

**Kocaeli Üniversitesi**  
**Bilgisayar Mühendisliği Bölümü**  
**Yazılım Laboratuvarı II Proje 1**  
**Son Teslim Tarihi : 18 Mart 2022**

1 Mart 2022

**İçindekiler**

<b>1 Amaç</b>	<b>2</b>
<b>2 Senaryo</b>	<b>2</b>
<b>3 Kabuller</b>	<b>3</b>
<b>4 Görev</b>	<b>3</b>
<b>5 Kısıtlar ve Kolaylıklar</b>	<b>3</b>
<b>6 Sistem Mimarisi</b>	<b>4</b>
<b>7 Kullanılacak Veri</b>	<b>4</b>
<b>8 Kullanıcı Senaryoları (Use Case)</b>	<b>7</b>
8.1 Müşterinin Araçlarını Takip Etmesi . . . . .	7
8.1.1 Tanım . . . . .	7
8.1.2 Aktörler . . . . .	7
8.1.3 Kabuller . . . . .	7
8.1.4 Ön Koşullar . . . . .	7
8.1.5 Normal Akış . . . . .	7
8.1.6 Alternatif Akışlar . . . . .	9
8.2 Sisteme araç verinin girişi ve işlenmesi . . . . .	9
8.2.1 Tanım . . . . .	9

8.2.2	Aktörler . . . . .	9
8.2.3	Ön koşullar . . . . .	9
8.2.4	Normal Akış . . . . .	9
<b>9</b>	<b>Taslak Arayüzler</b>	<b>9</b>
<b>10</b>	<b>Başarı Kriteri ve Notlandırma</b>	<b>11</b>
<b>11</b>	<b>Proje Teslimi</b>	<b>12</b>
<b>12</b>	<b>KAYNAKÇA</b>	<b>12</b>

## 1 Amaç

Yazılım Laboratuvarı 2 dersi projesi kapsamında çok katmanlı bir web uygulaması geliştirilecektir. Bu proje ile öğrencilerimizin aşağıdaki hususlarda gerçek dünya ihtiyaçlarını dikkate alan, uygulama yeteneği kazandırılması hedeflenmektedir.

- Model, Manzara/Arayüz ve Yönetim (Model,View,Controller) kalıbının kullanılması (Arayüz, İşlem Mantığı ve Verinin ayrıştırılması)
- Mesaj simsarı kullanarak yazılım fonksiyonel bölümleri arasındaki bağımlılıkların azaltılması
- Dağıtık bir sistem mimari kullanarak çözüm oluşturma
- Güncel bir web kütüphanesi, alt yapısının kullanımının öğrenilmesi ve uygulanması
- Değişik tipte veri tabanlarını kullanma ve veri tabanı tasarımı

## 2 Senaryo

Araç takip hizmeti vermek isteyen bir A firması, müşterilerine araçlarına takabilecekleri GPS destekli bir takip donanımı ve ilgili hizmetleri ücret karşılığı sunmak istemedir. Anılan takip donanım üzerinde GSM modülü bulunmakta olup, araçların mevkileri GSM üzerinden, A firmasının sunucularına aktarılacaktır. A firması, müşterilerine web arayüzü ve mobil uygulamalar üzerinden, ödeme planına göre, değişik güncelleme seviyelerinde, araçların mevki bilgilerini sunacaktır. A firması ilgili donanımı hazır ticari ürün olarak temin edecektir. Ancak A firmasının dağıtık bir sistem kurma ve ilgili servisleri hazırlama ihtiyacı bulunmaktadır. A firmasının yerel bir teknik alt yapı işletmek istemediğinden oluşturulacak çözüm bulut hizmetleri üzerinde kullanılabilir olması gerekmektedir.

Siz A firmasının teknik çözüm ortağı B firmasını temsil ediyorsunuz. A firmasının ihtiyaçları kapsamında, ilgili sistem çözümünü, aşağıda verilen temel ihtiyaçlar ve tasarım kısıtları kapsamında oluşturmakla ve demosunu yapmakla görevlisiniz.

### 3 Kabuller

- Araçlardan gelecek verinin GSM alt yapısı ve ilgili operatarün sağladığı hizmetler kapsamından Bölüm-7 kısmında anlatılan formatta ulaştırıldığı kabul edilecektir.

### 4 Görev

Temel fonksiyonel isterler:

1. Birden fazla kaynaktan (araçlardan) akan GPS verileri alınarak bir veri tabanına depolanacaktır. Verilerin alınması için bir mesaj simsarcısı (message broker) kullanılacaktır. *Bu proje kapsamında Apache Kafka vb. büyük veri ortamları için hazırlanmış ağır ürünler yerine ZeroMQ<sup>1</sup> veya benzeri hafif bir yazılımın seçilmesi tavsiye edilir.*
2. Gelen anlık veriler bir NoSQL veri tabanına depolanacaktır. *Bu maksatla örneğin Redis <sup>2</sup>kullanılabilir.*
3. Müşteri hesap bilgileri bir SQL veri tabanında tutulacaktır. SQL veri tabanı MySQL, Postgre SQL gibi bir veritabanı sunucusu olabileceği gibi, proje kapsamında sqllite<sup>3</sup> gibi gömülü sistemler için tasarlanmış basit bir veri tabanı da olabilir.
4. Gelen anlık araç bilgileri harita alt yapısının olduğu bir web arayüzünde aşağıda belirtildiği gibi sergilenecektir. Google api kullanılabileceği gibi uygun bir coğrafi bilgi sistemi hizmeti de kullanılabilir.
5. Müşteri kendisine verilen hesap bilgileri kapsamında sisteme giriş yapabilecektir.
6. Her müşteriye sahibi olduğu veya hizmetini aldığı araçların anlık veya en son mevki bilgisi gelen güncellemeler kapsamında, **harita altlığı üzerinde** sunulacaktır.
7. Araçları geçmiş mevki bilgileri, müşteri tarafından verilen zaman aralığı kapsamında, veri tabanından çekilerek çekilerek sergilenecektir. Hiç bir şey belirtilmemiş ise en son yarım saatlik mevki bilgileri her bir araç için sergilenecektir.
8. Müşteri bir aracı seçip, 1-24 saat arasında bir seçim yaparak, sadece o araca yönelik geçmiş güzergahı sergileyebilecektir.

### 5 Kısıtlar ve Kolaylıklar

Çözümün oluşturulmasında aşağıdaki kısıtlara dikkat edilecektir.

---

<sup>1</sup><https://zeromq.org/>

<sup>2</sup><https://redis.io/>

<sup>3</sup><https://sqlite.org/index.html>

1. Tüm kullanılacak yazılımsal araçlar ve uygulamalar (kütüphaneler, veri tabanları, mesaj simsarı vs.) açık kaynak kodlu olacaktır. Buradaki amaç; sektörde geniş bir kullanımı bulunan açık kaynak kodlu yazılım ve araçların kullanımına ve yönetimine alışmanızdır.
2. Kullanılacak araçlar konusunda tavsiye olarak verilen araçların kullanımı şart değildir. Kullanımı tavsiye edilen uygulamalar, genel olarak kurulum ve yönetimi nispeten kolay olduğu değerlendirilen uygulamalardır. Verilen tavsiyeler projeyi zamanında bitirebilmeniz içindir. Belirtilen mimariye uygun olacak şekilde, mimariye ve amaca uygun aşına olduğunuz veya öğrenmek istediğiniz farklı araçları kullanılabiliyorsunuz.
3. Yazılımsal araçların (veri tabanı, mesaj broker vb), kolay kurulumu veya yönetimi için Docker<sup>4</sup> gibi container veya sanallaştırma teknolojilerinden yararlanabilirsiniz.
4. Uygulanabilecek sistem mimarisi Bölüm-6'de belirtilmiştir. Söz konusu sistem mimarisi, her ne kadar bulut ortamına aktarılabilir, modüler dağıtık yapıya uygun olarak tasarlanmış ise de, tek bir bilgisayar üzerinde geliştirilmesi, sunumu ve demosu mümkündür. Teslim edilecek kaynak kodda söz konusu ayırım veya modülerlik demonun bir parçası olarak sunulacaktır ve notlandırılacaktır.
5. Proje kapsamında İsveç taksi verisi kullanılacaktır. Detaylı bilgi Bölüm-7 kısmında belirtilmiştir.
6. Kullanılacak programlama dili veya dilleri konusunda bir kısıtlama getirilmemiştir. Öğrenciler java,python, php, ruby, go, rust, c++, javascript vb. dilleri ve uygun bir web kütüphanesini kullanarak projeyi yapabilir.

## 6 Sistem Mimarisi

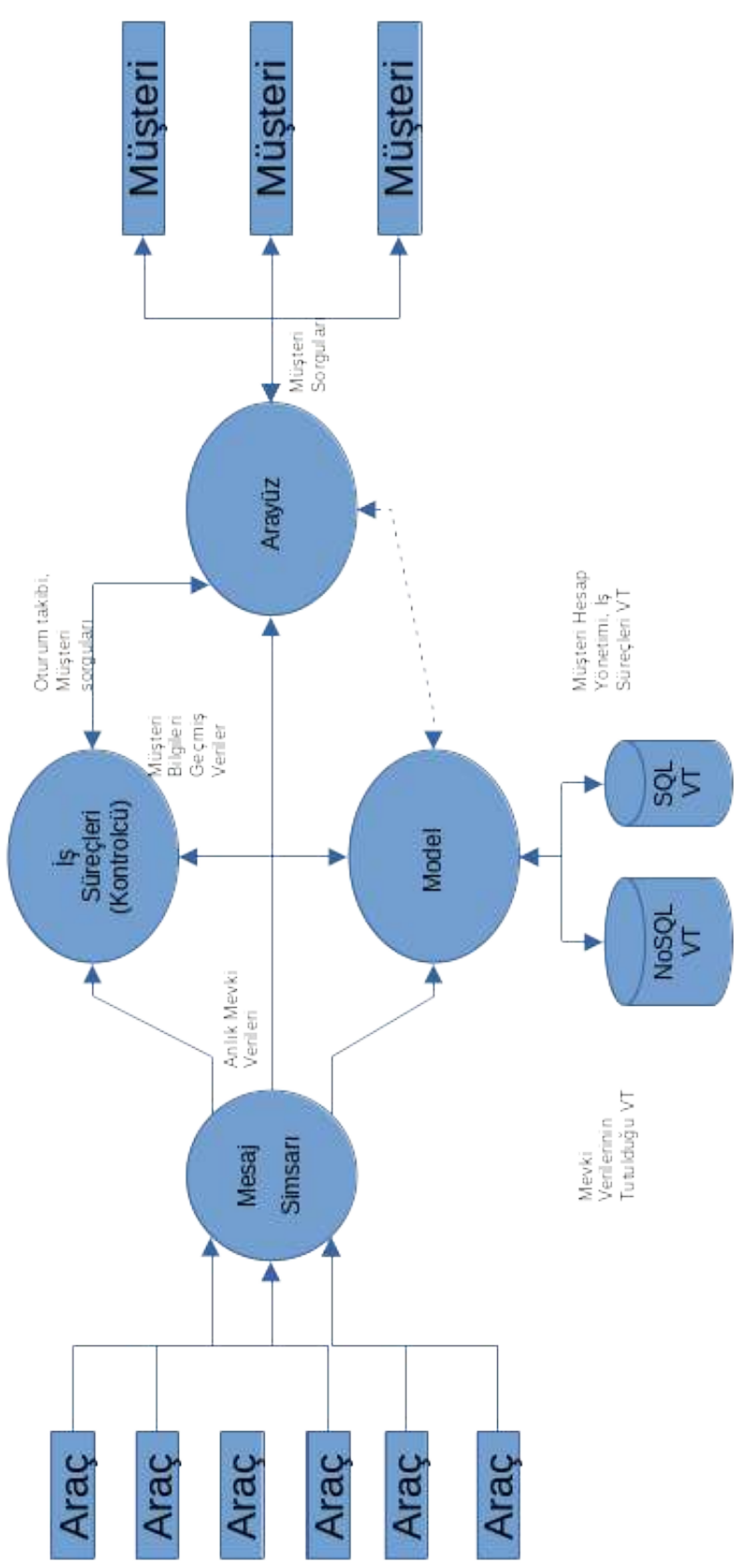
Öngörülen sistem mimarileri Şekil-1,2'de gösterilmiştir. İki alternatif sunulmuştur. Daha kapsamlı mimariler oluşturulabilir. Gerçek ortamlarda Arayüz, Model, Yönetim gibi fonksiyonel gruplar ideal olarak bağımsız olarak geliştirilir, çoğu zaman farklı fiziksel makineler üzerinde ve yedekli olarak çalıştırılır. Birbirlerinin hizmetlerin iyi tanımlı arayüzleri üzerinden ulaşırlar. Genelde ileşimleri dahi bir mesaj simsarcısı üzerinden gerçekleştirilir. Ancak notlandırmada buna dikkat edilmeyecektir. Proje kapsamında sistem mimarisinde belirtildiği gibi Mesaj Simsarcısı sadece gelen anlık araç bilgilerinin sistem modüllerine dağıtımı için kullanılmasına bakılacaktır.

## 7 Kullanılacak Veri

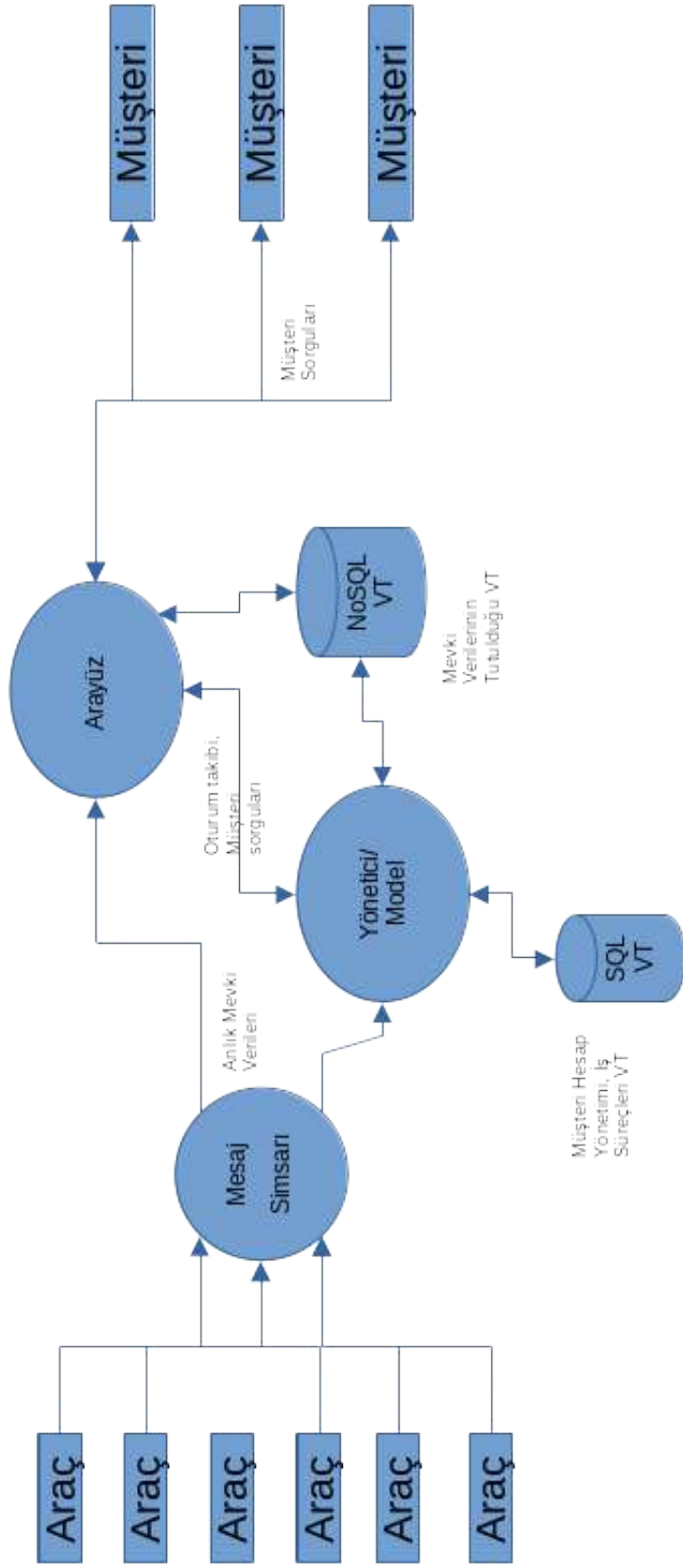
Proje kapsamında İsveç taksi verisi [1] kullanılacaktır. Söz konusu taksi verisinde yaklaşık 100 civarında taksinin bir aylık verileri bulunmaktadır. Genel olarak veriler 1dk. aralıklarla

---

<sup>4</sup><https://hub.docker.com>



Şekil 1: Sistem Mimarisi-1



Şekil 2: Sistem Mimarisi-2

ile kayıt edilmiştir. Ancak gerçek veriler olması nedeni ile, kayıt hatası, gönderim hatası vs. nedenler ile hatalı veriler güzergah içinde çıkabilmektedir. Mesala bazen veride 1dk. yüzlerce millik bir atlama yapıp geri dönmektedir. Bazen 1dk. süreler daha saatlik sürelerle dönüşmektedir. Bu tür atlamaları veya bariz hatalı kısımları veriyi inceleyip, önceden temizleyebilirsiniz veya hız/mesafe vb. değişik kriterler koyarak çalışma esnasında veriyi silebilirsiniz. Gerçek verilerde bu tür hatalar, mesleki yaşamınızda karşılayacağınız ve çözüm bulmanız gereken meselelerdir. Diğer taraftan projenin süre kısıtı nedeni, hatalı veriye karşı gerçek zamanlı bir dayanıklılık beklenmemektedir. Notlandırmada bu hususa dikkat edilmeyecektir. Önceden seçtiğiniz araç verilerin gözden geçirmeniz, düzeltmeniz veya temizlemeniz tavsiye olunur.

**Proje kapsamında iki farklı müşteri ve her müşteri için iki araç, yani en az 4 farklı araç verisini söz konusu veriden çıkararak kullanabilirsiniz.**

## 8 Kullanıcı Senaryoları (Use Case)

Aşağıda notlandırmaya temel olacak bazı temel kullanıcı senaryoları sunulmuştur.

### 8.1 Müşterinin Araçlarını Takip Etmesi

#### 8.1.1 Tanım

Müşterinin web sitesine giriş yapmasını ve sahibi olduğu araçların mevkilerini görmesini tanımlar

#### 8.1.2 Aktörler

**Birincil Aktör:** Nihai müşteri

**İkincil Aktörler:** Sistem modülleri

#### 8.1.3 Kabuller

Müşteri Hesabı bu proje kapsamında önceden açılmış olacak ve bilgileri SQL veri tabanında bulunacaktır. Hesap yönetimi vb. hususlar için ek arayüz/kabiliyetler geliştirilmesine gerek yoktur. Sadece var olan bilgileri sorgulayan bir giriş sayfası geliştirilecektir.

#### 8.1.4 Ön Koşullar

Bölüm-6'de belirtilen tüm sistem alt modülleri ayakta, birbirleri ile iletişim halinde ve hizmete hazır durumdadır.

#### 8.1.5 Normal Akış

1. Müşteri internet gezgini üzerinde A şirketinin araç takip hizmeti bağlantısını girer veya daha önceden kayıt edilmiş bir sayfa işaretisini tıklar.

2. Arayüz modülü ana giriş sayfasını üreterek bir kullanıcı adı ve şifre giriş alanının bulunduğu giriş sayfasını müşteriye sergiler.
3. Müşteri daha önceden açılmış olan hesabı kapsamında kendisine verilen kullanıcı adı ve şifre bilgilerini girer.
4. Arayüz modülü, girilen bilgileri Yönetim/Model modülüne gönderir, Yönetim/Model modülü kullanıcı bilgilerini doğrular.
5. Yönetim/Model modülü tarafından müşterinin sistem giriş zamanı SQL veri tabanının kayıt edilir. (Analitik, muhasebe vb. ihtiyaçlar nedeni ile)
6. Yönetim/Model modülü kullanıcı hesabına göre araç bilgilerin ve her bir araç için son 30dk. verileri hazırlayarak Arayüz modülüne gönderir. Son 30dk veriler performans açısından Arayüz modülü tarafından NoSQL veritabanından direk çekilebilir.
7. Arayüz modülü müşterinin sahip olduğu araçlara yönelik güncel durumları ve 30dk.lık geçmiş bilgileri ile gösterir.
8. Yönetim modülü müşteri oturumunun açıklığını takip eder ve oturum açık kaldığı sürece 30dk bilgileri ve son mevkileri günceller, güncellettir. **Performans veya gerçekleştirme kolaylığı için arayüz modülü anlık veriyi mesaj brokardan direk alabilir.**
9. Müşteri ekranda bir liste vb. kabiliyet üzerinde gösterilen araç listesinden bir aracı seçerek çift tıklar.
10. Arayüz modülü zaman aralığı bilgilerinin seçilmesini gösterecek bir dialog sunar. Arayüz dialog bitiş zamanını otomatik doldurur. Başlangıç zaman varsayılan olarak bir saat öncesi girilir. Bilgilerin müşterinin değiştirilmesine olanak tanınır.
11. Müşteri seçtiği araç için zaman aralığını girer ve giriş tuşuna basar veya varsayılanı onaylar.
12. Arayüz modülü girilen bilgileri Yönetim/Model modülüne gönderir, Yönetim/Modülü girilen zaman aralığına karşılık gelen bilgileri çıkarıp Arayüz modülüne gönderir.
13. Arayüz modülü gerekli sergilemeyi sağlar. Bu sayfa açık kaldığı sürece araç bilgilerine güncelleme yapılmaz ve sadece sorgulanan araç bilgisi gösterilir.
14. Müşteri çıkış linkine tıklayarak sistemden çıkar. Müşteri oturumu sona erer.
15. Yönetim/Model tarafından müşterinin çıkış zamanları veri tabına kayıt edilir.



### 8.1.6 Alternatif Akışlar

Normal Akış Madde-4

1. Müşteri bilgileri hatıdır.
2. Sistem üç defaya kadar müşteriyi doğrulamaya çalışır. Üç defadan fazla hatalı girişte, doğrulamayı bir diyalog ile bırakır.

## 8.2 Sisteme araç verinin girişi ve işlenmesi

### 8.2.1 Tanım

Bu senaryo verinin sisteme girişini ve sistem içinde işlenmesine yönelik akışı ifade eder.

### 8.2.2 Aktörler

**Birincil Aktör:** Yönetim/Model Modülleri ve Mesaj Simsarcısı

### 8.2.3 Ön koşullar

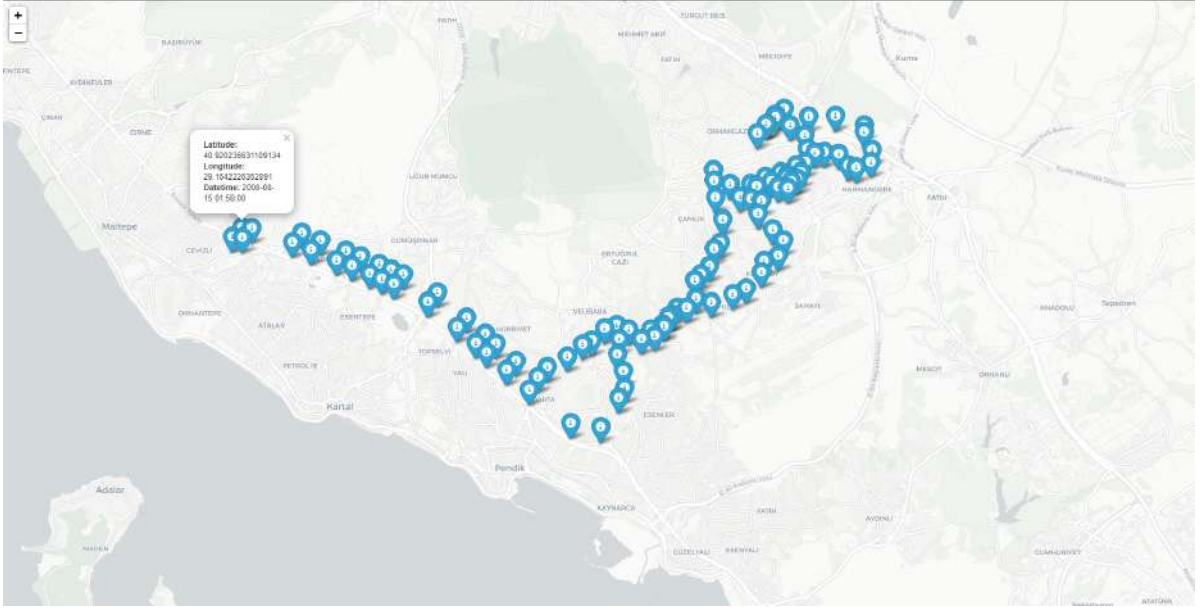
1. Sistem modülleri ayakta ve hizmete hazırdır, modüller birbiri ile iletişim halindedir. GSM operatör alt yapısı araç verileri mesaj simsarcısını bulduğu internet adresine yönlendirilmektedir.

### 8.2.4 Normal Akış

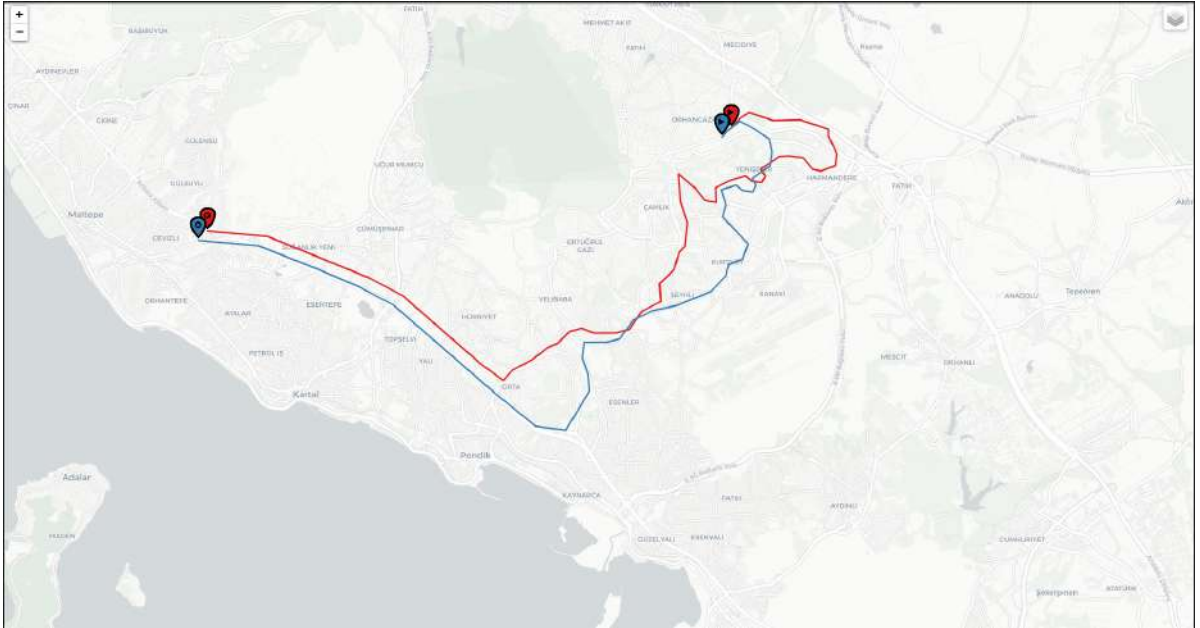
1. Araç verisi mesaj simsarcısına ulaşır.
2. Mesaj simsarcısı söz konusu veriyi abone olan modüllere aktarır. (Yönetici/Model, Arayüz)
3. Yönetici/Model veriyi alarak, nosql veritabanına depolar.
4. Yönetici/Model gelen veriyi etkileyen bir sorgu var ise, model sorgunun cevabını güncelleyerek Yönetimci/Arayüze güncel bilgiyi gönderir.

## 9 Taslak Arayüzler

Fikir vermesi açısından Şekil-3,figure-sample-plotting-2’de örnek plotlamalar gösterilmiştir. Plotlar araçların id’leri ile markalanmalıdır.



Şekil 3: Örnek bir güzergah plotlama - 1



Şekil 4: Örnek bir güzergah plotlama - 2

## 10 Başarı Kriteri ve Notlandırma

- Minimum değerlendirme koşulu, bir müşteriye ait bir araç verisinin anlık olarak veri tabanında depolanması ve web arayüzünde sergilenmesi, gelen güncellenmelerin web arayüzünde gösterilebilmesidir. Bu kabiliyeti sergilemeyen projeler “0” (Sıfır) alacaktır.
- 100 tam puan: Aşağıdaki kriterlerin karşılanması kapsamında verilecektir.
  1. Şekil-6’de belirtilen alt yapının veya belirtilen esneklikler kapsamının muadilinin gerçekleştirilmesi
  2. Harita alt yapısını kullanılması/sergilenmesi
  3. Kullanıcı Senaryosu 8.1 demosunun yapılması
  4. 2 müşteri için iki aracın bilgilerinin aynı anda gerçek zamanlı sergilenmesi
  5. Kaynak kodun modüllerliği/esnekliği açıklanması sağladığının gösterilmesi
  6. Raporun teslim edilmesi
- Değerlendirmeye alınan projelerde, verilen isterlere uygunsuzluk ölçüsünde tam nottan puan düşülecektir. Genel presipler aşağıda belirtilmiştir.
  1. Mimariye uygunsuzluk/modüler mimari kullanılmaması (monolitik uygulama): -10
  2. NoSQL veri tabanı kullanılmaması: -10
  3. SQL veri tabanı kullanılmaması: -10
  4. Mesaj simsarcısı kullanılmaması: -10
  5. Harita altlığının sergilenememesi: -10
  6. Giriş ekranı (kullanıcı adı, şifre) ve giriş ile ilgili kullanıcı senaryosundaki belirtilen fonksiyonları yerine getirmemesi: -10
    - Hatalı şifre girişi fonksiyonları : -5
    - Girişin ve çıkışın veri tabanına kayıt edilmemesi: -5
  7. İki farklı hesaba ait araç bilgilerini eş zamanlı güncellendiğinin gösterilememesi: -20
    - Bir hesaba yönelik iki araç izinin birlikte gösterilememesi: -10
  8. Modüler olmayan, özensiz spaghetti kaynak kod: -10
  9. Rapor teslim edilmemesi: -10
  10. Yukarıda belirtilen sınırlar dahilinde, her bir özellik için takdir yetkisi kullanılacaktır.

## 11 Proje Teslimi

- Rapor IEEE formatında (önceki yıllarda verilen formatta) 4 sayfa, akış diyagramı veya yalancı kod içeren, özet, giriş, yöntem, deneysel sonuçlar, sonuç ve kaynakça bölümünden oluşmalıdır.
- Dersin takibi projenin teslimi dahil edestek.kocaeli.edu.tr sistemi üzerinden yapılacaktır. edestek.kocaeli.edu.tr sitesinde belirtilen tarihten sonra getirilen projeler kabul edilmeyecektir.
- Proje ile ilgili sorular edestek.kocaeli.edu.tr sitesindeki forum üzerinden Arş.Gör. Muhammet Ahmet Demirtaş veya Arş.Gör. Yılmaz Dikilitaş'a sorulabilir.
- Sunum tarihleri daha sonra duyurulacaktır.
- Sunum sırasında algoritma, geliştirdiğiniz kodun çeşitli kısımlarının ne amaçla yazıldığı ve geliştirme ortamı hakkında sorular sorulabilir.
- Kullandığımız herhangi bir satır kodu açıklamanız istenebilir.

## 12 KAYNAKÇA

- [1] *taxiMovementConcatenated* / Kaggle. URL: <https://www.kaggle.com/henrikengdahl/taximovementconcatenated>.