# Backend Developer Case Study – Banka Ekstre Yükleme ve KPI Raporlama

Backend Developer Case Study – Banka Ekstre Yükleme ve KPI Raporlama

1. Proje Tanımı  
Bu proje, kullanıcıların bankadan indirdikleri ekstre dosyalarını (CSV) sisteme yükleyerek verilerini kaydedebilecekleri,  
filtreleyebilecekleri ve belirli tarih aralıklarında finansal KPI’larını görüntüleyebilecekleri bir REST API geliştirilmesini amaçlamaktadır.

Ana iş akışı:  
1. Kullanıcı sisteme kayıt olur ve giriş yapar.  
2. Banka ekstresini CSV formatında yükler.  
3. Sistem verileri doğrular, normalize eder ve veritabanına kaydeder.  
4. Kullanıcı işlemlerini filtreleyerek görüntüler.  
5. Kullanıcı belirli bir tarih aralığında finansal KPI raporlarını alır.

2. Teknik Gereksinimler  
- Backend Framework: Python 3.10+ ve Django 4.x, Django REST Framework  
- Veritabanı: PostgreSQL  
- Kimlik Doğrulama: JWT veya DRF Token Authentication  
- Dokümantasyon: Swagger veya OpenAPI  
- Test: Django Test Framework veya Pytest (temel senaryolar)

3. API Gereksinimleri

3.1. Kullanıcı Yönetimi  
- POST /auth/register (email, password)  
- POST /auth/login (email, password)  
- Tüm veri işlemleri kullanıcıya özel olmalı.

3.2. Banka Ekstre Yükleme  
- POST /transactions/upload  
 - Headers: Authorization, Idempotency-Key  
 - multipart/form-data ile CSV yükleme  
 - Duplicate kontrolü ve atomic işlem  
 - Örnek CSV:  
 date,amount,currency,description,type  
 2025-07-01,4500.00,TRY,"Satış: Fatura #1023",credit  
 2025-07-02,-1200.00,TRY,"Kira Ödemesi",debit

3.3. İşlem Listeleme  
- GET /transactions (filtre parametreleri)

3.4. KPI Raporlama  
- GET /reports/summary (start\_date, end\_date)  
- Dönen bilgiler: total\_income, total\_expense, net\_cash\_flow, top\_expense\_categories

4. Veri Modeli  
- User (Django default)  
- Transaction: user, date, amount, currency, type, description, category, unique\_hash  
- ImportBatch (opsiyonel): batch bilgileri

5. Teslim Formatı  
- GitHub repo linki  
- README.md (kurulum, kullanım, örnek istekler)  
- Migration dosyaları  
- Swagger/OpenAPI endpoint  
- Örnek CSV  
- Minimum 5–6 test

6. Değerlendirme Kriterleri  
- Veri modeli: 20p  
- API doğruluğu: 20p  
- Kod kalitesi: 20p  
- Test kapsamı: 15p  
- Hata yönetimi: 15p  
- Dokümantasyon: 10p

7. Ekstra Puan  
- Docker Compose  
- Otomatik kategorileme  
- Celery ile haftalık rapor  
- Para birimi dönüştürme desteği

Not: Bu proje, gerçek hayattaki MMM platformunun bir bölümünü simüle eder.