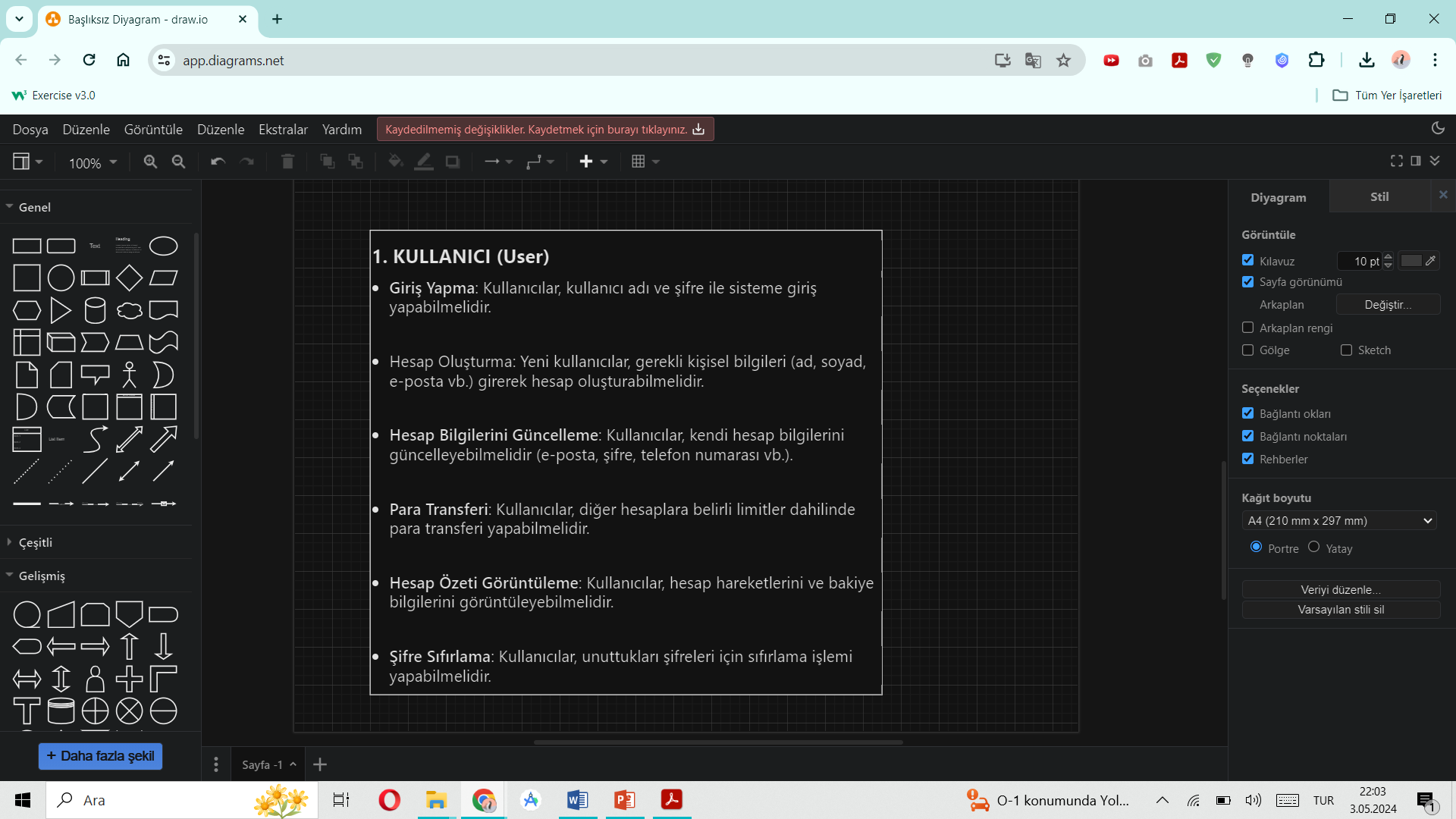
Bankalar, uygulama üzerinden kullanıcılara hizmet sunar ve işlemleri gerçekleştirir.

Bankalar, güvenli bir şekilde kullanıcı bilgilerini ve işlemlerini yönetir.

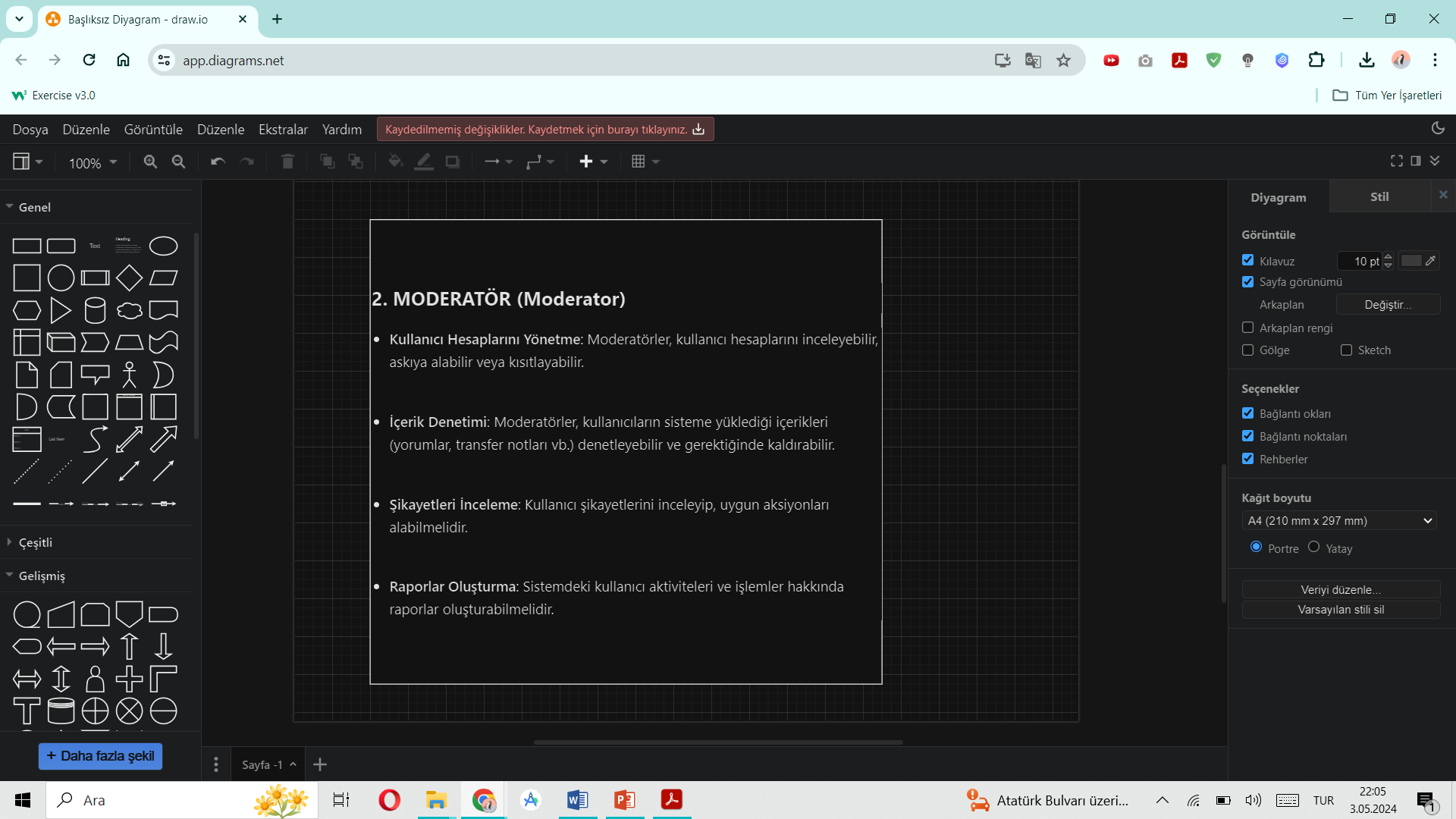
Bankalar, müşterilerin yaşadığı sorunlarla yakinen ilgilenir.

**BEŞİNCİ BÖLÜM**

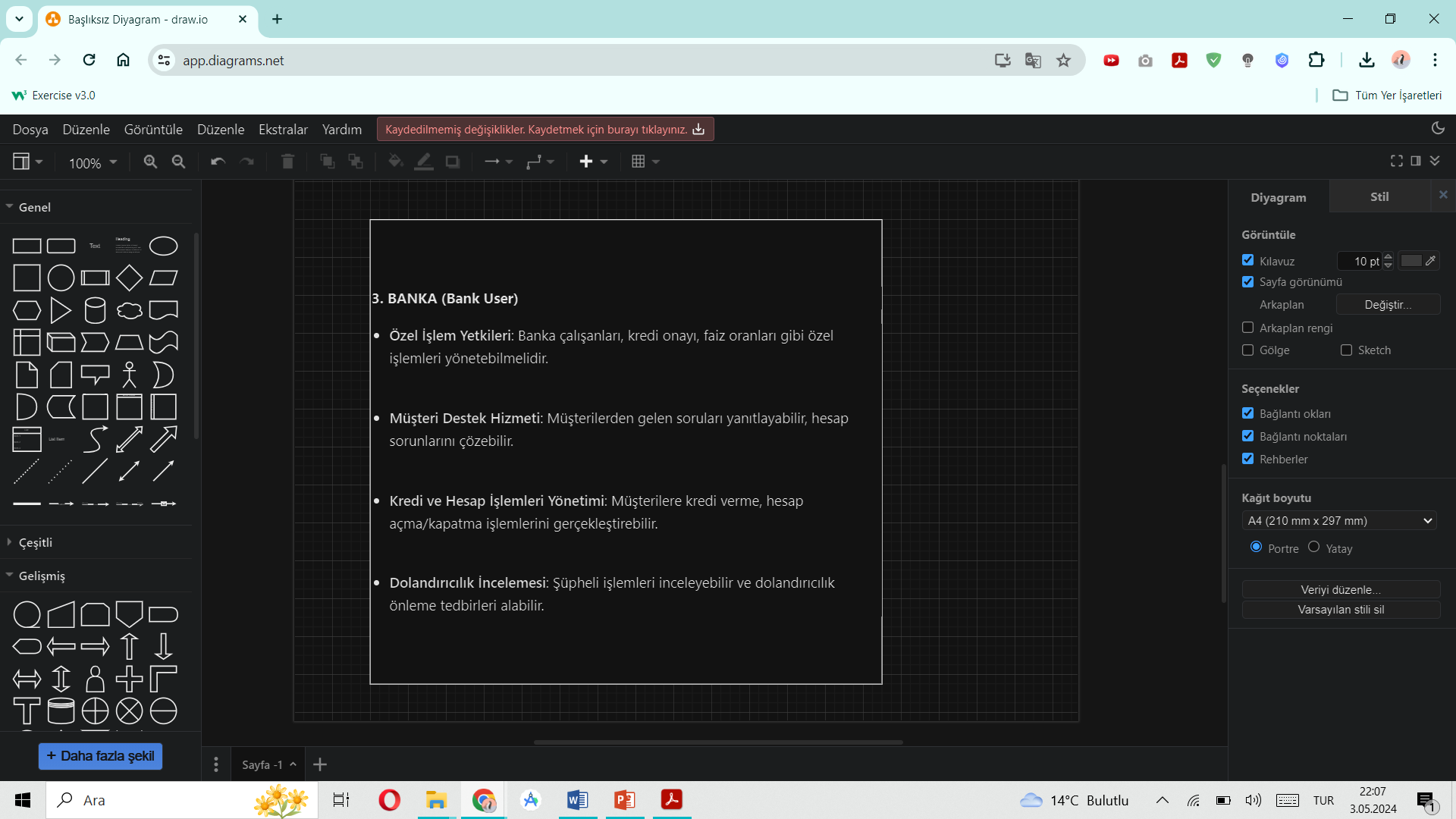
**5. İŞLEVSEL ve KALİTE GEREKSİNİMLERİ**



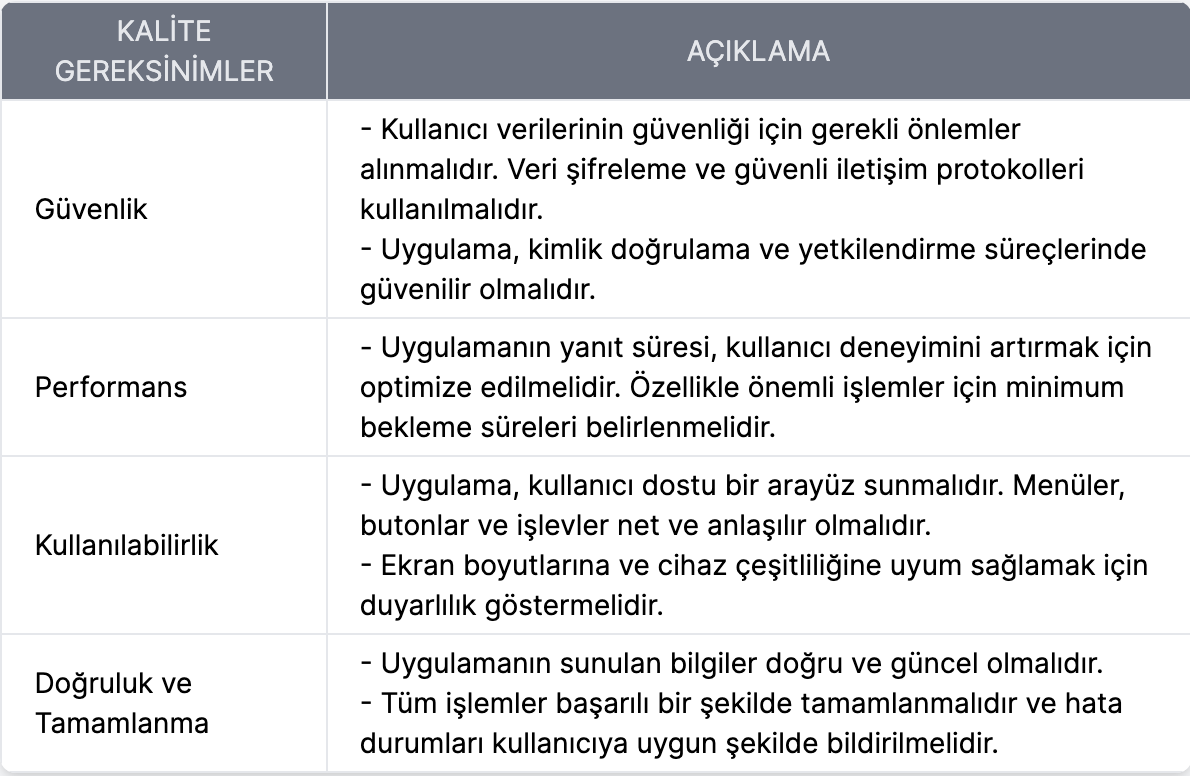
**Şekil 5a:** Kullanıcı için işlevsel gereksinimler



**Şekil 5b:** Moderatör için işlevsel gereksinimler



**Şekil 5c:** Banka için işlevsel gereksinimler



**Şekil 5b:** Kalite Gereksinimleri

**6. SENARYOLAR**

**6.1 Her bir Usecase için Oluşturulan Senaryo**

**6.1a Senaryo: Login**

* Başlatıcı: User Başlangıç Koşulu: Kullanıcı giriş ekranına gelir.
* Ana Akış: Kullanıcı kullanıcı adı ve şifresini girer.
  + - Sistem (Moderator) bilgileri doğrular.
    - Bilgiler doğruysa, sistem kullanıcıyı ana ekrana yönlendirir.
* Alternatif Akış:
  + - 3a. Bilgiler yanlışsa, sistem hata mesajı gösterir ve tekrar giriş yapılmasını ister.
* Bitiş Koşulu: Kullanıcı başarılı bir şekilde giriş yapar veya hata mesajı alır.

**6.1.b Senaryo: View Account**

* Başlatıcı: User veya Bank
* Başlangıç Koşulu: Kullanıcı veya banka çalışanı, hesap bilgilerini görüntülemek ister.
* Ana Akış:
  + - Kullanıcı veya banka çalışanı "View Account" seçeneğini seçer.
    - Sistem hesap bilgilerini getirir ve ekranda gösterir.
* Bitiş Koşulu: Hesap bilgileri başarıyla görüntülenir.

**6.1.c Senaryo: Perform Transaction**

* Başlatıcı: User veya Bank
* Başlangıç Koşulu: Kullanıcı veya banka çalışanı bir işlem gerçekleştirmek ister.
* Ana Akış:
  + - Kullanıcı veya banka çalışanı işlem bilgilerini girer.
    - Sistem işlem bilgilerini doğrular ve gerçekleştirir.
    - Sistem işlem sonucunu kullanıcıya bildirir.
* Bitiş Koşulu: İşlem başarıyla gerçekleştirilir ve kullanıcı bilgilendirilir.

**6.1.d Senaryo : Make Payment**

* Başlatıcı: User veya Bank
* Başlangıç Koşulu: Kullanıcı veya banka çalışanı ödeme yapmak ister.
* Ana Akış:
  + - Kullanıcı veya banka çalışanı ödeme bilgilerini girer.
    - Sistem ödeme bilgilerini doğrular ve ödemeyi gerçekleştirir.
    - Sistem ödeme sonucunu kullanıcıya bildirir.
* Bitiş Koşulu: Ödeme başarıyla gerçekleştirilir ve kullanıcı bilgilendirilir.

**6.1e Senaryo: Make Transfer**

* Başlatıcı: User veya Bank
* Başlangıç Koşulu: Kullanıcı veya banka çalışanı para transferi yapmak ister.
* Ana Akış:
  + - Kullanıcı veya banka çalışanı transfer bilgilerini girer.
    - Sistem transfer bilgilerini doğrular ve transferi gerçekleştirir.
    - Sistem transfer sonucunu kullanıcıya bildirir.
* Bitiş Koşulu: Transfer başarıyla gerçekleştirilir ve kullanıcı bilgilendirilir.

**6.1f Senaryo: Create Profile**

* Başlatıcı: User veya System (Moderator)
* Başlangıç Koşulu: Kullanıcı veya sistem, yeni bir profil oluşturmak ister.
* Ana Akış:
  + - Kullanıcı veya sistem profil bilgilerini girer.
    - Sistem profil bilgilerini doğrular ve profili oluşturur.
    - Sistem profil oluşturma sonucunu kullanıcıya bildirir.
* Bitiş Koşulu: Profil başarıyla oluşturulur ve kullanıcı bilgilendirilir.

**6.1g Senaryo: Approve Account Creation**

* Başlatıcı: Bank
* Başlangıç Koşulu: Banka çalışanı yeni hesap oluşturma talebini onaylamak ister.
* Ana Akış:
  + - Banka çalışanı hesap oluşturma talebini görüntüler.
    - Banka çalışanı talebi onaylar veya reddeder.
    - Sistem sonucu kullanıcıya bildirir.
* Bitiş Koşulu: Hesap oluşturma talebi onaylanır veya reddedilir ve kullanıcı bilgilendirilir.

**6.1h Senaryo: View Customer Information**

* Başlatıcı: Bank
* Başlangıç Koşulu: Banka çalışanı müşteri bilgilerini görüntülemek ister.
* Ana Akış:
  + - Banka çalışanı müşteri bilgilerini görüntüler.
    - Sistem müşteri bilgilerini getirir ve ekranda gösterir.
* Bitiş Koşulu: Müşteri bilgileri başarıyla görüntülenir.

**6.1j Senaryo: Add/Edit Customer Information**

* Başlatıcı: System (Moderator)
* Başlangıç Koşulu: Sistem yöneticisi müşteri bilgilerini eklemek veya düzenlemek ister.
* Ana Akış:
  + - Sistem yöneticisi müşteri bilgilerini girer veya düzenler.
    - Sistem bilgileri doğrular ve günceller.
    - Sistem sonucu kullanıcıya bildirir.
* Bitiş Koşulu: Müşteri bilgileri başarıyla eklenir veya düzenlenir ve kullanıcı bilgilendirilir.

**6.1k Senaryo: Add/Edit Bank Employee Information**

* Başlatıcı: System (Moderator)
* Başlangıç Koşulu: Sistem yöneticisi banka çalışanı bilgilerini eklemek veya düzenlemek ister.
* Ana Akış:
  + - Sistem yöneticisi banka çalışanı bilgilerini girer veya düzenler.
    - Sistem bilgileri doğrular ve günceller.
    - Sistem sonucu kullanıcıya bildirir.
* Bitiş Koşulu: Banka çalışanı bilgileri başarıyla eklenir veya düzenlenir ve kullanıcı bilgilendirilir.

**6.1m Senaryo : View System Logs (Monitor Activities)**

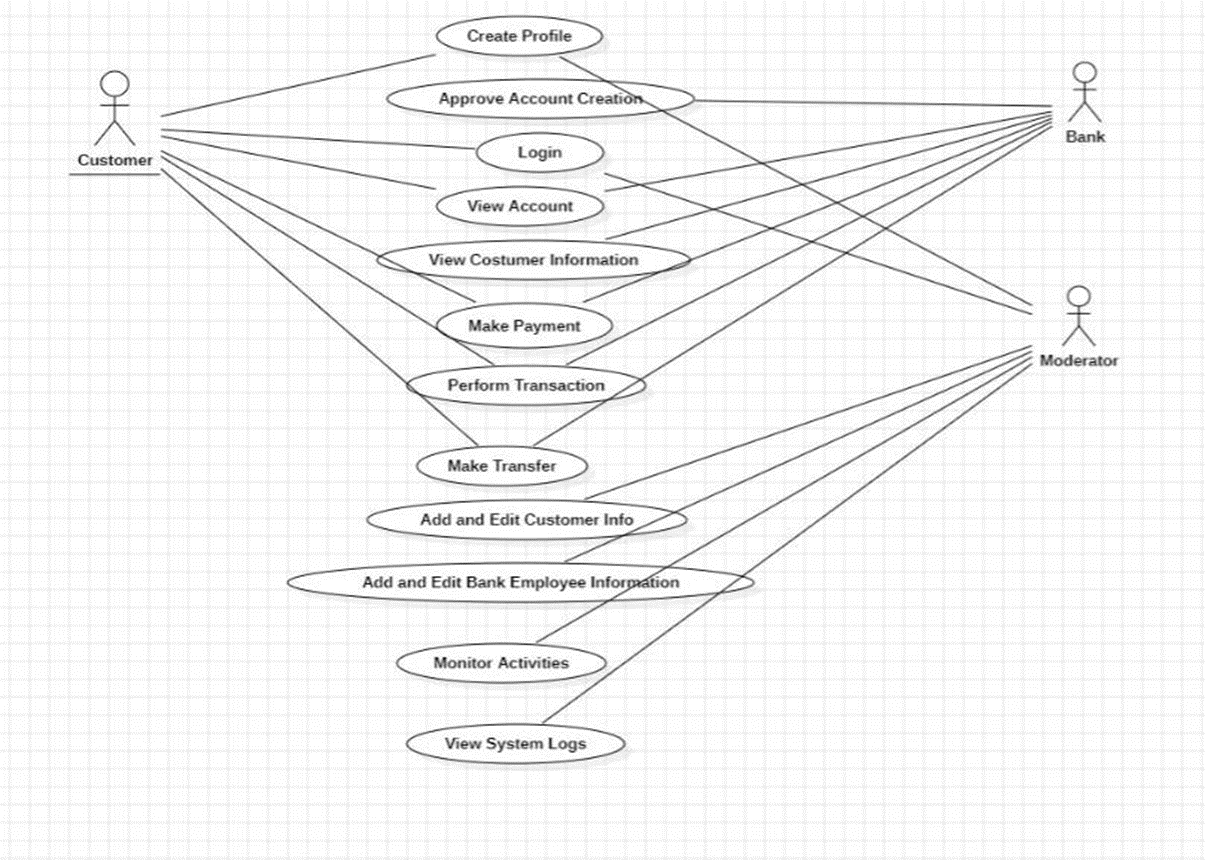
* Başlatıcı: System (Moderator)
* Başlangıç Koşulu: Sistem yöneticisi sistem loglarını görüntülemek ister.
* Ana Akış:
  + - Sistem yöneticisi sistem loglarını görüntüler.
    - Sistem log bilgilerini getirir ve ekranda gösterir.
* Bitiş Koşulu: Sistem logları başarıyla görüntülenir.

**6.1n Senaryo: Monitor Activities**

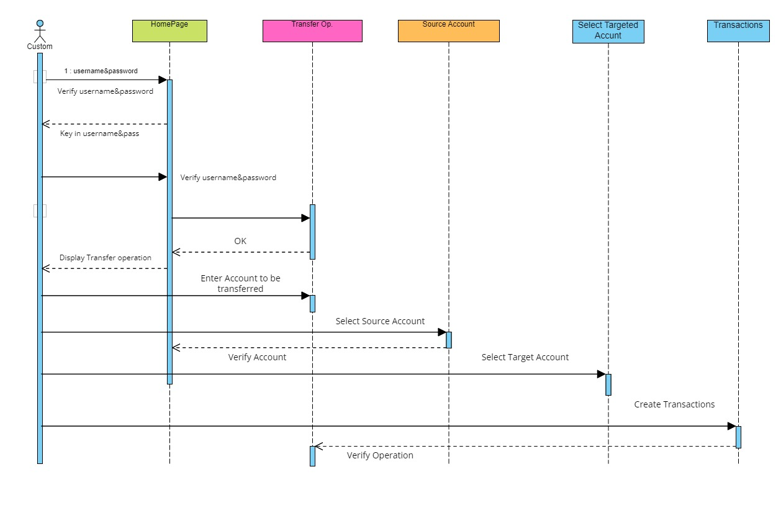
* Başlatıcı: System (Moderator)
* Başlangıç Koşulu: Sistem yöneticisi, kullanıcı ve banka çalışanlarının aktivitelerini izlemek ister.
* Ana Akış:
  + - Sistem yöneticisi izlemek istediği aktiviteleri seçer.
    - Sistem aktiviteleri izler ve raporlar.
    - Sistem yöneticisi aktiviteleri analiz eder ve gerekli aksiyonları alır.
* Bitiş Koşulu: Aktiviteler başarıyla izlenir ve gerekli aksiyonlar alınır.

**7. TABLOLAR VE DİAGRAMLAR**

**7.1 Use Case Diagram**



**7.2 Sequence Diagram**

****

**7.3 SWOT Tablosu**

**metin, ekran görüntüsü, yazı tipi, iş kartı içeren bir resim

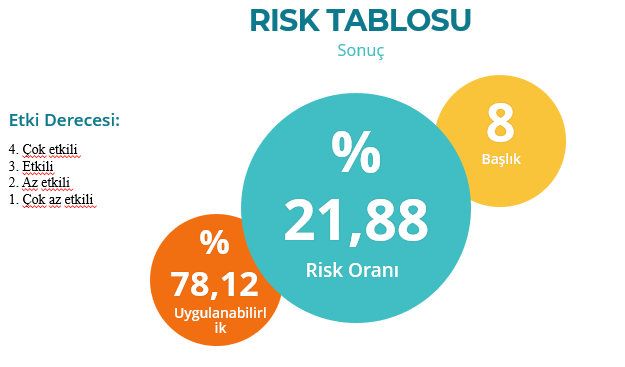
Açıklama otomatik olarak oluşturuldu**

**7.4 Risk Tablosu**

**metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu**

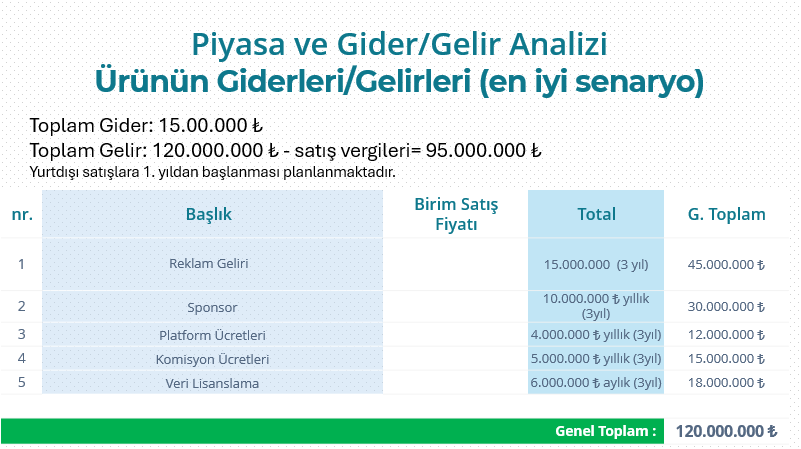
****

****

**7.5 Gelir Gider Modelleri**

**metin, ekran görüntüsü, yazı tipi, web sayfası içeren bir resim

Açıklama otomatik olarak oluşturuldu**

****

**metin, ekran görüntüsü, yazı tipi, sayı, numara içeren bir resim

Açıklama otomatik olarak oluşturuldu**

**7.6 Ms Project Management**

**metin, ekran görüntüsü, öykü gelişim çizgisi; kumpas; grafiğini çıkarma, sayı, numara içeren bir resim

Açıklama otomatik olarak oluşturuldu**

**7. KAYNAKÇA**

1. Developers. (2024, Nisan 30). *Android Studio* *ile Tanışın.* Developers. <https://developer.android.com/studio/intro>

- Erişim tarihi: 1 Mayıs 2024.

2. Niranjanky, K. (2023, Aralık 31). *Java in Android Development: A Comprehensive Guide*. Medium. <https://medium.com/@niranjanky14/navigating-the-java-%EF%B8%8Flandscape-in-android-development-a-comprehensive-guide-a35aa687b9eb>

- Erişim tarihi: 1 Mayıs 2024.

3. Okafor, V. (2015, Eylül 17). *Ultimate Guide to Android SQLite Database*. Medium. <https://medium.com/@valokafor/ultimate-guide-to-android-sqlite-database-44cc8636a4ec>

- Erişim tarihi: 30 Nisan 2024.

4. Aydın, A. (2020, Mart 12). *Android SQLite Kullanımı*. Medium. <https://ayselaydin.medium.com/android-sqlite-kullan%C4%B1m%C4%B1-1477a89fc0ad>

- Erişim tarihi: 1 Mayıs 2024.

5. Shah, D. (2024, Şubat 12). *Mobile Banking Application Development: Services, Innovation, and Best Practices*. Medium. <https://dishita-shah1714.medium.com/mobile-banking-application-development-services-innovation-and-best-practices-c42d080a2735>

- Erişim tarihi: 1 Mayıs 2024.