

İÇİNDEKİLER

| | |
|--|-----------|
| 1. GİRİŞ | 2 |
| 1.1. PROJE HAKKINDA..... | 2 |
| 1.2. PROJE YAPIM SÜRECİNDE TESLİM EDİLECEK DÖKÜMANLAR..... | 2 |
| 2. PROJE ORGANİZASYONU | 2 |
| 2.1. YAZILIM SÜREÇ MODELİ..... | 2 |
| 2.2. YETKİ VE SORUMLULUKLAR..... | 3 |
| 2.3. KULLANILACAK ARAÇ VE TEKNİKLER | 3 |
| 3. PROJE YÖNETİM PLANI..... | 3 |
| 3.1. GÖREVLER..... | 3 |
| 3.1.1. <i>Object Detection Modeli</i> | 4 |
| 3.1.1.1. Görevin tanımı..... | 4 |
| 3.1.1.2. Görev basamakları..... | 5 |
| 3.1.1.3. Kaynaklar | 5 |
| 3.1.1.4. Bağımlılık ve kısıtlamalar | 5 |
| 3.1.1.5. Risklerin tanımlanması..... | 5 |
| 3.1.2. <i>Depth Estimation Modeli</i> | 6 |
| 3.1.2.1. Görevin tanımı..... | 6 |
| 3.1.2.2. Görev basamakları..... | 6 |
| 3.1.2.3. Kaynaklar | 6 |
| 3.1.2.4. Bağımlılık ve kısıtlamalar | 6 |
| 3.1.2.5. Risklerin tanımlanması..... | 7 |
| 3.1.3. <i>Mobil Arayüz</i> | 8 |
| 3.1.3.1. Görevin tanımı..... | 8 |
| 3.1.3.2. Görev basamakları..... | 8 |
| 3.1.3.3. Kaynaklar | 8 |
| 3.1.3.4. Bağımlılık ve kısıtlamalar | 8 |
| 3.1.3.5. Risklerin tanımlanması..... | 9 |
| 3.2. GÖREVLENDİRME | 10 |
| 3.3. ZAMAN ÇİZELGESİ..... | 10 |
| 4. EK MATERYALLER | 11 |

1. GİRİŞ

1.1.Proje Hakkında

BM-314 Yazılım Mühendisliği dönem projesi kapsamında görme engelli bireylerin hayatını kolaylaştırmaya yönelik Görme Engelliler için Nesne Tespiti uygulaması yapılması planlanmıştır. Projede amaç; görme engelli bireylerin etraflarındaki objelerden, bunların yakınlıklarından ve bunların ne olduğundan haberdar olup çevresel farkındalıklarının artarak günlük hayatlarını kolaylaştırmaktır.

Oluşturulacak yazılım dört kısımdan oluşacaktır ve bunların hepsi birbirini destekler şekilde çalışacaktır. İlk olarak nesne tespiti yapay zekâ modeli etrafta bir nesne olup olmadığı ve varsa bu nesnenin nerede ve ne olduğunu tespit edecektir. Bundan sonra bir derinlik tahmini modeli bulunan nesnenin uzaklığını tahmin edecektir. Son olarak bulunan nesne, nesnenin ne tarafta olduğu ve ne kadar uzakta olduğu sesli raporla görme engelli bireye haber verilecektir.

Sonuç olarak hazırlanacak uygulama görme engelli bireylerin etrafındaki nesneleri fark etmelerini sağlayarak günlük hayata dahil olmalarını kolaylaştıracaktır. Bu dokümantasyonda projenin belli başlı özellikleri, süreç modeli ve ekip elemanlarının görevlendirilmesi konuları işlenmiştir.

1.2.Proje Yapım Sürecinde Teslim Edilecek Dokümanlar

Proje kapsamında teslim edilmesi hedeflenen belge ve diğer unsurlar aşağıdadır:

- Software Project Management Plan (SPMP)
- Software Requirements Specification (SRS)
- Software Design Description (SDD)
- Software Test Documentation (STD)
- Otel Yönetim Sistemi
- Sunum Dosyası

2. PROJE ORGANİZASYONU

2.1. Yazılım Süreç Modeli

BM-314 Yazılım Mühendisliği dersi kapsamındaki projede yazılım süreç modellerinden Scrum Modeli'nin kullanılmasına karar verildi. Bu süreç modeli, sık ve

hızlı toplantılar yapılarak o anki gereklilikler için iş bölümlerinin rahatça yapılması ve belirsiz durumlarda hızlı aksiyon alınması avantajlarına sahiptir. Buradaki seçimde rol oynayan en önemli madde yapay zekâ modellerindeki belirsizliklerdi. Bir model için gereken veri setini bulamamamız, istediğimiz performansta çalışacak bir model oluşturamamamız veya çalışma sürelerinin çok uzun olması durumunda planın değişmesi gerekeceğinden bu durumlar yaşanırsa hızlı aksiyon alma gerekliliği bizi bu modeli seçmeye yönlendirdi. Bunların yanında Scrum modelinin ekip üyeleriyle sürekli iletişim gerektiren yapısı 2 kişi olmamız ve sürekli iletişim halinde olmamız sayesinde ulaşılabilir. Bunlar göz önünde bulundurulduğunda Scrum modelinin kullanılmasına karar verildi.

2.2. Yetki ve Sorumluluklar

Proje kapsamında Furkan Egecan Nizam ve Dilay Ece Maral, yazılımın optimum çözümlere ulaşmasını yazılımın optimum çözümlere ulaşmasını hedefleyerek, zamanında teslim edilmesini amaçlamaktadır. Proje ekibinde hiyerarşik bir yapı söz konusu değildir; dolayısıyla bir proje lideri belirlenmemiştir. Ancak sorumluluklar ekip üyeleri arasında uzmanlık alanlarına göre paylaşılmıştır. Bu kapsamda, Furkan Egecan Nizam veri bilimi ile ilgili çalışmalardan, Dilay Ece Maral ise projenin geri kalan yazılım ve geliştirme süreçlerinden sorumlu olacaktır.

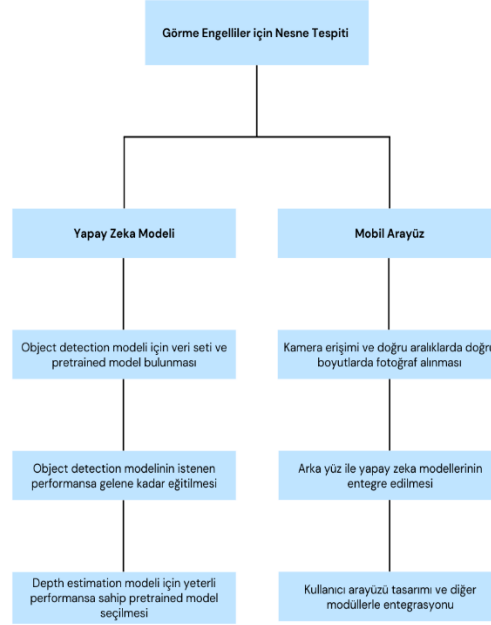
2.3. Kullanılacak Araç ve Teknikler

İlgili projenin yapay zekâ modellerinin geliştirilmesi yüksek hesaplama gücü gerekliliği sebebiyle Google Colab üstünde gerçekleştirilecektir. Bunun yanında uygulamanın mobil backend ve frontend tasarımı Swift üzerinden IOS platformu için yapılacaktır. Proje geliştirme sürecinde Git ve GitHub ile sürüm kontrolü sağlanacaktır.

3. PROJE YÖNETİM PLANI

3.1. Görevler

Görevlerin tanımlanması işlemi projedeki belirsizliklerin fazla olması ve seçilen yazılım geliştirme modelinin de bunu destekleyen yapıda olması sebebiyle görevlerin belirsizlikleriyle birlikte tanımlanmasına ihtiyaç duymuştur. Projede ihtiyaç duyulacak modüller şekil 3.1.1’de görüntülenmiştir.



Şekil 3.1.1. Gereksinimler Şeması

Gereksinimlerin belirlenmesinin ardından yapılması gereken bu gereksinimlerin nasıl gerçekleştirileceği konusundaki belirlemeleri yapmaktır. Burada yapay zekâ modellerinin geliştirilmesi ile mobil arayüzün geliştirilmesi işlemleri iki farklı kişide olduğundan dolayı paralel olarak yürütülecekti. Bunun ilk sebebi, ne kadar içleri belli olmasa da hem yapay zekâ modelinin hem mobil arayüzün girdileri ve çıktıları belliydi. Burada ilk olarak object detection modeli fotoğraflar alacak; bu fotoğraflarda bir nesne var mı, varsa nerede ve bu nesne ne sorularına cevap verecek. Bunun için gerekli girdi olan fotoğraflar yani canlı video da mobil arayüz kullanılarak telefonun kamerasından sağlanacak. Bundan sonra modelden alınan çıktı depth estimation modeline verilecek, bu model de tespit edilen bir nesne varsa bu nesnenin ne kadar uzaklıkta olduğunu tahmin edecek. Bu bilgi de elde olduğunda mobil arayüze verilecek, mobil arayüz de bu bilgileri sesli bir şekilde kullanıcıya iletecek.

3.1.1. Object Detection Modeli

3.1.1.1. Görevin tanımı

Object detection modeli projenin belkemiği sayılabilecek bir kısımdır. Model kullanıcının kamerasından sürekli olarak fotoğraf alacak, bunları analiz edecek,

içinde nesne olup olmadığı, nerede olduğu ve ne olduğu bilgilerini döndürecektir. Bu bilgiler projenin vaat ettiklerinin genelini kapsadığından çokça dikkatli olunmalıdır. Bunun yanında sürekli çalışma ve mobilde çalışma gereklilikleri de hafif bir model olması gerekliliğini ortaya çıkarmaktadır.

3.1.1.2. Görev basamakları

- Model için gerekli nesneleri içeren veri seti seçilmesi
- Hafif ve yüksek doğruluklu pretrained model seçilmesi
- Seçilen modelin fine-tune edilmesi

3.1.1.3. Kaynaklar

3.1.1.3.1. Teknik kaynaklar

- Kaggle
- Google Colab
- Hugging Face

3.1.1.4. Bağımlılık ve kısıtlamalar

Object detection modeli projenin mihenk taşıdır. Projenin geri kalanı bu modelin performansının iyi olduğu göz önünde bulundurularak tasarlanacaktır. Bunun sebebi bundan sonra gelen depth estimation modelinin çalışmasının bu modele bağlı olması, sonrasında da kullanıcıya sesli olarak verilen çıktının bu iki modelin çıktılarına bağlı olmasıdır. Sayılan unsurlar projenin tüm bileşenlerinin object detection modeline bağımlı olmasına sebep olmaktadır. Bu model üç adet kısıtlamaya tabidir. Bunlardan ilki, günlük hayatta karşılaşılan nesnelerin genelini kapsayan bir veri seti bulma zorluğudur. İkinci olarak bahsedilen veri setinin boyutunun yüksek olması muhtemel olması sebebiyle pretrained bir model kullanılsa bile fine-tune işleminin uzun sürebilecek olması ve bunun çok sayıda deneme yapmayı zorlaştırmasıdır. Sonuncu olarak ise yüksek performansla çalışması gereken bu modelin mobil cihazlarda çalışma gerekliliği düşük hesaplama gücünden dolayı doğruluğunu etkilemektedir.

3.1.1.5. Risklerin tanımlanması

Bağımlılıklar ve kısıtlamalar başlığı altında da belirtildiği gibi object detection modeli projenin temelini oluşturmaktadır. Bu yüzden model oluşturulurken karşılaşılabilecek hataları öngörerek buna uygun önlemler alınması karşılaşılan sorunlara karşı bir B planı daha olduğundan projenin akışının bozulmasını engellerken, yazılımın kalitesini de artırır. Bu başlık altında karşılaşılabilecek riskler ve çözümleri aşağıdadır:

Risk1. Gerekli veri setinin bulunamaması.

Çözüm1. Birden çok veri setindeki gerekli nesneleri içeren verilerin toplanarak özel bir veri seti oluşturulması.

Risk2. Modeli eğitmenin fazla uzun sürmesi.

Çözüm2. Kullanılan modelin mimarisinin değiştirilmesi, modelin eğitim mantığının değiştirilmesi veya veri setinden küçük örnekler alınarak eğitilmesi.

Risk3. Modelin hesaplama gücü gerekliliğinin yüksek olması.

Çözüm3. Modelin Tensorflow Lite modele dönüştürülmesi veya 8 bite dönüştürme gibi yöntemlerle küçültülmesi.

3.1.2. Depth Estimation Modeli

3.1.2.1. Görevin tanımı

Depth estimation modeli adından da anlaşılabilceği gibi derinlik tespiti yapacaktır. Proje özelinde bu işlem, object detection modelinin tespit ettiği nesnelerin kullanıcıya ne kadar uzaklıkta olduğunu tespit etmek için kullanılacaktır. Object detection modeli bir nesne tespit ettiği durumda model çalışacak, girdi olarak object detection modeline verilmiş olan fotoğrafın depth estimation modelinin girdi boyutlarına göre değiştirilmiş halini alacak, çıktı olarak ise tespit edilen nesnenin uzaklığını verecektir.

3.1.2.2. Görev basamakları

- Pretrained bir depth estimation modeli seçilmesi
- Seçilen modelde performans testleri yapılması
- Gerekli olduğu takdirde modelin fine-tune edilmesi

3.1.2.3. Kaynaklar

3.1.2.3.1. Teknik kaynaklar

- Google Colab
- Hugging Face
- Kaggle

3.1.2.4. Bağımlılık ve kısıtlamalar

Bir yerden bir yere giderken karşımıza çıkan nesnelere nasıl tepki vereceğimizi sadece

karşımıza çıkan nesnenin ne olduğu ve ne tarafta olduğu etkilemez, buradaki en önemli unsurlardan biri de nesnenin uzaklığıdır. Yakındaki nesne için ani ve keskin bir tepki vermemiz gerekir. Bu projede görme engelli bireye sağlanacak uzaklık bilgisi bu modelin sorumluluğundadır. Modelin en önemli bağımlılığı ve kısıtlandığı unsur, kendisinden önce çalışacak olan object detection modelinin çıktısıdır. Çünkü model, object detection modelinin nesne olarak tanımladığı yerin uzaklığını kullanıcıya bildirecektir. Projenin bu modele bağımlılığı ise kullanıcıya verilecek çıktıdaki oldukça önemli bir noktanın bu modele bağımlı olmasıdır. Modelin kısıtlarından bir diğeri ise düşük hesaplama gücüne sahip mobil cihazlarda yüksek performansla çalışması gerekliliğidir.

3.1.2.5. Risklerin tanımlanması

Modelin barındırdığı risklerin en belirginini daha başlangıçta object detection modeline bağımlı olmasından gelir. Bunun yanında bu model de bir yapay zekâ modeli olduğundan dolayı object detection modelinde görülebilecek hatalar burada da görülebilir.

Risk1.Object detection modelinin çıktısının yanlış olması.

Çözüm1. Bu basamağa geçilmeden object detection modelinin performansından emin olunması

Risk2. Modelin çalışmasının fazla uzun sürmesi.

Çözüm2. Modelin Tensorflow Lite modele ve 8 bit'e dönüştürülmesi gibi performans arttırmaları yapılması ya da yapay zekâ yerine matematiksel hesaplama temelli bir yaklaşıma geçilmesi.

Risk3. Modellerden istenen sonuç alınamaması.

Çözüm3. Projedeki olası girdilere yakın veri setleri bulup modellerin fine-tune edilmesi.

3.1.3. Mobil Arayüz

3.1.3.1. Görevin tanımı

Burada ilk olarak mobil cihazın kamerasından belli aralıklarda görüntü alınmalı, bunlar ilk model olan object detection modelinin kabul ettiği formata dönüştürülmeli, sonrasında object detection modeline verilmeli. Bundan sonra object detection modelinin sonuçları kontrol edilmeli, eğer bir nesne bulunduyorsa görüntü bu sefer depth estimation modelinin kabul ettiği formata dönüştürülmeli, bundan sonra depth estimation modeline verilmeli. Bu iki modelin çıktıları alınıp kullanıcıya verilecek şekilde formatlanmalı, sonrasında kullanıcıya sesli olarak bu bilgiler iletilmeli.

3.1.3.2. Görev basamakları

- Mobil cihazın kamerasından belli aralıklarda görüntü alınması
- Alınan görüntülerin object detection modeline uygun formatlanması
- Object detection modelinin çıktısının yorumlanması
- Görüntünün depth estimation modeline uygun formatlanması
- Çıktıların alınıp kullanıcıya verilecek şekilde formatlanması
- Bilgilerin kullanıcıya sesli şekilde iletilmesi

3.1.3.3. Kaynaklar

3.1.3.3.1. Teknik kaynaklar

- Swift
- AVFoundation (kamera erişimi ve görüntü işleme için)
- Core ML (AI modellerinin entegrasyonu için)
- Text-to-Speech (Sesli geri bildirim için)
- Xcode (Geliştirme ortamı)

3.1.3.3.2. Yazılı kaynaklar

- IEEE Standarts
- Apple Developer Dokümantasyonu

3.1.3.4. Bağımlılık ve kısıtlamalar

Modellerin girdilerinin kaynağı olan kamera erişimi ve görüntülerin alınması, bu

görüntülerin modellerin girdi boyutlarına uygun şekilde formatlanması, modellerin birbirine bağlanması, bu sırada yapılması gereken kontrol, modellerin çıktılarının formatlanması ve kullanıcıya sesli olarak bildirilmesi bu arayüzünün sorumluluğundadır. Görülebileceği gibi projenin tamamı bu arayüze bağımlı olarak çalışmaktadır, bu yüzden projenin iskeleti diyebileceğimiz kısım burasıdır. Bunun yanında modellerin performansına direkt etkisi yoktur, bu da arayüzün performansını modellerin performansı ile kısıtlar.

3.1.3.5. Risklerin tanımlanması

Sistemdeki her şeyi birbirine bağlayan mobil arayüz; kamera erişimi, görüntü işleme, model çıktılarının formatlanması ve sesli geri bildirim gibi tüm işlemlerin doğru sırada ve hatasız gerçekleşmesini sağlamalıdır. Arayüzde meydana gelen bir hata, görüntülerin işlenememesi, model çıktılarının yanlış yorumlanması veya sesli geri bildirim gecikmesi gibi sorunlara yol açarak sistemin yanlış veya gereksiz çalışmasına sebep olabilir.

Risk1. Kamera erişim problemleri nedeniyle görüntü alınamaması.

Çözüm1. Kamera izinlerinin doğru yapılandırıldığından ve cihaz uyumluluğundan emin olunmalı. AVFoundation yapılandırması test edilerek olası erişim sorunları giderilmeli.

Risk2. Alınan görüntülerin object detection ve depth estimation modellerine uygun formata dönüştürülebilmesi.

Çözüm2. Format dönüşüm işlemleri için önceden test yapılmalı ve format uyumsuzlukları için hata yönetimi mekanizmaları oluşturulmalı.

Risk3. Model çıktı formatlarının kullanıcıya uygun şekilde formatlanamaması.

Çözüm3. Çıktı formatları için standart bir yapı belirlenmeli ve format dönüşüm işlemi otomatikleştirilmeli.

Risk4. Text-to-Speech ile sesli geri bildirim sırasında gecikme veya yanlış seslendirme olması.

Çözüm4. Text-to-Speech yapılandırması test edilmeli ve gecikmeler için tamponlama (buffering) mekanizması kullanılmalı.

Risk6. İşlem sırasında cihazın aşırı ısınması veya pilin hızlı tükenmesi.

Çözüm6. İşlem yükü optimize edilmeli, arka planda gereksiz işlem yükü oluşturan kodlar kaldırılmalı.

Risk7. Swift, AVFoundation veya Core ML sürüm güncellemeleri sonrası uyumsuzluk yaşanması.

Çözüm7. Sürüm güncellemelerinden önce test ortamında kontrol yapılmalı ve uyumluluk sağlanmadan güncelleme yapılmamalı.

3.2. Görevlendirme

Projenin veri bilimi kısmından Furkan Egecan Nizam, mobil arayüz geliştirme kısmından ise Dilay Ece Maral sorumludur.

3.3. Zaman Çizelgesi

Scrum yazılım geliştirme modelinin doğası gereği kesin bir zaman çizelgesi vermek doğru olmayacak olsa da tahmini bir zaman çizelgesi Tablo 3.3.1’de verilmiştir.

| Tarih | Görev | Sorumlu Eleman |
|-------------------------|---|---------------------------------------|
| 21.03.2025 / 28.03.2025 | Object detection modeli için gerekli veri seti ve pretrained modele karar verilmesi | Furkan Egecan Nizam |
| 18.03.2025 / 21.03.2025 | Kamera erişiminin çözülmesi | Dilay Ece Maral |
| 28.03.2025 / 04.04.2025 | Object detection modelinin gerekli veri setiyle fine-tune edilmesi | Furkan Egecan Nizam |
| 28.03.2025 / 04.04.2025 | Sesli çıktı verme özelliğinin çözülmesi | Dilay Ece Maral |
| 04.04.2025 / 11.04.2025 | Depth estimation için kullanılacak pretrained modele karar verilmesi | Furkan Egecan Nizam |
| 04.04.2025 / 11.04.2025 | Kameradan fotoğraf alıp object detection modelindi çalıştırma testleri | Dilay Ece Maral |
| 16.04.2025 | Software Requirements Specification Teslimi | Furkan Egecan Nizam / Dilay Ece Maral |

| | | |
|----------------------------|---|--|
| 11.04.2025 / 18.04.2025 | Object detection modeli için geri bildirimler ışığında performans iyileştirmeleri | Furkan Egecan Nizam |
| 11.04.2025 / 18.04.2025 | Object detection modeli ile depth estimation modeli arasında köprünün kurulması | Dilay Ece Maral |
| 18.04.2025 / 25.04.2025 | Depth estimation modeli için geri bildirimler ışığında performans iyileştirmeleri | Furkan Egecan Nizam |
| 18.04.2025 / 25.04.2025 | Alınan çıktıların formatlanıp kullanıcıya sesli şekilde iletilmesi | Dilay Ece Maral |
| 25.04.2025 / 02.05.2025 | Tam sistem testleri | Furkan Egecan Nizam / Dilay Ece Maral |
| 30.04.2025 | Software Design Description Teslimi | Furkan Egecan Nizam / Dilay Ece Maral |
| 02.05.2025 / 09.05.2025 | Testler ışığında performans iyileştirmeleri | Furkan Egecan Nizam / Dilay Ece Maral |
| 09.05.2025 / 14.05.2025 | Son sistem testleri | Furkan Egecan Nizam / Dilay Ece Maral |
| 14.05.2024 | Software Test Documentation Teslimi | Furkan Egecan Nizam / Dilay Ece Maral |
| ~ | Proje Sunumu | Furkan Egecan Nizam / Dilay Ece Maral |

Tablo 3.3.1. Zaman Çizelgesi

4. EK MATERYALLER

Bu başlık altında ek olarak verilmesi istenen bir materyal bulunmamaktadır.