



YAZILIM BİLİŞİM AKADEMİSİ

BITİRME PROJESİ RAPORU

Go Influence

Influencer Marketing Platformu / Marka - Influencer İş Birliği Yönetim Sistemi

Furkan Yorulmaz

Temmuz 2026

İstanbul

İÇİNDEKİLER

1. GİRİŞ	3
1.1. Projenin Amacı	3
1.2. Proje Kapsamı	3
1.3. Proje Kısıtları	3
2. MATERYAL VE YÖNTEM	4
2.1. Kullanılan Teknolojiler	4
2.2. React ve Vite Yapısı	4
2.3. Firebase Kullanımı	4
2.4. Redux Toolkit Kullanımı	5
2.5. Firestore Security Rules	5
3. SİSTEM TASARIMI	6
3.1. Genel Mimari	6
3.2. Rol Bazlı Kullanıcı Akışı	6
3.3. Veritabanı Koleksiyonları	7
3.4. Kategori Bazlı Eşleşme Algoritması	8
3.5. Kampanya Yaşam Döngüsü	8
4. MODÜL GELİŞTİRME	9
4.1. Landing ve Auth Modülü	9
4.2. Marka Paneli	10
4.3. Influencer Paneli	10
4.4. Admin Paneli	10
4.5. Teklif ve Başvuru Modülü	10
4.6. Mesajlaşma Modülü	10
5. GÜVENLİK	11
6. TEST VE DOĞRULAMA	12

7. SONUÇ

13

8. KAYNAKÇA

13

1. GİRİŞ

Dijital pazarlama dünyasının en dinamik ve hızlı büyüyen alanlarından biri olan influencer marketing, markaların hedef kitlelerine daha samimi ve organik yollarla ulaşmasını sağlamaktadır. Ancak geleneksel ajans modelleri ve manuel yürütülen iş birliği süreçleri, hem markalar hem de bağımsız içerik üreticileri (influencer) için zaman alıcı ve yönetilmesi zor bir yapı oluşturmaktadır. Bütçe planlamasından içerik doğrulamasına, tekliflerin takibinden gerçek zamanlı mesajlaşmaya kadar olan süreçlerin dijitalleştirilmesi gerekliliği, Go Influence projesinin çıkış noktasını oluşturmuştur.

1.1. Projenin Amacı

Go Influence, markalar ile kreatif içerik üreticilerini tek bir ortak platformda buluşturan modern, web tabanlı bir influencer marketing paneli ve iş birliği yönetim sistemidir. Projenin temel amacı; markaların kampanya başlatma, uygun hedef kitleye sahip influencer'ları bulma, teklif iletme, başvuru toplama ve güvenli iletişim kanalları kurma süreçlerini merkezi, veritabanı destekli, şeffaf ve ölçülebilir bir yapıya kavuşturmasıdır.

Sistem, influencer'ların ajans bağımlılığını azaltarak kendi portfolyolarını sergilemelerine, fiyatlandırma ve hizmet detaylarını doğrudan belirlemelerine imkan tanımaktadır. Aynı zamanda admin rolü sayesinde platform genelindeki işlemler denetlenmekte, güvenli ve aktif bir topluluk yapısı korunmaktadır.

1.2. Proje Kapsamı

Go Influence projesi, uçtan uca rol bazlı erişim kontrolü sağlayan web arayüzleri, gerçek zamanlı NoSQL veritabanı katmanı ve güvenli dosya depolama mekanizmalarını kapsamaktadır. Geliştirilen uygulama; Landing tanıtım sayfası, kimlik doğrulama modülü, marka, influencer ve admin kontrol panelleri, gelişmiş kategori bazlı eşleşme algoritması, teklif ve başvuru takip mekanizması ile Firestore tabanlı anlık mesajlaşma sistemini içermektedir.

1.3. Proje Kısıtları

Geliştirme sürecinde dikkate alınan ve projenin sınırlarını belirleyen temel kısıtlar şunlardır:

- **Yapay Zeka Yapısının İptali:** Projenin başlangıç aşamasında planlanan yapay zeka tabanlı otomatik eşleşme modeli, test edilebilirliği zorlaştırması, yüksek API maliyetleri ve şeffaflık kısıtları nedeniyle iptal edilmiştir. Sistem, tamamen deterministik ve optimize edilmiş manuel kategori/hizmet eşleşme motoruna dönüştürülmüştür.
- **Gerçek Ödeme Altyapısı:** Finansal işlemler ve komisyon takipleri için sistem altyapısı tasarlanmış ancak BDDK kuralları ve test ortamı kısıtları nedeniyle fiziksel sanal pos entegrasyonu simüle edilmiş (mock) veri yapısı üzerinden gerçekleştirilmiştir.
- **Sosyal Medya API Sınırlandırmaları:** Instagram, TikTok ve YouTube gibi platformların katı veri gizliliği politikaları ve API erişim izinleri nedeniyle, influencer'ların takipçi sayıları ve etkileşim oranları platform içerisinde beyan usulü ve seed veriler aracılığıyla doğrulanmıştır.

2. MATERİYAL VE YÖNTEM

Go Influence projesinin geliştirilmesinde, modern web standartlarına uygun, yüksek performanslı ve ölçeklenebilir yazılım teknolojileri tercih edilmiştir. Geliştirme sürecinde "Single Page Application (SPA)" mimarisi ve "Backend-as-a-Service (BaaS)" modeli entegre edilerek hızlı ve güvenli bir platform yapısı elde edilmiştir.

2.1. Kullanılan Teknolojiler

Projenin tamamlanmasında kullanılan kütüphane, framework ve servis listesi aşağıda sunulmuştur:

- **React 19 & Vite 6:** İstemci tarafı uygulama mimarisinin kurulması ve hızlı modül sıcak yükleme (HMR) süreçleri için kullanılmıştır.
- **Redux Toolkit & React Redux:** Global state (durum) yönetimi, kullanıcı oturumlarının takibi ve paneller arası senkronizasyon için tercih edilmiştir.
- **Firestore Auth:** Güvenli e-posta ve şifre tabanlı kimlik doğrulama süreçleri için kullanılmıştır.
- **Firestore Firestore:** NoSQL yapısında, anlık (real-time) veri senkronizasyonu sağlayan veritabanı çözümü olarak konumlandırılmıştır.
- **Firestore Storage:** Kampanya banner'ları ve influencer profil görsellerinin bulut üzerinde depolanması amacıyla entegre edilmiştir.
- **Tailwind CSS 4:** Minimalist, modern ve tamamen duyarlı (responsive) arayüz bileşenlerinin tasarlanmasında kullanılmıştır.
- **Lucide React & Motion React:** Platform genelindeki ikonografi ve akıcı kullanıcı arayüzü animasyonları için tercih edilmiştir.

2.2. React ve Vite Yapısı

Vite 6, geliştirme ortamında sağladığı esneklik ve esbuild tabanlı derleme hızı sayesinde proje omurgası olarak seçilmiştir. React 19'un getirdiği yenilikçi eş zamanlı render yetenekleri ve performans optimizasyonları sayesinde, büyük veri setlerine sahip influencer listeleri ve gerçek zamanlı sohbet ekranları donma yaşanmadan akıcı bir şekilde çalıştırılmaktadır.

2.3. Firestore Kullanımı

Uygulamanın veri tabanı, dosya depolama ve kimlik yönetimi ihtiyaçları Firestore ekosistemi üzerinden çözülmüştür. Sunucusuz (Serverless) mimari sayesinde ek sunucu maliyetleri ve veri tabanı yönetim karmaşası engellenmiştir. Firestore'un döküman ve koleksiyon yapısı, ilişkisel olmayan ancak esnek veri yapıları tanımlamamıza imkan tanımıştır. Firestore Auth entegrasyonu ile JWT (JSON Web Token) tabanlı oturum yönetimi otomatik olarak sağlanmış, Firestore Storage ile de yüksek boyutlu medya dosyalarının güvenli yüklenmesi gerçekleştirilmiştir.

2.4. Redux Toolkit Kullanımı

Kullanıcı rolleri (brand, influencer, admin) ve oturum verileri Redux mimarisi ile kontrol edilmektedir. Veri tabanından çekilen anlık bildirimler, kampanya durumları ve platform ayarları "slice" yapıları altında toplanarak veri kirliliği ve gereksiz render işlemlerinin önüne geçilmiştir. `authSlice.js`, `brandPanelSlice.js` ve `influencerPanelSlice.js` gibi modüller sayesinde uygulama temiz ve sürdürülebilir bir kod tabanına kavuşmuştur.

2.5. Firestore Security Rules

İstemci tabanlı Firebase doğrudan erişimini sınırlandırmak ve veri güvenliğini en üst düzeye çıkarmak amacıyla `firestore.rules` dosyası yapılandırılmıştır. Rol bazlı erişim yetkilendirmeleri veri tabanı seviyesinde kontrol edilerek, kötü niyetli kullanıcıların yetkileri dışındaki dokümanları okuması veya değiştirmesi kesin olarak engellenmiştir.

3. SİSTEM TASARIMI

Sistem tasarımı, kullanıcıların platformu en verimli şekilde kullanabilmesi ve verilerin bütünlüğünün korunması amacıyla modüler ve katmanlı olarak kurgulanmıştır.

3.1. Genel Mimari

Go Influence, sunucusuz bulut mimarisi üzerine inşa edilmiştir. Kullanıcı tarayıcı üzerinden React arayüzü ile etkileşime geçtiğinde, doğrudan Firebase servisleriyle konuşur. İstemci yetkilendirmesi Firebase Auth üzerinden doğrulanarak Firestore ve Storage güvenlik kurallarına takılır. Bu sayede harici bir API katmanına gerek kalmadan yüksek hızlı veri akışı sağlanır.

3.2. Rol Bazlı Kullanıcı Akışı

Kullanıcı Firebase Auth ile kayıt olduktan veya giriş yaptıktan sonra, otomatik olarak Firestore üzerindeki ``users/{uid}`` dökümanı okunur. Bu dökümanda yer alan ``role`` alanına göre sistem yönlendirme yapar:

- **Brand (Marka):** Marka dashboard ekranına ve kampanya yönetim araçlarına yönlendirilir.
- **Influencer:** Eğer ilk girişi ise onboarding (profil tamamlama) sürecine, profili onaylı ise influencer paneline yönlendirilir.
- **Admin:** Sistem genelindeki onay bekleyen kullanıcıları ve platform istatistiklerini izleyebileceği admin paneline yönlendirilir.

Kullanıcıların erişim hakları ayrıca ``status`` (aktif, onay bekliyor, reddedildi, donduruldu) alanlarına göre dinamik olarak kısıtlanır. Örneğin; onay bekleyen (pending) veya reddedilen (rejected) bir influencer, kampanya fırsatlarına erişemez.

3.3. Veritabanı Koleksiyonları

Platformda kullanılan temel Firestore koleksiyonları ve şemaları aşağıda detaylandırılmıştır:

Koleksiyon Adı	Açıklama ve Kritik Alanlar
users	Kullanıcı rolü ve durum bilgisi. Alanlar: uid, email, role, status, onboardingComplete, isActive.
brands	Marka profili ve kurumsal detaylar. Alanlar: uid, companyName, website, categories, sector, logoUrl.
influencers	Influencer portfolyo, fiyat ve kitle bilgileri. Alanlar: uid, username, categories, services, pricing, profileVisibility.
campaigns	Markaların oluşturduğu kampanyalar. Alanlar: id, brandUid, title, categories, budgetRange, bannerUrl, status.
campaignMatches	Kampanyalar ile influencer'lar arasındaki uyum eşleşmeleri. Alanlar: id, campaignId, influencerUid, score, reasons.
campaignOffers	Gönderilen teklifler ve başvurular. Alanlar: campaignId, brandUid, influencerUid, amount, status (Pending, Accepted, Rejected).
chatThreads	Marka ve influencer sohbet odaları. Alanlar: id, participants, messages (array), lastMessageText.
transactions	Finansal komisyon ve bütçe transfer kayıtları. Alanlar: campaignId, grossAmount, feePercent, status.

3.4. Kategori Bazlı Eşleşme Algoritması

Yapay zeka katmanının iptal edilmesinden sonra, `src/utils/interestMatchingEngine.js` dosyası altında tamamen performans odaklı çalışan bir statik eşleşme motoru geliştirilmiştir. Bu motor, marka kampanyaları ile influencer niteliklerini matematiksel formüllerle analiz eder.

Eşleşme motorunun çalışma kriterleri şunlardır:

- **Kategori Uyumu (Ağırlık: %70):** Kampanya kategorileri (örn: Teknoloji, Moda) ile influencer ilgi alanları karşılaştırılır. Kesişim kümesinin boyutu skurun bu kısmını belirler.
- **Hizmet/İçerik Türü Uyumu (Ağırlık: %30):** Kampanyanın talep ettiği içerik türü (Reels, Story, YouTube Post vb.) ile influencer'ın sunduğu hizmet listesi karşılaştırılır.
- **Eşik Değer:** Toplam uyum skoru sıfır olan veya influencer'ın ayarlarında belirlediği `minimumMatchScore` limitinin altında kalan eşleşmeler sistem tarafından süzülür ve listelenmez.

Uyum Skoru Formülü:

$$Skor = (Kategori_Uyumu_Orani \times 70) + (Hizmet_Uyumu_Orani \times 30)$$

3.5. Kampanya Yaşam Döngüsü

Bir kampanyanın oluşturulmasından sonlanmasına kadar geçen süreç dinamik olarak izlenir. Sistem, `src/utls/campaignStatus.js` yardımcı dosyası aracılığıyla her yüklemede başlangıç ve bitiş tarihlerini kontrol eder. Bitiş tarihi geçmiş olan kampanyalar otomatik olarak `isExpired: true` ve `isActive: false` statüsüne geçirilir. Bu kampanyaların durum etiketleri "Süresi Geçmiş" olarak güncellenir ve influencer arama listelerinden kaldırılır.

4. MODÜL GELİŞTİRME

Uygulama, yüksek modülerlik prensibiyle geliştirilmiştir. Projenin kaynak kod yapısı, temiz kod standartlarına uygun olarak yapılandırılmıştır:

```
src/  
  app/  
    store.js  
  components/  
    common/  
    guards/  
  features/  
    admin/  
      adminService.js  
      adminSlice.js  
    auth/  
      authService.js  
      authSlice.js  
      AuthBootstrap.jsx  
    brand/  
      brandPanelSlice.js  
    influencer/  
      influencerPanelSlice.js  
    settings/  
      platformSettingsSlice.js  
  pages/  
    admin/  
      AdminPage.jsx  
    brand/  
      BrandPanelPage.jsx  
    influencer/  
      InfluencerPanelPage.jsx  
    landing/  
      LandingPage.jsx  
  utils/  
    campaignStatus.js  
    interestMatchingEngine.js
```

4.1. Landing ve Auth Modülü

Ziyaretçileri karşılayan Landing sayfası, platformun çalışma mantığını anlatan minimalist animasyonlar (Motion React) içerir. Auth modülü ise kayıt adımlarında rollerine göre kullanıcıları ayırıştırır. Onay bekleyen influencer'lar için geliştirilen `PendingApproval.jsx` ekranı, admin onayı gelene kadar panellere erişimi engeller.

4.2. Marka Paneli

Marka kullanıcıları için tasarlanmış panelde; dashboard veri özetleri, yeni kampanya oluşturma sihirbazı, kampanya düzenleme ekranı, banner yükleme ve anlık önizleme alanları yer alır. Markalar, kampanyalarını yayına aldıkları an arka planda çalışan eşleşme algoritması sayesinde en uygun kreatif listesine anında ulaşabilir ve tek tıkla teklif gönderebilirler.

4.3. Influencer Paneli

Influencer'lar kendilerine özel panel üzerinden gelen teklifleri görebilir, marka ihtiyaçları (aktif kampanyalar) havuzunu tarayabilir, fiyat ve profil bilgilerini anlık olarak güncelleyebilirler. Geliştirilen şifre değiştirme modülü sayesinde kullanıcılar mevcut şifrelerini doğrulayıp yeni şifre atayabilirler, bu işlemten sonra güvenlik amacıyla oturumları sonlandırılır.

4.4. Admin Paneli

Sistem yöneticileri (admin); bekleyen influencer başvurularını inceler, onay süreçlerini yönetir, aktif kampanyaları takip eder ve platform genel komisyon oranlarını güncelleyebilir. Bir influencer admin tarafından onaylandığında, sistem arka planda onun için otomatik olarak kampanya eşleşmelerini tetikler.

4.5. Teklif ve Başvuru Modülü

Marka ile influencer arasındaki finansal ve iş birliği adımları bu modülle kontrol edilir. Teklifler doğrudan `campaignOffers` koleksiyonuna yazılır. Durum değişiklikleri (Pending, Accepted, Rejected) her iki tarafın panelinde de anlık olarak güncellenir.

4.6. Mesajlaşma Modülü

Kullanıcılar arası iletişim, Firestore tabanlı anlık mesajlaşma sistemiyle yürütülür. Marka teklif gönderdiğinde veya influencer başvuru yaptığında `ensureChatThread` fonksiyonu çalışarak benzersiz bir sohbet odası (`brandUid_influencerUid`) oluşturur. Mesajlar, Firestore'un `arrayUnion` yapısı kullanılarak anlık olarak iletilir ve dinlenir.

5. GÜVENLİK

Bir SaaS platformu olarak Go Influence, veri gizliliği ve rol güvenliğine üst düzeyde önem vermektedir. Tüm veri tabanı işlemleri sunucu tarafında doğrulanmış kurallara tabidir.

5.1. Firebase Auth ve Oturum Güvenliği

Kimlik doğrulama süreçleri tamamen Firebase Auth tarafından yönetilir. Şifreler istemci tarafında veya veritabanında düz metin olarak asla tutulmaz, şifreli JWT anahtarları üzerinden oturumlar doğrulanır. Ayarlar sayfasında şifre güncelleme yapıldığında, kullanıcının oturumu güvenlik amacıyla otomatik olarak kapatılır ve tekrar giriş yapması istenir.

5.2. Veritabanı Erişim Kuralları (Firestore Security Rules)

Firestore security rules aracılığıyla tanımlanan kritik erişim kuralları şunlardır:

- **users/{uid}**: Yalnızca dökümanın sahibi olan kullanıcı kendi bilgilerini okuyabilir ve güncelleyebilir.
- **brands/{uid} & influencers/{uid}**: Sadece oturum açmış kullanıcılar okuyabilir; yazma/güncelleme yetkisi ise yalnızca döküman sahibine ve admin rolündeki kullanıcılara aittir.
- **campaigns**: Oturum açmış tüm kullanıcılar okuyabilir; ancak yeni kampanya oluşturma ve güncelleme yetkisi yalnızca ilgili markanın kendisine tanımlanmıştır.
- **chatThreads**: Bir sohbet odasına yalnızca odanın katılımcıları (`participants` dizisi içinde UID'si yer alan kullanıcılar) erişebilir ve yeni mesaj yazabilir.

5.3. Service Account Güvenliği

Platformda test amaçlı demo veri üreten seed betikleri (scripts) Firebase Admin SDK kullanmaktadır. Bu betiklerin çalışması için gerekli olan `service-account.json` dosyası, kesinlikle ortak depolara (git repoları) eklenmeyecek şekilde `.gitignore` dosyasına dahil edilmiş ve lokal ortamda izole edilmiştir.

6. TEST VE DOĞRULAMA

Platformun üretim ortamına (production) hazır hale getirilmesi sürecinde kapsamlı test ve doğrulama aşamaları tamamlanmıştır.

6.1. Build ve Dağıtım Testleri

Projenin derleme süreçleri `npm run build` komutuyla test edilmiştir. Derleme işlemi hatasız tamamlanmış, Vite tarafından verilen büyük dosya boyutu (chunk size) uyarıları için kod bölme (code-splitting) optimizasyonları uygulanarak sayfa yükleme hızları artırılmıştır.

6.2. Veritabanı Bağlantı ve İzin Testleri

Firestore kuralları yerel emülatörler ve test projesi üzerinde doğrulanmıştır. Yapılan testlerde, influencer'ların teklif kabul etmesi durumunda veya markaların yeni teklif oluşturması aşamasında ortaya çıkan Firestore yetkilendirme (permission-denied) hataları, güvenlik kuralları güncellenerek tamamen çözülmüştür.

6.3. Kampanya ve Teklif Akışı Doğrulaması

Oluşturulan 50 test markası ve 30 test influencer'ı ile sistem simüle edilmiştir. Kampanya oluşturulduğunda ilgili influencer'ların ekranına tekliflerin anlık düştüğü, kabul edilen tekliflerin sohbet odalarını başarıyla tetiklediği gözlemlenmiştir. Tarihi geçen kampanyaların listeden çıkarılması süreci de başarıyla doğrulanmıştır.

7. SONUÇ

Go Influence projesi, markalar ile influencer kullanıcıları arasındaki kampanya, teklif, başvuru ve mesajlaşma süreçlerini tek bir web uygulamasında başarıyla birleştirmiştir. Proje Firebase tabanlı gerçek zamanlı veri yapısı, rol bazlı güvenlik kontrolleri ve React tabanlı modern kullanıcı arayüzü ile stabil bir şekilde çalışmaktadır. Başlangıçta planlanan yapay zeka destekli eşleşme sistemi daha sonra manuel kategori ve hizmet uyumuna dayalı bir eşleşme modeliyle değiştirilmiştir. Bu tercih, sistemin daha şeffaf, daha kolay test edilebilir ve düşük maliyetli çalışmasını sağlamıştır.

Proje; gelecek aşamalarda gerçek bankacılık ve ödeme geçidi entegrasyonları, sosyal medya ağlarının resmi API'leri üzerinden canlı veri doğrulama sistemleri, gelişmiş grafiksel kampanya raporlamaları ve mobil uygulama sürümleriyle genişletilmeye oldukça uygun bir altyapıya sahiptir.

8. KAYNAKÇA

1. React Documentation. *React 19 New Features and Architecture*. Web: react.dev
2. Vite Documentation. *Vite 6 Frontend Build Tooling*. Web: vite.dev
3. Redux Toolkit Guide. *Modern State Management in React applications*. Web: redux-toolkit.js.org
4. Firebase Documentation. *Cloud Firestore, Authentication and Security Rules*. Web: firebase.google.com/docs
5. Tailwind CSS Documentation. *Tailwind CSS v4.0 Framework Specifications*. Web: tailwindcss.com
6. Lucide React Documentation. *Open Source Vector Icon Library for React*. Web: lucide.dev