

# MOBİL UYGULAMA GELİŞTİRME PROJE RAPORU

**AD SOYAD: METİN AYDIN**

**NUMARA: G221210383**

**GRUP : 2A**

## 1. GİRİŞ VE PROJENİN AMACI

Bu projenin temel amacı, kullanıcıların çalışma verimliliğini artırmak ve dijital dikkat dağılığını ile mücadele etmektir. "Odaklanma Takibi" uygulaması, Pomodoro tekniği benzeri bir zamanlayıcı ile kullanıcıların belirli sürelerde odaklanması sağlar, uygulama arka plana atıldığında (başka uygulamaya geçildiğinde) bunu "dikkat dağılığı" olarak tespit eder ve kayıt altına alır. Elde edilen veriler, bulut tabanlı bir veritabanında saklanarak kullanıcıya grafiksel raporlar halinde sunulur.

## 2. SİSTEM MİMARİSİ

Uygulama, modern mobil uygulama geliştirme standartlarına uygun olarak **Sunucusuz (Serverless)** mimari ile tasarlanmıştır.

- İstemci (Client):** React Native (Expo) çerçevesi kullanılarak geliştirilen mobil arayüz.
- Sunucu/Veritabanı (Backend):** Google Firebase platformu kullanılmıştır. Veri saklama işlemi için NoSQL tabanlı **Cloud Firestore** tercih edilmiştir.
- Kimliklendirme Mimarisi:** Kullanıcı deneyimini zorlaştıran "Kayıt Ol/Giriş Yap" ekranları yerine, **Cihaz Bazlı Kimliklendirme (Device-Based Authentication)** mimarisi kullanılmıştır. Uygulama ilk açılışta cihaza özgü benzersiz bir ID (UUID) üretir ve tüm verileri bu ID ile eşleştirir. Bu sayede veri güvenliği sağlanırken kullanım kolaylığı maksimize edilmiştir.

## 3. KULLANILAN TEKNOLOJİLER VE KÜTÜPHANELER

Proje içerisinde aşağıdaki temel modüller ve kütüphaneler kullanılmıştır:

- React Native & Expo:** Çapraz platform (iOS/Android) mobil uygulama geliştirme altyapısı.
- Firebase Firestore:** Odaklanma oturumlarının JSON formatında bulutta saklanması.
- AsyncStorage:** Cihaz ID'sinin ve kullanıcı tarafından eklenen özel kategorilerin yerel hafızada tutulması.
- Expo Haptics:** Kullanıcı butonlara bastığında veya süre bittiğinde titreşimsel geri bildirim sağlamak için.
- React Native Chart Kit:** Verilerin Pasta (Pie) ve Çubuk (Bar) grafiklerle görselleştirilmesi.

6. **React Navigation:** Alt menü (Tab Navigation) ile sayfalar arası geçişin yönetilmesi.
7. **AppState API:** Uygulamanın ön planda mı yoksa arka planda mı olduğunu (dikkat dağılıklığı) tespiti.

## 4. PROJE MODÜLLERİ VE EKRANLAR

Uygulama temel olarak 2 ana ekranlı ve yardımcı modüllerden oluşmaktadır:

### 4.1. Ana Sayfa (Zamanlayıcı Ekranı - HomeScreen.js)

Bu modül, uygulamanın operasyonel merkezidir. İçerisinde şu bileşenler bulunur:

- **Zamanlayıcı Motoru:** setInterval kullanılarak oluşturulan geri sayım mekanizması.
- **Kategori Yönetimi:** Kullanıcının listeden seçim yapabildiği veya manuel olarak yeni kategori ekleyebildiği modül (AsyncStorage destekli).
- **Süre Ayarlama Modülü:** Kullanıcının 15, 25, 60 dk gibi hazır butonları kullanabildiği veya manuel (örn: 43 dk) giriş yapabildiği alan.
- **Motivasyon Bileşeni:** Her açılışta rastgele bir motivasyon sözü gösteren algoritma.
- **Dikkat Takip Algoritması:** Kullanıcı uygulamadan çıkış yaparsa sayacı duraklatan ve sayaç değerini artıran olay dinleyicisi (Event Listener).

## Odaklanma Takibi



# 25:00

Hedef: 25 dk

Başlat

Vazgeç ve Sıfırla

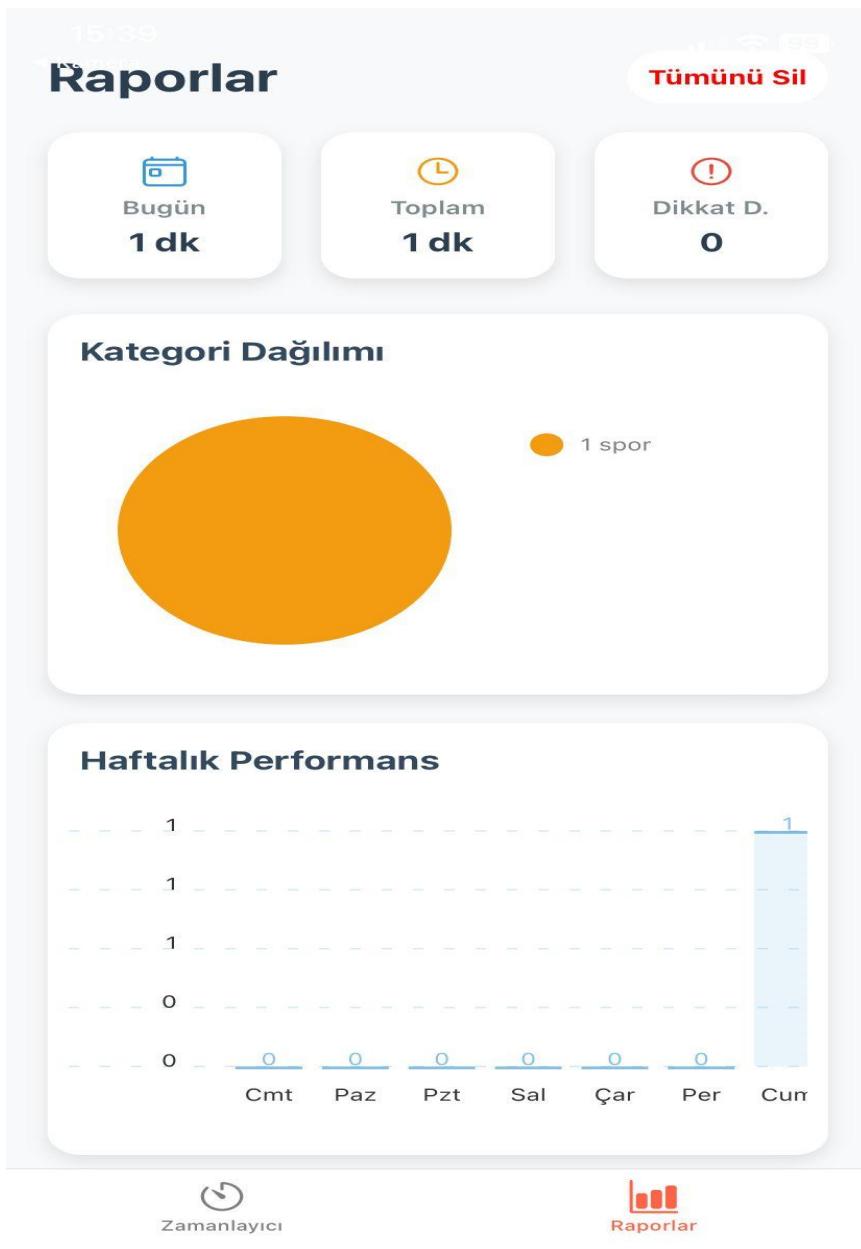


"Zaman en değerli sermayendir."

## 4.2. Raporlar Ekranı (Dashboard - ReportsScreen.js)

Bu modül, verilerin analiz edildiği kısımdır.

- Veri Çekme Servisi:** Firestore'dan sadece o cihaza ait (userId filtresi ile) verileri çeker.
- İstatistik Motoru:** Çekilen ham verileri işleyerek "Bugünkü Toplam Süre", "Haftalık Dağılım" ve "Kategori Yüzdeleri"ni hesaplar.
- Görselleştirme:** Hesaplanan verileri Grafik kütüphanesi aracılığıyla kullanıcıya sunar.
- Geçmiş Listesi ve Yönetim:** Son aktiviteleri listeleyen ve tekil/toplu silme işlemi yapan yönetim paneli.



## 5. VERİ MODELİ (DATABASE SCHEMA)

Firestore veritabanında focusSessions koleksiyonu altında tutulan her bir dökümanın yapısı şöyledir:

Alan Adı	Veri Tipi	Açıklama
id	String (Auto)	Firestore tarafından üretilen benzersiz kayıt ID'si.
userId	String	Cihaza özgü üretilen benzersiz kullanıcı kimliği.
category	String	Odaklanma kategorisi (Örn: "Ders", "Spor").
duration	Number	Çalışılan süre (dakika cinsinden).
distractions	Number	Uygulamadan kaç kez çıktıgı sayısı.
date	String	İşlemin yapıldığı tarih (YYYY-MM-DD).
createdAt	Timestamp	Sıralama işlemleri için tam zaman damgası.

The screenshot shows the Firestore console interface. On the left, the 'focusSessions' collection is listed with several documents: 4npRak4at18xHnCRLT60, Kgcyt1pX5KAovGsg2vVZ, TvwU5pcUvxRkD93CQ3l8 (selected), WpBdvIbpUdp6Czu2LsGx, and rA1UOCRAvieBbemxv5Hz. The selected document 'TvwU5pcUvxRkD93CQ3l8' has its details expanded on the right. The document contains fields: category: "spor", createdAt: "2025-12-12T11:59:50.784Z", date: "2025-12-12", distractions: 0, duration: 1, and userId: "user-1765540414030-508".

## 6. İŞ AKIŞ ŞEMASI (ALGORİTMA)

1. **BAŞLAT:** Uygulama açılır.
2. **KİMLİK KONTROLÜ:** Cihaz hafızasında ID var mı?
  - *Hayır:* Yeni ID üret -> Hafızaya Kaydet.
  - *Evet:* Mevcut ID'yi yükle.
3. **KULLANICI GİRİŞİ:** Kullanıcı Kategori seçer ve Süreyi belirler.
4. **SAYAC:** "Başlat" butonuna basılır -> Geri sayım başlar.
5. **OLAY DİNLEME (DÖNGÜ):**
  - *Uygulama Arka Plana mı geçti?* -> EVET -> Sayacı Durdur & Dikkat Dağılıklığı Sayacını Artır.
  - *Süre Bitti mi?* -> EVET -> Adım 6'ya git.
6. **KAYIT:** Veriler (userId ile birlikte) Firebase'e gönderilir.
7. **RAPORLAMA:** Raporlar ekranına gidildiğinde Firebase'den userId ile sorgu yapılır ve grafikler çizilir.

