

# Mobil Sorgular

## Mobile Queries

Cumali TOPRAK

Mühendislik Fakültesi, Bilgisayar Mühendisliği  
Kocaeli Üniversitesi  
Kocaeli, Türkiye  
cumalitoprakk@gmail.com

Berkay Efe ÖZCAN

Mühendislik Fakültesi, Bilgisayar Mühendisliği  
Kocaeli Üniversitesi  
Kocaeli, Türkiye  
berkayefeozyan@gmail.com

**Özetçe—** Bu proje kapsamında bulut bilişim ve map api kullanarak android platformunda bir uygulama geliştirmemiz beklenmektedir. Taksi gezing (trajectory) verileri kullanarak android platformunda farklı sorguların yapılabildiği bir uygulama geliştirmemiz amaçlanmaktadır. Biz de bulut bilişim olarak verilerimizi veri tabanında saklamak için mongodb veri tabanını seçtik. Server side kodumuzu da bizim için daha uygun bir platform olan heroku'ya deploy ettik. Bu projede bulut bilişim teknolojilerini kullanarak nasıl mobil bir uygulamayı geliştireceğimizi görmüş olduk.

**Anahtar Kelimeler—** Mobil sorgular, React Native, Nodejs, MongoDB, Map api.

**Abstract —** Within the scope of this project, you are expected to develop an application on the android platform using cloud computing and google map api. We aim to develop an application where different queries can be made on the android platform using the data of taxi Gezing (trajectory). As cloud computing, we chose the mongodb database to store our data in the database. We also deployed our server side code to heroku, which is a more suitable platform for us. In this project, we saw how we can develop a mobile application using cloud computing technologies.

**Keywords—** Mobile queries, React native, Nodejs, MongoDB, Map api.

## I. GİRİŞ

Bu proje kapsamında bizden istenilen mobil geliştirme frameworklerinden birini seçip bunla bir bilişim tabanlı mobil uygulama geliştirmek istenmektedir. Biz de öncelikle react native'in popülerliğinden dolayı bu frameworku seçmeye karar verdik. Bize verilen veriler üzerinde sorgular gerçekleştireceğimiz için bu verileri depolayacağımız bulut bilişim teknolojilerinden mongodb ile çalışmaya karar verdik. Mongodb deneme sürümü 500 mb veriye kadar ücretsiz kullanıma izin verildiği için bu bizim için gayet yeterli bir miktar oldu. Bize verilen verileri öncelikle ihtiyacımız olmayan kısımları filtrelemekle işe başladık. Veriyi ihtiyacımız doğrultusunda filtreledikten sonra bu csv

dosyasını üçüncü parti ui yazılımlar aracılığı ile mongodb dokümanına çevirerek daha önce oluşturduğumuz mongodb server'ına gönderdik. Artık veriler üzerinde çalışılmaya hazır hale gelmiş oldu.

Daha sonra backend tarafında hangi teknolojiyi kullanmalıyız diye düşündük. Bu düşünceler ve tartışmalar sonucunda nodejs'in bizim geliştireceğimiz uygulama için gayet yeterli ve doğru bir seçim olacağına karar verdik. Nodejs'de query'leri handle etmek için kendi built-in mekanizmasını değil de daha verimli ve yaygın kullanılan Express kütüphanesini kullandık. Mongodb içerisine depolamış olduğumuz verileri burada filtreledik. Yani bu demek değil ki tüm verileri buraya çekip burada filtreleme yaptık, mongodb'nin mongoose adlı kütüphanesi yardımıyla mongodb'ye istekte bulunurken filtreleme işlemini yaptık. Yani bu da bizim için çok önemli bir performans getirisi oldu. Bu iş için backend tarafını da yormamış olduk.

## II. TEMEL BİLGİLER

### Mongodb Nedir?

MongoDB 2009 yılında geliştirilmiş açık kaynak kodlu bir NoSQL veri tabanıdır. Bugün piyasada Cassandra, BigTable, Dynamo gibi birçok NoSQL veri tabanı bulunmaktadır. MongoDB'de her kayıt bir doküman olarak ifade edilir. Ve bu dokümanlar json formatı şeklinde saklanır. Daha önce ilişkisel veri tabanlarıyla ilgilenenlerin bildiği table yapısını burada collection, row yapısını document, column yapısını ise field alır. Günümüzde aktif olarak kullanılan pek çok programlama dili için driver desteği bulunması bakımından bugün NoSQL sistemler içerisinde en çok tercih edilenlerden biridir.

### React nedir, nerelerde kullanılır?

React Native için kısaca Facebook tarafından üretilen, cross-platform mobil uygulama geliştirme olanağı sağlayan bir framework diyebiliriz.

### Express kütüphanesi nedir, niin kullanılır?

Express.js modl paketi, nodejs tabanlı bir web uygulama sunucu atısıdır. Express.js'nin sunduėu sınırsız HTTP yardımcı araları ve katmanlar sayesinde saėlam bir API oluřturmak olduka hızlı ve kolaydır.

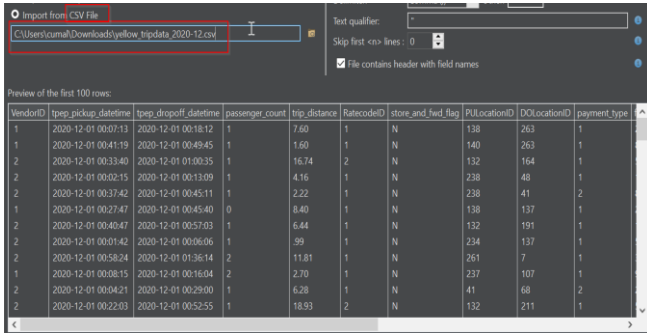
### Heroku nedir, niin kullandık?

Heroku, bir bulut biliřim (cloud computing) uygulama altyapısı servis saėlayıcısıdır. Platform as a Service (PaaS) olarak ifade edebileceėimiz Heroku, JavaScript, Ruby, Java, PHP, Python, Golang, Scala ve Clojure ile geliřtirdiėimiz web tabanlı uygulamalarımızı internete tařımamızı (deploy, manage ve scale) saėlar. Bu iřlemler iin Dyno adı verilen Linux temelli kk ve hafif sanal ortamlar kullanılmaktadır.

## III. YNTEM

### Mongodb : Verilerin Depolanması Ve Sorguların Yapılması

Verileri oluřturmuř olduėumuz mongodb server'ına stdyo 3t adı verilen bir yazılım ile csv'den mongodb dokmanına cevirdik. Bunun ncesinde ihtiyaımız olmayan stnları ıkardık ki hem sorgu hızı performansı aısından hem de mongodb alan kullanımı aısından bize faydalı olsun. Ařaėıdaki bir kesit gsterilmiřtir.



The screenshot shows the Studio 3T interface. At the top, there's a section for 'Import from CSV File' with a file path 'C:\Users\cumali\Downloads\yellow\_tripdata\_2020-12.csv'. Below this, a 'Text qualifier' is set to 'Skip first <no> lines: 0'. A checkbox 'File contains header with field names' is checked. The main area displays a 'Preview of the first 100 rows' of data. The data is organized into columns: VendorID, tpep\_pickup\_datetime, tpep\_dropoff\_datetime, passenger\_count, trip\_distance, RatecodeID, store\_and\_fwd\_flag, PULocationID, DOLocationID, and payment\_type. The first few rows show data for different vendors and locations, with passenger counts and distances.

VendorID	tpep_pickup_datetime	tpep_dropoff_datetime	passenger_count	trip_distance	RatecodeID	store_and_fwd_flag	PULocationID	DOLocationID	payment_type
1	2020-12-01 00:07:13	2020-12-01 00:18:12	1	7.60	1	N	138	263	1
1	2020-12-01 00:41:19	2020-12-01 00:49:45	1	1.60	1	N	140	263	1
2	2020-12-01 00:33:40	2020-12-01 01:00:35	1	16.74	2	N	132	194	1
2	2020-12-01 00:02:15	2020-12-01 00:13:09	1	4.16	1	N	238	48	1
2	2020-12-01 00:37:42	2020-12-01 00:45:11	1	2.22	1	N	238	41	2
1	2020-12-01 00:27:47	2020-12-01 00:45:40	0	8.40	1	N	138	137	1
2	2020-12-01 00:40:47	2020-12-01 00:57:03	1	6.44	1	N	132	191	1
2	2020-12-01 00:01:42	2020-12-01 00:06:06	1	99	1	N	234	137	1
2	2020-12-01 00:58:24	2020-12-01 01:36:14	2	11.81	1	N	261	7	1
1	2020-12-01 00:08:15	2020-12-01 00:16:04	2	2.70	1	N	237	107	1
2	2020-12-01 00:04:21	2020-12-01 00:29:00	1	6.28	1	N	41	68	2
2	2020-12-01 00:22:03	2020-12-01 00:52:55	1	18.93	2	N	132	211	1

řekil 1 : Studio 3T programının ekran grnts

Sorgularımızı yazarken mongodb'nin kendine ait fonksiyonlarını kullanarak filtreleyerek ilgili verileri ektik. Bu da bize performans aısından baya faydalı oldu. Ařaėıda rnek bir mongodb filtreleme sorgusu bulunmaktadır.

```
app.get('/queryOne', async (req, res) => {
  try{
    var data = await TripDataModel
    .aggregate([
      {
        $group: {
          _id: {
            $dateToString: {
              format: '%Y-%m-%d',
              date: '$tpep_pickup_datetime',
            },
          },
          totalPassengerCount: { $sum: '$passenger_count' },
        },
      },
      {
        $sort: { totalPassengerCount: -1 },
      },
      {
        $limit: 5,
      },
    ]),
  }
```

řekil 2: Backend de yazılmıř olan veri tabanı sorgusu

### MapBox Api Kullanımı

Mapbox, Foursquare, Lonely Planet, Financial Times, The Weather Channel, Instacart Inc. ve Snapchat gibi web siteleri ve uygulamalar iin zel bir evrimii harita saėlayıcısıdır. Mapbox geliřtiriciler iin konum verilerini saėlama, harita sdıkları ve navigasyon gibi hizmetleri sunar.

Projede Mapbox' ın Geocoding ve Directions API leri kullanılmıřtır.

- Geocoding API: İleri coėrafi kodlama, konum metnini coėrafi koordinatlara dnřtrerek “2 Lincoln Memorial Circle NW”yi -77.050,38.889'a dnřtrr.
- Directions API: Trafiėe ve olaya duyarlı ynlendirmeyi kullanarak optimum srř, yryř ve bisiklet rotalarını hesaplar.

Yukarıdaki API leri kullanımı eriřim anahtarını ile mmkn olmaktadır.

### Heroku'ya Uygulamanın deploy edilmesi

Uygulamamızın geliřtirme ařamasını bittikten sonra aws ile heroku arasında kaldık fakat yaptığımız arařtırmalar sonucu herokunun bizim iin daha faydalı olacaėına karar verdik. ncelikle herokuya uygulamamızı deploy etmek iin heroku cli adlı yazılımını indirdikten sonra git ile uygulamamızı herokuyu uzak repoya attığımız zaman heroku ilgili projeyi derleyip direkt olarak web sitesini kullanıma sundu.

#### IV. GERÇEKLEŞTİRİLEN TESTLER VE DEĞERLENDİRME

Uygulamamızı hem development ortamında hem de production ortamında denediğimiz zaman uygulamamızın hatasız olarak çalışmakta, kendisine gelen requestleri sorunsuz olarak handle etmektedir. Uygulama halen gelişime açık olup bu proje kapsamında bizden istenilen şeyleri sorunsuz olarak gerçekleştirmektedir.

#### V. ÖZET VE SONUÇLAR

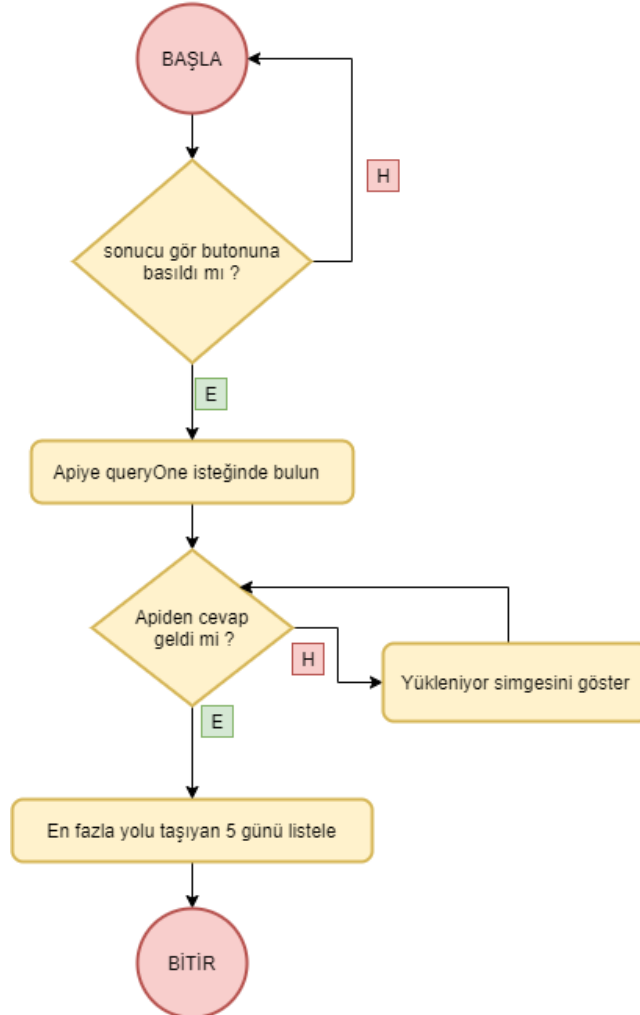
Bu proje kapsamında nasıl bir mobil uygulama yapmanın yansıra bu mobil projeyi nasıl bulut bilişim teknolojileri kullanarak yapacağımızı da öğrendik. Birçok yeni teknoloji kullandık.

#### VI. KAYNAKÇA

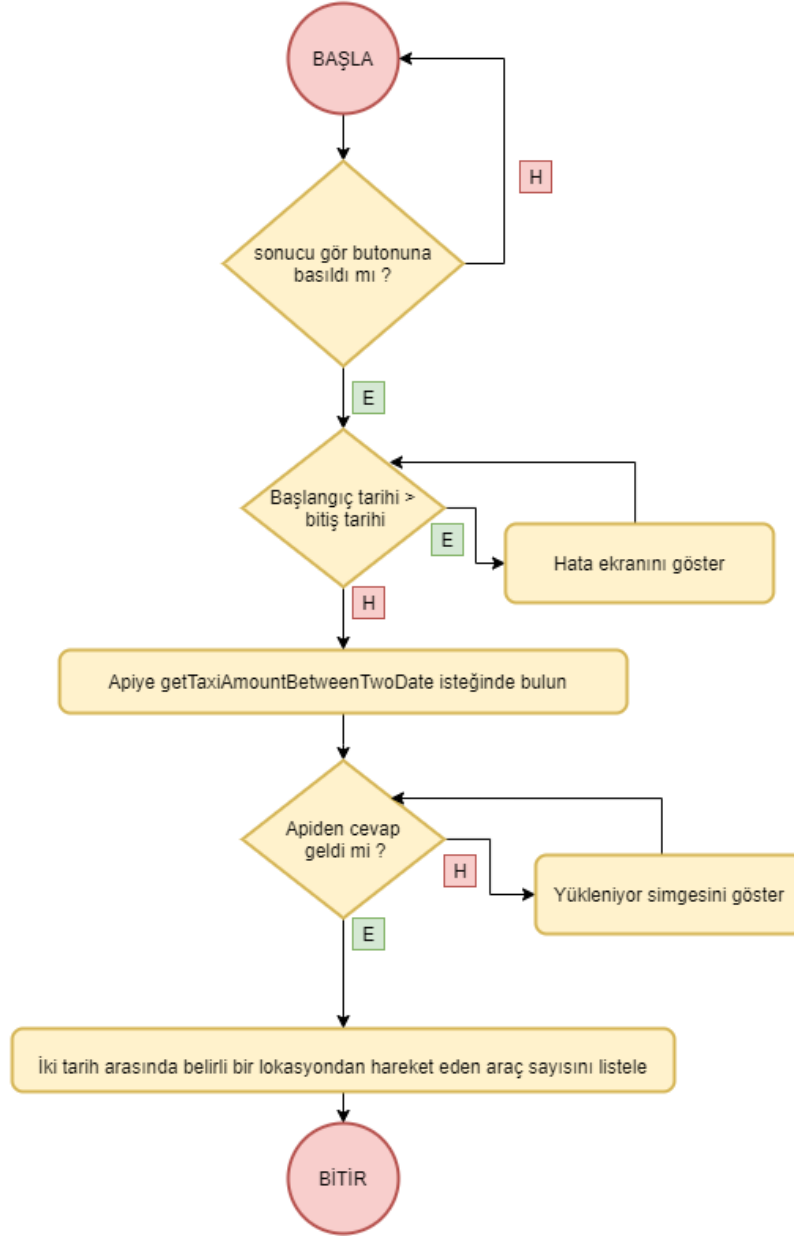
- 1- <https://www.youtube.com/>
- 2- <https://reactnative.dev/>
- 3- <https://www.heroku.com/>
- 4- <https://expressjs.com/>
- 5- <https://docs.mapbox.com/api/navigation/directions/>
- 6- <https://docs.mapbox.com/api/search/geocoding/>
- 7- <https://www.mongodb.com/>

#### VII. AKIŞ ŞEMALARI

##### Tip 1 - En fazla yolcu taşıyan 5 gün



## Tip 2 - İki tarih arasında belirli bir lokasyondan hareket eden araç sayısı



### Tip 3 - Belirli bir günde en uzun seyahatin harita üzerinde çizimi

