

TICK - HABER UYGULAMASI

Ata GÜLALAN - Oğuzhan TÜRKER

Bilgisayar Mühendisliği Bölümü

Kocaeli Üniversitesi

xavaneo@gmail.com oguzturker8@gmail.com

Özet

Uygulama, kullanıcıların mobil platformda kullanabilecekleri bir haber uygulamasıdır.

1. Giriş

Haberler sisteme bir haber yükleme arayüzünden yüklenmektedir. Bunun için kişinin önce arayüze, uygulama yöneticileri tarafından belirlenen kullanıcı adı ve şifre ile bu sisteme giriş yapması gerekir. Her haberin birer resim, başlık, etiket ve tarih kısmı vardır. Bu bilgiler haberin sisteme eklendiği esnada veya daha sonra belirlenir. Kullanıcı isterse etiket ekleme kısmından haberle ilgili yeni bir etiket oluşturabilir. Yeni haber için tarih otomatik belirlenir ancak isterse kullanıcı özel bir tarih belirleyebilir.

Sisteme eklenen haber, kullanıcılara haberin içerik kısmının sağ üstünde bulunan uçak butonu ile bildirim olarak gönderilebilir. Uygulama kullanıcısı bildirim tıklayarak ilgili habere ulaşabilir. Eğer kullanıcı zaten uygulamada ise, bildirim uygulama ekranında belirir.

Mobil kısımda, sunucu tarafından sisteme yüklenen tüm haberler en yeni haberden en eski habere doğru olmak üzere sıralanmaktadır. Bu bölümde kullanıcı okumak istediği haberin üzerine dokunur, böylece haberin içerik sayfasına yönlendirilir. Bu bölümde, üstten aşağı olmak üzere istenen haberin; başlık, içerik ve istatistik bölümü bulunmaktadır. Kullanıcı isterse haberin sonunda bulunan bu kısımdan haberi beğenip/beğenmeyerek fikrini sunabilir.

2. Temel Bilgiler

Proje gelişiminde;
Programlama dili olarak “JavaScript” kullanılmıştır.
Tümleşik geliştirme ortamı olarak “Android Studio ve Visual Studio Code” kullanılmıştır.
Çalışma ortamı olarak “NodeJS” kullanılmıştır.
Hibrit uygulama köprüsü olarak “React-Native” kullanılmıştır.

3. Tasarım

Proje aşağıdaki başlıklar altında geliştirilmiştir.

3.1 Arayüz

3.1.1 Sunucu Arayüzü

Haberlerin oluşturulduğu sunucunun arayüz bileşenleri aşağıda listelenmiştir.

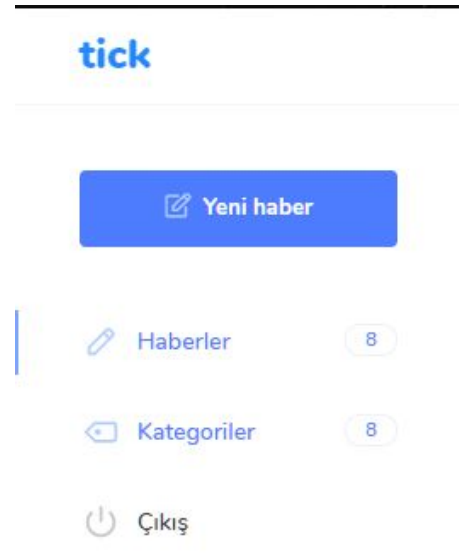
Sunucu arayüzünün sol kısmında yönetim paneli bulunmaktadır (Şekil 1). Burada;

Yeni Haber butonu ile sisteme yeni bir haber eklenebilir.

Haberler sekmesinden sistemdeki mevcut haberler listelenebilir.

Kategoriler sekmesinden haberlere eklenen kategoriler sıralanabilir ve yeni bir kategori eklenebilir.

Çıkış bölümünden ise uygulama sunucusundan güvenli çıkış yapılır.



(Şekil 1 : Yönetim Paneli)

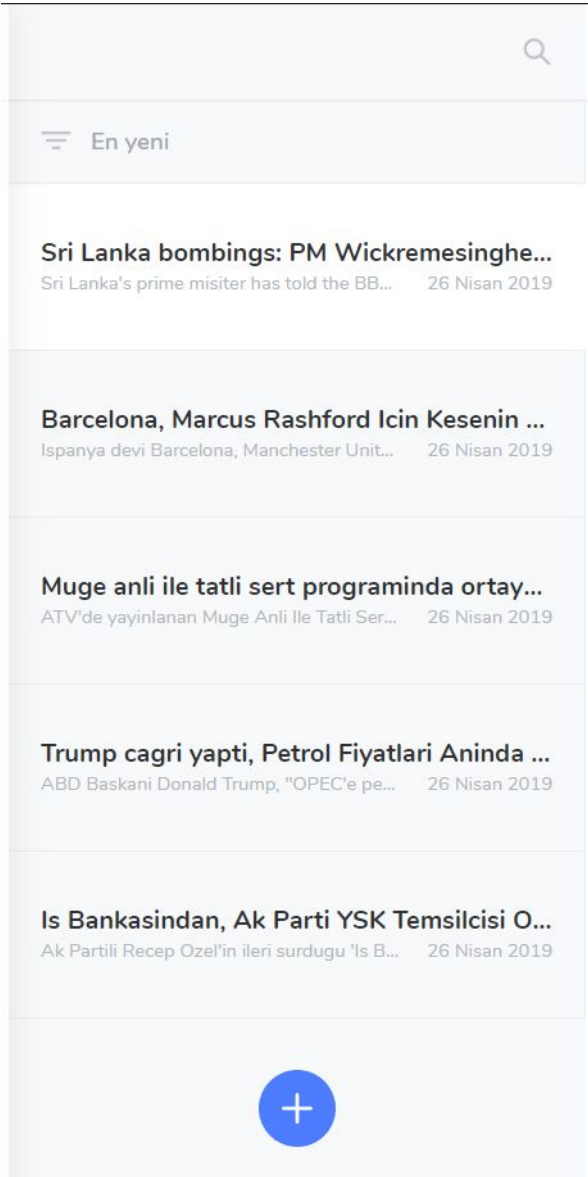
Orta bölümde sistemdeki haber listesi bulunmaktadır (Şekil 2).

Burada;

Arama kutucuğu, sistemdeki haberleri girdiğini karakterlere göre filtreler.

En yeni / En eski butonu, sistemdeki haberleri en yeniden eskiye veya en eskiden yeniye olarak sıralar. Başlangıç ayarı en yenisidir.

Yeni Haber butonu ile sisteme yeni bir haber eklenilebilir.



(Şekil 2 : Haber Listesi)

Sağ bölümde sistemdeki haber bilgileri bulunmaktadır (Şekil 3).

Orta bölümünde bulunan haberlere tıklanılarak ilgili haberin içerik bölümüne erişilir.

Burada;

Tarih bölümü haberin gözükeceği tarihi ayarlar.

Başlık mobil uygulamadaki haberin başlık kısmını oluşturur.

Etiketler kısmında haberlere istenilen etiketler eklenebilir. Kullanıcılar bu etiketlere göre haberleri filtreleyebilirler.

İçerik kısmında haberin ilgili içeriği yazılır.

Görsel yükle, haberin görselini sisteme yükler.

Görsel kaldır, haberde kullanılan ilişkili görseli kaldırır.

Kaydet butonu yapılan yeni haber girişini ya da haber değişikliğini kaydeder. Butona basılmadan yapılan değişiklikler geçerli sayılmaz ve kullanıcıyı uyan bir pop-up çıkmasına neden olur. Kullanıcı isterse yaptığı değişiklikleri iptal edebilir.

Çöp kutusu seçilen haberi sistemden kaldırır.

Gönder butonu haberi mobil kullanıcılara bildirim olarak gönderir.



(Şekil 3 : Haber İçerik Bölümü)

3.1.1 Mobil Arayüz

Ana sayfada listelenen haberlerin arayüz bileşenleri aşağıda listelenmiştir.

tick uygulama logosudur.

Tümü haberlerin, girilen etiketlere göre filtelenmesini sağlar.

Haber resmi, sunucuda seçilen haber resmi burada gösterilir.

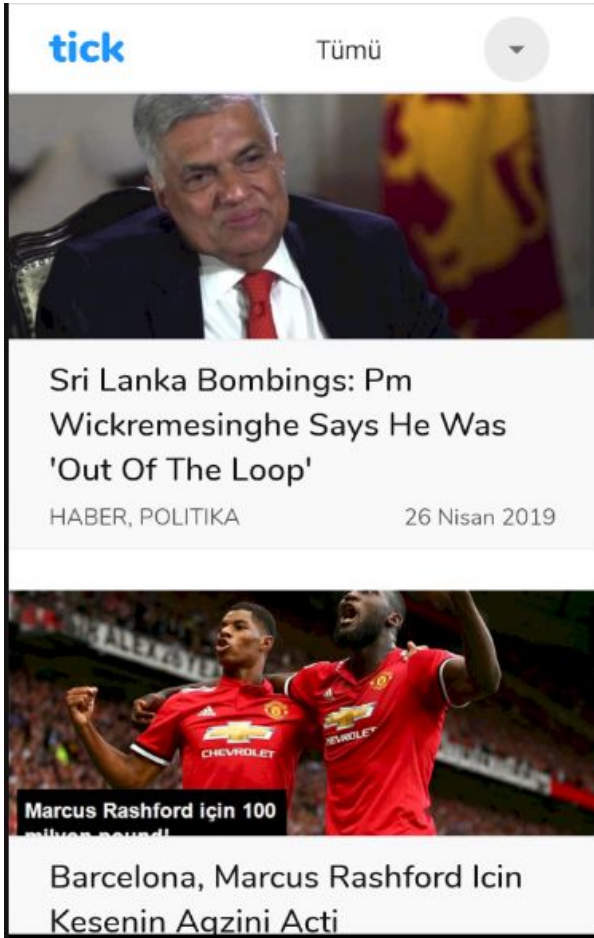
Haber başlığı, sunucuda belirlenen haber başlığı bu bölümde gösterilir.

Etiket haberin etiketleri bu bölümde gösterilir.

Tarih, haberin belirlenen tarihi bu bölümde gösterilir.

Kullanıcı geri tuşuna basarak uygulamada yer alan haberleri güncelleyebilir.

Uygulamanın herhangi bir bölümünde, yeni haber geldiğinde kullanıcıya, uygulamanın içerisinde bildirim gösterilir. Eğer kullanıcı uygulamada değilse, bu bildirim sistem bildirimi olarak gönderilir ve tıklanıldığında uygulamayı açar.



(Şekil 4 : Uygulama Ana Sayfası)

Habere tıklanıldığında açılan mobil içerik kısmının arayüz bileşenleri aşağıda listelenmiştir.

Geri butonu ana sayfaya döndürür.

Başlık haberin başlığını görüntüler.

Görsel sunucuda haber için yüklenen görseldir.

Görüntülenme kullanıcıların haberi

görüntülediklerinde artan, gösterim sayacıdır.

Beğen butonu, kaç kullanıcının bu haberi beğendiğini görüntüler ve kullanıcılar bu butona tıklayarak haberi beğenebilir.

Beğenme butonu, kaç kullanıcının bu haberi beğenmediğini görüntüler ve kullanıcılar bu butona tıklayarak haberi beğenmeyebilir.



(Şekil 5 : Haber Sayfasının İstatistik Bölümü)

3.2 İletişim

URL’de gönderilen parametreler iki nokta (:) ile ifade edilmiştir. Sunucu arayüzünde ve mobilde iletişim için bulunan endpointler aşağıda listelenmiştir:

- **POST "/login"** : Veritabanında istenen kullanıcı bilgisinin bulunup bulunmadığını kontrol eder ve buna göre token döndürür. Kullanıcı adı ve şifre bekler.
- **POST "/admin/add"** : Yeni bir haber ekler veya eğer haber var ise günceller. Token ve haber ile ilgili parametreleri bekler.
- **POST "/admin/remove"** : Belirtilen haberi veritabanından siler. Token ve haber kimlik numarasını bekler.
- **POST "/vote"** : Mobil uygulamadan gönderilir, ilgili haberi beğenip beğenmediğini belirtir. Kullanıcı kimlik bilgisi ve oy bekler.
- **GET "/view/:slug"** : Haberi döndürür ve görüntüleme sayısını bir artırır. Haber kimlik numarasını bekler.
- **POST "/image"** : Sunucuya bir resim yükler ve resmin kimliğini döndürür. Resim bayt dizgisi bekler.
- **GET "/image/:imageid"** : Belirtilen kimlikteki resmi döndürür.
- **GET "/list"** : Tüm haberlerin listesini döndürür.
- **GET "/list/:categoryid"** : Belirtilen kategorideki haberlerin listesini döndürür.

3.1.4 Dosya Yapıları ve Fonksiyonlar

Uygulamada kullanılan fonksiyonlar aşağıda dosyalarıyla birlikte listelenmiştir.

tick-lite/containers/list.js

Mobil uygulamanın ana sayfa bölümüdür. Sistemdeki haberler burada listelenir.

- **getItems()** : Mevcut filtreye göre sunucudan haber listesini çeker.
- **handleBackPress(e)** : Ana sayfa güncelleme kontrolcüsüdür. Geri tuşuna basıldığında istenen etkinliği engeller ve anasayfayı günceller.
- **componentWillMount()** : Uygulama açılmadan önce **getItems()** fonksiyonunu çağırır ve listeyi hazırlar.
- **componentDidMount()** : Uygulamaya etkinlik dinleyicilerini atar. Böylece sayfa geçişlerinde geri tuşunun işlevi değiştirilebilir.
- **componentWillUnmount()** : Atanan etkinlik dinleyicileri görevden alınır.
- **onReceived()** : Yeni bir haber eklendiğinde çağırılır. **getItems()** fonksiyonunu çağırır.
- **onOpened(openResult)** : Yeni bir haber eklendiğinde çıkan bildirime tıklandığında çağırılan fonksiyondur. Habere yönlendirir.
- **newsList({item})** : Haber listesini işler ve anasayfaya ekler.

tick-lite/containers/post.js

Seçilmiş haberin içerik sayfasıdır.

- **getNew(slug)** : Tıklanılan haberin bilgilerini sunucudan alır.
- **vote(diff)** : Habere oy verir. Diff parametresi, pozitif veya negatif oy verdiğini belirtir.
- **componentdidMount()** : Haberi açan cihazın kimliğini alır ve **getNew(slug)** fonksiyonunu çağırır.

tick-server/database/db.json

Kullanıcı veritabanı dosyasıdır.

tick-server/public/index.html

Sunucu giriş ekranını bulundurur. "/login" endpointine POST isteği atar.

tick-server/public/admin/index.html

Sunucu admin ekranını bulundurur. Haberler ile ilgili tüm işlemler bu sayfada gerçekleşir.

tick-server/src/

Endpoint modüllerini bulunduran klasör.

- **admin.js** : "/admin/add" ve "/admin/remove" endpointlerinin çağırıldığı fonksiyonların bulunduğu modül.
- **category.js** : "/list/:categoryid" endpointinin çağırıldığı fonksiyonların bulunduğu modül.
- **error.js** : Hata durumunda çağırılan fonksiyonların bulunduğu modül.
- **image.js** : "/image" ve "/image/:imageid" endpointlerinin çağırıldığı fonksiyonların bulunduğu modül.
- **index.js** : Tüm modülleri birbirine bağlayan modül.
- **list.js** : "/list" endpointinin çağırıldığı fonksiyonların bulunduğu modül.
- **login.js** : "/login" endpointinin çağırıldığı fonksiyonların bulunduğu modül.
- **view.js** : "/view/:slug" endpointinin çağırıldığı fonksiyonların bulunduğu modül.
- **vote.js** : "/vote" endpointinin çağırıldığı fonksiyonların bulunduğu modül.

3.1.5 Karşılaşılan Sorunlar ve Çözümleri

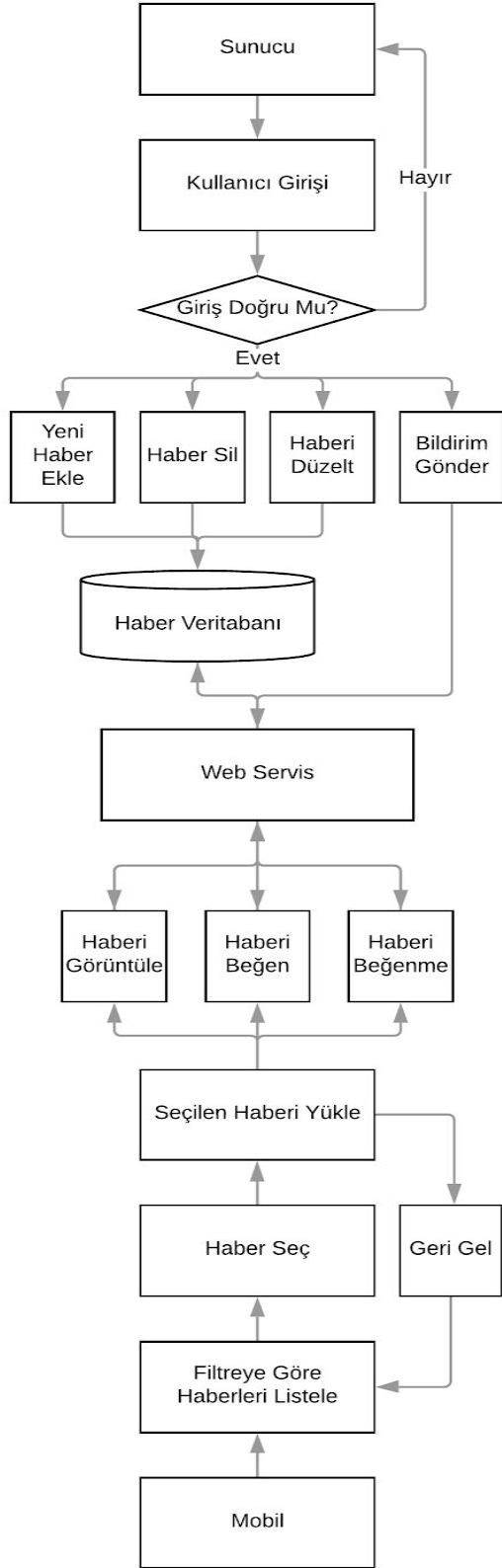
[1] NoSQL Veritabanı İçin Oy Listesi

Kullanıcıların oylarının haber listesinden bağımsız olması, ancak oy sayısının haber ile beraber mobil uygulamaya yollanması gerekiyordu. Kullanıcıları da haber listesinde tutmayı düşünsek de işlem yükü fazla bir yöntemdi. Bunu çözmek adına oy verirken "votes" tablosundaki veriler kontrol edildikten sonra haberin oy sayısını güncelledik. Bu sayede tüm oyların sayılması yerine ilgili oyun artmasını sağladık. Bu geliştirim işlemci ve disk yükünü azalttı.

[2] Kullanıcı oylarının birden fazla sayılması

Bu sorunu çözmek için başta aklımıza iki yöntem geldi. Ya kullanıcının, mobil uygulamaya giriş yapmasını sağlayacaktık, böylece kullanıcıya ait bir token belirlenecek ve kullanıcı bu token ile vote işlemini gerçekleştirecekti; ya da tokensiz bir şekilde beğen/beğenme butonlarını yapacaktık, ki bu uygulama tekrar başlatıldığında aynı habere birden fazla oy verebilmek demektir. Bu iki yöntem de isteklerimizi karşılamadı, çünkü kullanıcıya sürtünmesiz ve stabil bir deneyim sağlamak istiyorduk. Böylece yeni bir yöntem düşündük ve kullanıcının bir etkileşimi olmadan, cihaz kimlik numarası ile işlem yapmaya karar verdik. Böylece kullanıcı uygulamayı kaldırıp tekrar yüklese dahi, oy verdiği haberler belli olacak ve bu haberlere tekrar oy veremeyecekti.

4. Akış Şeması



(Şekil 6 : Akış Şeması)

5. Kazanımlar

5.1 Mobil Uygulama Tecrübesi

Yaptığımız ilk mobil uygulamaydı. Sonucunda mobil uygulamaların, diğer yaptığımız uygulamalardan farkını ve gereksinimlerini anladık. Yeni bir işletim sistemlerinde geliştirme yapmanın hazzını yaşadık.

5.2 React-Native

Hibrit uygulama nasıl geliştirilir, native uygulama ile JavaScript arasında nasıl bridge kurulur, gradle neye yarar, bunları öğrendik.

5.3 Emülatör Kullanımı

İlk defa Android SDK ve AVD kullandık. Bir Android emülatörü nasıl kurulur, nasıl çalıştırılır ve limitasyonları nelerdir, bunları öğrendik.

5.4 Backend Tecrübesi & Web Servis

Sunucu tarafında bir port dinleyen uygulama geliştirdik ve gelen isteklere göre istenen işlemleri yapan bir uygulama geliştirdik. POST, GET ve benzeri türlerdeki HTTP istek yapısını öğrendik. Request ve Response tiplerini öğrendik.

6. Kaynakça

react-native-device-info - GitHub

[1]<https://github.com/react-native-community/react-native-device-info>

Erişim Tarihi: 23.04.2019

Understanding React Native Component Lifecycle Api - The Journal Blog

[2]<https://blog.usejournal.com/understanding-react-native-component-lifecycle-api-d78e06870c6d>

Erişim Tarihi: 23.04.2019

Android Asset Studio - Launcher icon generator

[3]<http://romannurik.github.io/AndroidAssetStudio/icons-launcher.html>

Erişim Tarihi: 23.04.2019

React Native SDK Setup - OneSignal

[4]<https://documentation.onesignal.com/docs/react-native-sdk-setup>

Erişim Tarihi: 23.04.2019

Remove top Navigation Bar - Stack Overflow

[5]<https://stackoverflow.com/questions/46065819/remove-top-navigation-bar-for-certain-screens>

Erişim Tarihi: 23.04.2019

React Native Örnek Login Uygulaması #3 - Ajax kullanımı, İstekler ve Uyarılar - Avare Kodcu
[6]<http://avarekodcu.com/konu/17/react-native-ornek-login-uygulamasi-3-ajax-kullanimi-istekler-ve-uyarilar>
Erişim Tarihi: 23.04.2019

React Native: How to Setup Your First App - Hacker Noon
[7]<https://hackernoon.com/react-native-how-to-setup-your-first-app-a36c450a8a2f>
Erişim Tarihi: 23.04.2019

Draw horizontal rule in React Native - Stack Overflow
[8]<https://stackoverflow.com/questions/43380260/draw-horizontal-rule-in-react-native>
Erişim Tarihi: 23.04.2019

How can you float: right in React Native? - Stack Overflow
[9]<https://stackoverflow.com/questions/32030050/how-can-you-float-right-in-react-native>
Erişim Tarihi: 23.04.2019

How to add custom fonts in React-Native Project - YouTube
[10]<https://www.youtube.com/watch?v=22LEiBYBiTw>
Erişim Tarihi: 23.04.2019

Scroll to top of a FlatList - GitHub
[11]<https://github.com/facebook/react-native/issues/13560>
Erişim Tarihi: 23.04.2019

React-Native Networking - Facebook GitHub
[12]<http://facebook.github.io/react-native/docs/network>
Erişim Tarihi: 23.04.2019

React Native Get Unique ID of Device - About React
[13]<https://aboutreact.com/react-native-get-unique-id-of-device>
Erişim Tarihi: 23.04.2019

Add a custom font to your React Native app - Bam
[14]<https://blog.bam.tech/developper-news/add-a-custom-font-to-your-react-native-app>
Erişim Tarihi: 23.04.2019

How can I generate an apk that can run without server with react-native? - Stack Overflow
[15]<https://stackoverflow.com/questions/35935060/how-can-i-generate-an-apk-that-can-run-without-server-with-react-native>
Erişim Tarihi: 23.04.2019

Build and Install unsigned apk on device without the development server? - Stack Overflow
[16]<https://stackoverflow.com/questions/35283959/build-and-install-unsigned-apk-on-device-without-the-development-server>
Erişim Tarihi: 23.04.2019

Gradient CSS Generator - CSSMatic
[17]<https://www.cssmatic.com/gradient-generator>
Erişim Tarihi: 23.04.2019

Prevent Line/Paragraph Breaks in contentEditable - Stack Overflow
[18]<https://stackoverflow.com/questions/425274/prevent-line-paragraph-breaks-in-contenteditable>
Erişim Tarihi: 23.04.2019

Window.confirm() - Mozilla Developers
[19]<https://developer.mozilla.org/en-US/docs/Web/API/Window/confirm>
Erişim Tarihi: 23.04.2019

The Definitive Guide to DateTime Manipulation - Toptal
[20]<https://www.toptal.com/software/definitive-guide-to-datetime-manipulation>
Erişim Tarihi: 23.04.2019

Express-Fileupload - GitHub
[21]<https://github.com/richardgirges/express-fileupload/tree/master/example#basic-file-upload>
Erişim Tarihi: 23.04.2019

Express - GitHub
[22]<https://github.com/expressjs/express>
Erişim Tarihi: 23.04.2019

Create Notification & Send to Specific Devices - OneSignal
[23]<https://documentation.onesignal.com/reference#create-notification>

Upload a image using Ajax and FormData - CoderWall
[24]<https://coderwall.com/p/cjinag/upload-a-image-using-ajax-and-formdata>
Erişim Tarihi: 23.04.2019

Convert ISO date to milliseconds in JavaScript - Stack Overflow
[25]<https://stackoverflow.com/questions/9229213/convert-iso-date-to-milliseconds-in-javascript>
Erişim Tarihi: 23.04.2019