|  |  |  |
| --- | --- | --- |
|  |  |  |

T.C.

CUMHURİYET ÜNİVERSİTESİ

MÜHENDİSLİK FAKÜLTESİ

BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

|  |  |  |
| --- | --- | --- |
|  | HAVA KALİTESİ İZLEME PROGRAMI  ERHAN ARSLAN  LİSANS BİTİRME PROJESİ |  |

HAZİRAN-2018

SİVAS

TEZ BİLDİRİMİ

Bu tezdeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edildiğini ve tez yazım kurallarına uygun olarak hazırlanan bu çalışmada bana ait olmayan her türlü ifade ve bilginin kaynağına eksiksiz atıf yapıldığını bildiririm.

İmza

ERHAN ARSLAN

Tarih:

# 

# ÖZET

HAVA KALİTE İZLEME WEB UYGULAMASI

ERHAN ARSLAN

Danışman: Yrd. Doç. Dr. Ahmet Gürkan YÜKSEK

Komisyon

Yrd. Doç. Dr. A. Gürkan YÜKSEK

Yrd. Doç. Dr. Halil ARSLAN

Arş. Gör. Emre DELİBAŞ

Ne yazık ki sanayi devriminden sonra gelişen globalleşen dünyamızı daha fazla sera gazları ve diğer sağlığımızı etkileyen zararlı gazlarla kirlettik. Bünyesi hassas olan insanlar bu durumdan kolayca etkilenmektedirler. Bunun takibini sağlayabilmek, yaşanılan, solunan havanın kirlilik seviyesini belirten bir uygulama yaparak, bu konuda yapılmamış olan bir hizmeti kullanıcılara sunmak istiyorum.

Amaçlar;

- Solunum rahatsızlığı olan insanların havanın kalite takibini sağlayabilmesi

- Farkındalık yaratarak toplumun çevreye karşı duyarlılığını arttırmak

- Tüm mobil platformlarda çalışabilen bir uygulama geliştirmek

Anahtar Kelimeler:

havakalite, airquality, temizhava

İçindekiler

[ÖZET iv](#_Toc515990938)

[1. GİRİŞ 1](#_Toc515990939)

[1.1. Program ve Altyapısının Çalışması 1](#_Toc515990940)

[1.1.1. Program Kullanımı 2](#_Toc515990941)

[1.1.2 Programın Altyapı ve Çalışma Mantığı 2](#_Toc515990942)

[2. KAYNAK ARAŞTIRMASI 2](#_Toc515990943)

[2.1. Front-End Teknolojileri 2](#_Toc515990944)

[2.1.1. JavaScript 3](#_Toc515990945)

[2.1.2 Leaflet JS 4](#_Toc515990946)

[2.1.3 OpenStreetMap 5](#_Toc515990947)

[2.1.4 Bootstrap 6](#_Toc515990948)

[2.1.5 HTML5 CSS3 7](#_Toc515990949)

[2.2 Backend Teknolojileri 8](#_Toc515990950)

[2.2.1 PHP 8](#_Toc515990951)

[2.2.2 MySQL 10](#_Toc515990952)

[3.MATERYAL VE YÖNTEM 11](#_Toc515990953)

[3.1 Front End Materyal ve Yöntemler 11](#_Toc515990954)

[3.2 Back End Materyal ve Yöntemler 11](#_Toc515990955)

[4. ARAŞTIRMA SONUÇLARI VE TARTIŞMA 11](#_Toc515990956)

[4.1 Front-End Araştırma ve Sonuçları 12](#_Toc515990957)

[4.2 BackEnd Araştırma ve Sonuçları 21](#_Toc515990958)

[5. SONUÇLAR VE ÖNERİLER 23](#_Toc515990959)

[5.1 Sonuçlar 23](#_Toc515990960)

[5.2 Öneriler 23](#_Toc515990961)

[KAYNAKLAR 24](#_Toc515990962)

# 1. GİRİŞ

Günümüzde tüm sistemlerin otonom haline gelmesi, daha fazla makina gücü ve daha fazla üretim anlamına gelmektedir. Bu üretimin artması, çeşitli ekonomik model oluşumları, sanayilerin üretim aşamaları ve sonrasında doğayı kirletecek atıkların artmasına ve doğanın bunu kendi kendine temizleyemez bir hal almasını sağlamıştır. Bu artışın üretimden kaynaklı olmayan bir diğer sebebi de fosil yakıt kullanımıdır. Bunun sonucu olarak hava, sera gazları ve diğer kimyasal gazlarla kirlenmiştir. Her canlı bu durumdan etkilenmektedir, özellikle bu gazlara maruz kalmış ve biz insanoğlu gibi akciğer solunumu yapan canlılarda solunum yolu rahatsızlıkları meydana gelmiştir.

Bunun önüne geçmek adına bir web sitesi yapmaya karar verdim. Bu site sayesinde hava kalitesinden rahatsızlık duyan insanlar için havanın solunabilir olup olmadığı, sağlıklı olan insanların hastalanma olasılıklarını azaltmak ve bu sayede daha sağlıklı nesiller yetişmesi ve insanlığın devamının sürdürülebilmesine bir destek olması amacıyla oluşturmak istediğim programın çoklu işlem kapasiteli ve tüm internet bağlantısı olan cihazlardan erişim ve kullanımını sağlamak istedim. Bu amacımın karşılığı olarak arka tarafında sunucu bir sistem kullanarak derlemeye karar verdim. Bu sayede web sitesi son aşamaya geldiğinde internet bağlantısı sağlayabilen tüm cihazlar tarafından kullanılabilen, tamamen işlevsel her kesimin rahatlıkla kullanabileceği bir forma ulaşacaktır.

## 

## 1.1. Program ve Altyapısının Çalışması

Programın tanıtımını iki başlık şeklinde anlatacağım. Adresi internet tarayıcısından giren kullanıcının sitede neler yapabileceklerini anlatacağım. Bir diğer kısımda da kullanıcının bu işlemleri yaparken arkasında çalışmakta olan sistem hakkında kısaca bilgilendireceğim.

### Program Kullanımı

Web sitesine girilen adresin ardından açılan sayfada hava kalitesi izleme yeteneğine sahip olan istasyonların harita üzerindeki konumları belirtilmiştir. Bulunulan şehirdeki yada merak edilen şehrin kirlilik değerlerini konum simgesine tıklanarak solda açılacak olan bilgi ekranında o istasyona ait ve sunucu tarafından desteklenen son güncellenmiş değerlerle birlikte derlenmiş hava kalitesi değerleri görüntülenecektir. Seçilen istasyon sürekli olarak saat başı güncellenen ve istasyonun kalitesine göre elde edilmiş olan verilerini grafiksel olarak ve otoriteler tarafından belirlenen limitini aştığında rengi değişecek bar çubuklar ile havanın kirli olup olmadığını anlayabilecektir.

### 1.1.2 Programın Altyapı ve Çalışma Mantığı

Uygulamanın sunucu destekli olması sebebiyle hem ön taraf (frontend) hem de arka taraf (backend) yazılım teknolojilerine ihtiyaç vardır. Ön taraf genel olarak kullanıcıya kolaylık sağlaması ve estetik olarak bir görünüm olması amaçlanan kısımdır. Javascript ve Jquery görsel desteği ve arka kısım ile iletişimi kurmaktadır. Arka taraf kısmında Php programlama dilini kullanmaktayım. Web servislerinin yazımı bu dil ile gerçekleştirilmiştir, ancak sunucu ile bağlantı kısmında verileri almamda doğrudan katkısı bulunan Aqi topluluğuna ait bir API kullanmaktayım. API ‘nin servis yazımı da javanın web teknolojilerinden Spring Boot ile yazıldığını daha sonra öğrendim.

# 2. KAYNAK ARAŞTIRMASI

Kullanılacak teknolojilerle ilgili olarak belgelemeler aşağıdaki gibi verilmiştir.

## 2.1. Front-End Teknolojileri

### 2.1.1. JavaScript



Kısaca JS olarak bilinen JavaScript yüksek seviye yorumlanabilir nesneye dayalı, dinamik prototip tabanlı bir betik dilidir. HTML ve CSS ile birlikte World Wide Web teknolojisinin 3 temel bileşeninden birisi olarak kabul edilir ve web sitelerinin birçoğunda etkin olarak kullanılmaktadır. Oluşturulmuş olan API’si sayesinde metin, dizi, zaman yada düzenli ifadeler ile birlikte doküman obje modeli gibi birçok etmenin kullanılabilmesine olanak sağlar.

JS önceleri sadece client tarafında kullanılıyor olsa da , javascript motorunun gömülü olduğu sunucu tarafı yazılım, veritabanı gibi sunucu tarafı yazılımlar ve word ve pdf gibi web tabanlı olmayan yazılımlar ve benzeri şekilde çalıştırabilmesi, dolayı günümüzde mobil ve masaüstü uygulamaları için bile uygun bir hale gelmiştir.

#### 2.1.1.1 Tarihçe

İlk olarak Brenden Eich tarafından geliştirilmiş olup, ilk defa Aralık 1995 tarihinde Netspace Navigator 2.0 ile birlikte piyasaya sürüldü. Geliştirme aşamasında adı Mocha olarak geçmesine rağmen Eylül 1995 tarihine kadar resmi olarak *LiveScript* olarak isimlendiriliyordu. Netscape 2.0 Beta3 ile birlikte ismi JavaScript olarak değiştirildi.

JavaScript'in isim değişikliği ile Netscape'in Java teknolojilerine (Java Applet) destek vermesi yaklaşık olarak aynı tarihlere denk gelir. Microsoft ile Netscape'in web teknolojileri ve platformları alanında yarıştığı bir dönemde, Netscape Sun Microsystems 'ın geliştirdiği Java platformuna destek vererek dağıtık bir işletim sistemi yaratmayı öngörüyordu. JavaScript ise, Visual Basic 'in yaptığı gibi, profesyonel olmayan programcıları hedefleyen daha basit bir betik dili olarak düşünülmekteydi.

#### 2.1.1.2 Standartlaşma

1996 yılının kasım ayında Netscape firması JavaScript'in endüstri standardı olarak belirlenmesi amacıyla Ecma International firmasına başvuruda bulunduğunu ilan etti. Bunun sonucunda standardize edilen sürüm ECMAScript olarak isimlendirildi ve 1997 yılının haziran ayında Ecma International, ECMA-262 belirtiminin ilk sürümünü yayımladı. Haziran 1998'de ISO/IEC-16262 standardına uyumlu hale getirilecek değişiklikler yapılarak ikinci sürüm, Aralık 1999'da ise üçüncü sürüm yayımlandı.

ECMAScript'in dördüncü sürümü üzerinde ciddi bir çalışma yapılmasına rağmen asla tamamlanamadı, ancak beşinci sürüm için önemli bir ilham kaynağı oldu. Beşinci sürüm 2009 yılının aralık ayında yayımlandı. ECMAScript'in şu an geçerli olan son standart belirtimi, Haziran 2011 tarihinde yayımlanan 5.1 sürümüdür ve Javascript 1.8.5 tarafından desteklenmektedir.

### 2.1.2 Leaflet JS



Leaflet açık kaynaklı bir JavaScript kütüphanesidir ve web dünyasında haritalama uygulamaları için çoğunlukla tercih edilir. İlk kez 2011 de piyasaya sürülen Leaflet JS, çoğu mobil ve masaüstü platformlarını, HTML5 ve CSS3 ü OpenLayers ve Google Maps API sayesinde desteklemektedir. FourSquare, Pinterest ve Flicker gibi bilinen haritalama uygulamaları bu JavaScript kütüphanesi katkısıyla oluşturulmuştur.

Leaflet son sürümü olan Ocak 2018 1.3.1 sürümü ile stabil hale gelmiştir. 2013 yılında MapBox a dahil olan Vladimir Agafonkin tarafından geliştirilmiştir.

Leaflet kullanıcılara public sunucularda bulununan web haritalarını diğer katmanları da opsiyonel olacak şekilde kolayca gösterimlerine izin verir. GeoJSON dosyalarından özellikler yükleyerek yeni katmanlar oluşturulabilmesine ve tıklandığında linke göndererek kullanılmasına izin verir.

### 2.1.3 OpenStreetMap



OpenStreetMap (OSM) özgür yazılım şartları altında oluşturulan açık kaynaklı bir dünya çapında harita oluşturma projesidir. GPS alıcıları ve diğer kamu malı kaynaklardan toplanan bilgiler ile oluşturulur.

OSM açık kaynaklı olan ve dünyanın her yerinden gönüllü katılımcılarının oluşturdukları bir dünya haritası projesidir. Tıpkı Vikipedi gibi kullanıcıların içeriklerini oluşturdupu bir yapıya sahiptir. Haritalar kullanıcıların sunmuş olduğu GPS izleri, uydu fotoğrafları ve yerel yönetimler tarafından sağlanan .eşitli veriler doğrultusunda oluşturulmaktadır. Yapılan değişiklikler anında yayınlanır.

OpenStreetMap’de bulunan bütün içerik Open Database Licence lisansı ile korunmaktadır. Katkılar üreticilerine ait olmaya devam etmekle birlikte, telif feragatlı lisansı, içeriğin serbestçe dağıtımına olanak sağlamaktadır. Tamamen ücretsiz ve reklamsızdır. Yaklaşık 3.7 milyon kişi tarafından kullanılmakta ve desteklenmektedir.

### 2.1.4 Bootstrap



Twitter Bootstra**p** (ya da kısaca **Bootstrap**) açık kaynak kodlu, web sayfaları veya uygulamaları geliştirmek için kullanılabilecek araçlar bütünü ve önyüz çatısıdır. Bootstrap, web sayfaları veya uygulamalarında kullanılabilecek, HTML ve CSS tabanlı tasarım şablonlarını içerir. Bu şablonlar form, navigasyon çubuğu, buton gibi arayüz bileşenleri oluşturmakta kullanılabilmektedir.

Boostrap orijinal adıyla Twitter Blueprint adıyla Mark Otto ve Jacob Thornton tarafından Twitter için bir framework olarak kullanılmıştır.

### 2.1.5 HTML5 CSS3



**HTML** internet sayfalarının en temel kod yapısıdır. Yani bir internet sayfasının iskeletidir diyebiliriz. Bir sayfa sunucu tarafından işlendikten sonra geriye html sayfa yapısı döndürülür. Tarayıcılar da (Chrome, Firefox, Explorer, Safari, Opera vs) html kodlarına göre sayfayı görüntülerler.

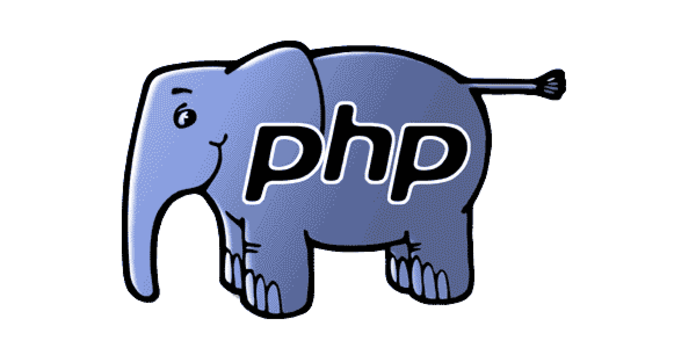
HTML yapısı zaman içinde geliştirilerek son sürüm olan **HTML5** adı altında bir standart oluşturuldu. Bu geliştirmenin temel sebebi ise her türlü tarayıcıya ve cihaza uyum sağlayarak ortak ve iyi bir görünüm sağlamaktı.

Bu temel HTML iskeletini görsel anlamda iyileştirmek için stil kodları bulunur. Bu kodlara ise CSS kodları denir. Sayfa görünümü, menü, resim, yazı tipleri, sayfa içi açılır pencereler vs. gibi bir çok özelliklerin bu kodlar ile tanımlanır. CSS’in son sürümü olan **CSS3** ile HTML5 mükemmel bir uyum sağlamış, önceki sürümlerde oluşan hatalardan ayıklanmıştır.

## 2.2 Backend Teknolojileri

Programın arka tarafında bulunan kullanıcının görsel olarak göremediği verinin yönetimi, saklanması ve front end kısmına olan data gönderimini sağlayan ve asıl işin yapıldığı kısım olarak değerlendirilen bölümdür.

### 2.2.1 PHP



Aslen: Personal Home Page - Kişisel Ana Sayfa , internet için üretilmiş, sunucu taraflı, çok geniş kullanımlı, genel amaçlı, HTML içerisine gömülebilen betik bir programlama dilidir. İlk kez 1995 yılında Rasmus Lerdorf tarafından yaratılan PHP'nin geliştirilmesi bugün PHP topluluğu tarafından sürdürülmektedir. Ocak 2013 itibarıyla 244 milyondan fazla web sitesi PHP ile çalışırken, 2.1 milyon web sunucusunda PHP kurulumu bulunmaktadır.

PHP kodları PHP işleme modülü bulunan bir web sunucusu tarafından yorumlanır ve çıktı olarak web sayfası üretilir. Bu kodlar veriyi işlemek üzere harici bir dosyaya kaydedilerek çağırılabildiği gibi doğrudan HTML kodunun içine de gömülebilir. PHP zaman içinde bir komut satırı arayüzü sunacak şekilde evrilmiştir, PHP-GTK yardımıyla grafiksel masaüstü uygulaması geliştirmek de mümkündür.

PHP özgür bir yazılım olup PHP Lisansı ile dağıtılmaktadır. Bu lisans kullanım şartları kısmında GNU Genel Kamu Lisansı ile örtüşmese de, PHP tüm web sunuculara ve hemen hemen tüm işletim sistemi ve platforma ücretsiz olarak yüklenebilir.

#### 2.1.1.1 Tarihçe

PHP'nin geliştirilmesi 1994 yılında Rasmus Lerdorf ‘un kişisel web sitesini yönetmek için bir takım CGI (Common Getaway Interface) Perl betikleri yazmasıyla başladı. Rasmus bu betikler yardımıyla özgeçmişini sitede görüntülüyor ve sitenin ziyaretçi trafiğini kaydediyordu. Bir süre sonra web formları ile etkileşime girebilecek, veritabanları ile iletişim kurabilecek ve daha hızlı çalışabilecek şekilde bu betikleri C ile yeniden kodladı ve bu uygulamaya Kişisel Ana Sayfa/Form Yorumlayıcı (PHP/FI) adını verdi. PHP/FI yardımıyla basit anlamda dinamik web siteleri oluşturmak oldukça kolaydı. Rasmus, 8 Haziran 1995 tarihinde hata ayıklama sürecini hızlandırmak ve kaynak kodu iyileştirmek üzere bir Usenet tartışma grubunda PHP/FI uygulamasını Personal Home Page Tools (PHP Tools) adıyla ilk kez duyurdu. 2013 yılı itibarıyla PHP içinde mevcut olan Perl benzeri değişken tanımlama, form yönetimi ve betik kodları arasına HTML kodu yerleştirebilme gibi temel özellikler bu ilk sürümde de mevcuttu. Sözdizimi genel olarak Perl'e benzese de daha kısıtlı ve basit ancak daha tutarsızdı. Bir geliştirme takımıyla aylar süren çalışma ve test aşamalarının ardından PHP/FI 2 sürümü 1997 Kasım ayında resmi olarak duyuruldu.

Zeev Suraski ve Andi Gutmans 1997 yılında ayrımlayıcı’yı baştan yazdılar ve dilin ismini PHP: Hypertext Preprocessor olarak değiştirerek PHP 3 ün temellerini attılar. Bunu PHP 3 ün herkese açık olan test sürecinin başlaması izledi ve 1998 yılının Haziran ayında PHP 3 resmi olarak duyuruldu. Suraski ve Gutmans bu sürecin ardından PHP'nin çekirdeğini sil baştan kodlamak suretiyle 1999 yılında Zend Motorunu geliştirdiler ve İsrail'in Ramat Gan kentinde Zend Technologies isimli bir şirket kurdular.

22 Mayıs 2000 tarihinde temelleri Zend Engine 1.0 ile güçlendirilmiş PHP 4 duyuruldu ve Ağustos 2008 itibarıyla 4.4.9 sürümüne ulaşana kadar aktif olarak geliştirildi. Bugün PHP 4 ün geliştirilmesi durdurulmuştur ve güvenlik güncellemeleri dahil olmak üzere üzerinde hiçbir iyileştirme yapılmamaktadır.

13 Temmuz 2004 tarihinde yeni Zend Engine II ile güçlendirilmiş PHP 5 yayınlandı. PHP 5, nesneye yönelik programlama için çok daha fazla olanak sağlıyor, PHP Veri Nesneleri (PDO) eklentisi yardımıyla veritabanlarına erişim için oldukça tutarlı ve hızlı bir arayüz yanında performansa yönelik çok sayıda iyileştirme barındırıyordu. 2008 yılında PHP 4 ün tutarlı sürümünün yayınlanması durduruldu ve PHP 5 geliştirilmesi devam eden tek tutarlı sürüm oldu. PHP duruk içselleştirim (Late static binding) özelliği barındırmıyordu ancak 5.3 sürümü ile bu önemli eksik giderildi.

### 2.2.2 MySQL



6 milyondan fazla sistemde yüklü bulunan çoklu iş parçacıklı, çok kullanıcılı, hızlı ve sağlam bir veritabanı yönetim sistemidir.

Ücretsiz dağıtımları mevcut olup, ticari lisans ile birlikte kullanmak isteyenler için de ücretli bir lisans seçeneği de mevcuttur. Linux altında daha hızlı bir performans sergilemektedir. Açık kaynak kodludur ve ikilik kod halinde indirilebilir sürümleri de mevcuttur. Ayrıca ODBC sürücüleri de bulunduğu için birçok geliştirme platformunda rahatlıkla kullanılabilir.

# 3.MATERYAL VE YÖNTEM

## 3.1 Front End Materyal ve Yöntemler

Backend kısmından gelen JSON formatındaki datayı JavaScript ile Sublimetext editör kullanarak işlemenin ardından OSM ve leaflet yardımıyla son görünüm elde edilir.

## 3.2 Back End Materyal ve Yöntemler

SpringBoot framework ü kullanan bir API vasıtasıyla waqi.com adresinden çekilen veri PHP servisinde derlenir ve veritabanına frontend kısmının kullanılması için aktarılır. Her iki taraf için de SublimeText editörünü kullandım ve sunucu olarak localhost un içinde bulunduğu WAMP server ı kullandım.

# 4. ARAŞTIRMA SONUÇLARI VE TARTIŞMA

Proje fikrinin ardından örneklerini araştırdım fakat bu anlamda tutarlı bir örneğe rastlamadım. [www.havaizleme.gov.tr](http://www.havaizleme.gov.tr) adresinde bulunan verileri kullanmaya karar verdim fakat bunu sağlayamadım. [www.waqi.com](http://www.waqi.com) adresinde yer alan verilerden Türkiye’ye ait istasyonları saptayıp bunlarla ilgili verileri çekmeyi başardım. waqi.com sitesinin havaizleme.gov.tr adresinden veriyi çektiğini ve bunu saatlik olarak yaptığını öğrendim. Bu sebeple projemi geliştirdikten sonra çektiğim verilerin yaklaşık 2 saat öncesine ait olduğunu belirledim.

## 4.1 Front-End Araştırma ve Sonuçları

Projemde tecrübelerime ve bilgi düzeyime bağlı olarak 2 temel değişiklik yaptım. Ön kısımda Angularjs ve Cordova Ionic kullanmak ve sadece hibrit mobil platforma yönelmek yerine ön kısım tasarımımı JavaScript, Boostrap, HTML5 ve CSS3, LeafletJS ve OpenStreetMap teknolojilerine kullanarak görsel olarak daha anlamlı bir görünüm elde etmeyi başardım. Kaynak kodları aşağıdaki gibi yer almaktadır.

<?php

require 'controller/Station.php';

$st = new Station();

//Multi Station

$stDataAll = $st->getStationInfoAll();

?>

<!DOCTYPE html>

<html>

<head>

<title>HAVA KALİTE</title>

<meta name="viewport" content="width=device-width, initial-scale=1.0, maximum-scale=1.0, user-scalable=no">

<link rel="stylesheet" href="https://use.fontawesome.com/releases/v5.0.13/css/all.css" integrity="sha384-DNOHZ68U8hZfKXOrtjWvjxusGo9WQnrNx2sqG0tfsghAvtVlRW3tvkXWZh58N9jp" crossorigin="anonymous">

<link rel="stylesheet" href="https://unpkg.com/leaflet@1.0.1/dist/leaflet.css" />

<link rel="stylesheet" href="css/leaflet-sidebar.css" />

<link href="css/themify-icons.css" rel="stylesheet" type="text/css" media="all" />

<link href="css/bootstrap.css" rel="stylesheet" type="text/css" media="all" />

<link href="css/theme.css" rel="stylesheet" type="text/css" media="all" />

<link href="css/custom.css" rel="stylesheet" type="text/css" media="all" />

<link href='http://fonts.googleapis.com/css?family=Lato:300,400%7CRaleway:100,400,300,500,600,700%7COpen+Sans:400,500,600' rel='stylesheet' type='text/css'>

<style>

body {

padding: 0;

margin: 0;

}

html, body, #map {

height: 93%;

width: 100%;

font: 10pt "Helvetica Neue", Arial, Helvetica, sans-serif;

}

</style>

</head>

<body>

<div class="nav-container">

<nav class="nav-centered">

<div class="nav-bar text-center" style="height: 165px;">

<div class="module widget-handle mobile-toggle right visible-sm visible-xs">

<i class="ti-menu"></i>

</div>

<div class="module-group text-left" style="z-index: 999999;">

<div class="module left" style="z-index: 999999;">

<ul class="menu" style="z-index: 999999;">

<li class="has-dropdown" style="z-index: 999999;">

<a href="#">

ERHAN ARSLAN

</a>

<ul>

<li>

<a href="#">

Web Sitesi

</a>

</li>

</ul>

</li>

</ul>

</div>

</div>

</nav>

</div>

<div id="sidebar" class="sidebar collapsed">

<!-- Nav tabs -->

<div class="sidebar-tabs">

<ul role="tablist">

<li><a href="#home" role="tab"><i class="fa fa-info"></i></a></li>

</ul>

<ul role="tablist">

<li><a href="#settings" role="tab"><i class="fa fa-gear"></i></a></li>

</ul>

</div>

<div class="sidebar-content">

<div class="sidebar-pane" id="default">

<h1 class="sidebar-header">İstasyon Bilgisi<span class="sidebar-close"><i class="fa fa-caret-left"></i></span></h1>

<div>

<br>

<h3 id="stname" style="margin-left: 25px;"></h3>

<h3 id="date" style="margin-left: 25px;"></h3>

<br>

<h3 id="pm101" style="margin-left: 25px;">Pm10 24 Saatlik Grafiği</h3>

<canvas id="pm10"></canvas>

<h3 id="pm251" style="margin-left: 25px;">Pm25 24 Saatlik Grafiği</h3>

<canvas id="pm25"></canvas>

<h3 id="o31" style="margin-left: 25px;">O3 24 Saatlik Grafiği</h3>

<canvas id="o3"></canvas>

<h3 id="so21" style="margin-left: 25px;">SO2 24 Saatlik Grafiği</h3>

<canvas id="so2"></canvas>

<h3 id="no21" style="margin-left: 25px;">NO2 24 Saatlik Grafiği</h3>

<canvas id="no2"></canvas>

<h3 id="co1" style="margin-left: 25px;">CO 24 Saatlik Grafiği</h3>

<canvas id="co"></canvas>

</div>

</div>

<div class="sidebar-pane" id="home">

<p>ERHAN ARSLAN KİMDİR?</p>

</div>

<div class="sidebar-pane" id="settings">

<h1 class="sidebar-header">Settings<span class="sidebar-close"><i class="fa fa-caret-left"></i></span></h1>

</div>

</div>

</div>

<div id="map" class="sidebar-map"></div>

<script

src="https://code.jquery.com/jquery-3.3.1.min.js"

integrity="sha256-FgpCb/KJQlLNfOu91ta32o/NMZxltwRo8QtmkMRdAu8="

crossorigin="anonymous"></script>

<script src="https://unpkg.com/leaflet@1.0.1/dist/leaflet.js"></script>

<script src="js/leaflet-sidebar.js"></script>

<script src="js/pako.js"></script>

<script src="js/moment.js"></script>

<script src="https://cdnjs.cloudflare.com/ajax/libs/Chart.js/2.7.2/Chart.bundle.js"></script>

<script src="js/jquery.min.js"></script>

<script src="js/bootstrap.min.js"></script>

<script src="js/parallax.js"></script>

<script src="js/scripts.js"></script>

<script>

var stationAll=<?php echo json\_encode($stDataAll)?>;

var points=[];

var point=[];

for(var i in stationAll)

{

point.push(parseFloat(stationAll[i]['stationId']));

point.push(parseFloat(stationAll[i]['stationLat']));

point.push(parseFloat(stationAll[i]['stationLong']));

points.push(point);

point=[];

}

var singleStData;

var dataSaat = [];

var dataY = [];

var dataColor = [];

var chartType=["pm10","pm25","o3","so2","no2","co"];

function createChartData(){

for(var i = 0; i < stationAll.length; i++)

{

if(stationAll[i].stationId == singleStData['data'][0].idx)

{

document.getElementById("stname").innerHTML ="İstayson Adı: "+stationAll[i].stationName;

}

}

document.getElementById("date").innerHTML ="Ölçüm Tarihi: "+singleStData['data'][0]['time']+" / "+ moment(singleStData['data'][0]['date']).format('D-MM-YYYY');

for(var k in chartType){

dataSaat = [];

dataY = [];

dataColor = [];

var stData=singleStData["analizData"][singleStData["data"][0]["idx"]][chartType[k]];

for(var i in stData)

{

dataSaat.push(stData[i]['label']);

dataY.push(stData[i]['y']);

dataColor.push(dataGetColor(stData[i]['y']));

}

if(dataY[0] != "-" || dataY[3] != "-" || dataY[5] != "-" || dataY[7] != "-" || dataY[9] != "-" || dataY[11] != "-" || dataY[13] != "-" || dataY[15] != "-" || dataY[17] != "-" || dataY[19] != "-" || dataY[21] != "-" || dataY[23] != "-"){

document.getElementById(chartType[k]).style.display = "block";

document.getElementById(chartType[k]+"1").style.display = "block";

var old\_element = document.getElementById(chartType[k]);

var new\_element = old\_element.cloneNode(true);

old\_element.parentNode.replaceChild(new\_element, old\_element);

createChart(chartType[k]);

}else{

document.getElementById(chartType[k]).style.display = "none";

document.getElementById(chartType[k]+"1").style.display = "none";

}

}

}

var map = L.map('map');

map.setView([39, 34.5], 6);

L.tileLayer('http://{s}.tile.openstreetmap.org/{z}/{x}/{y}.png', {

maxZoom: 18,

attribution: 'Map data &copy; OpenStreetMap contributors'

}).addTo(map);

var sidebar = L.control.sidebar('sidebar').addTo(map);

<?php for ($i=0; $i <count($stDataAll) ; $i++) {?>

marker = new L.marker([<?= $stDataAll[$i]['stationLat']?>,<?= $stDataAll[$i]['stationLong']?>])

.addTo(map).on('click', function () {

sidebar.toggle();

document.getElementById("default").classList.add("active");

document.getElementById("home").classList.remove("active");

getStationData(<?= $stDataAll[$i]['stationId']?>);

});

<?php }?>

function getStationData(val)

{

var settings = {

"async": true,

"crossDomain": true,

"url": "http://45.55.134.38:8091/station/data?idx="+val,

"method": "GET"

}

$.ajax(settings).done(function (response) {

singleStData= JSON.parse(decompressData(response));

createChartData();

});

}

function decompressData(data)

{

// Get some base64 encoded binary data from the server. Imagine we got this:

var b64Data = data;;

// Decode base64 (convert ascii to binary)

var strData = atob(b64Data);

// Convert binary string to character-number array

var charData = strData.split('').map(function(x){return x.charCodeAt(0);});

// Turn number array into byte-array

var binData = new Uint8Array(charData);

// Pako magic

var data = pako.inflate(binData, { raw: true });

// Convert gunzipped byteArray back to ascii string:

var strData = String.fromCharCode.apply(null, new Uint16Array(data));

return strData;

}

function createChart(val){

var ctx = document.getElementById(val).getContext('2d');

var myChart = new Chart(ctx, {

type: 'bar',

data: {

labels: dataSaat,

datasets: [{

label: 'Değer',

data: dataY,

backgroundColor: dataColor

}]

},options: {

legend: {

display: false

},

tooltips: {

callbacks: {

label: function(tooltipItem) {

return tooltipItem.yLabel;

}

}

}

}

});

}

function dataGetColor($val)

{

return ($val <=50) ? "rgba(43, 251, 0, 1)" :

(($val <=70) ? "rgba(255, 254, 0, 1)" :

(($val <=100) ? "rgba(254, 157, 0, 1)" :

(($val <=120) ? "rgba(255, 0, 0, 1)" :

(($val <=150) ? "rgba(97, 3, 128, 1)" : "rgba(97, 3, 77, 1)"))));

}

</script>

</body>

</html>

## 4.2 BackEnd Araştırma ve Sonuçları

Projemi ilk araştırmalarıma göre Spring Boot ve Hibernate ile oluşturmayı uygun görmüş olsam da sonrasında bunun için daha sade bir yapı olan PHP yi kullanmaya karar verdim. Spring Boot ile yazılmış olan gelen datayı API vasıtası ile PHP servisime alıp daha sonra bunu veritabanına front kısmının kullanabilmesi için gönderdim. Kaynak kodları aşağıdaki gibi yer almaktadır.

<?php

class Station {

public function getStationData($id)

{

$curl = curl\_init();

curl\_setopt\_array($curl, array(

CURLOPT\_PORT => "8091",

CURLOPT\_URL => "http://45.55.134.38:8091/station/data?idx=".$id,//uzak sunucudan id ye göre veri çektim

CURLOPT\_RETURNTRANSFER => true,

CURLOPT\_ENCODING => "",

CURLOPT\_MAXREDIRS => 10,

CURLOPT\_TIMEOUT => 30,

CURLOPT\_HTTP\_VERSION => CURL\_HTTP\_VERSION\_1\_1,

CURLOPT\_CUSTOMREQUEST => "GET",

));

$response = curl\_exec($curl);

$err = curl\_error($curl);

curl\_close($curl);

if ($err) {

return "err";

} else {

$data=gzinflate(base64\_decode($response));

return json\_decode($data,true);

}

}//func end

public function calculatePm10($val)

{

if($val=='-')

{

return null;

}else

return ($val <=50) ? "rgba(43, 251, 0, 0.5)" :

(($val <=70) ? "rgba(255, 254, 0, 0.5)" :

(($val <=100) ? "rgba(254, 157, 0, 0.5)" :

(($val <=120) ? "rgba(255, 0, 0, 0.5)" :

(($val <=150) ? "rgba(97, 3, 128, 0.5)" : "rgba(97, 3, 77, 0.5)"))));

}//calculatePm10 End

public function getStationInfoAll()

{

$curl = curl\_init();

curl\_setopt\_array($curl, array(

CURLOPT\_PORT => "8091",

CURLOPT\_URL => "http://45.55.134.38:8091/station/info",

CURLOPT\_RETURNTRANSFER => true,

CURLOPT\_ENCODING => "",

CURLOPT\_MAXREDIRS => 10,

CURLOPT\_TIMEOUT => 30,

CURLOPT\_HTTP\_VERSION => CURL\_HTTP\_VERSION\_1\_1,

CURLOPT\_CUSTOMREQUEST => "GET",

));

$response = curl\_exec($curl);

$err = curl\_error($curl);

curl\_close($curl);

if ($err) {

echo "cURL Error #:" . $err;

} else {

$data=gzinflate(base64\_decode($response));

return json\_decode($data,true);

}

}

}// class end

?>

# 5. SONUÇLAR VE ÖNERİLER

## 5.1 Sonuçlar

PHP JavaScript diline olan hakimiyetimi arttırmış oldum. Bununla birlikte havakalitesi projesinin sonucu olarak anlık olmasa bile yakın zamanın havasını takip edilebilen bir websitesi oluşturulmuş oldu ve internete bağlanabilen ve bir tarayıcısı olan tüm cihazlardan erişim olanağı bulunmaktadır.

## 5.2 Öneriler

Proje mobile derlenebilir, push notification ile bulunan konumun hava kalite değerleri kullanıcıya aktarılabilir.

# KAYNAKLAR

[1]<https://tr.wikipedia.org/wiki/JavaScript>

[2]<https://en.wikipedia.org/wiki/JavaScript>

[3]<https://getbootstrap.com/docs/4.1>

[4]<https://wiki.openstreetmap.org/wiki/Main_Page>

[5]<https://gelecegiyazanlar.turkcell.com.tr/konu/web-programlama/egitim/301-javascript/javascript-nedir>

[6]<https://en.wikipedia.org/wiki/Leaflet_(software)>

[7]<https://leafletjs.com/reference-1.3.0.html>

[8]<https://en.wikipedia.org/wiki/OpenStreetMap>

[9]<https://www.mapsmarker.com/wp-content/uploads/jseventsapi/>

[10]<https://en.wikipedia.org/wiki/Bootstrap_(front-end_framework)>

[11]<https://tr.wikipedia.org/wiki/Bootstrap_(önyüz_çatısı)>

[12]<https://getbootstrap.com/docs/4.1/about/overview/>