

DigiBank & Smart City – Proje Raporu

- **Öğrenci:** Ferdi Özaydın / 2420003042
- **Ders:** Nesne Tabanlı Programlama
- **Tarih:** 27.12.2025
- **Proje Kaynak Kodları:** <https://github.com/ferdiozaydin/digibank>

1. Özet

Bu proje, geleneksel bankacılık işlemlerini modern "Akıllı Şehir" (Smart City) konseptleriyle birleştiren, güvenli, modüler ve ölçeklenebilir bir yazılım sistemi prototipidir. Çalışma iki ana evreden oluşmaktadır:

1. **Analiz Evresi:** Referans alınan "digibank.exe" isimli konsol tabanlı C++ uygulaması analiz edilmiştir. Bu uygulama, projenin temel gereksinimlerini belirlemek için bir kıyaslama noktası olarak kullanılmış, ancak tarafımızca geliştirilmemiştir.
2. **Geliştirme Evresi:** Referans sistemin sunduğu temel fonksiyonlar baz alınarak, tamamen modern teknolojilerle (Java 17, Python Flask, PostgreSQL, Docker) donatılmış, "DigiCityBank" adı verilen özgün bir sistem geliştirilmiştir.

Proje, Nesne Tabanlı Programlama (OOP) prensiplerini, yaygın Tasarım Kalıplarını (Design Patterns) ve modern güvenlik uygulamalarını (MFA, kriptografik hashing) pratik bir senaryoda göstermeyi amaçlar. Ayrıca Docker konteynerizasyonu, veritabanı entegrasyonu (PostgreSQL) ve e-posta simülasyonu (Mailpit) gibi endüstri standartları da projenin altyapısına dahil edilmiştir.

2. Giriş

2.1. Problem Tanımı

Günümüzde bankacılık sistemleri sadece para transferi yapılan finansal araçlar olmaktan çıkmıştır. Dijitalleşen dünyada bankalar, kullanıcıların fatura ödemelerini, aboneliklerini, akıllı ev sistemlerini ve hatta kripto varlıklarını tek bir merkezden yönettiği "Süper Uygulamalara" dönüşmektedir. Ancak bu entegrasyon süreci; veri güvenliği, sistemlerin birbiriley konuşması (interoperability) ve ölçeklenebilirlik gibi karmaşık mühendislik problemlerini beraberinde getirmektedir. Geleneksel monolitik yapılar bu ihtiyaçlara cevap vermekte zorlanmaktadır.

2.2. Projenin Amacı

Bu projenin temel amacı, modern bir finansal teknolojinin (FinTech) ihtiyaç duyduğu altyapıyı akademik bir perspektifle simüle etmektir. Hedefler şunlardır:

1. **OOP Prensiplerinin Uygulanması:** Soyutlama, kalıtım, çok biçimlilik ve kapsülleme gibi prensiplerin gerçek hayat problemlerinde nasıl kullanıldığı göstermek.
2. **Mimari Dönüşüm:** Basit bir konsol uygulamasından (digibank.exe), çok katmanlı ve servis odaklı modern bir web mimarisine (DigiCityBank) geçiş sürecini yönetmek.
3. **Tasarım Kalıpları:** Singleton, Strategy, Observer, Adapter ve Command gibi kalıpları kullanarak kodun esnekliğini ve bakım kolaylığını artırmak.

4. **Güvenlik:** Kullanıcı verilerini korumak için endüstri standarı güvenlik önlemlerini (Salted Hashing, TOTP MFA) entegre etmek.

3. Sistem Tasarımı ve Yöntem

3.1. Yazılım Mimarisi

Sistem, modern mikroservis yaklaşımılarına benzer şekilde, sorumlulukların ayrıldığı (Separation of Concerns) dağıtık bileşenlerden oluşur:

- **Sunucu Katmanı (Backend - Java 17):** Sistemin beyni olarak görev yapar. Harici bir web sunucusu (Tomcat vb.) yerine, Java'nın yerleşik **HttpServer** sınıfı kullanılarak native HTTP işleme yeteneği sergilenmiştir. Tüm iş kuralları, veri doğrulama, veritabanı iletişim ve güvenlik kontrolleri bu katmanda yönetilir.
- **İstemci Katmanı (Frontend - Python Flask):** Kullanıcı etkileşiminin gerçekleştiği web arayüzüdür. Modern HTML5/CSS3 ve JavaScript kullanılarak tasarlanmış, Python Flask framework'ü üzerinde çalışmaktadır. Backend ile RESTful API üzerinden JSON formatında haberleşir.
- **Veritabanı Katmanı (PostgreSQL):** Veri kalıcılığını sağlamak için ilişkisel veritabanı yönetim sistemi olan PostgreSQL kullanılmıştır. Kullanıcı bilgileri, işlemler ve sistem logları burada tutulur.
- **Yan Servisler (Mailpit):** Sistemin e-posta gönderme yeteneklerini test etmek için kullanılan, gerçek dünya senaryolarını simüle eden bir SMTP sunucusudur.

3.2. Kullanılan Tasarım Kalıpları (Design Patterns)

Projede karşılaşılan yapısal sorunlar, kanıtlanmış tasarım kalıpları ile çözülmüştür:

- **Singleton:** **AuditLogger** sınıfı ile tüm sistemde tek bir loglama mekanizması garanti edilmiş, böylece eşzamanlı erişim sorunları önlenmiştir.
- **Strategy:** Ödeme sisteminde esneklik sağlamak için **PaymentStrategy** arayüzü kurgulanmıştır. Bu sayede FIAT para veya Kripto para ile ödeme algoritmaları çalışma zamanında (runtime) seçilebilir hale gelmiştir (Open/Closed Principle).
- **Adapter:** Farklı arayzlere sahip dış kripto servislerini sisteme entegre etmek için **CryptoAdapter** kullanılmıştır.
- **Observer:** Akıllı ev (Smart Home) modülünde, sensör verilerindeki değişimlerin ilgili cihazlara (klima, ışıklandırma) otomatik bildirilmesi için bu desen kullanılmıştır.
- **Command:** Yönetici komutları ve ev otomasyon işlemleri birer nesne (Command Object) olarak modellenmiş, bu sayede işlemlerin sıraya alınması, geri alınması (undo) ve loglanması kolaylaşmıştır.

3.3. Güvenlik Altyapısı

Güvenlik, projenin en kritik bileşenlerinden biridir:

- **Parola Güvenliği:** Kullanıcı parolaları asla düz metin (plaintext) olarak saklanmaz. SHA3-512 algoritması ve rastgele üretilen "Salt" değerleri ile hashlenerek veritabanında tutulur.
- **Çok Faktörlü Kimlik Doğrulama (MFA):** Giriş güvenliğini artırmak için Zaman Tabanlı Tek Kullanımlık Şifre (TOTP) algoritması uygulanmıştır.
- **Yetkilendirme:** Rol tabanlı erişim kontrolü (RBAC) ile "Admin" ve "Standart Kullanıcı" yetkileri ayrıstırılmış, kritik operasyonlar sadece yetkili kullanıcılarla açılmıştır.

4. Kullanıcı Arayüzleri ve Uygulama Görüntüleri

Projenin evrimini ve geldiği noktayı belgelemek adına, hem referans alınan eski sistem hem de geliştirilen yeni sistemin ekran görüntüleri aşağıda sunulmuştur.

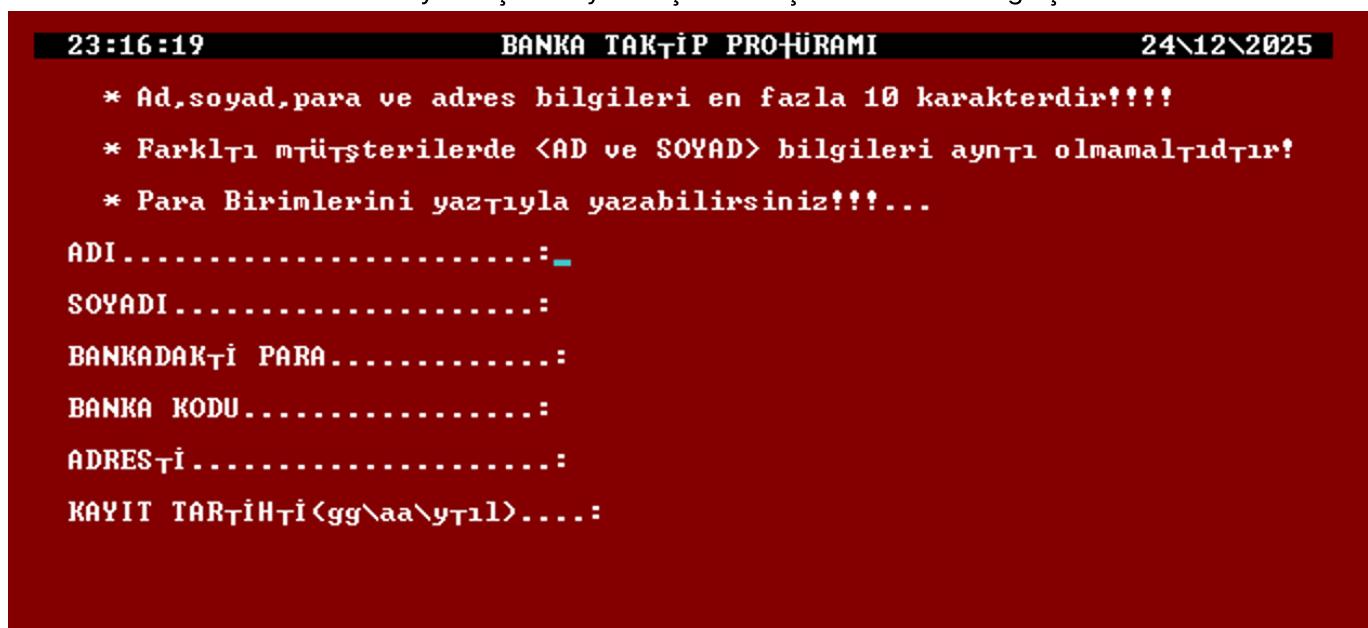
4.1. Bölüm 1: DigiBank Konsol Uygulaması (Referans Proje)

Bu projenin geliştirilmesinde analiz ve kıyaslama amacıyla kullanılan, C++ tabanlı **digibank.exe** referans uygulamasının arayüzleri aşağıdadır. Bu uygulama tarafımızca geliştirilmemiş, sadece gereksinim analizinde kullanılmıştır.

1. Ana Menü ve Giriş Kullanıcıyı karşılayan ana menü, tüm operasyonlara (Ekleme, Silme, Listeleme vb.) erişim sağlayan basit bir metin arayüzüdür.



2. Veri Ekleme Yeni bir banka veya müşteri kaydı oluşturmak için kullanılan veri giriş ekranı.



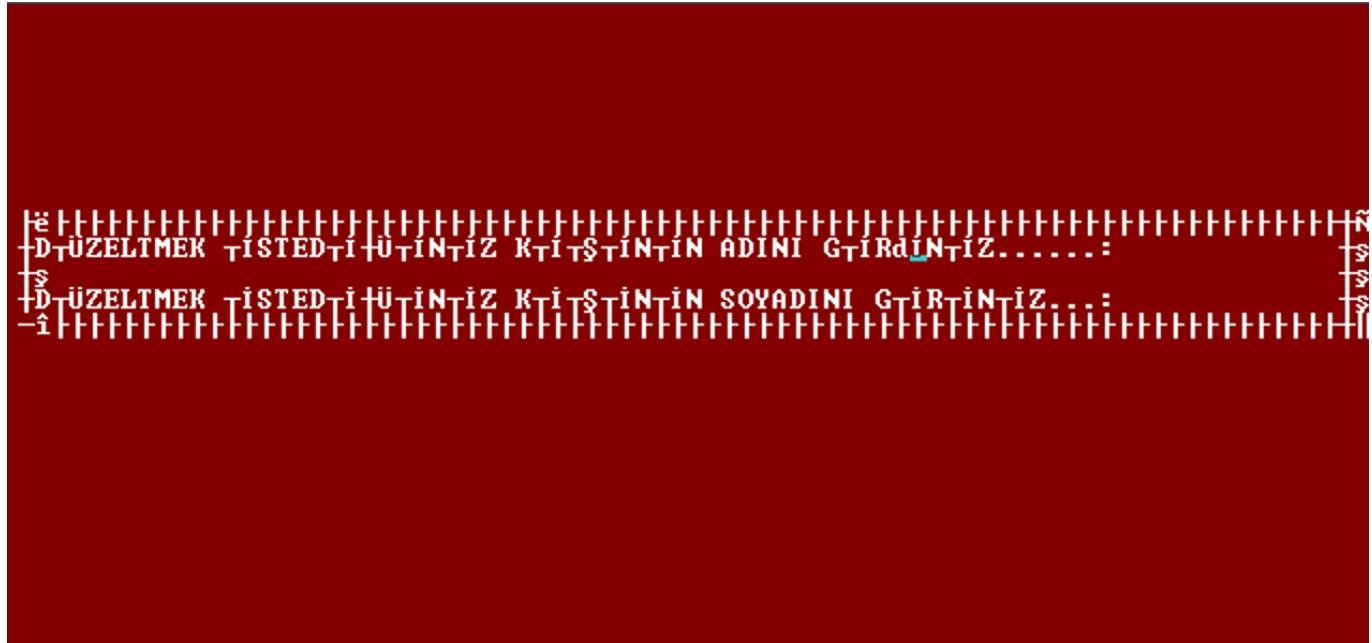
3. Veri Listeleme Sistemde kayıtlı olan tüm verilerin listelendiği ekran.

test	test	233	112233	deneme
------	------	-----	--------	--------

Tüm kayıtlar listedendi ana menü tictin <ENTERA> BASINIZ...:->

4. Arama İşlemi Belirli kriterlere göre kayıtlar arasında arama yapılmasını sağlayan fonksiyon.

İSTİYORUM SİLAHINI İSTİYORUM
SARADI ÜİNİZ KİTSİNTİN ADINI GİRTİNTİZ.....a.....
-İSTİYORUM SİLAHINI İSTİYORUM
SARADI ÜİNİZ KİTSİNTİN SOYADINI GİRTİNTİZ.....:

5. Kayıt Düzenleme Mevcut bir kaydın bilgilerini güncellemek için kullanılan ekran.**6. Kayıt Silme** Veritabanından bir kaydın silinmesi işlemi.

4.2. Bölüm 2: DigiCityBank Web Platformu (Geliştirilen Modern Sistem)

Referans projenin kısıtları aşılarak geliştirilen, modern web teknolojileri ile donatılmış DigiCityBank platformunun arayüzleri aşağıdadır.

- 1. Güvenli Giriş Ekranı** Kullanıcı adı ve parola doğrulamasının yapıldığı, arka planda MFA ve güvenli oturum yönetiminin çalıştığı giriş kapısı.

DigiBank'e Giriş

Lütfen hesap bilgilerinizi girin.

Kullanıcı Adı

Şifre

TOTP Kodu

Authenticator uygulamasındaki 6 haneli TOTP kodunu girin.

Giriş Yap

2. Yönetim Paneli (Dashboard) Kullanıcının finansal durumunu, varlıklarını ve son hareketlerini bir bakışta görebildiği, kullanıcı deneyimi odaklı ana ekran.

DigiBank & SmartCity

Digibank Akıllı Şehir Ayarlar admin Çıkış

Hoş geldin, admin

Finansal özetin burada.

TOPLAM TL BAKİYESİ
₺5,000.00
Bu ay +2.4%

KRIPTO VARLIKLER
฿0.5
≈ \$34,250.00 USD

Hızlı İşlemler
Kısayollar

Para Gönder **Fatura Öde**
İşlem Kayıtları **Döviz/Swap**
Akıllı Ev

Performans Göstergeleri

Demo verisi

API Latency 120 ms	Ödeme Başarı 98.5%	Sensör Uyarıları 3 aktif
Kripto Kullanımı 42%		

Son İşlemler

İşlem bulunmuyor
Güncel kayıt yok

0.00

3. Performans ve Analitik Yöneticiler için hazırlanan, sistemin işlem hacmini, finansal tahminleri (Forecast) ve sunucu sağlık durumunu gösteren grafiksel analiz ekranı.

DigiBank & SmartCity

Digibank Akıllı Şehir Ayarlar admin Çıkış

Performans ve İzleme

API gecikmesi, ödeme başarı oranı ve altyapı sağlığı tek ekranda.

API Gecikmesi
120 ms
Hedef < 150 ms

Ödeme Başarı
98.5%
Son 1 saat

Sensör Uyarıları
3
Aktif saha alarmı

Kripto Kullanımı
42%
Toplam işlemlerde pay

Gecikme ve Throughput
Dakikalık

The chart displays two data series: 'Gecikme (ms)' (latency in ms) and 'Throughput (txn/dk)' (transactions per second). The latency starts at approximately 140 ms at 09:00, dips to 120 ms at 11:00, rises to 130 ms at 12:00, and then fluctuates between 100-150 ms throughout the day. The throughput starts at 100 txns/dk at 09:00, peaks at 220 txns/dk at 19:08, and generally shows an upward trend with some fluctuations.

Başarı / Hata Oranı
Son ölçümler

The chart shows the ratio of successful transactions to failed ones. The success rate is consistently high, around 95-100%, while the error rate is very low, near 0%. The bars represent the percentage of successful transactions for each hour from 09:00 to 19:08.

4. Akıllı Şehir (Smart Government) Ödemeleri Elektrik, su, doğalgaz gibi temel faturaların ödendiği, şehir altyapısı ile entegre modül.

The screenshot shows the 'Akıllı Devlet Servisleri' section of the application. It displays various service statistics and payment method information.

- Aktif Entegrasyonlar:** E-Devlet bağlı, Trafik sistem bağlı, Son güncelleme: 27.12.19:07
- Bekleyen Toplam:** Borç, Bekleyen Toplam: ₺1,950.00, 2 adet ödeme.
- Yaklaşan Son Tarih:** Takip, Yaklaşan Son Tarih: 20.11.2025, En yakın vade.
- Ödenen (Listede):** Tamam, Ödenen: 0, ₺0.00.
- Ödeme Kanalları:** Demo, TL / BTC, ETH / Stablecoin.

5. Para Transferi Kullanıcılar arası para transferini (EFT/Havale) simüle eden, bakiye kontrolü yapan arayüz.

The screenshot shows the 'Para Gönder' (Send Money) section of the application. It includes a transfer form and a list of bank accounts.

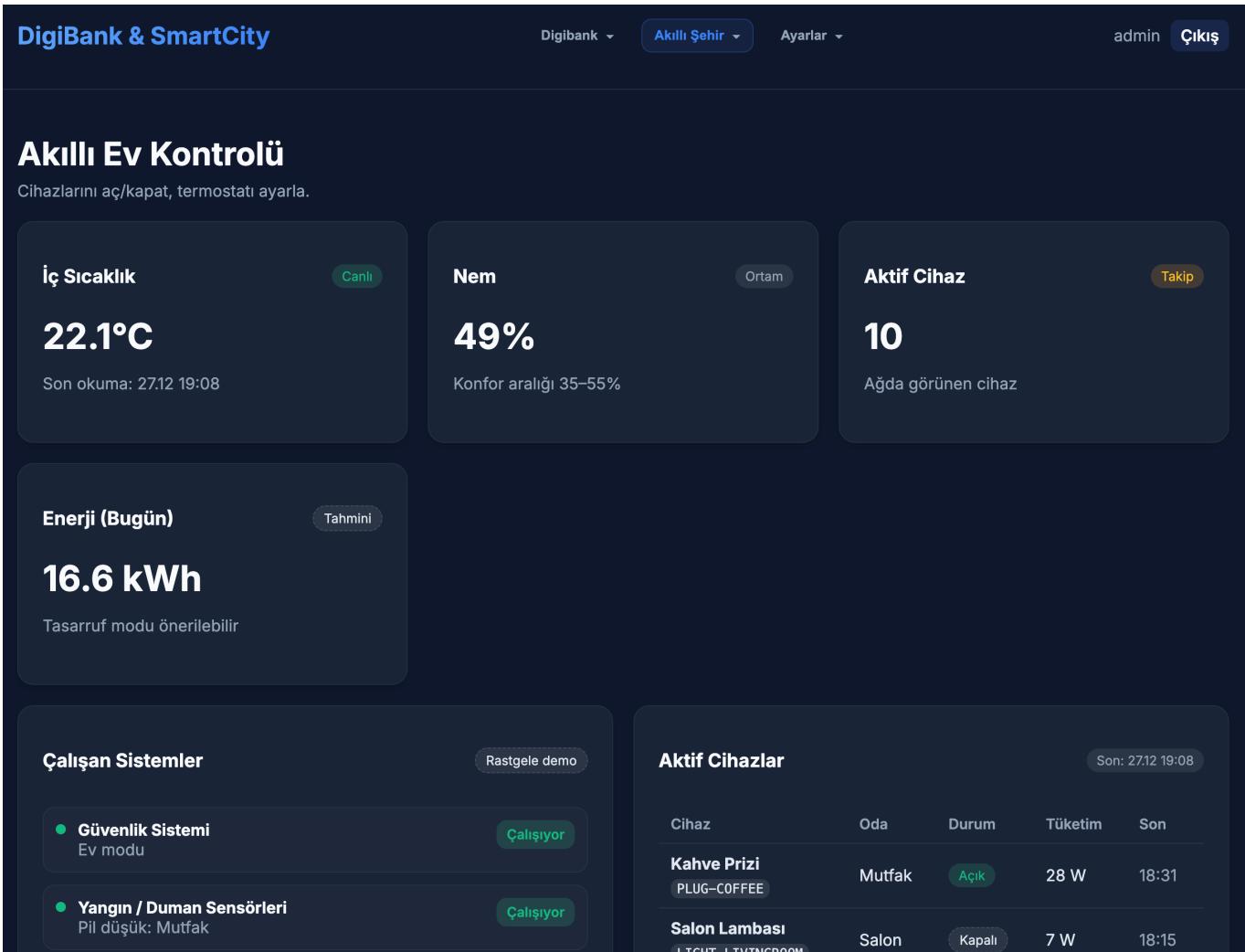
Transfer Formu:

- Alicı Adı: Fuat Alp
- IBAN: TR09 0000 1000 0003 0009 10000 34
- Tutar (TL): 790
- Açıklama: Fatura

Hesaplarım (Demo):

Hesap Tipi	Hesap Numarası	Bakiye (TL)
Vadesiz TL	TR12 0001 0000 1234 5678 0001	18,500.75 TL
USD Hesabı	TR34 0001 0000 1234 5678 0002	5,200.00 TL
EUR Hesabı	TR56 0001 0000 1234 5678 0003	3,100.50 TL

6. Akıllı Ev (Smart Home) Kontrolü Bankacılık uygulaması üzerinden evdeki IoT cihazlarının (Termostat, Işıklar) yönetildiği, Observer deseninin uygulandığı modül.



7. Yönetici: Kullanıcı Listesi Sistem yöneticisinin kayıtlı kullanıcıları yönettiği, yetki ataması yapabildiği panel.

DigiBank & SmartCity

Kullanıcı Yönetimi
Sisteme kayıtlı kullanıcılar: ekle / düzelt / ara / listele / sil.

Yeni Kullanıcı Ekle Admin yetkisi gereklidir

Kullanıcı adı: ornek.kullanici
Şifre: Güçlü bir şifre
Rol: RESIDENT

Kullanıcı Listesi DB veya cache

Ara Sıfırla

ID	Kullanıcı	Rol	TL Bakiye	Kripto
1	admin	ADMIN	5000.0	0.5
2	test	RESIDENT	700	0
5	user	RESIDENT	0	0

TOTP için 000000 demo kodu geçerlidir.

8. Yönetici: Finansal İşlemler Gerçekleşen tüm para transferlerinin ve ödemelerin dökümünün alındığı denetim (audit) ekranı.

ID	Ad Soyad	Para	Banka Kodu	Adres	Kayıt Tarihi	İşlem
19	Galip Şen	240	VAKIFBANK	Yalova	27.12.2025	<button>Ara</button> <button>Excel İndir</button> <button>PDF İndir</button> <button>Mail Gönder</button> <button>Güncelle</button> <button>Sil</button>
18	Fırat Dicle	1400	YAPIKREDİ	Beşiktaş	24.12.2025	<button>Ara</button> <button>Excel İndir</button> <button>PDF İndir</button> <button>Mail Gönder</button> <button>Güncelle</button> <button>Sil</button>
17	Sedat Kul	1390	AKBANK	İstanbul	25.12.2025	<button>Ara</button> <button>Excel İndir</button> <button>PDF İndir</button> <button>Mail Gönder</button> <button>Güncelle</button> <button>Sil</button>

9. Excel Raporlama Finansal verilerin muhasebe sistemlerine aktarımı için Excel formatında dışa aktarılmasını sağlayan özellik.

id	fullName	amount	bankCode	address	recordDate
20	Tuncay Tuy	4500	QNB	İzmir	2025-12-27
19	Galip Şen	240	VAKIFBANK	Yalova	2025-12-27
18	Fırat Dicle	1400	YAPIKREDİ	Beşiktaş	2025-12-24
17	Sedat Kul	1390	AKBANK	İstanbul	2025-12-25

10. E-Posta Servisi Entegrasyonu Sistemden gönderilen bildirimlerin yakalandığı sanal SMTP sunucusu (Mailpit) arayüzü.

 Mailpit

Mark unread Delete

[Return to inbox](#)

digibank@local.test a few seconds ago
To: transactions@local.test
DigiBank Hesap Hareketleri (all)

digibank@local.test 25 minutes ago
To: transactions@local.test
DigiBank Hesap Hareketleri (all)

From <digibank@local.test>
To <transactions@local.test>
Subject DigiBank Hesap Hareketleri (all)
Date Sat, 27 Dec 2025, 10:06 pm (10.2 kB)
Tags Add tags...

[Text](#) [Headers](#) [Raw](#) [Link Check](#)

Merhaba,

DigiBank hesap hareketleri erteledir.
Filtre: all
Kayıt sayısı: 4
Tarih: 2025-12-27 19:06:24


[digibank_transactio...](#)


[digibank_transactio...](#)

11. İşlem Bildirimi Kullanıcıya işlemin başarılı olduğuna dair gönderilen e-posta ve web bildirimi.

DigiBank & SmartCity

Digibank Akıllı Şehir Ayarlar admin Çıkış

Mail gönderildi: transactions@local.test

İşlem Kayıtları

Digibank - İşlem Listesi

Yeni İşlem

Açıklama ara (örn: Transfer, Odeme)

Ara Excel İndir PDF İndir Mail Gönder

ID	Ad Soyad	Para	Banka Kodu	Adres	Kayıt Tarihi	İşlem
20	Tuncay Tuy	4500	QNB	İzmir	27.12.2025	Güncelle Sil
19	Galip Şen	240	VAKIFBANK	Yalova	27.12.2025	Güncelle Sil
18	Fırat Dicle	1400	YAPIKREDİ	Beşiktaş	24.12.2025	Güncelle Sil
17	Sedat Kul	1390	AKBANK	İstanbul	25.12.2025	Güncelle Sil

5. Sonuç ve Değerlendirme

Bu proje çalışması ile "DigiBank" referans fikrinden yola çıkılarak, modern yazılım mühendisliği prensiplerine tam uyumlu "DigiCityBank" ekosistemi başarıyla hayatı geçirilmiştir. Süreç boyunca elde edilen teknik ve akademik kazanımlar aşağıda detaylandırılmıştır:

1. Tam Yığın (Full-Stack) Mimari Yetkinliği:

- Proje, sadece tek bir dilde kod yazmanın ötesine geçerek, **Java** (Backend) ve **Python** (Frontend) gibi farklı teknolojilerin bir arada nasıl uyum içinde çalışabileceğini (Interoperability) kanıtlamıştır.
- REST API standartlarına uygun servisler kurgulanarak, istemci ve sunucu arasındaki bağımlılık azaltılmış, sistem modüller hale getirilmiştir.

2. Nesne Tabanlı Tasarımın Gücü:

- Teorik olarak öğrenilen Tasarım Kalıplarının (Design Patterns), kodun kalitesini, okunabilirliğini ve değiştirilebilirliğini ne denli artırdığı pratikte gözlemlenmiştir. Özellikle **Observer** deseni ile sistemin reaktif (tepkisel) hale gelmesi ve **Strategy** deseni ile yeni ödeme yöntemlerinin sisteme kolayca eklenebilmesi, doğru tasarımın önemini ortaya koymuştur.

3. Güvenlik Farkındalığı:

- Bir fintech uygulamasında "güvenlik sonradan eklenemez, tasarımla gelir" (Security by Design) ilkesi benimsemisti. Parolaların hashlenmesi, MFA kullanımı ve yetki kontrolleri gibi önlemler, güvenli yazılım geliştirme süreçlerinin bir parçası olmuştur.

4. Operasyonel Mükemmellik (DevOps):

- Uygulamanın **Docker** konteynerleri üzerinde çalıştırılması, "benim makinemde çalışıyordu" sorununu ortadan kaldırılmış ve dağıtım süreçlerini standartlaştırmıştır. Bu sayede proje, herhangi bir ortamda (Local, Cloud) hızlıca ayağa kaldırılabilir hale gelmiştir.

Gelecek Çalışmalar ve Öneriler: Mevcut prototip başarıyla çalışmakla birlikte, ticari bir ürüne dönüşmesi için şu geliştirmeler planlanmaktadır:

- Gerçek blockchain ağıları (Ethereum vb.) ile entegrasyon sağlanarak kripto ödemelerinin simülasyondan çıkarılması.
- Mikroservis mimarisinin daha ileri seviyeye taşınarak her modülün (User Service, Payment Service, Notification Service) ayrı konteynerlerde çalıştırılması.
- Yük testleri yapılarak sistemin yüksek trafik altındaki davranışının analiz edilmesi.

Özetle; DigiCityBank projesi, modern yazılım geliştirme araçlarını ve yöntemlerini bir araya getiren, akademik derinliği olan ve endüstriyel standartları yakalayan kapsamlı bir mühendislik çalışmasıdır.

6. Ekler

- Kaynak Kodlar:** GitHub Reposu / Proje Klasörü
- Veritabanı Şeması:** [docs/ek_a_database_schema.mmd](#)
- Sınıf Diyagramları (UML):** [docs/2_uml_class.mmd](#)
- Kurulum ve Çalıştırma:** [docs/0_Steps.md](#)