



TEKNOFEST

HAVACILIK, UZAY VE TEKNOLOJİ FESTİVALİ

TURİZM TEKNOLOJİLERİ YARIŞMASI

PROJE DETAY RAPORU

Takım İsmi: GöksuTech

Proje Adı: TravellIt

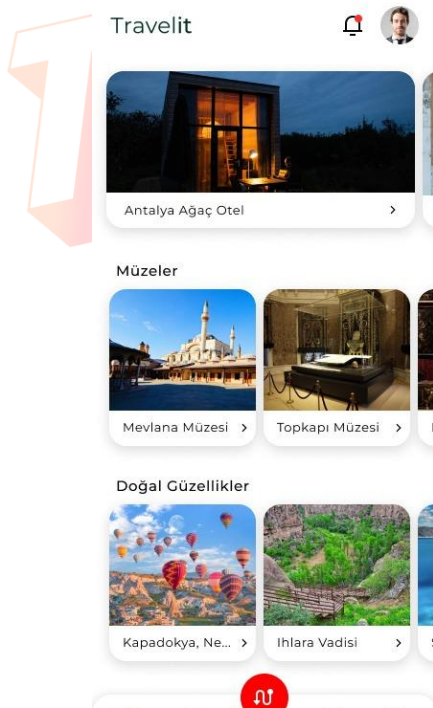
Başvuru Id: 317075

İçindekiler

Proje Ekibi/Proje Özeti	2
Algoritma ve Tasarım	3
Sistem Mimarisi	4
Hayata Geçirilebilirlik/Uygulanabilirlik	5
Yenilikçilik/Özgünlük	5
Bir Soruna/İhtiyaca Çözüm Üretmesi	6
SWOT Analizi	7
Kaynakça	8

1. Proje Ekibi/Proje Özeti

Projede yapay zeka ve arttırılmış gerçeklik geliştiricisi olarak Muhammed Talha Semerci, mobil uygulama ve UI/UX geliştirici olarak Emre Özcan, Backend geliştiricisi olarak ise Berkay Kulak yer almaktadır. Projenin asıl amacı turizme yeni bakış açıları getirerek turistleri daha çok gezmeye teşvik etme ve bunu olabildiğince efektif yollarla gerçekleştirebilmektir. Projede kullanılan temel teknolojiler ise yapay zeka, mikroservis mimarisi, Azure bulut ortamları, ArKit ve Flutter ile geliştirilmektedir. Kullanılan mimariler için en güncel yapılar kullanılmaktadır. Projenin tanımı en basit şekilde; **ülkemizi gezmek isteyen turistlere ilgi alanları ve isteklerine göre, bir gezi yol haritası çizmektir. Kullanıcı uygulamaya giriş yapar ve rota oluşturma özelliği ile ilk başta ilgili olduğu alanları seçer. Sonrasında gitmek istediği illeri seçerek işlemine devam eder. Bu adımdan sonra seçmiş olduğu tür ve illere göre uygun turistik yerler listelenir. Buradan kullanıcı istediği yerleri manuel olarak ta ekleyebilir veya uygulama için geliştirilen yaş, tür ve şehir parametrelerine uygun**



Şekil 1 Ana Ekran

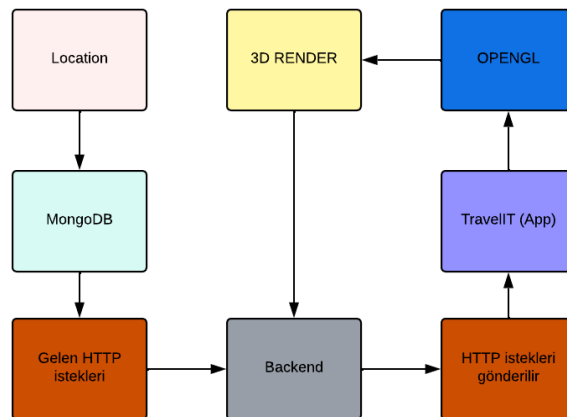


Şekil 2 Başlangıç Konumu

olarak otomatik gezme haritası da oluşturur. Son adımda ise ilk gitmek istediği ili seçerek yol haritası en ideal şekilde çizilir.

2. Algoritma ve Tasarım

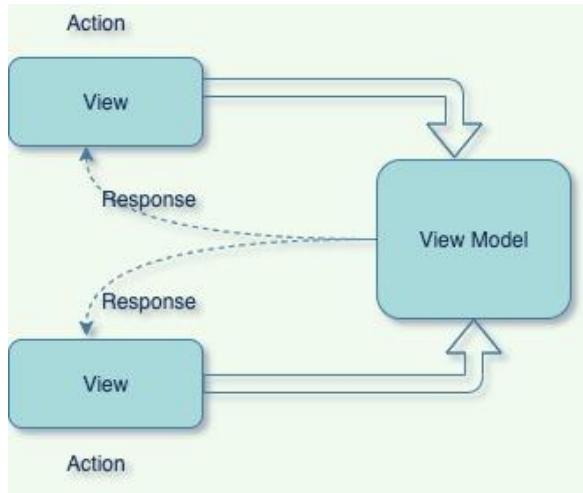
Bu projenin arka plan tarafında kullanılacak algoritmalar sıralama, çoklu filtreleme, düzenleme, sayfalama algoritmaları temel olarak tasarlanmıştır ve entegrasyonu yapılmıştır. Veritabanı ile etkileşimi sağlamak için ORM tabanlı Entity Framework çatısı altında geliştirmeler sağlanmıştır. Yapılan geliştirmeler ile veritabanı tabloları Code First yaklaşımı ile ele alınmaktadır. Mikroservis mimarisi konseptinde ilerleyen, docker kullanımı ve çoklu containerlar ile PostgreSQL, MongoDB, MSSQL veritabanları ile uygulama geliştirilmiştir. NoSQL tarafında daha çok kullanıcı konumları ve bilgileri tutulurken, ilişkisel veritabanı tabloları Kullanıcılar, Admin, Roller, Claims, Adres tabloları üzerine yoğunlaşmıştır ve tablo ilişkileri ortaya konulmuştur. RabbitMQ aracılığı ile Python, NodeJS, .Net dilleri birbiriyle etkileşimli olarak haberleşmektedir.[10] Python tarafında kullanıcı bilgilerini işleyerek daha iyi önerilerde bulunan yapay zeka algoritması ile CNN, RCNN, Yapay Sinir Ağları kullanılarak doğruluk oranı yüksek tahminlerde bulunulmaktadır.[4] Proje algoritması şu sırayla gerçekleşmektedir. Kullanıcı Giriş yapar ve konumu belirlenir. Kullanıcı ilgili olduğu türleri işaretler. Sonrasında gideceği şehirleri seçer. Seçilen şehre göre lokantalar, müzeler, tarihi yerler vb. listelenir. Listelenen yerlerden birisi seçilerek bütçe durumu girilir ve il haritasına göre rota oluşturulur. Oluşturulan rota veri yapıları algoritmaları sayesinde en kısa yolu bulma yöntemi ile çizilir ve yüksek kullanıcı deneyimi sağlanır. Kullanıcı uygulamayı kullandıkça rolü yükselir ve belirlenen müzelere girişlerinde indirimler yapılır. Projede kullanılan yöntemlerin başında yapay zeka ve veri yapıları algoritmaları gelmektedir. Yapay zekanın tahmini algoritmasında Keras kütüphanesinden faydalanılması planlanmaktadır. [5] Yapılan testlerde tahmin doğruluk oranları %85.2 civarında gerçekleşmektedir. Mobil uygulamadan kullanıcıya gitmesi gereken en iyi yolu veren algoritmada ise Deep First Search ve Breadth First Search algoritmaları esas alınmıştır. C# servislerinin iş katmanında gerçekleştirilen olayda veritabanına düşen uzunluklar hesaba katılır ve ülke düzleminde arama yapmaya başlanır. Node.js ile üretilecek(produce) olan Restful API içerisinde JWT yapısı ile Authorization yapılır. Giriş yapılırken validation kütüphanesi ile parametreler kontrol edilir. Kullanıcı her giriş yaptığında SharedPreferences içerisinde saklanan JWT ile sistem içerisindeki işlemlere ulaşabilir ve kullanabilir. Bu yöntem ile aynı zamanda kullanıcının rolünü kontrol edilir ve ona göre puanlandırma mekanizması çalıştırılır.



Şekil 3 Uygulamanın ana işleyiş şeması

3. Sistem Mimarisi

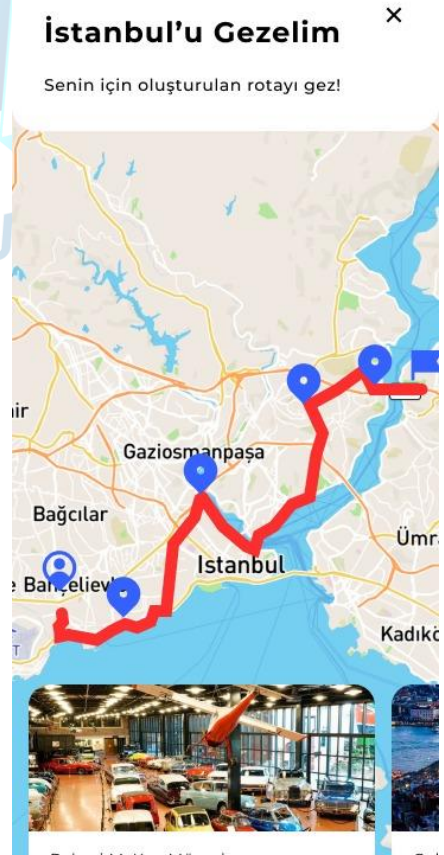
Projenin .Net Core backend ana mimarisi N-Tier mimari olarak geçmektedir. Kullanılan katmanlardan Veri erişim (Data Access) katmanı ile veritabanı etkileşimi sağlanır, Repositoryler ile database katmanında CRUD operasyonları yapılır. SOLID prensiplerine uymak için ise Generic Repository deseni (pattern) ve Unit Of Work deseni kullanılarak transaction yapısı sağlanır.[13] Entity (Varlık) katmanında Veritabanı tabloları ve sütunları code first yapısına uygun olarak tasarlanır., Business (İş) katmanı Veri erişim katmanı ile bağlantı kurar ve Dependencjy Injection



Şekil 4 Flutter Uygulama Mimarisi

aracılığı ile gevşek bağlılık sağlanır. En son katmanda ise UI katmanında da Mobil tarafına API endpointleri sayesinde hem iOS hem de Android için HTTP protokolleri sağlanır.[14] Kullanıcı resimleri Azure Blob Storage servislerinde tutulur ve projenin canlıya alınması Azure servisleri sayesinde yapılır. Mikroservisler aracılığı ile projenin ödeme, raporlama, loglama ve sisteme kayıt etme gibi faktörleri farklı servisler aracılığı ile gerçekleştirilir. [7] Bu servisler gerçekleştirilirken API Gateway aracılığı ile servis haberleşmeleri sağlanır. Tüm bunlar gerçekleştirilirken dillerin birbiri ile haberleşmesi RabbitMQ ve Apache Kafka ile sağlanmaktadır. Python gelen HTTP isteklerini okuyarak kendisi yorumlar ve tahmin çıktıları üretir. Python tarafında Django ile veri işleme ve görselleştirme teknikleri uygulanmaktadır. Son çıktı Mobil uygulamaya API endpointi ile iletilir. CI/CD uygulama geliştirme aşamalarına otomasyon dahil edilerek uygulama sık sık müşterilerden geri dönüşler ile güncellenmektedir. DevOps çözümleri ile yüksek bir hızda hareket ederek turistler için daha hızlı yenilik yapılabilir. Değişen pazarlara daha iyi uyum sağlanması ve projenin gidişata yönelik hızlanması sağlanmıştır. Node.js uygulamasında Nest.js kütüphanesinin sunduğu 3-Tier Mimarisi kullanılmaktadır. 3-Tier temel olarak 3 ana başlığa sahiptir. Presentation Tier, Application Tier, Data Tier. Presentation Tier kullanıcı ile uygulama ile etkileşime girdiği, uygulamanın kullanıcı arayüzü ve iletişim katmanıdır. Application Tier da Presentation Tier'da toplanan bilgiler (bazen Data Tier'deki diğer bilgilere karşı) iş mantığı, belirli bir iş kuralları kümesi kullanılarak işlenir. Uygulama katmanı ayrıca veri katmanındaki verileri ekleyebilir, silebilir veya değiştirebilir. Data Tier, bazen veri erişim katmanı veya arka uç olarak adlandırılan, uygulama tarafından işlenen bilgilerin depolandığı ve yönetildiği yerdir. Flutter ile geliştirilen uygulama sayesinde Dart diliyle yazılan kodla hem Android işletim sisteminde hemde iOS işletim sisteminde performanslı bir mobil uygulama çıkartılmıştır.[15] Projenin kullanıcı ile etkileşimde olan yeri mobil uygulama olacağından, arayüz tarafındaki kullanıcı

aracılığı ile gevşek bağlılık sağlanır. En son katmanda ise UI katmanında da Mobil tarafına API endpointleri sayesinde hem iOS hem de Android için HTTP protokolleri sağlanır.[14] Kullanıcı resimleri Azure Blob Storage servislerinde tutulur ve projenin canlıya alınması Azure servisleri sayesinde yapılır. Mikroservisler aracılığı ile projenin ödeme, raporlama, loglama ve sisteme kayıt etme gibi faktörleri farklı servisler aracılığı ile gerçekleştirilir. [7] Bu servisler gerçekleştirilirken API Gateway aracılığı ile servis haberleşmeleri sağlanır. Tüm bunlar gerçekleştirilirken dillerin birbiri ile haberleşmesi RabbitMQ ve Apache Kafka ile sağlanmaktadır. Python gelen HTTP



Şekil 5 İl Rota Oluşturma

deneyimi çok önemlidir. Bunun için asenkron servis isteği fonksiyonları geliştirilmiştir. Kullanıcı kolay bir şekilde gezilecek yerlere ulaşabilecek ve bunlar hakkında detaylı bilgi elde edecektir. Rota Belirleme özelliği ile öncelikle kullanıcı gezmek istediği türleri ve illeri işaretleyecek. Bu veriler servise gönderilip işlenip uygun olan turistik yerler servis tarafından döndürülecektir. Kullanıcı bu yerleri görüp detaylı bilgilerine ulaşp, ilgili yerin 3D arttırılmış gerçeklik imajı Flutter ArKit ile telefon kamerası aracılığıyla gösterilecektir.[11] Bu sayede turist gezmek istediği yerle ilgili detaylı bilgilere sahip olacaktır. Ayrıca kendisi bu listeyi oluşturmak istemeyen kullanıcı otomatik seç özelliğini kullanabilecektir. Bu özellik kullanıcının yaşını geldiği ülkeyi ve türleri ele alarak otomatik bir seyahat listesi oluşturacaktır. Sonraki adımda ise listeye uygun özel il rota hesaplaması oluşturulup kullanıcıya sunulacaktır.

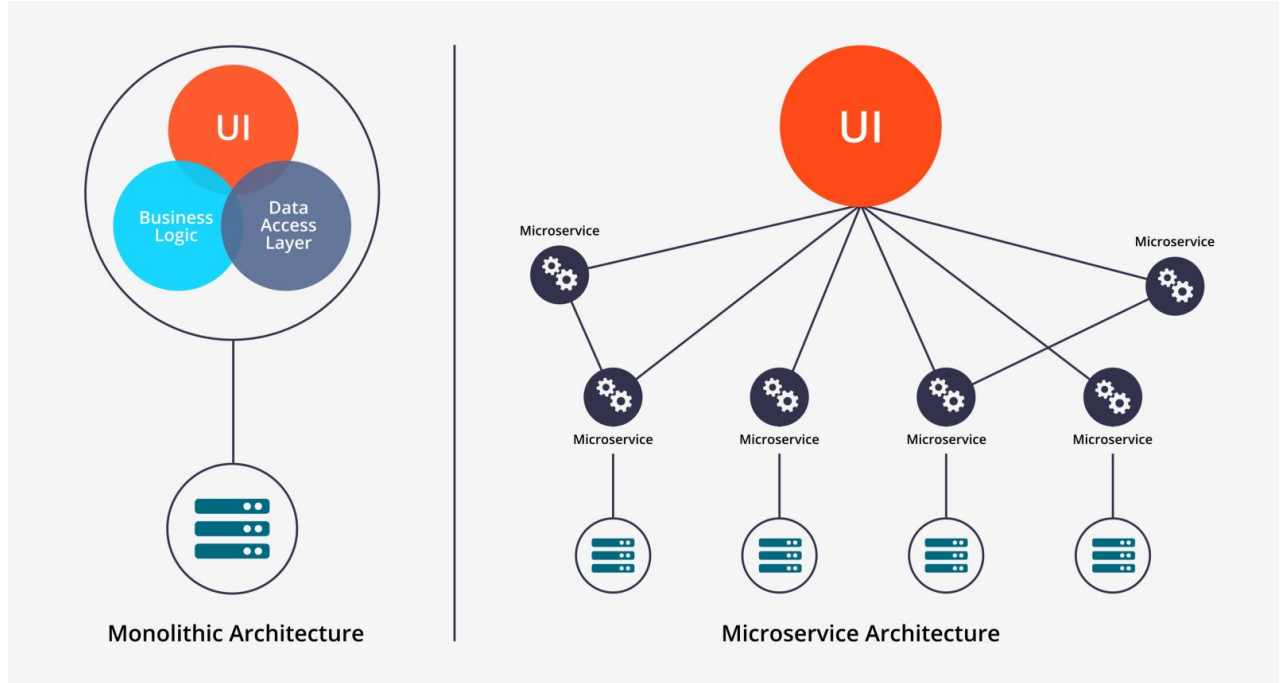
4. Hayata Geçirilebilirlik/Uygulanabilirlik

Uygulamanın ana prensiplerinden olan rota oluşturma mekanizması mobil arayüz tasarımı ile iyileştirilmiştir. Mobildeki kullanıcı butonları ile daha iyi sonuçlar alındığı backend loglamaları ile database üzerinden Kibana servisi ile izlenmiş olup daha iyi sonuçlar verebilmek için görsel iyileştirilmelere gidilmiştir. Bunun sonucunda %32 oranında daha çok kullanıcı uygulamayı indirmiş ve kullanmaya başlamıştır. Rota oluşturma ekranında kullanıcıların daha iyi görseli anlayabilmesi için Google haritalar kütüphanesi kullanılmıştır. Ve bu sayede daha efektif çizimler elde edilerek gerçekçi sonuçlar elde edilmiştir. Uygulamanın çok istek aldığı durumlarda bazı mobil servisler kontrol edilmiştir ve kullanıcıyı etkileyecek herhangi uzun süren bir yavaşlığın meydana gelmediği gözlemlenmiştir. Docker ortamda her bir container imajeleri tek tek kontrol edilerek bütünlüğü sağlanmaktadır ve bu durumda stabiliteyi %12 oranında arttırdığı gözlemlenmiştir. Flutter ArKit kütüphanesi ile turistik yerlere özgün olan 3 boyutlu imaj görselleri kullanıcıya cep telefonu kamerası aracılığıyla gösterilecektir. Bu imaj görselleri ilgili olan turistik yere uygun olarak servis tarafından döndürülecektir. Bu sayede imaj dosyaları uygulama içinde tutulmayarak hafıza tarafından verim elde edilmiştir.

5. Yenilikçilik/Özgünlük

TravellIt projesinin en yenilikçi yönleri 3D render, arttırılmış gerçeklik ve Yapay Zeka tahminleri ile kullanıcıya özgü ve UI dostu bir uygulama ve ayrıca turistlere atanan roller ile belirli indirimlerin sağlanması ve uygulamanın daha çok kullanılması ve geniş kitlelere yayılması yenilikçi yönlerindendir. [12] Projenin yenilikçi yönlerinden bir başkası da mikroservis mimarisi kullanılarak geliştirilmiş ölçeklenebilirlik, daha esnek uygulamalar için daha iyi hata izolasyonu, daha iyi veri güvenliği, uyumluluk, DevOps için daha fazla iş çevikliği ve desteği sağlanacaktır.[8] Tüm bunların yanı sıra uygulamanın performans girdileri de oldukça iyi çıkmıştır. Tüm HTTP isteklerini çok daha hızlı gerçekleştirmek için Elastic Search altyapısı kullanılacaktır. Bütün bu durumlar pozitif yönde değerlendirilebilir. Çoklu filtreleme özellikleri ile karmaşık yapıdan daha basit yapıya da dönüştürülmesi amaçlanmıştır ve uygulanmıştır. Yapay zeka tahminleri ile birçok işlemi yapay zeka yapacaktır. Rota hesaplama ve güvenliği turizm açısından oldukça etkili olması ve bu tarz yöntemlerin çok gelişmiş olmaması uygulamanın ana çıkış amaçlarından bir tanesidir. Turistlerin 3

boyutlu görseller ile daha gitmeden mobil ekranda gösterilmesi ise gezmek istediği yeri seçmesinde oldukça etkili olmaktadır. Yapay zeka tahmin algoritmaları ile de kullanıcılar için çözüm önerilerinde bulunması gerekse de işlemlerin manuel bir şekilde gerçekleştirilmesi hedeflenmektedir. [1]



Şekil 6 Monolithic ve Mikroservis Tasarımı



6. Bir Soruna/İhtiyaca Çözüm Üretmesi

Projenin Türkiye için önemi oldukça büyüktür. Turislerin ülkemize geldiğinde nasıl efektif ve uygun maliyetlerde gezeceğini tam bilemediği saptanmıştır. Gitmek istediği yerleri bilse bile nasıl ve hangi koşullar altında gideceği, o yolun en efektif olup olmama durumu tam olarak kestirilmemektedir. Ayrıca plansız gezmeler ile tatili verimsiz geçirdikleri gözlemlenmiştir. Tüm bunlara çözüm olarak ise TravellIt uygulaması hayata geçirilmiştir. Turizm ülkesi olan Türkiye'nin doğal güzelliklerinin daha da efektif bir şekilde seyahat edilmesi amaçlanmıştır. Kullanıcılara daha uygun yol haritaları çizilerek gitmesi gereken yerler ve yapay zeka tahminleri ile bütçe dostu yerlerin tahmini ile kullanıcı deneyimi iyileştirilmiştir. [2] Turistler uygulamayı kullandıkça kendilerine ait bazı özellik geliştirmeleri ve rol atamalar ile belirli düzeylerde seyahat indirimleri kazanacaklardır. Kazandıkları indirimler sayesinde hem uygulamayı daha çok kullanacak, hem daha çok gezecek hem de çeşitli indirimler ile giderlerini daha efektif yönetebileceklerdir. İlk kullanıcıdan son kullanıcıya kadar UI dostu ile kullanımı oldukça kolaydır ve turistler bu konuda zorlanmayacaklardır. Ayrıca projede artırılmış gerçeklik ile sağlanan 3D renderlar ile kullanıcıların daha doğru kararlar vermesi sağlanacaktır. Turislere gidebilecekleri yerlere daha hızlı ve efektif bir şekilde ulaşmasını sağlayacak bir uygulama geliştirilmiştir. Ülke turizmine katkıda bulunacağına elde edilen istatistikler ile katkı sağlanacağı görülmektedir.

Rotanı Belirleyelim

Gezmek istediğin yerlere göre listeleme yaptık. Hoşuna giden yerleri ekleyebilir, veya bize bırakabilirsin.



İstanbul

Topkapı Sarayı

Rahmi M. Koç



Samsun

Bandırma Gemi-Müze ve Millî Mücadele Açık Hava Müzesi

Gazi Müzesi

Antalya

Devam Et

Otomatik Seç

Rotanı Belirleyelim

Gezmek İstedğin Yerleri Seç

Modern

Tarihi

Müze

Doğa

Dini

Yemek

Devam Et

Rotanı Belirleyelim

Gezmek İstedğin Şehirleri Seç

Otomatik Seç

İstanbul

Ankara

İzmir

Antalya

Aydın

Diyarbakır

Gaziantep

Nevşehir

Bolu

Hatay

Mardin

Samsun

Van

Zonguldak

Isparta

Konya

Burdur

Bursa

Devam Et

7. SWOT Analizi

Projenin güçlü yönleri olarak Yapay zeka, Mikroservis mimarisi, 3D render, Arttırılmış gerçeklik, Raporlama, Çoklu filtreleme, Sıralama, Sayfalama, UI desteği, iOS ve Android çoklu platform desteği, Raporlama, Kullanıcılara özgü yapılan indirimler ve deneyimi arttırmayı hedefleyen algoritmalar projenin avantajı olarak gösterilebilir. Projenin zayıf yönü olarak ise Azur servislerine verilen ücret tutarının kullanıcıların çokluğuna göre fazla olma ihtimali ve yeterli geri dönüt alamaması halinde değişik uygulama yaptırımlarına gidilebilmesi.

Projenin fırsatları ise uygulamanın Turizm sektöründe faaliyet gösteren kuruluşların siteme kayıt olarak kendi otellerini, lokanta ve restoranlarını ve birçok müze , doğal ve tarihi yerlerin gösterimi ile çeşitli iş imkanlarının doğabilmesine ve bütçe dostu olması ile projeyi çekici kılmaktadır.

Yapılacak iyileştirmelerden bir tanesi ise rotaların daha düzgün hesaplanabilmesi için veri yapıları algoritmalarının iyileştirilmesi, yapay zekanın gerekli veriler ile beslenmesi ve tüm bunların yanında kullanıcıların uygulamayı daha da sade bir şekilde kullanabilmesi hedeflenmektedir.[3] Projenin fazla debug edilmesiyle beraber doğan sorunlar unit testler yazılarak giderilmiştir.

Projenin riskleri arasında mikroservis mimarilerin birbiriyle uyum içerisinde çalışmaması, belirli deployment süreçlerinde gecikmeler gösterilebilir. Fazla kullanıcı girdilerinde servislerin stres testlerinde problemler çıkabilecektir. [9]

Rota Hesaplandı

Gezmek istediğin yer ve şehirlere göre rotan hesaplandı.

İstanbul Tamamlandı

Bir Sonraki Rota Samsun

Hemen Başla!

Şimdi Başla!

Tamam

Ana Sayfa

Ana Sayfa

Rotaların

- ☒ İstanbul
- ☒ Samsun
- ☐ Bolu Şimdi Başla!
- ☐ Isparta
- ☐ Antalya
- ☐ Diyarbakır

8. Kaynakça

1. Winston, P. H. (1992). *Artificial intelligence*. Addison-Wesley Longman Publishing Co., Inc..
2. Tussyadiah, I. (2020). A review of research into automation in tourism: Launching the Annals of Tourism Research Curated Collection on Artificial Intelligence and Robotics in Tourism. *Annals of Tourism Research*, 81, 102883.
3. Zsarnoczky, M. (2017). How does artificial intelligence affect the tourism industry?. *VADYBA*, 31(2), 85-90.
4. Samara, D., Magnisalis, I., & Peristeras, V. (2020). Artificial intelligence and big data in tourism: a systematic literature review. *Journal of Hospitality and Tourism Technology*.
5. Tsaih, R. H., & Hsu, C. C. (2018). Artificial intelligence in smart tourism: A conceptual framework. *Artificial Intelligence*, 2.

6. Kazak, A. N., Chetyrbok, P. V., & Oleinikov, N. N. (2020). Artificial intelligence in the tourism sphere. In *IOP Conference Series: Earth and Environmental Science* (Vol. 421, No. 4, p. 042020). IOP Publishing.
7. Alshuqayran, N., Ali, N., & Evans, R. (2016, November). A systematic mapping study in microservice architecture. In *2016 IEEE 9th international conference on service-oriented computing and applications (SOCA)* (pp. 44-51). IEEE.
8. Nadareishvili, I., Mitra, R., McLarty, M., & Amundsen, M. (2016). *Microservice architecture: aligning principles, practices, and culture*. " O'Reilly Media, Inc."
9. Pulli, K., Aarnio, T., Miettinen, V., Roimela, K., & Vaarala, J. (2007). *Mobile 3D graphics: with OpenGL ES and M3G*. Elsevier.
10. Dossot, D. (2014). *RabbitMQ essentials*. Packt Publishing Ltd.
11. Chen, Y., Wang, Q., Chen, H., Song, X., Tang, H., & Tian, M. (2019, June). An overview of augmented reality technology. In *Journal of Physics: Conference Series* (Vol. 1237, No. 2, p. 022082). IOP Publishing.
12. Zlatanova, S. (2002). Augmented reality technology. *GIST Report No. 17, Delft, 2002, 72 p.*
13. Sammon, J. W. (1969). A nonlinear mapping for data structure analysis. *IEEE Transactions on computers*, 100(5), 401-409.
14. Goadrich, M. H., & Rogers, M. P. (2011, March). Smart smartphone development: iOS versus Android. In *Proceedings of the 42nd ACM technical symposium on Computer science education* (pp. 607-612).
15. Napoli, M. L. (2019). *Beginning Flutter: A Hands On Guide To App Development*. John Wiley & Sons.