

**ANKARA ÜNİVERSİTESİ
MÜHENDİSLİK FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**



**Kişisel Finans Takip Sistemi
Final Raporu**

**BLM4537 - IOS ile Mobil Uygulama Geliştirme
20.12.2025**

**Rabia Şevval Yaşar
22290015**

Github: Rabia-23

Project: <https://github.com/Rabia-23/FinanceApp>

Video: <https://drive.google.com/file/d/12x32sYsxd7Vp2Oob-lurqDwPZtyu49Xu/view?usp=sharing>

ÖZET

Kişisel Finans Takip Sistemi, kullanıcıların parasal işlerini dijital ortamda kolayca izlemesini sağlayan kapsamlı bir web ve mobil uygulamadır. Uygulama - gelir gider kaydı, birden çok hesabın yönetimi, bütçe planı, aboneliklerin takibi, finansal hedef koyma ve anlık döviz kuru gösterme özelliklerini tek yerde toplar.

Sunucu tarafında ASP.NET Core 8 Web API, PostgreSQL veritabanı ve Entity Framework Core ORM kullanıldı. JWT tabanlı kimlik doğrulama sistemi, kullanıcıların güvenli oturum açmasını sağlar. İstemci tarafında Flutter ile yazılan arayüz, hem web hem de mobil cihazlarda tek kodla çalışır.

RESTful API, MVC mimarisi ve Code-First yaklaşımı gibi güncel yazılım ilkelerine uygun biçimde tasarlandı; sade görünümü ve geniş özellik setiyle bireysel finans yönetiminde etkili bir araç sunar.

Anahtar Kelimeler - Finans Yönetimi, ASP.NET Core, Flutter, PostgreSQL, Web API, Mobil Uygulama

İÇİNDEKİLER

ÖZET	1
İÇİNDEKİLER	2
1. GİRİŞ	3
1.1. Projenin Amacı ve Kapsamı	3
1.2. Kullanılan Teknolojiler	3
1.3. Raporun Yapısı	3
2. VERİTABANI TABLOLARI	4
2.1. Users Tablosu	4
2.2. Accounts Tablosu	4
2.3. Budgets Tablosu	5
2.4. Subscriptions Tablosu	5
2.5. Goals Tablosu	6
2.6. İlişkisel Diyagram (ERD)	7
2.7. Cascade Delete Davranışı	7
3. REGISTRATION PAGE	8
4. HOME PAGE	9
4.1. Accounts	9
4.2. Net Worth + Graph	9
4.3. Transactions + Budgets	9
5. TRANSACTIONS PAGE	11
6. SUBSCRIPTION PAGE	12
7. GOALS PAGE	13
8. CURRENCY PAGE	14

1. GİRİŞ

1.1. Projenin Amacı ve Kapsamı

İnsanların parasını düzenli tutması, hedefine ulaşması için şarttır. Defter tutmak ya da Excel'de işlem yazmak hem zaman yer hem de yanlış yazma ihtimali yüksektir. Bu sıkıntıyı çözmek için hazırlanan Kişisel Finans Takip Sistemi, tüm parasal işleri dijital ortamda basit biçimde yürütür. Uygulama şu işleri yapar:

- Hesap Yönetimi - İstediği kadar banka hesabını ekler, bakiyeyi anında görür
- Gelir-Gider Takibi - Harcamasını ve parasal girdisini kategorileyip, dilediği filtreyi kullanarak listeler
- Bütçe Planlama - Haftalık, aylık ya da yıllık bütçe oluşturur - dileyen zaman otomatik yeniler
- Abonelik Yönetimi - Düzenli çıkan faturaları tanımlar, vaktinden önce bildirim gönderir
- Hedef Belirleme - Birikim hedefi koyar, ne kadar kaldığını izler
- Döviz Kurları: Canlı kurları anında görür

1.2. Kullanılan Teknolojiler

Yazılım, güncel web ve mobil araçlarla oluşturuldu

Backend

- ASP.NET Core 8 - Hızlı, her ortamda çalışan web API altyapısı
- PostgreSQL - İlişkisel veritabanı
- Entity Framework Core - Code-First yaklaşımıyla ORM
- JWT Authentication - Token tabanlı güvenli oturum
- BCrypt - Parola koruması

Frontend

- Flutter - Web ve mobil için tek kodla çalışan arayüz çatısı
- Dart - Flutter'ın dili
- fl_chart - Grafik ve rapor görselleştirme
- http - REST haberleşmesi

Dış API

- ExchangeRate API - Anlık döviz kuru bilgisi

1.3. Raporun Yapısı

Bölüm 1: Veritabanı şeması ve tablo bağlantıları

Bölüm 2-7: Kullanıcı ekranları ve açıklamaları

2. VERİTABANI TABLOLARI

Sistemde toplamda 6 tablo bulunur.

2.1. Users Tablosu

Kullanıcı bilgilerini saklar. Şifreler BCrypt algoritması kullanılarak hashlenmiş olarak saklanmaktadır. Plain text şifre saklanmaktadır..

Sütun Adı	Veri Tipi	Açıklama	Kısıtlamalar
UserId	INTEGER	Birincil anahtar	PRIMARY KEY, AUTO INCREMENT
Username	VARCHAR(100)	Kullanıcı adı	NOT NULL
Email	VARCHAR(150)	E-posta adresi	NOT NULL, UNIQUE
PasswordHash	VARCHAR(500)	Hashlenmiş şifre	NOT NULL
CreatedAt	TIMESTAMP	Hesap oluşturma tarihi	NOT NULL, DEFAULT CURRENT_TIMESTAMP

2.2. Accounts Tablosu

Kullanıcıların banka hesaplarını saklar. Her hesap bir kullanıcıya aittir (Many-to-One).

İlişkiler:

- Her işlem bir kullanıcıya aittir (Many-to-One)
- Her işlem bir hesaba bağlıdır (Many-to-One)

Kategoriler: Market, Ulaşım, Eğlence, Faturalar, Maaş, Serbest Gelir, vb.

Sütun Adı	Veri Tipi	Açıklama	Kısıtlamalar
TransactionId	INTEGER	Birincil anahtar	PRIMARY KEY, AUTO INCREMENT
UserId	INTEGER	İşlemi yapan kullanıcı	FOREIGN KEY → Users(UserId)
AccountId	INTEGER	İşlemin yapıldığı hesap	FOREIGN KEY → Accounts(AccountId)
TransactionType	VARCHAR(20)	İşlem türü (Income/Expense)	NOT NULL
TransactionAmount	DECIMAL(18,2)	İşlem tutarı	NOT NULL
Category	VARCHAR(50)	Kategori	NOT NULL

Title	VARCHAR(200)	İşlem başlığı	NOT NULL
Notes	TEXT	Opsiyonel notlar	NULL
TransactionDate	TIMESTAMP	İşlem tarihi ve saati	NOT NULL
CreatedAt	TIMESTAMP	Kayıt oluşturma tarihi	NOT NULL

2.3. Budgets Tablosu

Kullanıcıların oluşturduğu bütçeleri saklar.

İlişki: Her bütçe bir kullanıcıya aittir (Many-to-One).

Özel Özellik: Bütçeler otomatik olarak yenilenir. Bitiş tarihi geçtiğinde sistem otomatik olarak yeni bir periyot başlatır.

Sütun Adı	Veri Tipi	Açıklama	Kısıtlamalar
BudgetId	INTEGER	Birincil anahtar	PRIMARY KEY, AUTO INCREMENT
UserId	INTEGER	Bütçe sahibi kullanıcı	FOREIGN KEY → Users(UserId)
BudgetLimit	DECIMAL(18,2)	Bütçe limiti	NOT NULL
SpentAmount	DECIMAL(18,2)	Harcanan tutar	DEFAULT 0
PeriodType	VARCHAR(20)	Periyot tipi (Weekly/Monthly/Yearly)	NOT NULL
StartDate	TIMESTAMP	Bütçe başlangıç tarihi	NOT NULL
EndDate	TIMESTAMP	Bütçe bitiş tarihi	NOT NULL
CreatedAt	TIMESTAMP	Kayıt oluşturma tarihi	NOT NULL

2.4. Subscriptions Tablosu

Tekrarlayan abonelikleri (Netflix, Spotify, vb.) saklar. Her abonelik bir kullanıcıya aittir (Many-to-One).

Sütun Adı	Veri Tipi	Açıklama	Kısıtlamalar
SubscriptionId	INTEGER	Birincil anahtar	PRIMARY KEY, AUTO INCREMENT
UserId	INTEGER	Abonelik sahibi kullanıcı	FOREIGN KEY → Users(UserId)
SubscriptionName	VARCHAR(100)	Abonelik adı	NOT NULL

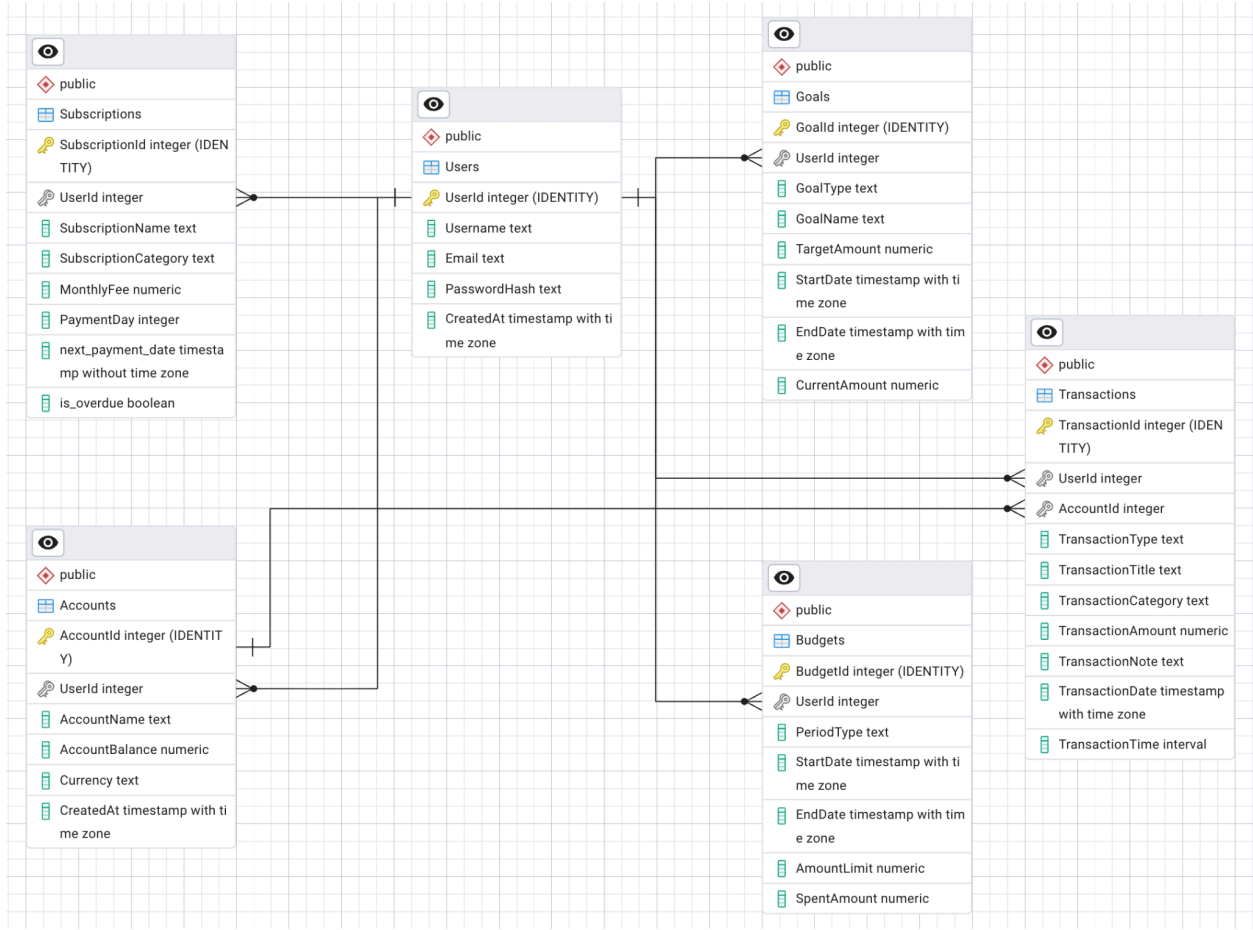
Category	VARCHAR(50)	Kategori	NOT NULL
MonthlyAmount	DECIMAL(18,2)	Aylık tutar	NOT NULL
PaymentDay	INTEGER	Ödeme günü (1-31)	NOT NULL
NextPaymentDate	TIMESTAMP	Bir sonraki ödeme tarihi	NOT NULL
IsActive	BOOLEAN	Aktif durumu	DEFAULT TRUE
CreatedAt	TIMESTAMP	Kayıt oluşturma tarihi	NOT NULL

2.5. Goals Tablosu

Kullanıcıların finansal hedeflerini/birikimlerini saklar. Her hedef bir kullanıcıya aittir (Many-to-One).

Sütun Adı	Veri Tipi	Açıklama	Kısıtlamalar
GoalId	INTEGER	Birincil anahtar	PRIMARY KEY, AUTO INCREMENT
UserId	INTEGER	Hedef sahibi kullanıcı	FOREIGN KEY → Users(UserId)
GoalType	VARCHAR(50)	Hedef türü	NOT NULL
GoalName	VARCHAR(100)	Hedef adı	NOT NULL
TargetAmount	DECIMAL(18,2)	Hedef tutar	NOT NULL
CurrentAmount	DECIMAL(18,2)	Mevcut birikim	DEFAULT 0
StartDate	TIMESTAMP	Başlangıç tarihi	NOT NULL
EndDate	TIMESTAMP	Bitiş tarihi	NOT NULL
CreatedAt	TIMESTAMP	Kayıt oluşturma tarihi	NOT NULL

2.6. İlişkisel Diyagram (ERD)



2.7. Cascade Delete Davranışı

Veritabanında **Cascade Delete** politikası uygulanmıştır:

Örnek: Bir kullanıcı sistemden silindiğinde, ona ait tüm hesaplar, işlemler, bütçeler, abonelikler ve hedefler otomatik olarak silinir. Bu veri bütünlüğünü korumak için kritik öneme sahiptir.

3. REGISTRATION PAGE

- Kullanıcı adı, email adresi ve şifresi girilerek işlemi yapılır.
- Email ve şifre girilerek giriş işlemi yapılır.

Kullanıcı giriş kısmı Şekil 3.1’de, Kullanıcı kayıt kısmı Şekil 3.2’de gösterilmektedir.

15:17

Hoş Geldiniz

Devam etmek için giriş yapın veya kayıt olun

Giriş Yap Kayıt Ol

E-posta

ornek@email.com

Şifre

.....

Giriş Yap

Şekil 3.1 Kullanıcı giriş kısmı

15:18

Hoş Geldiniz

Devam etmek için giriş yapın veya kayıt olun

Giriş Yap Kayıt Ol

Ad Soyad

Adınız Soyadınız

E-posta

ornek@email.com

Şifre

.....

Kayıt Ol

Şekil 3.2 Kullanıcı kayıt kısmı

4. HOME PAGE

4.1. Accounts

- Bakiye ve hesap adı bilgileri girilerek yeni hesap eklenebilir ve oluşturulmuş hesap kartının üstüne basılı tutularak hesap silinebilir.
- Ana sayfada hesap kartları halinde hesap adı ve o hesaba ait bakiye gösterilir.

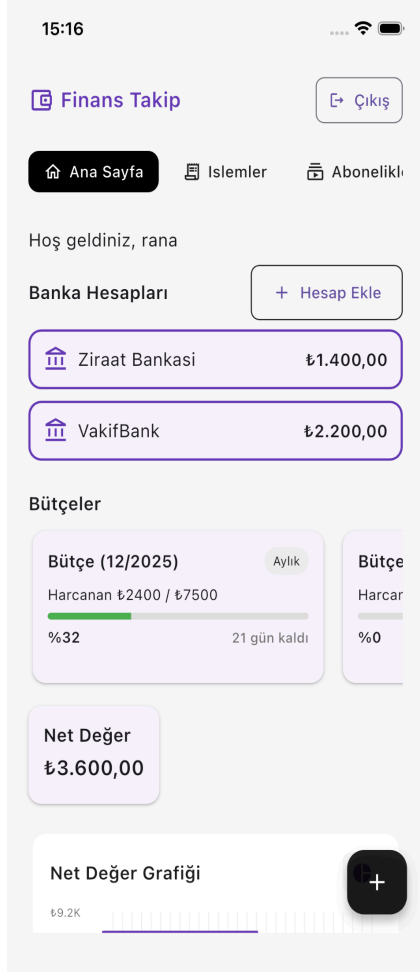
4.2. Net Worth + Graph

- Kullanıcının toplam bakiyesi gösterilir.
- Gelir ve giderler tek çizgi grafik ile gösterilir.
- Kategori bazlı pasta grafiği gösterilir.

4.3. Transactions + Budgets

- Kullanıcının son 30 işlemi görüntülenir.
- İşlemler tarihlerine göre gruplandırılır.
- İşlemlerin gider veya gelir olduğu, ne kadar olduğu ve ne zaman işleme geçildiği gösterilir.
- Hızlı işlem oluşturmak için “+ İşlem Ekle” butonu ana sayfada da yer alır (butonun içeriği Transactions kısmında açıklanacak).
- Yeni bütçe oluşturma kartı ile bütçenin periyot tipi (haftalık, aylık, yıllık), bütçenin başlangıç tarihi (bitiş tarihi belirlenen periyot türüne göre otomatik oluşturulur) ve bütçe limit bilgileri girilerek kolay bir şekilde yeni bütçe oluşturulur.
- Oluşturulan bütçe kartında bütçenin limitiyle birlikte harcanan miktar, progress çubuğu, bütçenin yüzde kaçının harcandığı bütçe bitimine ne kadar kaldığı ve periyot tipi bulunur.
- Bütçe kartına basıldığında bütçe geçmişleri görünür.
- Bütçe kartına basılı tutulduğunda bütçe güncelleme ve silme seçenekleri bulunur.
- Bütçe silinmediği sürece bütçe seçilen periyodik süre türünde otomatik yenilenmeye devam eder.

Ana sayfa Şekil 4.1, Şekil 4.2’ de gösterilmektedir.



Şekil 4.1 Home Page

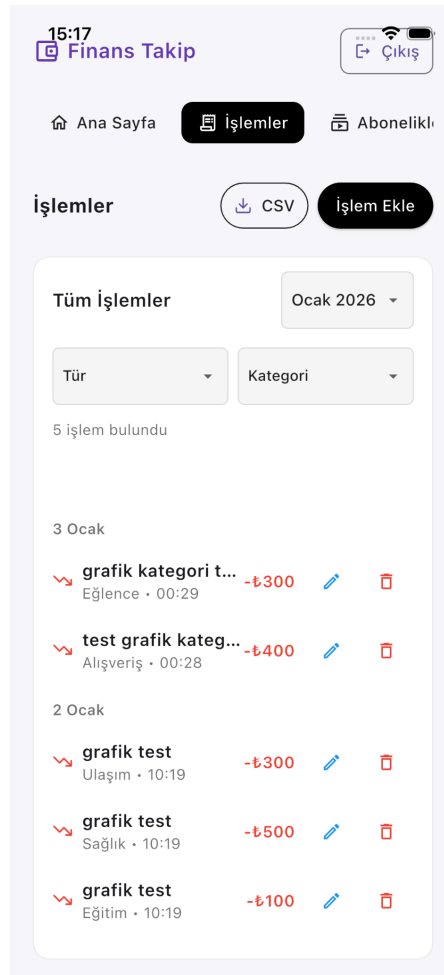


Şekil 4.2 Home Page

5. TRANSACTIONS PAGE

- İşlemler aylara, kategorilere ve gelir-gider olduğuna göre filtrelenebilir.
- “CSV İndir” butonu ile filtrelenen işlemler csv dosyası olarak indirilir.
- “+ Yeni İşlem” butonu ile işlem türü, tarih, saat, tutar, banka hesabı (kullanıcıya ait hesaplar dropdown ile seçilir), başlık kategori ve opsiyonel bir not bilgileri girilerek kolayca yeni bir işlem oluşturulur.
- Her bir işlem güncellenebilir ve silinebilir. Silinen transaction hangi hesaptan harcandıysa otomatik olarak o hesaba transaction’da harcanan para geri eklenir.

Transaction sayfası Şekil 5.1’de gösterilmektedir.

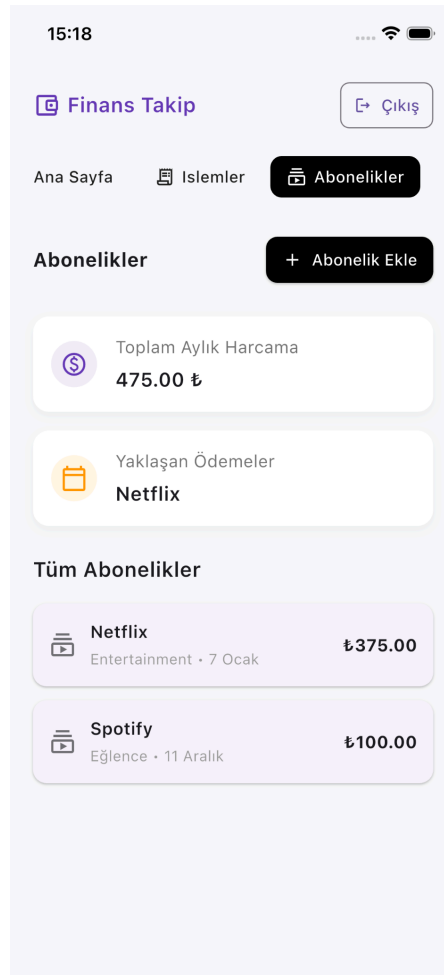


Şekil 5.1 Transaction sayfası

6. SUBSCRIPTION PAGE

- Aboneliklere giden toplam aylık harcama, aktif abonelik sayısı ve yaklaşan ödeme sayısı gösterilir.
- Tüm abonelikler kartlar şeklinde listelenir.
- Bir abonelik kartına tıklandığında aboneliğin ödemesi yapılabilir, aboneliğin o aylık ödemesi atlanabilir, abonelik silinebilir.
- “+ Yeni Abonelik” butonu ile abonelik adı, kategori, aylık tutar ve ödeme günü bilgileri girilerek kolayca yeni bir abonelik oluşturulur.

Subscription sayfası Şekil 6.1’de gösterilmektedir.

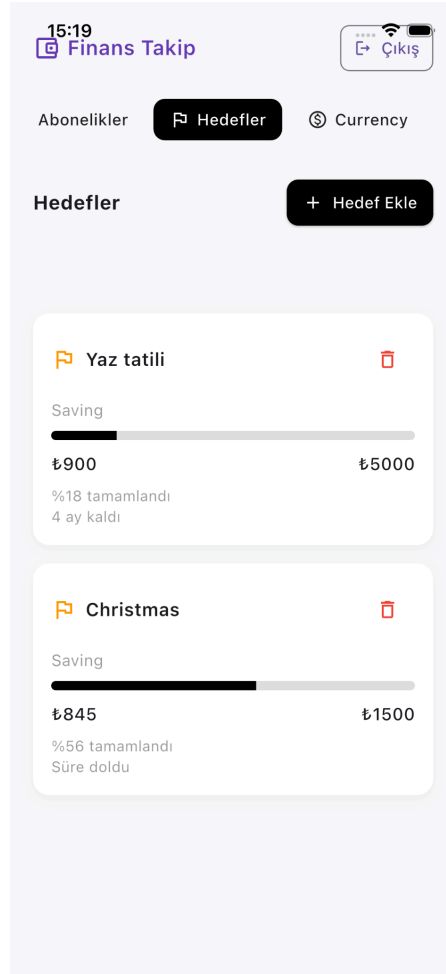


Şekil 6.1 Subscription sayfası

7. GOALS PAGE

- Oluşturulan hedef kartlarında progress çubuğu, yüzdelik, hedeflenen tutar, hedef adı ve hedefi silmek için bir ikon bulunur.
- Bir hedef kartına tıklanıldığında hedefte/birikimde olan mevcut para miktarı, hedeflenen tutar, hedefe ulaşmak için kalan miktar progress çubuğu ve yüzdelik ile birlikte verilir. Altında ise hedefe katkı eklemek için hesap (kullanıcının var olan hesaplarından), katkı miktarı ve opsiyonel bir not bilgileri girilir.
- “+ Yeni Hedef” butonu ile hedef türü, hedef adı, hedef miktarı, başlangıç tarihi ve bitiş tarihi bilgileri girilerek kolayca yeni bir hedef/birikim oluşturulur.

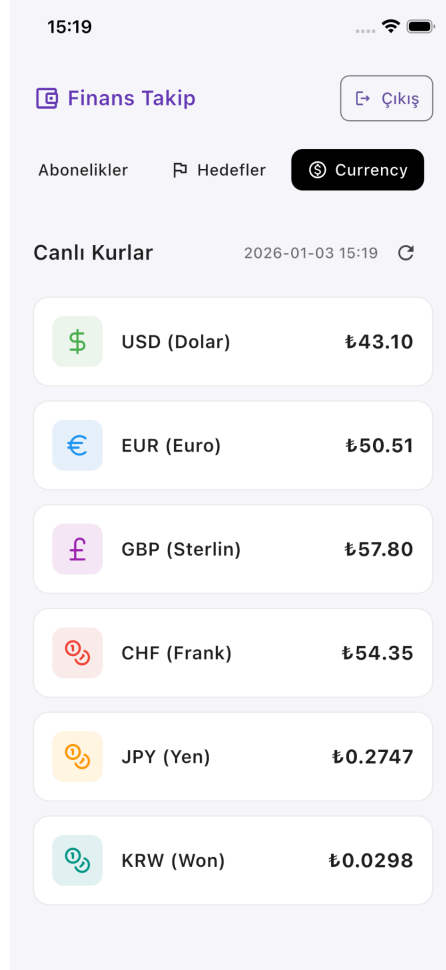
Goal sayfası Şekil 7.1’de gösterilmektedir.



Şekil 7.1 Goal sayfası

8. CURRENCY PAGE

- USD, EUR, GBP, CHF, JPY, KRW ekranda -TRY bazlı- gösterilir. Currency sayfası Şekil 8.1’de gösterilmektedir.



Şekil 8.1 Currency sayfası