**Yazılım Laboratuvarı Proje 3**

**Akıllı Reklam Yönetim**

**Uygulaması Raporu**

Remzi ATAY, Enes Börtehan NAYMAN

*Kocaeli Üniversitesi, Mühendislik Fakültesi*

*Bilgisayar Mühendisliği*

*160201106, 150201201*

***Özet — Bu projede bir mobil uygulama ile Cloud Platform üzerine kurulu Real Time Dateabase’i bir api ile bağlamamız isteniyor. Mobil uygulama, kampanya görmel isteyen kullanıcı, Databese de bir Cloud da kampanyada depolayan ve istekleri yorumlayan kısım olacaktır.***

**I. GİRİŞ**

Android geliştirme kısmında daha çok yeni olmamıza rağmen gerek Java ile ilgili olmamız gerek de daha önceden hobi olarak Android projeleri geliştirmiş olduğumuz için bu kısımda zorlanmadık.

Server kısmını yazarken kararsızlığa kapıldık. Bu durumda ne tarz bir API kullanacağımız ve hangi platformda geliştireceğimiz bir soru işareti idi. Sonuç olarak uygulama basitliğinden dolayı PHP kullanarak bir RESTful servis geliştirme kararı aldık.

Projeyi geliştirirken uygulama kısmı için Android Studio, Cloud üzerine kurulması gereken bir veritabanı için Firebase Real Time Dateabese, kullanıcı girişi için Firebase Authentication, harita için Mapbox, server kısmı için ise Visual Studio Code kullandık. Admin panelini geliştirmek için yine HTML, PHP ve tasarımı için Bootstrap kullandık.

**I.I. FİREBASE NEDİR**

Google tarafından yeni özelliklerin eklenmesiyle gelişen Firebase, bütün bu ihtiyaçları karşılayabilme iddiasında ücretsiz kullanım da sunan bir platform. Herhangi bir platformda uygulama geliştirmeye bir sebeple başlayıp daha sonra bir kontrol paneli ve her durumda ulaşılabilir kullanıcı veri deposuna ihtiyacınız olduğunu hemen hissetmişsinizdir. Günümüzde uygulamalar platform farketmeksizin aynı veriye her cihazdan erişmek istiyor. Uygulaması, birçok kullanıcı tarafından yüklenen geliştiricilerin de kayıt - oturum bilgilerini tutma, uygulamaların kullanım verilerini analiz etme, yeni duyurular yapmak için aynı zamanda kullanıcıya bildirim gönderme, uygulamayı test etme gibi işlemleri rahatlıkla yönetebileceği bir yönetim paneli gerekiyor. İşte Google tarafından yeni özelliklerin eklenmesiyle sürekli kendini geliştiren [Firebase](https://firebase.google.com/), bütün bu ihtiyaçları karşılayabilmek için uygulama geliştiricilerine [ücretsiz](https://firebase.google.com/pricing/) kullanım da sunan bir platformdur.

**I.II. API NEDİR**

API bir uygulamanın işlevlerini/varlıklarını başka uygulamalara sunmasıdır. Temel olarak bir istek alır ve bir yanıt döndürür. Yanıt genellikle JSON yada XML tipindedir.

**I.III. FİREBASE REAL TİME DATABASE**

Firebase Realtime Database, bulutta barındırılan bir veritabanıdır. Veriler JSON olarak saklanır ve bağlanan her istemciye gerçek zamanlı olarak senkronize edilir. İOS, Android ve JavaScript SDK'larımızla platformlar arası uygulamalar oluşturduğunuzda, tüm müşterileriniz bir Realtime Database örneğini paylaşır ve otomatik olarak en yeni verilerle güncellemeleri alır.

**I.IV. FİREBASE AUTHENTİCATİON**

Firebase Authentication, kullanıcıları uygulamanıza doğrulamak için arka uç hizmetleri, kullanımı kolay SDK'lar ve hazır UI kitaplıkları sunar. Şifreleri, telefon numaralarını, Google, Facebook ve Twitter gibi popüler birleşik kimlik sağlayıcılarını ve daha fazlasını kullanarak kimlik doğrulamayı destekler.

**I.V. JSON NEDİR**

JSON (JavaScript Object Notation) nesne yönelimli bir veri formatıdır. JavaScript uygulamaları için oluşturulmuş ancak günümüzde ufak bant genişliği ihtiyacı ve okuma kolaylığı ile popüler olmuş ve XML’i tahtından etmiştir.

**II. YÖNTEM**

**II.I. API TASARIMI**

API’mizi PHP kullanarak gerçekleştirdik. Api için nesneye yönelik bir tasarım yoluna gittik. Firma ya da kampanya eklemek için POST parametresini okuduk ve oluşturduğumuz Firma nesnesinde uygun metod ile işlettik. Dönen veriyi String olarak alıp mobil uygulamamızda gösterdik.

**II.II.MOBİL UYGULAMA TASARIMI**

Mobil uygulama’yı Android Studio kullanarak Java dili ile yazdık. Tasarım kısmını kişisel zevklerimize göre dizayn ettik. Aynı zamanda nesneye yönelik bir tasarım yoluna gittik. FirmaBox classımızı bir ListView e kendi yazdığımız adapter yardımıyla istediğimiz kadar ekleyebiliyoruz. Tasarımsal değil, programsal bir arayüz tasarladık. Tasarımı yaparken çok fazla sorunla karşılaştık ve dolayısıyla çok fazla çözüm ile muhatap olduk. Bu vesile ile yardım aldığımız birçok kaynak oldu.[3][4][5][6][7][8][9]

**II.III. ADMİN ARAYÜZÜ**

Admin arayüzünü basit bir PHP/HTML form sayfası ile hallettik. Tasarımı Bootstrap ile yaptık. Formu oluşturmak için hazır kod üreten bir websitesi[10] kullandık ve PHP ile form verisi gönderme ve okuma işlemi için de ayrı bir kaynaktan yardım aldık.[11]

**III. KARŞILAŞILAN ZORLUKLAR**

**III.I. BİLDİRİM YOLLAMA**

Kullanıcı mevcut bir kampanyaya yaklaştığında bildirim gönderilmesi isteniyor. Bunu yapmak için çözüm yolları aradığımızda Push Notifications ile karşılaştık ancak zaman bakımından oldukça maliyetli bir yoldu bu.

Service kullanmaya karar vermemizin sebebi uygulamasının ve mantığının basit olması idi. Uygulama belirtilen aralık ile server’a istek atar. Bu işlemi bir threadi sonsuz bir döngü gibi belli bir aralıkla çalıştırarak yapar. Eğer Mevcut yarıçap içerisinde kampanya görürse bildirir. Service kullandığımız için telefon kapalıyken bile çalışacak bir sınıfımız oluyordu. Ancak bunu nasıl uygulayacağımız bir sorundu. Bu konuda internetten yardım aldık.[12]

Bildirimi oluşturma kısmı hakkında da bilgimiz olmadığından bu kısım için de yardım aldık.[13] Bunun kesin bir çözüm olmadığını düşünüyoruz çünkü birimizin telefonunda bildirim oluşturabilen bu yöntem öbürümüzde çalışmıyor.

**III.II. INTENTLER ARASI GEÇİŞ**

Birkaç farklı Activity ve Service ile çalıştığımızdan dolayı bunlar arasında veri aktarımı zor olabiliyor. Tamsayı, metin gibi tipler rahatlıkla iletilebilirken bir nesne veya bir resim iletmek baş belası olabiliyor. Bu sorunu atlatmak için internetten yardım aldık.[14] Çözüm resmimizi kaydetip, dosya ismini intent ile iletmek idi. Bu sayede gereksiz bantgenişliği tüketiminden de kurtulmuş olduk.

**III.III. VERİ DEPOLAMA**

Veri depolama olarak Firebase Cloud Platformu’nun Real Time Database Kısmını kullandık. Firebase’in Authentication özelliği ile giriş yapıldıktan sonra özel kullanıcı bilgilerini tutmak için (Mevcut GPS konumu gibi) SharedPreferences gibi basit ve native bir yol izledik. Bu yöntem ile de dolaylı yoldan vektör, dizi depolamak mümkün olacağından her hangi bir databese’e ya da başka bir kütüphaneye ihtiyaç duymadık.

**IV. SONUÇ**

Sonuç olarak tüm isterler karşılandı. Elimizde bir Cloud Platformdaki Firma ya Kampanyanları çekebilen, bunları kategorize edebilen, GPS yardımıyla kullanıcının anlık konumuna göre kullanıcının belirttiği yarıçaptaki kampanyaları bulan listeleyen ve katagorize eden bir Uygulamamız oldu.

Ayrıca mevcut konumunda yakınında bir kampanya varsa ya da bir kampanya ya yaklaşıyorsa kullanıcıya bir bildirim gönderirilir.

Bu proje sonunda daha önce hiç uğraşmış olmadığımız ve merak ettiğimiz Realtime Databese, Cloud Platform kavramlarını öğrenmiş olduk.

**KAYNAKÇA(REFERANSLAR)**

1. <https://codinginfinite.com/restful-web-services-php-example-php-mysql-source-code/>
2. <https://firebase.google.com/docs/database?authuser=0>
3. <https://firebase.google.com/docs/auth?authuser=0>
4. <https://stackoverflow.com/a/8684666/>
5. <https://stackoverflow.com/a/3765577/>
6. <https://stackoverflow.com/a/17357449/>
7. <https://stackoverflow.com/a/18870853/>
8. <https://stackoverflow.com/a/28136027/>
9. <https://stackoverflow.com/a/10074612/>
10. <https://stackoverflow.com/a/9288544/>
11. <https://bootsnipp.com/forms>
12. <https://docs.mapbox.com/android/maps/examples/#getting-started>
13. <https://stackoverflow.com/a/37996926/>
14. <https://stackoverflow.com/a/27721837/>
15. <https://stackoverflow.com/a/11010565/>
16. <https://github.com/kcochibili/TinyDB--Android-Shared-Preferences-Turbo>
17. <https://stackoverflow.com/a/42780501/>
18. <https://stackoverflow.com/a/5162096/>

**Remzi ATAY**

**160201106**

**Enes Börtehan NAYMAN**

**150201201**