T.C.
İSTANBUL ÜNİVERSİTESİ-CERRAHPAŞA
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

YÜKSEK LİSANS TEZİ

TELE-NÖROREHABİLİTASYON SİSTEMİNİN GELİŞTİRİLMESİ VE UYGULANMASI

Osman Şakir KAPAR

DANIŞMAN
Dr. Öğr. Üyesi Muhammed Erdem İSENKUL

II. DANIŞMAN
Doç. Dr. Yonca ZENGİNLER YAZGAN

Bilgisayar Mühendisliği Anabilim Dalı

Bilgisayar Mühendisliği Programı

Haziran, 2023

Bu tez kapsamında geliştirilen sistemi gerçekleştirme süreçlerinden karşılaştığım her türlü problemlerde desteğini esirgemeyen sevgili eşim Demet AKYOL KAPAR başta olmak üzere aileme, arkadaşlarıma ve danışman hocam Muhammed Erdem İSENKUL'a ithaf ediyorum…

TEŞEKKÜR

Tez çalışması süresince beni destekleyen sevgili eşim başta olmak üzere Bilgisayar Mühendisi Demet AKYOL KAPAR, değerli danışmanım ve hocam Dr. Öğr. Üyesi Muhammed Erdem İSENKUL ve Doç. Dr. Yonca ZENGİNLER YAZGAN ve çalışmada ortak olarak çalıştığım diğer paydaşlarımdan Prof. Dr. Ela TARAKCI, Doç. Dr. Burcu ERSÖZ HÜSEYİNSİNOĞLU, Prof. Dr. Murat KÜRTÜNCÜ, Uzm. Fzt. Pelin VURAL, Fzt. Elif YILMAZ, Fzt. Recep ÖRMEN'e teşekkürlerimi en içten şekilde takdir ediyorum…

Haziran 2023 Osman Şakir KAPAR

İÇİNDEKİLER

Sayfa No

TEŞEKKÜR ..... **iii**
İÇİNDEKİLER ..... **iv**
ŞEKİL LİSTESİ ..... **vi**
TABLO LİSTESİ ..... **viii**
SİMGE VE KISALTMA LİSTESİ ..... **ix**
ÖZET ..... **x**
ABSTRACT ..... **xii**
1. GİRİŞ **1**
2. KAVRAMSAL ÇERÇEVE **3**
2.1. Tele-rehabilitasyon Tanımı 3
2.1.1. Tele-rehabilitasyon Sınıflandırma 4
2.1.2. Tele-rehabilitasyonun Faydaları 5
2.2. Proje Kapsamında Öncelik Verilen Hastalıkların Tanımı 6
2.2.1. İnme Hastalığının Tanımı 6
2.2.2. Multipl Skleroz Hastalığının Tanımı 7
2.2.3. Parkinson Hastalığının Tanımı 7
2.3. TeleNöroRehab 8
2.3.1. TeleNöroRehab Tanımı 8
2.3.2. TeleNöroRehab Yazılımsal Çerçevesi 9
2.4. Literatür Özeti 11
3. YÖNTEM **15**
3.1. Araçlar ve Teknolojiler 15
3.1.1. Visual Studio (2019) 15
3.1.2. Visual Studio Code 15
3.1.3. SQL Server Management Studio 16
3.1.4. Git 16
3.1.5. Github Desktop 17
3.1.6. Edge Browser Development Tools 17

3.1.7. ReSharper 17
3.1.8. Postman 18
3.1.9. Swagger 18
3.1.10. IIS 18
3.1.11. Node.js 19
3.1.12. Grunt 19
3.1.13. TypeScript 19
3.1.14. Single Page Application 20
3.1.15. Angular 20
3.1.16. Katmanlı Soğan Mimarisi 20
3.1.17. .NET Core (.NET 5) 22
3.1.18. SQL 22
3.1.19. ORM 23
3.1.20. Entity Framework 23
3.1.21. HTTP 24
3.1.22. RESTful API 25
3.1.23. JSON 25
3.1.24. Web Soket 26
3.1.25. SignalR 26
3.1.26. Bearer Authentication 27
3.1.27. JWT Token 27
3.1.28. Bootstrap 29
3.1.29. ImageSharp 29
3.2. Entegrasyon ve Mimari 29
3.2.1. Kurumsal Web Sitesi 29
3.2.2. Kullanıcı Arayüzü Web Uygulaması 33
3.2.3. Sunucu Tarafında Çalışan Algoritmik Program (Web Servis) 43
3.2.4. Veri Tabanı Tasarımı 49
4. BULGULAR 50
5. TARTIŞMA 52
6. SONUÇ VE ÖNERİLER 53
KAYNAKLAR 56
EKLER 60

ŞEKİL LİSTESİ

Sayfa No

Şekil 2.1. TeleNöroRehab çalışma prensibi 9
Şekil 2.2. TeleNöroRehab web uygulamaları tanımı 10
Şekil 2.3. TeleNöroRehab yazılımsal akış gösterimi 10
Şekil 3.1. Katmanlı Soğan Mimarisi katman tanımı 21
Şekil 3.2. Entity Framework çerçevesi için bir örnek .NET Core kodu ve SQL çıktısı..... 23
Şekil 3.3. RESTful çalışma prensibi..... 24
Şekil 3.4. Örnek bir JSON gösterimi..... 25
Şekil 3.5. SignalR çalışma prensibi..... 26
Şekil 3.6. Bearer Authentication kullanılan bir HTTP method Header örneği 27
Şekil 3.7. JWT token ve içerisinde yer alan veri gösterimi..... 28
Şekil 3.8. JWT çalışma akışı gösterimi 28
Şekil 3.9. Web site NPM paketleri için kullanılan package.json 30
Şekil 3.10. Örnek Grunt Bake kullanımı..... 31
Şekil 3.11. Web site proje dosyalama gösterimi 32
Şekil 3.12. Web sitesi için ekran görüntüleri..... 33
Şekil 3.13. TeleNöroRehab Angular projesi için dosyalama gösterimi 36
Şekil 3.14. Bir arayüz sayfasının çalışma prensibi..... 37
Şekil 3.15. Kullanıcı arayüzünde giriş ve anasayfa ekran görüntüleri 38
Şekil 3.16. Egzersiz programı görüntüleme sayfası için ekran görüntüleri..... 39
Şekil 3.17. Mesajlaşma arayüzü için ekran görüntüsü 40
Şekil 3.18. Kategori yönetimi sayfası için ekran görüntüleri 41
Şekil 3.19. Egzersiz Yönetimi sayfası için ekran görüntüleri 41

Şekil 3.20. Sistemde yer alan egzersiz videolarının hazırlanma aşamaları 42
Şekil 3.21. Program yönetimi sayfası için ekran görüntüleri 43
Şekil 3.22. İstek işleme akışı için çalışma prensibi 44
Şekil 3.23. Web Servis Katman mimarisi 45
Şekil 3.24. Web Servis iş mantığı katmanı için dosyalama gösterimi 48

TABLO LİSTESİ

Sayfa No

Tablo 1. TeleNöroRehab genel sayısal verileri 50

SİMGE VE KISALTMA LİSTESİ

Kısaltmalar	Açıklama
TR	: Tele-rehabilitasyon
Fzt.	: Fizyoterapist
MS	: Multipl Skleroz
SPA	: Single Page Application
HTTP	: Hyper-Text Transfer Protocol
IP	: İnternet Protocol
SP	: Serebral Palsi
IDE	: Integrated Development Environment
SSMS	: SQL Server Management Studio
DBMS	: Database Management System
SQL	: Stuctured Query Language
GUI	: Graphical User Interface
NPM	: Node Package Manager
SPA	: Single Page Application
IIS	: Internet Information Services
HTML	: Hyper-Text Markup Language
CSS	: Cascading Style Sheets
ORM	: Object-Relational Mapping
TCP	: Transmission Control Protocol
JSON	: JavaScript Object Nation
JWT	: JSON Web Token

ÖZET

YÜKSEK LİSANS TEZİ

TELE-NÖROREHABİLİTASYON SİSTEMİNİN GELİŞTİRİLMESİ VE UYGULANMASI

Osman Şakir KAPAR

İstanbul Üniversitesi-Cerrahpaşa

Lisansüstü Eğitim Enstitüsü

Bilgisayar Mühendisliği Anabilim Dalı

Bilgisayar Mühendisliği Programı

Danışman: Dr. Öğr. Üyesi Muhammed Erdem İSENKUL
II. Danışman: Doç. Dr. Yonca ZENGİNLER YAZGAN

Günümüz dünyasında nörolojik hastalıklara bağlı hastalıklar giderek artmaktadır. Nörolojik hastalıklara ve bunların yol açtığı semptomlara bağlı olarak gelişen rahatsızlıklar günlük yaşamları olumsuz etkilemektedir. Fizyoterapi ve rehabilitasyon tedavileri ile hastaların semptomları kontrol altına alınarak günlük yaşamlarında fiziksel aktivite fonksiyonları artırarak bireylere fiziksel kazanımlar sağlanmaktadır. Fizyoterapi ve rehabilitasyon tedaviler ile hastaların kişisel yaşam kalitesi ve sosyal hayata katılımları artmakta böylece fiziksel, sosyal ve psikolojik yönlerden desteklenmiş olmaktadır. Günümüzde hızla gelişen teknoloji ile rehabilitasyon uygulamaları da değişmekte ve gelişmektedir. Son yıllarda yapılan çalışmalarda tele-rehabilitasyon sistemleri ile yapılan egzersiz uygulamalarının hastaların tedaviye uyumunu ve katılımını arttırdığı ve standart rehabilitasyon uygulamaları kadar etkin olduğu bildirilmektedir. Nörolojik hastalıklara sahip bireylerin maddi ve manevi sorunları nedeni ile hastaneye ortamına ulaşıp alamadıkları tedavileri tele-rehabilitasyon sistemi sayesinde onlara ulaştırmak, ihtiyaç duydukları rehabilitasyon hizmetini almalarına kolaylık

sağlamaktadır. Bu sistemlerin yurtdışı uygulamalarında birçok farklı örneklerine rastlanmaktadır ancak ülkemizde bu hasta gruplarına özgü, Türkçe altyapı desteği sunan tele-rehabilitasyon sistemi örnekleri ve erişebilirliği oldukça sınırlıdır. Bu tez kapsamında geliştirilen sistemin amacı kişisel ve çevresel sebepler ile fizyoterapi ve rehabilitasyon hizmetlerine erişim zorluğu yaşayan kronik nörolojik hastalıklara sahip bireylerin değerlendirme, takip ve ihtiyaçlarına yönelik planlanan egzersizleri hastane ortamı dışında evlerinde kolayca uygulamalarına imkân sağlayan tele-rehabilitasyon sisteminin geliştirilmesi ve etkinliğinin incelenmesidir. Bu çalışmada geliştirilen sistemi kullanan her birey için kişisel demografik bilgiler ve bireyin geliştirilen sistemi kullanırken harcadığı süre, kullandığı içerik, katıldığı ve bitirdiği program gibi sistem özelinde gerçekleşen sayılar bilgiler ayrı ayrı toplanıp araştırmalar için veri bankası oluşturulmuştur. Geliştirilen sistem, bir nörolojik hastalığa sahip bireyin uzaktan çevrimiçi olarak tele-rehabilitasyon altyapısını kullanarak Türkçe olarak tedavi süreçlerini gerçekleştirmek amacı ile ileride yapılacak yerli uygulamalara basamak ve öncü olacağı düşünülmektedir.

Haziran 2023, 84 sayfa.

Anahtar kelimeler: Tele Sağlık, Tele-rehabilitasyon, Web Tabanlı Sistemler, Fizik Tedavi ve Rehabilitasyon

ABSTRACT

M.Sc. THESIS

DEVELOPMENT AND IMPLEMENTATION OF TELE-NEUROREHABILITATION SYSTEM

Osman Şakir KAPAR

Istanbul University-Cerrahpaşa

Institute of Graduate Studies

Department of Engineering Sciences

Program of Computer Engineering

Supervisor: Assist. Prof. Dr. Muhammed Erdem İSENKUL
Co-Supervisor: Assoc. Prof. Yonca ZENGÜNLER YAZGAN

In today's world, neurological diseases are on the rise. Disorders that develop due to neurological diseases and the symptoms caused by them negatively affect daily lives. Physiotherapy and rehabilitation treatments provide physical gains to individuals by controlling the symptoms of patients and increasing their physical activity functions in their daily lives. Physiotherapy and rehabilitation treatments increase patients' personal quality of life and their participation in social life, thus supporting them in physical, social and psychological aspects. Today, rehabilitation applications are changing and developing with the rapidly developing technology. In recent studies, it has been reported that exercise applications with tele-rehabilitation systems increase the compliance and participation of patients to treatment and are as effective as normal rehabilitation applications. Delivering the treatments that individuals with neurological diseases cannot reach and receive in the hospital environment due to financial and other problems through the tele-rehabilitation system makes it easier for them to receive the rehabilitation services they need. There are many different

examples of these systems in foreign applications, but in Turkey, the examples and accessibility of tele-rehabilitation systems that are specific to these patient groups and offer Turkish infrastructure support are very limited. The aim of the system developed within the scope of this thesis is to develop and detect the efficiency of a tele-rehabilitation system which is used to assess, follow and design home-based rehabilitation program among individuals with chronic neurological diseases who have difficulty in participating in physiotherapy and rehabilitation services because of personal or environmental reasons. In this study, personal demographic information for each individual using the developed system and system-specific numbers such as the time spent by the individual while using the developed system, the content used, the program attended and completed were collected separately and a data bank was created for research. It is thought that the developed system will be a step for future in Turkey and Turkish language applications with the aim of realizing the treatment processes of an individual with a neurological disease remotely using tele-rehabilitation infrastructure in Turkish.

June 2023, 84 pages.

Keywords: Telehealth, Telemedicine, Tele Rehabilitation, Web Based Systems, Physical Therapy and Rehabilitation,

1. GİRİŞ

21. yüzyılda nörolojik hastalıklar ve bunlara bağlı hastalıklar giderek artmaktadır. Hastalıkların semptomlarına ve özürlülük seviyesine etkileri neden ile nörolojik hastalığa sahip bir bireylerin düzenli olarak uzmanlar tarafından kontrol edilen bir fizyoterapi ve rehabilitasyon hizmeti almaları gereksinim haline gelmiştir (Melnick, 2016) ancak bu tedavilerin maliyetleri bireylere ve kişisel sağlık sigortası harcamalarında oldukça önemli bir yer tutmaktadır (Agostini, vd., 2015). Hastalar rehabilitasyon hizmetlerini, hastanelerde yatarak veya ayaktan, rehabilitasyon merkezlerinde ayaktan ya da evde bakım hizmetleriyle kendi evlerinde alabilmektedirler. Günümüzde gelişen teknoloji ile birlikte tele-rehabilitasyon hizmetleri de gelişmeye ve birçok açıdan yaygınlaşmaya başlamıştır. Hastaların rehabilitasyon hizmeti aldığı yere ulaşım zorlukları yaşaması, refakatçi bulamama, işten izin alamama, yoğun hastanelerde rehabilitasyon hizmetine başlayabilmek için uzun bekleme sürelerinin olması, sağlık kuruluşuna ulaşmak için harcanan zaman ve maliyet kaybı gibi birçok kişisel ve çevresel sorunlar nedeni ile tele-rehabilitasyon hizmetlerine ihtiyaç giderek artmaktadır. 2019 yılında ortaya çıkan ve bütün dünyayı sarsan COVID-19 pandemisi, kronik nörolojik hastalığa sahip bireylerin ihtiyaç duydukları rehabilitasyon hizmetine kolay bir şekilde ulaşamamalarına neden olmuş, hastaların fiziksel aktivite düzeyleri ve fiziksel becerileri etkilenmiştir (Fallon, 2020). Bu olumsuz durum sonrasında da hastaların fizyoterapi ve rehabilitasyon hizmetlerine erişimlerinin düzenli olmasını sağlayabilmek ve semptomlarını kontrol altına alabilmek amacıyla tele-rehabilitasyon sistemlerine duyulan ihtiyaç daha da artmıştır.

Tez kapsamında geliştirilen sistem çevrimiçi olarak uzaktan nörolojik hastalıklara sahip bireylerin tedavi aşamalarında teknoloji ve internet destekli bir alt yapı sunarak tele-rehabilitasyon uygulaması geliştirmek ve tedavi sürecine doğrudan destek sağlanmaktır. Geliştirilen sistem dinamik olarak bir tele-rehabilitasyon sistemi olarak tasarlansa da ana olarak toplumda en sık görülen (Licher, vd., 2019) İnme, Multipl Skleroz ve Parkinson hastalığına sahip bireylere öncelikli hitap etmektedir.

Kavramsal Çerçeve bölümünde, tez kapsamında geliştirilen tele-rehabilitasyon sisteminin genel tanımı, öncelik verilen hastalıkların tanımı, geliştirilen projenin tanımı ve bu konuda bugüne kadar yapılmış ve geliştirilmiş olan çalışmaları ve farkları anlatılmaktadır.

Yöntem bölümünde, geliştirilen sistem için kullanılan teknolojik alt yapılardan, yazılımlardan, veri toplama ve işleme araçlarından, sistemi kullanan katılımcı çalışma grubundan bahsedilmektedir.

Bulgular bölümünde, geliştirilen sistem sonucunda çalışmaya katılan İnme, Multipl Skleroz ve Parkinson hastalığına sahip katılımcıların sayısal veriler ile elde edilen istatiksel bulgular incelenmiştir.

Son olarak Tartışma ve Sonuç bölümünde, geliştirilen sistem ile elde edilen sonuçların katılımcılara sağladığı durumlardan bahsedilmiş olup ilerideki geliştirilmesi hedeflenen çalışmalar hakkında bilgi verilmektedir.

2. KAVRAMSAL ÇERÇEVE

2.1. Tele-rehabilitasyon Tanımı

Günümüzde gelişen teknoloji ile beraber ortaya çıkan gelişmeler, tüm sektörlerde olduğu gibi sağlık alanında da çarpıcı değişikliklere neden olmuştur. İnternet teknolojisinin gelişimi ve alt yapısının artması ile beraber sağlık hizmetlerinin daha uzak mesafelere bu teknolojileri kullanarak yaygınlaşması ortaya atılmıştır ve böylece günümüzde sıkça duyduğumuz "teletıp" teknolojisi gelişmeğe başlamıştır.

"Tele" kelimesi Yunanca kökenli olup, uzaktan anlamına gelmektedir (Yıldırım, 2021). Tele-tıp, sağlık hizmetlerinin elektronik ortamda internet ve iletişim araçları ile uzak mesafelere taşınmasıdır. Tele-tıp terimi, günümüzde daha kapsayıcı bir ifade olan "tele-sağlık" ismini almıştır (Akıncı, vd., 2015). Tele-tıbbın dallarından biri olan tele-rehabilitasyon (TR), bireylerin karşılaştığı olumsuz hastalıklar karşısında tedavi almasını ve sağlıklarının geliştirilmesi için teknolojik araçlar kullanılarak fizyoterapistler tarafından rehabilitasyon hizmetlerinin uzaktan verilmesidir (Ceylan, 2021). TR uygulamaları hastaların değerlendirilmesi, tedavi edilmesi ve takip edilmesine olanak sağlamaktadır. Son yıllarda hızla gelişen teknoloji ve internet iletişim araçlarının kullanımın yaygınlaşması sayesinde mobil cihaz tabanlı sağlık hizmetleri ortaya çıkmış, yüz yüze tedavi yöntemlerine alternatif ve maliyeti düşük rehabilitasyon uygulamaları geliştirilmeye başlanmıştır (Divyani, vd., 2020 & Seidler, vd. 2017). TR, yüz yüze sağlık hizmetine ulaşamayan kişiler için etkili, uygulanabilir ve sürdürülebilir bir yöntemdir (Ceylan, 2021).

Tele-rehabilitasyon (TR), iletişim araçlarının ve telekomünikasyon teknolojisi kullanılarak rehabilitasyon hizmetlerine uzaktan erişim sağlanmasıdır. TR sağlık hizmetlerindeki maliyetleri azaltmak, erişilebilirliği artırmak ve bireylerin sahip olduğu hastalıkları karşısında bağımsızlığını korumak için umut verici bir yöntem olarak günümüz teknolojisi ile beraber ortaya çıkmış bir gelişimdir. TR, sağlayıcıların tedaviyi uzaktan yönetmesine olanak vererek yetersiz hizmet alanlarına erişimi potansiyel olarak artırmaktadır (Knepley vd., 2020). Nörolojik rehabilitasyonda TR hizmetleri, hastaların değerlendirilmesi, takibi ve ihtiyaçlarına yönelik oluşturulan rehabilitasyon programları ile uzun süreli tedaviye imkân sunarken, tedavi

maliyetinin azaltılmasına da ve kişilere maddi olarak bütçelerine katkı sağlayabilmektedir (Agostini, vd. 2015). Bunların haricinde, evde etkin ve devamlılığı olan rehabilitasyon, kronik hastalıkları olan bu kişilerin hastalık yönetimini günlük yaşamlarıyla birleştirmesine olanak tanımaktadır (Isernia, vd. 2019).

Ülkemizde gibi gelişmekte olan ülkeler arasında bölgenin hava, yol, iklim ve coğrafik şartlarının belirli zaman içerisinde ya da her zaman kötüleşmesi hastane ulaşımı olumsuz etkilemektedir. Bu bölgede yaşan bireylere hizmet verecek sağlık hizmetlerindeki yetersizliklerden dolayı kişiler gerekli tedaviyi zamanında ve yeterli şekilde alamamaktadır. Tele-rehabilitasyonun asıl ortaya çıkma fikri ve amacı bu tür durumlardaki hastaların rehabilitasyon ve tedavi hizmetlerinden yararlanmasını sağlamak amacı ile faydalı bir şekilde kullanılabilinir (Rosen, 1999 & Ulicny, vd. 1996).

2.1.1. Tele-rehabilitasyon Sınıflandırma

Tele sağlık ve tele-rehabilitasyon, bireylerin ihtiyaçlarına ve bölgedeki alt yapının yeterlilik durumuna göre değişik servislerde tedavi hizmetini sunulabilir. Kalyon (1999) tele-rehabilitasyon üzerine yapmış olduğu çalışmada şu şekilde sınıflara ayırmıştır.

2.1.1.1. Evde Tele-rehabilitasyon

İletişim araçları ile sağlanan rehabilitasyon hizmeti bireylere doğrudan kendi evlerinde sağlık hizmeti sunan rehabilitasyon yöntemidir. Bireylerin doktor veya terapiste evlerinde ulaşması olanaksızıdır. Bireylerin evlerinde bilgisayar, telefon, televizyon ve kamera gibi teknolojik iletişim araçlarına ulaşımı imkânı vardır. Bireylerin tedavilere katılma oranın en üst düzeyde olduğu hizmettir (Kalyon, 1999).

2.1.1.2. Evde Rehberli Tele-rehabilitasyon

Bu tip uygulamalar bireyler sağlık hizmetlerinin uzağında kişisel ortamlarında yalnız değillerdir. Bireyin yakınında hemşire veya terapist vardır. Tedavi alacak birey bir merkez ile elektronik araçlar ile bağlantıdadır. Bu yöntemde hastanın yaşadığı bölgelerde küçük sağlık hizmetlerinin bulunmasının yanı sıra büyük çaplı sağlık hizmetleri olmadığı için ve kişinin hastalık tanımının yapıldığı bölgeye uzak olmasından dolayı gerçekleşecek tedaviyi kişisel ortamında alması gerekmektedir.

2.1.1.3. Toplumsal Tele-rehabilitasyon

Günümüzde en çok kullanılan tele-rehabilitasyon sistemidir. (Kalyon, 1999). Bireylerin toplu bir şekilde belirli bölgelerde oluşturulan bir sağlık hizmeti merkezinde elektronik araçlar ile

uzman kişilere bağlantı kurmasıdır. Bölgesel ve toplu bir şekilde olduğu için evde alınan tele-rehabilitasyon hizmetine kıyasla daha az maliyetli bir seçenektir.

2.1.1.4. Toplumsal Rehberli Tele-rehabilitasyon

Bireylerin evde aldığı rehberli tele-rehabilitasyon sistemine benzer bir şekilde bireylerin bulunduğu bölgelerde hastane veya klinikte bir doktor veya terapist rehberliğinde tedavi hizmeti almasıdır. Uluslararası tedavi ve eğitim bağlantılarında bu seçenek çoğunluklar tercih edilir.

2.1.1.5. Telekonsültasyon

Bu yöntem bir hastanın olmadığı, doğrudan iki sağlık hizmeti arasında uzman tıp mensupları arasında yapılan bağlantıdır. Burada taraflardan biri hasta olan bireyi doğrudan takip eder ve hastaya ait tetiklerin ayrıntılı incelemelerini yapar. Diğer taraf ise uzaktan alınan veri ile yardımcı olur. Bu yöntem bir tedavi seçeneğinden ziyade uzaktan eğitim ve öğretim amaçlıda kullanılır.

Tez kapsamında geliştirilen proje öncelik olarak Evde Tele-rehabilitasyon sınıfı için geliştirilmiştir ancak geliştirilen sistemin dinamik yönünden dolayı Toplumsal Tele-rehabilitasyon sınıfı içinde uygulanabilir.

2.1.2. Tele-rehabilitasyonun Faydaları

Kalyon (1999) tarafından yapılan çalışmanın sonucunda tele-rehabilitasyon hizmetinin sağladığı olanakları birçok açıdan büyük faydalar getirdiği gözler önüne serilmiştir. Tele tıp bir diğer adı ile tele sağlık ve alt dallarından biri olan tele-rehabilitasyon hizmetlerinin faydaları ve işlevleri konusunda karşılaşılan sorunlar günümüz teknolojileri ile çoğu giderilmiştir. 21. Yüzyılda gelişen teknoloji ile beraber yaygınlaşan iletişim araçlarının sağladığı olanaklar ile beraber en uzak ve en ücra köşelere en kaliteli sağlık hizmetlerinin ulaştırılması ve bilgi aktarımı olanaklı kılınmıştır (Kalyon, 1999).

Tele-rehabilitasyon sisteminin sağladığı önemli faydaları şu şekildedir;

- Sağlık hizmetinin ulaşılması güç olan bölgelere ve hastalara tedavi hizmetinin verilmesi

- Hastane tedavi masraflarının azalması

- Tedavisi süren ve taburcu hastalar, belirli bölgeye bağlı kalmadan tedavi hizmetlerine devam etmesi
- Bireylerin durumlarına göre farklı şekilde tedavi edilme imkânı
- Tedavi devamlılığının kolay ve sürdürülebilir olması

2.2. Proje Kapsamında Öncelik Verilen Hastalıkların Tanımı

Tez kapsamında geliştirilen proje toplumda en sık görülen (Licher vd., 2019) İnme, Multipl Skleroz ve Parkinson hastalığına sahip bireylere öncelikli olarak hitap etmektedir.

2.2.1. İnme Hastalığının Tanımı

İnme hastalığı, beyin damarına giden kan akışının belirli nedenler ile aniden kesilmesine bağlı olarak gelişen, zihinsel ve fiziksel yeti kaybına neden olan serebrovasküler bir hastalıktır (Serpil, vd. 2012). İnme hastalığı, iki sınıf altında incelenir;

- İskemik İnme
- Hemorajik İnme

İnme hastalığının, günümüzde uzun dönem olarak çeşitli zihinsel ve fiziksel kayıplara neden olduğu ve inme geçiren bireylerin hastalık sonrası fiziksel düzeyde yetersizlik yaşadığı bilinmektedir (Grenwood, vd. 2008 & Ilse vd. 2008 & Park, 2008). İnme, mortalite ve morbiditeye neden olan önemli bir rahatsızlıktır. İnme sonrası bireylerin sosyal yaşamlarını yaşamını sürdüren hastaların büyük bir kısmı kalıcı fiziksel yetersizlikler, anksiyete ve depresyon gibi psikolojik sorunlar yaşamaktadırlar (Lundström, vd. 2008 & Schreıner, vd. 2006 & Soyuer, vd. 2007).

İnme, bireylerin yaşam kalitesini olumsuz etkileyen ve ekonomik olarak toplumda gelir kaybına yol açan bir sorundur. Ülkemizde inme prevalansı ve inmeli bireylerin yaşam kalitesi ile ilgili veriler yetersizdir. Ancak toplumda birçok bireyin bu hastalık ile karşılaştığı bilinmektedir. Topçu vd. 2012 yaptığı bir çalışma sonucunda, inme hastalığının fiziksel ve fonksiyonel kayıplara hatta bireylerin yaşamlarının sona ermesine sebep veren önemli bir hastalıktır. Bireylerin tedavi sonrasında tekrardan tekrarlayan inmenin hastalığının etkilerinin görülme sıklığı ve bireylerin yaşamına etki eden olumsuzlukların artırdığından bahsedilmiştir.

İnme hastalığı için yapılan çalışmalarda tedavilerin sürekli ve devamlılığın olması gerektiği gözler önüne serilmiştir. Sürekli tedavi gerekliliği tele-rehabilitasyon uygulamalarının gerekliliği bir kez daha neden olması gerektiğini vurgulanmıştır.

2.2.2. Multipl Skleroz Hastalığının Tanımı

Multipl Skleroz (kısaca bilinen diğer adı ile MS) hastalığı, bireylerin nörolojik sorunlar nedenleri ile mobilite ve fonksiyonel işlevlerinde değişik derecelerde sorunlara ve kayıplara yol açan, bireyin sosyal ve yaşamsal faaliyetlerinde kısıtlamalara sebep olan ileri düzeyde engellilik oluşturan ve yaşam kalitesini bozan bir hastalıktır.

MS; genetik olarak yatkın bireylerde, çevresel faktörler, viral veya bakteriyel ajanlar, inflamatuvar ve otoimmün cevap esnasında salgılanan sitokinler ve henüz bilinmeyen bazı etiyolojik ajanların kombinasyonu sonucu oluşan bir hastalıktır (Öztürk, vd. 2017).

MS hastalığının seyri her bireyde farklı olması sebep ile rehabilitasyon programları bireylere özgü olarak hazırlanması gerektiği Mutluay (2006) tarafından yapılan çalışmada belirtilmiştir. Bireylerin mevcut engellilik seviyelerine bağlı olarak bireylerin gereksinimleri ve mevcut durum için hedeflerine uygun programlar tasarlanmalıdır.

MS hastalığına sahip bireylere uygulanan rehabilitasyon uygulamaları kişilerin günlüm yaşam aktivitelerine etki etmeyen bağımsız bir araç ile yürütülmesi gerektiği için tele-rehabilitasyon sisteminin önemi MS hastalığının tedavisinde önem arz etmektedir.

2.2.3. Parkinson Hastalığının Tanımı

1817 yılında literatüre James Parkinson tarafından tanımlanan Parkinson hastalığı, beyin hücrelerinde kayıp ile seyreden ve yavaş ilerleyen bir hastalıktır. Parkinson hastalığı insidansı ve prevalansı hızla artmakta olan bir nörodejeneratif bir hastalıktır. Günümüzde, dünya çapında toplamda 6 milyon kişiden fazlasını etkilediği bilinmektedir (Öksüz, 2022). Bu sayının 2040 yılında ortalama olarak 2 katına çıkması beklenmektedir.

Parkinson hastalığı genelde 50 yaş üstü bireylerde belirtileri ortaya çıkan ve bu belirtilerin yavaş seyredip artan bir şekilde ilerleyen nörolojik bir hastalıktır. Parkinson hastalığı ilk başlardaki belirtileri ile anlaşılmayabilir, çünkü hastalığı sebep olan sorunlar genellikle bireylere bir rahatsızlık vermez ya da göz ardı edilir (Waters, 2000).

Parkinson hastalığı bireylerin motor fonksiyonlarına doğrudan etki eden ve hastalarda farklı sorunlara yol açan nörodejeneratif bir hastalıktır. Bu hastalık bireylerin günlük yaşamına etkileri ve hastalığın artan evrelerinde bireylerin yaşam kalitesini önemli ölçüde azaltır. Erken tedavi ve fiziksel egzersizler ile semptomların yol açtığı sorunlar azaltılabilir ancak hastalara doğrudan uzman tarafından erişimleri zordur. Parkinson tedavisi için tele-rehabilitasyon kullanımı araştıran çok az çalışma olmasına rağmen günümüz teknolojileri ve iletişim araçlarındaki gelişmeler ile beraber tele-rehabilitasyon uygulamaları hastalar için önemli bir tedavi ve semptom etkilerini azaltmak için uygulanması gereken önemli bir sağlık hizmetidir (Vellata, 2021).

2.3. TeleNöroRehab

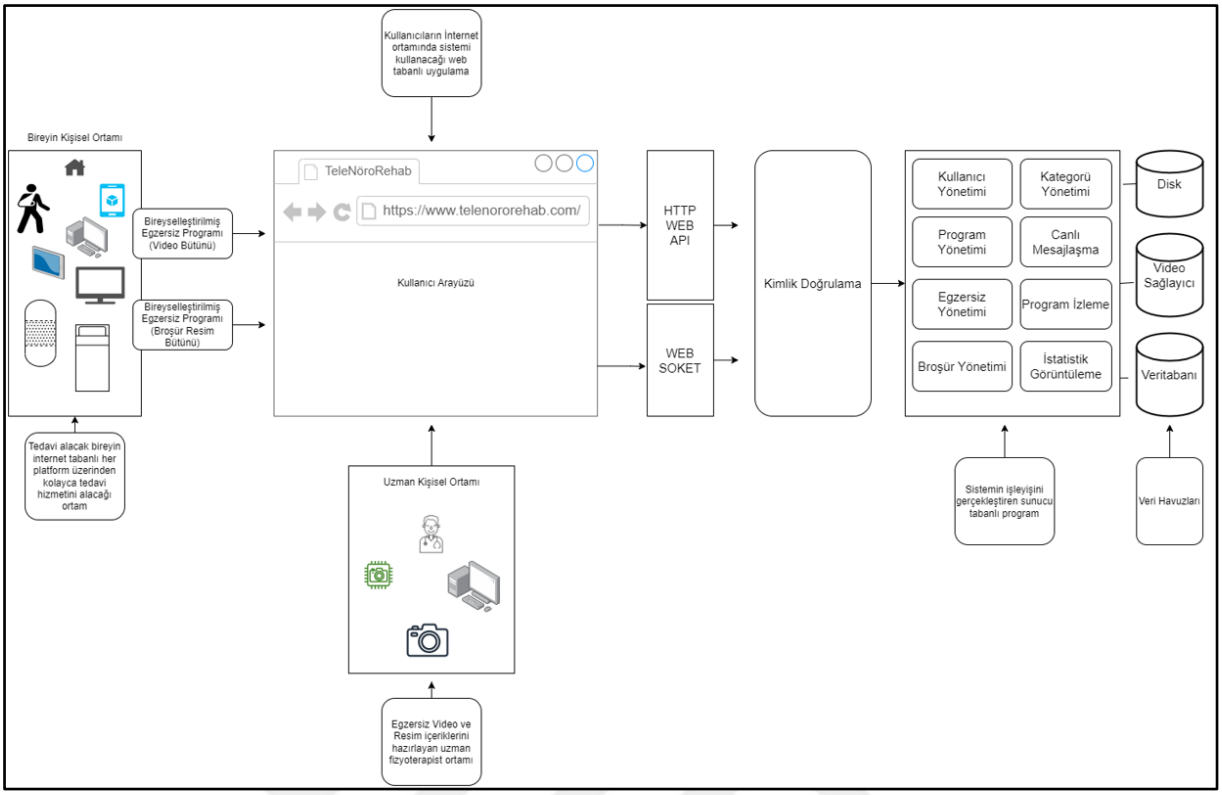
Bu tez kapsamında geliştirilen tele-rehabilitasyon projesinin ismi TeleNöroRehab'dır. Tezin bu bölümünden sonra geliştirilen proje TeleNöroRehab adı ile belirtilecektir.

2.3.1. TeleNöroRehab Tanımı

TeleNöroRehab, tele-rehabilitasyon uygulamasına örnek olarak geliştirilmiş özellikle Evde Tele-rehabilitasyon ve Toplu Tele-rehabilitasyon uygulamaları için geliştirilmiş dinamik bir sistemdir.

TeleNöroRehab, nörolojik hastalıklar sonucunda tanı almış olan kronik dönemdeki bireylerin uzaktan değerlendirilmesine ve egzersiz programları verilerek takip edilmesine imkân veren Türkçe dil desteği olan Türkiye'de geliştirilmiş kapsamlı ilk sistemdir.

TeleNöroRehab sistemi web sitesi üzerinden internete bağlanan tüm cihazlarda kolayca erişebilen bir uygulamalar bütünü olarak geliştirilmiştir. Sistem uzmanlar tarafından hazırlanmış egzersiz sisteme yüklenerek hastalara özgü olarak program sunarak uzaktan tedavi hizmeti sunmayı amaçlamaktadır.

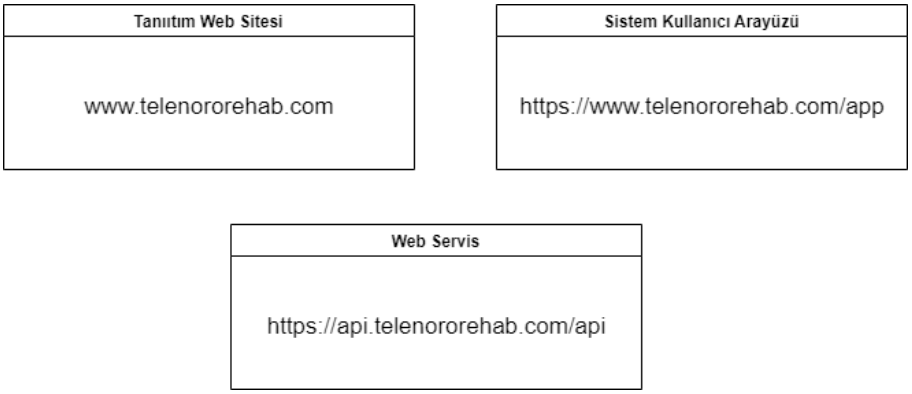


Şekil 2.1. TeleNöroRehab çalışma prensibi

Üç arayüzden oluşan sistemin uzman arayüzünde fizyoterapistler tarafından hazırlanmış egzersiz videoları veya broşürleri sisteme yüklenebilir, hasta listesine ve hastaların geri bildirimlerine ulaşılabilir, çeşitli ve hastalara özgü rehabilitasyon programları hazırlanabilir ve güncellenebilecek şekilde tasarlanmıştır. Hasta arayüzünde ise bireyler kendi bilgilerine ve tedavisine özel olarak hazırlanmış rehabilitasyon programına ulaşabilecek ve ihtiyaç duyduğunda devam eden programı hakkında iletişim kurabilmesini sağlayacak bir mesajlaşma paneli bulunmaktadır. Yönetici arayüzünde ise aynı anda farklı kurumlardan uzmanlar ekleme, bilgi düzenleme, egzersiz havuzu oluşturma gibi birçok adımın üst düzey kontrolü sağlanabilmektedir.

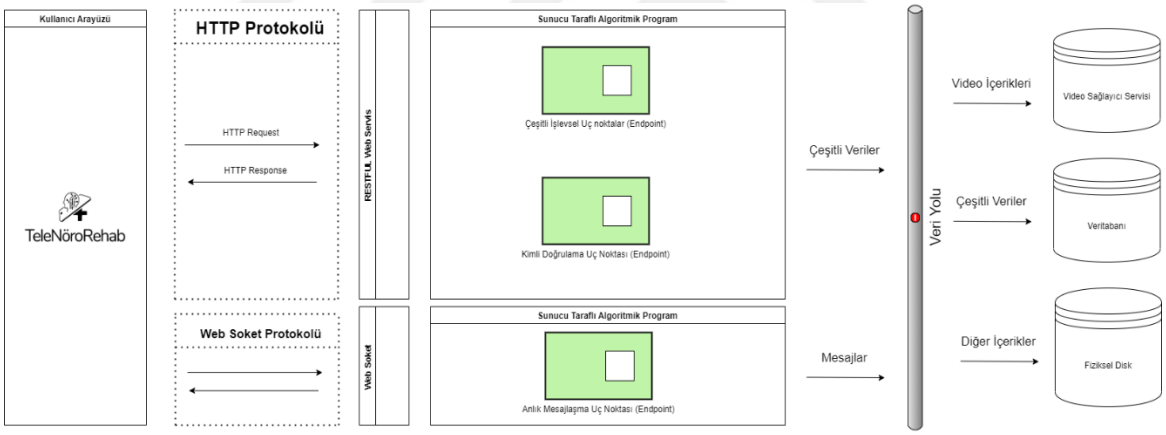
2.3.2. TeleNöroRehab Yazılımsal Çerçevesi

TeleNöroRehab sistemi kapsamlı bir web uygulaması olup iki front-end (Ön Yüz / Kullanıcı Tarayıcı Tabanlı Yazılım) ve bir back-end (Arka Yüz / Sunucu Tabanlı Yazılım) programından oluşmaktadır.



Şekil 2.2. TeleNöroRehab web uygulamaları tanımı

Proje geliştirilirken ve tasarlanırken günümüz teknolojilerin güncel olarak kullanılan birçok alt yapıyı hem front-end' hem de back-end'e kullanmaktadır. Kullanıcılara web tarayıcı ile hizmet veren sistem güncel web teknolojilerinden "Single Page Application" (SPA) geliştirilmiş olup sunucu taraflı program ile HTTP protokolü üzerinden inşa edilen web servisler ile veri alışverişi yapmaktadır (Şekil 2.2).



Şekil 2.3. TeleNöroRehab yazılımsal akış gösterimi

Şekil 2.2 de gösterildiği üzere TeleNöroRehab uygulaması kullanıcı tarayıcı tabanlı ve sunucu tabanlı çalışan 2 önemli programdan oluşmaktadır. Bu iki bölümler arasında oluşan verilerin transferi günümüzde kullanımı yaygın olan internet altyapısının temel protokolü olan İnternet Protokolü (IP) üzerine inşa edilmiş HTTP protokolü üzerinden web servis yardımları ile iletişim sağlamaktadır. Sistem üzerinde kullanılan teknolojik hizmetler Yöntem kısmında detaylıca anlatılmaktadır.

2.4. Literatür Özeti

Çevrimiçi olarak uzaktan nörolojik hastalıklara sahip bireylerin tedavi aşamalarında teknoloji ve internet destekli bir alt yapı sunarak tele-rehabilitasyon uygulaması geliştirmeden önce bu alanda yapılmış literatürdeki çalışmalar ve uygulamaların bir taramasını yapılmıştır. Yapılan bu taramada bireylere tele-rehabilitasyon hizmeti ile sağlanan tedavi uygulamaları ve çalışmaya katkıda bulunacak taramalar aşağıda özetlenmiştir.

Dodakian ve diğerleri tarafından 2017 yılında çalışmada inme hastalığına sahip bireyler erken teşhis döneminde tele-rehabilitasyon hizmetine dayanan oyunlaştırma tabanlı bir sistem geliştirerek sonuçlarını değerlendirmişlerdir. Bu çalışmaya katılan katılımcılar inmeden 3 ila 24 ay sonra stabil kol motor defisitleri olan hastalardır. Her hasta için evlerinde rahatça kullanabileceği 28 günlük bir tele-rehabilitasyon tedavisi uygulanmıştır. Hastalar bu 28 günlük program içerisinde her gün, özelleştirilmiş egzersizler ve oyunlara odaklanan yapılandırılmış 1 saat, vuruş eğitimi ve bir saatlik serbest oyundan oluşuyordu. Toplamda 336 katılımcı ile gerçekleşen çalışmaya 329/336 (%97,9) katılım gösterilmiştir. Bu çalışmanın sonuçlarına göre, tele-rehabilitasyon sistemi kullanılarak günlük inme eğitimi, inmeyi önleme bilgisinde %39'luk bir artışla gözlenmiştir. Yazarların görüşüne göre ev tabanlı bir tedavi sistemi, katılımcılara eğitim ve ikincil inme önleme sağlamada etkili olmuştur. Bilgisayar tabanlı bir arayüzün kullanılması, inme sonrası hastaların sağlığını izlemek ve iyileştirmek için birçok fırsat ve sunarak tedavilerin geliştirmesine olanaklar sunduğunu değerlendirmiştir.

Cason tarafından 2007 yılında yapılan çalışmasında “enTECH Telerehabilitation Program” adı altında tele-rehabilitasyon programının sonuçlarını araştırdı. Bu uygulamanın çıkış fikri, Amerika Birleşik Devletleri’ndeki Kentuck eyaletinin güneydoğusundaki çocukların erken müdahale hizmetleri ile uygun tedavi alabilecek ancak bölgenin coğrafik sorunları nedeni ile sağlık hizmetinde eksiklikler olan 2 ailenin belirlenmesi ile başlanmıştır. Bu iki aile sağlık hizmeti almak için yaklaşık 645 kilometrelik gidiş-dönüş yolculuğu gerçekleştirmeleri gerekiyordu. Bu çalışma başlamadan önce her iki aileye programın tanıtımı sağlayacağı faydaları ve elde edilen veriler ile tedavilerine yardımcı olacakları detaylı anlatılıp ailelerden imzalı onaylı bir izin alınmıştır. Program Kasım 2007 yılında 12 haftalık bir program düşünülerek başlanmış ancak bölgenin coğrafik sorunları ve hastalığın seyrinin değişmesi ile 8 haftalık bir çalışma yapılmıştır. Her aile için teknik destek, video konferans bağlatışı ve doğru konumlandırılmış kameralar ile ortam hazırlanmıştır. Bu program hastalar ve uzmanların video konferans ile uzaktan rehabilitasyon hizmeti ve tedavisine dayalıdır. Terapi

sırasında kurulan ortamda hasta, bakıcı ve uzaktan bağlantı kuran uzman vardır. Burada bakıcı tedavi sırasında çocuğa destek olması için ortamda bulunması uzaktan etkileşimi kolaylaştırmıştır. Her video konferans Skype programı üzerinden ortalama 30 dakika sürmüştür. Bu çalışmanın değerlendirmesinin sonuçları, tele-rehabilitasyonun sağlayıcı eksikliklerinin olduğu kırsal alanlarda yaşayan çocukların terapötik ihtiyaçlarını uygun maliyetli bir şekilde karşılama potansiyeline sahip olduğunu göstermiştir. enTECH Telerehabilitasyon Programı, sağlık eşitsizliklerini iyileştirmek ve rehabilitasyon hizmetlerine erişimi iyileştirmek için erken müdahale hizmetleri sunmak üzere tele-rehabilitasyonun fayda sağlayıcı bir model görevi olduğunu göstermiştir.

Ortiz-Gutiérrez ve arkadaşları tarafından 2013 yılında yapılan çalışmada geleneksel tedavinin mevcut olmadığı durumlarda fizik tedaviye uygun bir alternatifi temsil eden bir tele-rehabilitasyon programını sunan multipl sklerozlu hastalarda postüral kontroldeki potansiyel gelişmeleri göstermeyi amaçlamaktadır. Bu çalışmaya toplanda 50 hasta katılmış olup iki farklı gruba ayrılmışlardır. İlk grup haftada iki kez olmak üzere her seans ortalama 40 dakika olarak fizyoterapi tedavisi aldığı belirtilmektedir. Diğer grup ise Microsoft firmasına ait Xbox 360 ve Kinect oyun konsolu kullanılarak sanal gerçeklik üzerine kurulu video konferans aracılığıyla izlenen tele-rehabilitasyon tedavisi olarak haftada dört seans (seans başına 20 dakika) olmak üzere 40 seansa katılmışlardır. Tedavi programı her iki grup için toplamda 10 hafta sürmüştür. Tüm hastaları başlangıçta ve tedavi protokolünün sonunda değerlendirmek için bilgisayarlı dinamik “Sensory Organization Test” kullanılmıştır. Çalışmanın sonuçlarına göre her iki grupta da genel denge üzerinde bir gelişme göstermiştir. Görsel tercih ve vestibüler bilginin katkısı, deney grubunda önemli farklılıklar sağlamıştır. Sanal gerçeklik sistemine dayalı bir tele-rehabilitasyon programının, multipl sklerozlu kişilerde dengeyi ve postüral kontrolü sürdürmek için gerekli olan duyusal bilgi işleme ve entegrasyon sistemlerini optimize etmeye izin verdiğini göstermiştir. Bu çalışmada kullanılan sanal gerçeklik teknolojisinin öngörülü kişisel bilgisayar ve yanıt mekanizmalarını mümkün kıldığını ve geleneksel terapinin kolayca mevcut olmadığı durumlarda başarılı bir terapötik alternatif olarak hizmet edebileceğini önermiştir.

Şener ve arkadaşlarının 2021 yılında yaptığı çalışmada tele-rehabilitasyon ve bir sağlık hizmetinde yüz yüze yapılan tedavi değerlendirmelerin sonuçlarını karşılaştırarak, spastik Serebral Palsi rahatsızlığına sahip çocukların alt ekstremite fonksiyonel performanslarının bir sağlık hizmetin dışında da doğru ve etkin bir şekilde değerlendirilip değerlendirilemeyeceği

konusunda bilgi sahibi olmayı amaçlamaktadır. Bireylerin kliniğe ulaşamadığı farklı durumlarda (uzak mesafe, pandemi vb.) tele-rehabilitasyon yöntemi kullanılarak yapılacak hasta-terapist ilişkisinin devamlılığının ve tedavi bütünlüğünün korunabileceği ispatlamaya çalışmaktadır. Çalışma 3 spastik diparetik SP'li olgu alınmış ve her olgu hem klinik ortamında yüz yüze hem de tele-rehabilitasyon yöntemi kullanılarak değerlendirilmiştir. Yapılan yüz yüze ve tele-rehabilitasyon sonucunda, performans testlerinde farklılıklar görülürken, kişinin kendisi veya bakım veren tarafından yapılan değerlendirmelerin sonuçları arasında tutarlılık olduğu tespit edilmiştir. Başka bir deyişle, bireyin kendi kendini veya bakım verenin değerlendirmesiyle elde edilen sonuçlar arasında uyumlu bir görüntü ortaya çıkmıştır. Öte yandan, yüz yüze ve tele-rehabilitasyon arasında performans testlerinde farklılıklar belirlenmiştir.

Ferraris ve arkadaşlarının 2022 yılında yaptığı çalışma içerisinde geliştirilen proje REHOME adı verilen bir sistemin arka planını ve sistemi kullanan hastaların değerlendirilmesinden bahsedilmiştir. Nörolojik hastalıklarla ilişkili bilişsel, motor ve uyku bozukluklarının uzaktan izlenmesi ve rehabilitasyonu için yenilikçi metodolojileri ve cihazları entegre eden REHOME projesinin öncelikli hedeflerinden biri, sağlık kuruluşlarından hastanın evine kadar tedavinin sürekliliğini sağlayarak hasta ve klinisyen ihtiyaçlarını karşılamaktır. Bu çalışmada 70 katılımcı üzerinde yapılan kullanılabilirlik ve kullanıcı deneyimi anketlerine ilişkin sonuçlar üzerine değerlendirmeler yapmaktadır. Çalışmada geliştirilen projede katılımcılara belirli oyunlaştırılmış egzersiz programları bireylerin durumları incelenmektedir. Bu çalışmanın sonucundaki değerlendirmelere genel olarak, deneysel denemelerdeki katılımcıların deneyimlerini olumlu olarak değerlendirdiklerini göstermektedir. Geliştirilen proje bireylere uygulanan tedavinin ev ortamında etkili bir şekilde kullanılabileceğinden ve rehabilitasyon tedavilerinin devamlılığını ve hastaların yaşam kalitelerinin iyileştirilmesini sağlayabileceğinden bahsetmektedir.

Sobierajska-Rek PT ve arkadaşlarının 2020 yılında yaptığı çalışmada "Duchenne" kas distrofisi, (kademeli motor fonksiyon kaybı, solunum yetmezliği ve kardiyomiyopati ile karakterize genetik bir hastalıktır) hastalığına sahip bireylerin küresel koronavirüs pandemisi sırasında, sosyal mesafenin korunması ve kendi kendine izolasyon, düzenli multidisipliner bakımını zorlaştırdığını, ancak diğer yandan teknolojik geliştirmelerin ışığında tele sağlık çözümlerinin kullanarak tedavilerin devam edebileceğini araştırmıştır. Çalışmanın amacı, hastaların pandemide rehabilitasyona ilişkin durumlarını araştırmak, çevrimiçi rehabilitasyon

programı oluşturarak uzaktan tedavi sağlamak. Çalışma, Duchenne kas distrofisi olan 69 erkek çocuğu ile yapılmıştır. Rehabilitasyon programı, hastalar ve bakıcılar için çevrimiçi olarak belirli ortamlarda sunulmuştur. Aynı program videoya kaydedilip ve internette yayınlanmıştır. Çevrimiçi motor değerlendirme aracı altı motor testten oluşmaktan olup, bakıcılardan görevleri gerçekleştirmeleri ve tasarlanmış bir uygulama kullanarak hastanın duruşunun bir fotoğrafını paylaşmaları istenerek çalışmada uzaktan tedavini nasıl gerçekleştiğini görerek araştırma imkânı sunulmuştur. Çalışmanın sonuncunda tele-rehabilitasyon (çevrimiçi iletişim veya video) ile hastalar, bakıcıların yardımıyla evde rehabilitasyona devam edebilir. Çevrimiçi videolar/talimatlar/video kılavuzları, Duchenne musküler distrofisi olan hastaların ebeveynleri/bakıcıları tarafından yabancı klinik ortamından daha kabul edilebilir olduğunu ortaya atmıştır.

Literatürde, tele-rehabilitasyon ile ilgili birçok çalışma mevcuttur. Her çalışma belirli bir hastalık grubuna hitap etmektedir. Literatürde diğer çalışmaların ve bu tez kapsamında geliştirilen çalışmanın ortak özelliği, çeşitli coğrafik, maddi, konumsal, iklim ve pandemi gibi sorunlar yüzünden klinik ortamında sağlık hizmeti alamayan bireylerin kişisel ortamlarında aynı tedavileri aynı etkiler ile alabileceğini kanıtlamıştır. Çalışmalara katılan katılımcıların kişisel ortamlarında aldıkları tedavilerin daha sürdürülebilir ve devamlı olduğu dikkat çekmiştir. Tele-rehabilitasyon hizmetlerinin sağlık kuruluşlarının büyük maddiyat sorunlarına büyük bir çözüm olacağı birçok çalışmada önemle vurgulanmıştır. Literatürdeki çalışmalara vurgulanan bir diğer nokta ise bu hizmeti alacak bireylerin yaş ve teknoloji okur yazarlığına dikkat edilip yapılan çalışmaların bunlara uygun olarak her yaş ve okur yazarlık kesiminin kolayca hizmet alacağı şekilde geliştirilmesi gerektiği belirtilmiştir.

TeleNöroRehab, literatürdeki yapılan diğer çalışmalara istinaden Türkçe dil desteği olan Türkiye'de geliştirilmiş dinamik olarak uzaktan rehabilitasyon hizmeti sağlayacak kapsamlı ilk sistemdir. Bu çalışma "kaydet ve sonsuza dek kullan" mantığında uzmanlar tarafından açıklayıcı olarak egzersizlerin resim, video ve yazı olarak kaydedildikten sonra bireylere özgü programlar ile birçok hastalık grubu veya eğitimlerin sağlanabileceği bir hizmettir. Bu çalışmadaki geliştirilen sistem literatürdeki diğer çalışmalara göre özelleştirme daha ön plandadır.

3. YÖNTEM

TeleNöroRehab internet üzerinden hizmet veren kapsamlı bir web uygulamasıdır ve içerisinde birçok teknolojiyi barındırmaktadır. Bölüm 3.1'de kullanılan araçlar ve teknolojiler hakkında bilgi vermektedir. Bölüm 3.2 de ise kullanılan teknolojilerin entegrasyonu hakkında detaylı bilgi vermektedir.

3.1. Araçlar ve Teknolojiler

Tez kapsamındaki proje geliştirilirken programla araçları ve güncel teknolojiler kullanılarak geliştirilmiştir. Uygulama geliştirme araçlarından Visual Studio, Visual Studio Code, SQL Server Management Studio, Git, Github Desktop, Edge Browser Development Tools, ReSharper, Postman, Swagger gibi yardımcı geliştirme araçları kullanılmıştır. TeleNöroRehab kapsamında geliştirilen tüm uygulamalar IIS üzerinden barındırılıp sunulmaktadır.

3.1.1. Visual Studio (2019)

Visual Studio, Microsoft tarafından geliştirilen bir bütünleşmiş geliştirme ortamı (IDE) sunan bir yazılımıdır. Yazılım geliştirme için birçok farklı programlama dili ve platformu destekler. Visual Studio, kod yazımı, hata ayıklama, derleme ve dağıtım işlemlerini kolaylaştırmak için gerekli araçları sağlar.Visual Studio, Windows uygulamaları, web uygulamaları, mobil uygulamalar, oyunlar ve diğer birçok yazılım türü için kullanılabilir. İçinde birçok farklı özellik ve araç barındırdığı için, özellikle büyük ve karmaşık projeler üzerinde çalışırken çok faydalı olabilir. Ayrıca, Visual Studio, .NET Framework'ün bir parçasıdır ve bu nedenle C#, Visual Basic, F# ve diğer .NET dilleriyle kullanılabilir.

3.1.2. Visual Studio Code

Visual Studio Code, Microsoft tarafından geliştirilen açık kaynak kodlu bir metin düzenleyicisi ve kod editörüdür. Visual Studio'nun aksine, daha hafif bir yazılımdır ve temelde bir metin editöründen daha fazla özellik sunar.Bu editör içerisinde birçok farklı programlama dili ve platformu için kullanılabilir ve çeşitli eklentilerle özelleştirilebilir. Geliştiriciler, kod yazımı, hata ayıklama, derleme ve dağıtım işlemlerini kolaylaştırmak için gerekli eklentiler kullanabilirler.

TeleNöroRehab projesi geliştirme aşmalarında Visual Studio Code için birçok işleme yardımcı olan eklentiler kullanılmıştır.

Kullanılan eklentiler şunlardır:

1. Angular Language Service
2. Angular Snippets
3. Angular Support
4. Auto Rename Tag
5. Beautify
6. change-case
7. Code Spell Checker
8. CSharp2TS
9. Visual Studio Keymap
10. IntelliCode

3.1.3. SQL Server Management Studio

SQL Server Management Studio (SSMS), Microsoft tarafından geliştirilen bir veri tabanı yönetim sistemi (DBMS) aracıdır. Bu araç, Microsoft SQL Server veri tabanı yönetim sistemleri için bir GUI (grafik kullanıcı arayüzü) sağlar.

SSMS, veri tabanı yönetimi için birçok özellik sunar. Kullanıcılar, veri tabanı oluşturma, tablo oluşturma, veri ekleme, veri sorgulama, veri yedekleme ve geri yükleme, depolama yönetimi, kullanıcı yönetimi, güvenlik yönetimi ve diğer birçok veri tabanı yönetim görevini gerçekleştirebilirler. Ayrıca, SSMS, Transact-SQL (T-SQL) adı verilen SQL Server için özel bir sorgu diliyle birlikte gelir. Bu, veri tabanı yöneticilerinin ve geliştiricilerinin veri tabanı sorguları oluşturmasını ve çalıştırmasını sağlar.

3.1.4. Git

Git, bir sürüm kontrol sistemi ve kaynak kodu yönetim sistemidir. Git, Linux çekirdeğinin geliştirilmesi için Linus Torvalds tarafından 2005 yılında oluşturulmuştur. Git, bir projenin tarihçesini takip etmek, değişiklikleri yönetmek ve paylaşmak için kullanılır. Git, her değişikliğin bir sürüm olarak kaydedildiği bir yapıda çalışır ve bu nedenle kullanıcılar projelerinde yapılan her değişikliğin tarihçesini kolayca takip edebilirler. Git, kaynak

kodlarını ve belgeleri yüklemek, indirmek, değiştirmek ve farklı sürümlerini karşılaştırmak için kullanılabilir.

3.1.5. Github Desktop

Github Desktop, kullanıcıların Git ve Github ile etkileşimlerini kolaylaştırır. Git, bir sürüm kontrol sistemi olup, kodların takibini, yönetimini ve kaynak kodlarının paylaşımını kolaylaştırır. Github Desktop, kullanıcılara bu işlemleri kolayca yapma imkânı sunar ve ayrıca projeleri klonlama, pull ve push işlemleri yapma, dal ve birleştirme işlemleri gibi Git işlemlerini kolaylaştırır.Bu araç kullanıcıların kodlarını daha iyi yönetmelerini sağlar. Kullanıcılar, değişiklikleri görsel olarak takip edebilir, çatallar (fork) oluşturabilir, dal (branch) oluşturabilir ve birleştirebilir. Ayrıca, Github Desktop, kullanıcılara kodlarında meydana gelen değişikliklerin tarihçesini gösterir, yaptıkları değişikliklerin açıklamalarını ekleyebilme imkânı sunar ve kodların farklı sürümlerini karşılaştırabilir.

3.1.6. Edge Browser Development Tools

Edge Browser Development Tools, Microsoft Edge tarayıcısında bulunan bir dizi araçtır. Bu araçlar, web sayfalarının oluşturulması, test edilmesi ve hata ayıklanması için kullanılabilir. Edge Browser Development Tools, web sayfasının kodunu incelemeye, HTML, CSS ve JavaScript dosyalarını düzenlemeye, stil sayfalarını ve betikleri test etmeye, ağ trafiğini izlemeye ve hata ayıklamaya olanak tanır. Ayrıca, mobil cihazlarda web sayfalarını test etmek için simüle edilmiş cihazlar da içerir. Bu araçlar, web geliştiricilerinin daha hızlı ve daha etkili bir şekilde web sayfaları oluşturmalarına yardımcı olabilir ve sayfaların doğru bir şekilde çalıştığından emin olmak için kullanılabilir. Edge Browser Development Tools, diğer tarayıcılarda bulunan geliştirme araçlarına benzer şekilde çalışır ve web geliştirme işlemlerini daha kolay ve verimli hale getirir.

3.1.7. ReSharper

ReSharper, JetBrains firması tarafından geliştirilen bir Visual Studio eklentisidir. Bu eklenti, C# ve diğer .NET dilleriyle yazılmış kodların daha hızlı ve daha verimli bir şekilde yazılmasına, test edilmesine ve yönetilmesine olanak tanır.

ReSharper, bir dizi kod analizi, kod düzenleme ve otomatik kod üretimi aracı içerir. Bu araçlar, kodun daha okunaklı ve bakımı daha kolay hale getirilmesine yardımcı olabilir. Ayrıca, ReSharper, kodun daha iyi bir şekilde test edilmesine ve hata ayıklanmasına da yardımcı olabilir. ReSharper, ayrıca Visual Studio'da bulunan bazı özellikleri de geliştirir.

Örneğin, IntelliSense özelliğini daha zengin hale getirir, kod önerileri sunar, kodun daha hızlı bir şekilde yazılmasını sağlar ve kodun hızlı bir şekilde gezinilmesine olanak tanır.

3.1.8. Postman

Postman, web API'leri oluşturmak, test etmek ve belgelemek için kullanılan bir geliştirme aracıdır. Postman, kullanıcıların API isteklerini ve yanıtlarını görselleştirmelerine ve yönetmelerine olanak tanır. Postman, API'leri test etmek ve hata ayıklamak için birçok farklı araç sunar. Bu araçlar, istekleri oluşturma ve gönderme, yanıtları görüntüleme ve karşılaştırma, otomatik testler oluşturma ve API performansını test etme gibi işlemleri gerçekleştirmeyi kolaylaştırır.

3.1.9. Swagger

Swagger, birçok programlama dilinde kullanılabilen açık kaynak bir araçtır. Swagger, API tanımlama diline dayanan OpenAPI adlı bir spesifikasyon kullanır. OpenAPI, API'nin nasıl çalıştığını ve nasıl kullanıldığını açıklamak için bir açık standarttır. Swagger, kullanıcıların API'nin özelliklerini ve kaynaklarını keşfetmelerine, test etmelerine ve belgelemelerine yardımcı olur. Ayrıca, Swagger sayesinde, bir API kullanarak bir uygulama oluşturmak isteyen geliştiricilerin, API'nin tam olarak ne yapabileceği hakkında daha iyi bir fikir edinmeleri kolaylaşır.

3.1.10. IIS

IIS, Internet Information Services (İnternet Bilgi Servisleri) kısaltmasıdır. Microsoft tarafından geliştirilen bir web sunucusu yazılımıdır. IIS, web sayfalarını ve diğer internet protokolleriyle çalışan uygulamaları sunmak için kullanılır. IIS, Windows işletim sistemlerinin bir parçasıdır ve .NET Framework, ASP.NET ve Microsoft'un diğer web teknolojilerini destekler. IIS, geniş bir özellik yelpazesi sunar, örneğin FTP sunucusu olarak da kullanılabilir, dinamik web sayfalarını destekler, kullanıcı yetkilendirme ve kimlik doğrulama işlemleri yapabilir ve SSL/TLS şifrelemesi gibi güvenlik özellikleri sağlar. IIS, büyük ölçekli web uygulamaları için uygun bir seçenektir ve yüksek performans, güvenilirlik ve güvenlik sağlamak için tasarlanmıştır.

TeleNöroRehab projesi geliştirirken kullanılan araçlardan bahsedilmiş olup proje içerisinde birçok farklı teknolojiler kullanılmıştır. TeleNöroRehab bu alanda bir örnek model projesi olması için içerisinde kullanılan teknolojilerin ayrıntılı bir şekilde verilmesi önemlidir. Proje içerisinde kullanılan teknolojileri 3 bölüme ayırmak gerekir.

1. Front-End (Önyüz) Teknolojileri
2. Back-End (Arkayüz) Teknolojileri
3. Veri Haberleşme Teknolojileri

Front-End (Önyüz) yani kullanıcı arayüzleri geliştirirken birçok yardımcı teknoloji ve programlama ortamları kullanılmıştır.

3.1.11. Node.js

Node.js, JavaScript tabanlı bir programlama dili olan ve genelde sunucu tarafı uygulama geliştirme için kullanılan bir açık kaynaklı çalışma zamanı ortamıdır. Node.js, Google Chrome'un V8 JavaScript motorunu temel alır ve özellikle yüksek verimlilik gerektiren uygulamalar için tasarlanmıştır. Node.js, asenkron ve olay odaklı bir mimariye sahiptir, bu da uygulamaların daha hızlı çalışmasına ve daha az bellek kullanmasına olanak tanır. Node.js, birçok farklı paket ve modül içeren kapsamlı bir kütüphane deposu olan NPM (Node Package Manager) ile birlikte gelir. NPM, geliştiricilere hazır paketlerin kullanımını sağlar ve uygulamaların daha hızlı geliştirilmesine olanak tanır.

TeleNöroRehab içerisindeki front-end (önyüz) katmanlarında sıkça Node.js ortamının araçları kullanılmaktadır.

3.1.12. Grunt

Grunt, JavaScript projelerindeki tekrarlayan işleri otomatize etmek için kullanılan bir JavaScript görev çalıştırıcı aracıdır. Grunt, projelerdeki sık tekrarlanan işleri (örneğin, dosya birleştirme, kod sıkıştırma, testlerin çalıştırılması vb.) otomatikleştirmek için kullanılır. Grunt, Node.js tarafından desteklenenen bu araç NPM ile birlikte kullanılabilir. Grunt, özellikle büyük ve karmaşık JavaScript projeleri için idealdir. Projenin geliştirme sürecini hızlandırmak ve hataları azaltmak için kullanılabilir.

3.1.13. TypeScript

TypeScript, Microsoft tarafından geliştirilen açık kaynak kodlu bir programlama dilidir. TypeScript, JavaScript'e ek olarak, statik tipleri ve sınıf tabanlı nesne yönelimli programlama (OOP) özelliklerini de içerir ve bünyesinde barındırdığı derleyici sayesinde, yazılan kodu JavaScript koduna çevirir. TypeScript gerek istemci taraflı gerekse sunucu taraflı yazılım geliştirmede kullanılabilmektedir.

3.1.14. Single Page Application

SPA (Single Page Application), tüm uygulamanın tek bir web sayfası içinde çalıştığı bir web uygulama türüdür. Bu tür uygulamalarda, web sayfası, kullanıcının etkileşimleriyle dinamik olarak içerik değiştirir ve sayfayı yenilemeden uygulamanın çalışmasını sağlar. SPA, JavaScript çerçeveleri veya kütüphaneleri kullanarak geliştirilir. Bu araçlar, uygulamaların hızlı ve sorunsuz bir şekilde çalışmasını sağlar. SPA uygulamaları, RESTful API'ler veya GraphQL API'ler gibi web hizmetleri ile iletişim kurarak, veri alışverişi yapar.

3.1.15. Angular

Angular, Google tarafından geliştirilen bir JavaScript platformudur. Web uygulamaları için kullanılan bir çerçeve (framework) olarak kabul edilir. Angular, SPA (Single Page Application) uygulamaları, mobil uygulamalar ve masaüstü uygulamaları gibi farklı platformlarda kullanılabilir. Bu çerçeve, MVC (Model-View-Controller) desenini kullanarak, veri, görsel ve işlevleri ayrı ayrı yönetir. Angular, HTML, CSS ve JavaScript kodlarının birlikte çalışmasıyla uygulama geliştirme sürecini hızlandırır ve kolaylaştırır. Angular, TypeScript ile yazılan bir platformdur. TypeScript, JavaScript'in üzerine inşa edilen bir programlama dili olarak kullanılır ve güçlü bir tip kontrolü sağlar. Bu da hata ayıklama ve geliştirme sürecinde kolaylık sağlar.

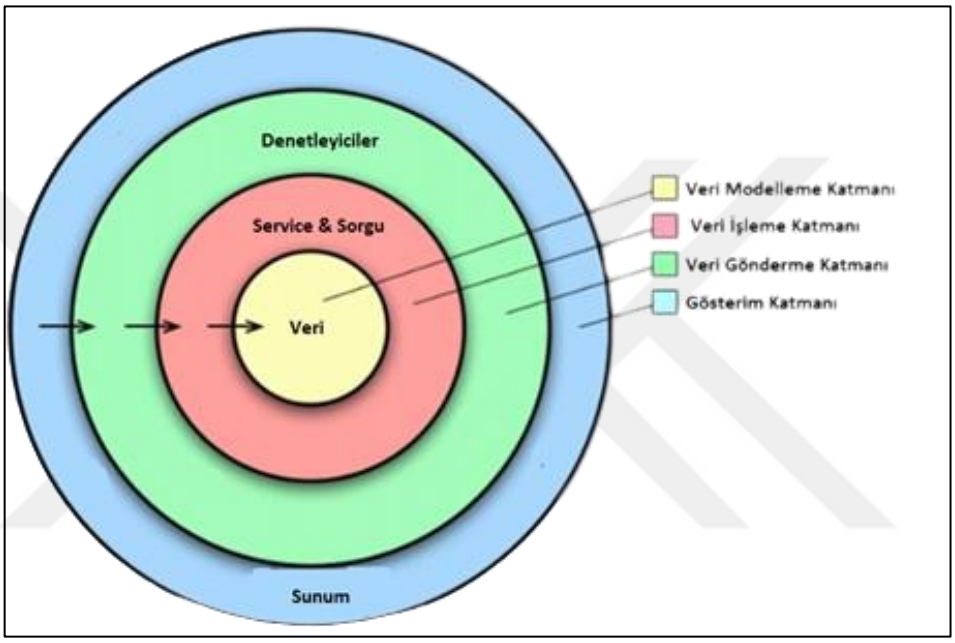
TeleNöroRehab uygulmasının kullanıcı arayüzleri Angular ile SPA alt yapısı kullanılarak HTML, CSS, Javascript ve TypeScript yazılım dillerini ile dinamik ve sürdürülebilir şekilde geliştirilmiştir.

Projenin sunucu tarafında çalışan algoritmik program, günümüzde yaygın olarak kullanılan teknolojilerin güncel standartlarına uygun şekilde katmanlı soğan mimarisi üzerinden geliştirilmiştir. Bu program geliştirilirken kullanılan teknolojiler şunlardır;

3.1.16. Katmanlı Soğan Mimarisi

Katmanlı soğan mimarisi, yazılım uygulamalarının katmanlar halinde ayrılması ve her katmanın diğerine bağımlılığının minimize edilmesi esasına dayanan bir yazılım mimarisi yaklaşımıdır. Bu yaklaşım, uygulamaların daha modüler, ölçeklenebilir, esnek ve bakımı kolay hale getirilmesini amaçlar. Katmanlı soğan mimarisi, bir uygulamanın genellikle en üst katmandan en alt katmana doğru sıralanmasıyla oluşur. En üst katman genellikle kullanıcı arayüzüdür. Bu katman, kullanıcı girişi ve çıkışıyla ilgili işlemleri yapar ve diğer katmanlarla iletişim kurar. Alt katmanlar, iş mantığını uygulamak ve veri erişimi işlemlerini

gerçekleştirmekle görevlidir. İş mantığı katmanı, uygulamanın işlevselliğiyle ilgili işlemleri gerçekleştirir. Veri erişimi katmanı, veritabanı veya diğer veri kaynaklarıyla iletişim kurar ve verilerin saklanması, okunması ve güncellenmesi işlemlerini gerçekleştirir. Katmanlar arasındaki bağımlılıklar, her bir katmanın yalnızca doğrudan alt katmanla iletişim kurmasıyla sınırlandırılır. Bu, katmanların birbirlerinden bağımsız hale getirilmesini ve bir katmanın diğerine bağımlı olmamasını sağlar. Bu yaklaşım, kodun bakımını ve test edilmesini kolaylaştırır ve kodun daha okunaklı ve anlaşılır hale gelmesine yardımcı olur.



Şekil 3.1. Katmanlı Soğan Mimarisi katman tanımı

Katmanlı soğan mimarisi genellikle aşağıdaki katmanlardan oluşur:

Kullanıcı arayüzü katmanı: Bu katman, kullanıcının uygulama ile etkileşim kurduğu arayüzü oluşturur. Bu katman, kullanıcının girdiği verileri işlemek ve uygulamanın çıktısını göstermek için diğer katmanlarla iletişim kurar. Web uygulamalarında, bu katman genellikle sunucu tarafı web teknolojileri kullanılarak oluşturulur.

Uygulama mantığı katmanı: Bu katman, uygulamanın işlevselliğini sağlar. Bu katman, kullanıcı arayüzünden gelen verileri işler, veritabanından veri alır veya veritabanına veri yazar. Uygulama mantığı katmanı, iş mantığı kurallarını uygulamak için kullanılır.

Veri erişim katmanı: Bu katman, veritabanı veya diğer veri kaynaklarıyla iletişim kurar. Bu katman, veri okuma, ekleme, güncelleme ve silme işlemlerini gerçekleştirir. Veri erişim katmanı, veritabanı bağlantılarını yönetir ve veritabanı bağımsızlığını sağlar.

Altyapı katmanı: Bu katman, uygulamanın alt seviye bileşenlerini oluşturur. Bu katman, uygulamanın konfigürasyonunu yönetir, günlük kayıtlarını tutar, hata yönetimini sağlar ve diğer alt seviye işlemleri gerçekleştirir.

Katmanlı soğan mimarisi, farklı projelerde farklı şekillerde uygulanabilir. Bazı projelerde, katmanlar farklı şekillerde isimlendirilir veya daha fazla katman eklenebilir. Ancak, genel olarak, katmanlı soğan mimarisi, uygulamanın bölünmesi ve her katmanın yalnızca kendinden önceki katmanla iletişim kurması gerektiği prensibine dayanır.

3.1.17. .NET Core (.NET 5)

.NET Core, Microsoft tarafından geliştirilen, açık kaynaklı, ücretsiz, platform bağımsız bir yazılım çatısıdır. .NET Core, .NET Framework'ten farklı olarak, Windows, Linux ve macOS gibi farklı işletim sistemlerinde çalışabilen uygulamaların geliştirilmesine olanak tanır. .NET Core, C#, F# ve Visual Basic gibi popüler programlama dillerini destekler ve .NET Standard'a uygun .NET kitaplıklarını kullanabilir. Bu, mevcut .NET kodlarının .NET Core ile uyumlu hale getirilebileceği anlamına gelir. Ayrıca, .NET Core, ASP.NET Core adı verilen bir web uygulama geliştirme çerçevesiyle birlikte gelir.

3.1.18. SQL

SQL, Structured Query Language (Yapılandırılmış Sorgu Dili) kısaltmasıdır. Veritabanı yönetim sistemleri (RDBMS) ile iletişim kurmak için kullanılan bir programlama dilidir. SQL, verilerin tutulduğu ve yönetildiği bir veritabanında, veri ekleme, silme, güncelleme ve sorgulama işlemleri yapmak için kullanılır. SQL, sorguların hızlı ve etkili bir şekilde yapılmasını sağlar. Sorgular, veritabanındaki verileri seçmek, filtrelemek, sıralamak ve gruplamak gibi işlemleri yapabilir. Ayrıca, SQL, veriler arasındaki ilişkileri yönetmek, veritabanı yapısını oluşturmak, veri bütünlüğünü sağlamak ve güvenliği sağlamak gibi diğer veritabanı yönetimi işlemlerini de yapabilir.

Proje içerisinde verilerin depolanması gereken veri tabanı olarak yine Microsoft ürünü olan MsSQL veri tabanı kullanılarak depolama işlemleri gerçekleştirilmektedir.

3.1.19. ORM

ORM, Object-Relational Mapping (Nesne İlişkisel Eşleme) kısaltmasıdır. Bir yazılım uygulamasında nesne yönelimli programlama (OOP) ile ilişkisel veritabanı arasındaki köprüyü sağlayan bir teknolojidir. ORM, veritabanındaki tabloları nesnelere ve sınıflara dönüştürür ve bu nesneler üzerinde nesne yönelimli programlama teknikleri kullanarak veritabanı işlemlerini gerçekleştirmeyi mümkün kılar.

3.1.20. Entity Framework

Entity Framework, Microsoft tarafından geliştirilen bir nesne yönelimli programlama (OOP) ve veri erişim teknolojisidir. Entity Framework, .NET Framework ile birlikte sunulur ve .NET Core'da da kullanılabilir. Bu teknoloji, veritabanına erişim işlemlerini nesne yönelimli bir yapıyla gerçekleştirmeyi sağlar. Entity Framework, veritabanındaki tabloları ve ilişkileri nesnelere ve sınıflara dönüştürür. Bu, geliştiricilerin veri erişimi işlemlerini nesne yönelimli bir yapıyla gerçekleştirmelerine olanak tanır. Ayrıca, Entity Framework, veritabanı işlemlerinde kullanılan SQL sorgularını otomatik olarak oluşturur ve veri işlemlerini optimize eder. Ayrıca veritabanı işlemlerinin yanı sıra, veri doğrulama ve veri bütünlüğü gibi diğer veri yönetimi işlemlerini de yapabilir. Bu teknoloji, LINQ (Language Integrated Query) sorgularını destekler ve geliştiricilerin LINQ sorgularını kullanarak veri sorgulama işlemlerini gerçekleştirmelerine olanak tanır.

Örnek Entity Framework .NET (C#) Kodu

```
var conversation = await _baseRepository
    .GetQueryable<Conversation>()
    .Join(_baseRepository.GetQueryable<ConversationParticipant>(), i => i.Id, j => j.ConversationId,
        (i, j) => new
        {
            Conversation = i,
            ConversationParticipant = j
        })
    .Where(i => i.Conversation.Id == model.ConversationId && i.ConversationParticipant.UserId == model.UserId)
    .Select(i => i.Conversation)
    .FirstOrDefaultAsync();
```

SQL Dönüşmüş Hali

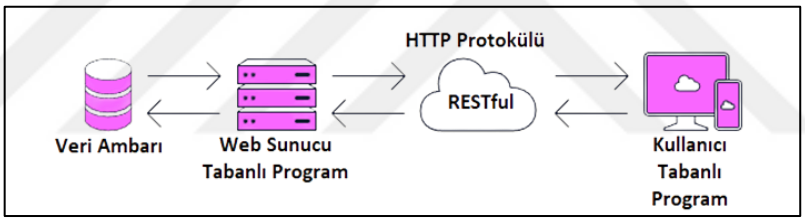
```
SELECT TOP (1) C.*
FROM Conversation AS "C"
INNER JOIN ConversationParticipant "CP" ON C.Id = CP.ConversationId
WHERE C.Id = 1 AND CP.UserId = 1
```

Şekil 3.2. Entity Framework çerçevesi için bir örnek .NET Core kodu ve SQL çıktısı

Şekil 3.1 de gösterildiği gibi .NET ortamından C# programla dilinin sunduğu LINQ ve Lambda Expression ile yazılan bir kodun SQL diline dönüşmüş hali gösterilmektedir. Entity Framework çok güçlü bir yardımcı ORM geliştirme ortamı kütüphanesidir. Büyük projelerde bu kütüphane kullanarak daha hızlı ve optimize veri işlemeler oldukça kısa zamanda geliştirilebilir. Tez kapsamında geliştirilen TeleNöroRehab'ın neredeyse tüm veri sorgu işlemleri bu ortam üzerine inşa edilerek hızlı ve optimize hale getirilmiştir.

Geliştirilen projedeki front-end (önyüz) ile back-end (arkayüz) arasında veri haberleşmesi iki farklı yöntem kullanılmıştır. Veri haberleşmesinden biri İstek-Cevap mimarisine dayanan HTTP yöntemlerine ile sağlanırken diğer ise soket mimarisi ile anlık veri yolu kurulması ile sağlanmaktadır.

TeleNöroRehab, sunucu tarafından çalışan uygulama ile kullanıcı tarayıcısı tarafında çalışan uygulama RESTful API üzerine inşa edilmiş JSON veri formatı ile istek-cevap mimarisine sahiptir.



Şekil 3.3. RESTful çalışma prensibi

3.1.21. HTTP

HTTP (Hypertext Transfer Protocol), web sayfalarının ve diğer internet kaynaklarının aktarılması için kullanılan bir protokoldür. HTTP, bir istemci-sunucu modeline dayanır, yani bir istemci (genellikle bir web tarayıcısı) bir sunucudan kaynak talep eder ve sunucu istemi yanıtlar. TCP/IP (Transmission Control Protocol/Internet Protocol) protokol ailesinde yer alan genellikle TCP bağlantıları kullanır. İstemci, bir sunucudan bir kaynağı almak için bir HTTP isteği gönderir ve sunucu, isteği alır, isteği işler ve istemciye bir yanıt gönderir.

HTTP, metin, resim, video, ses ve diğer verileri aktarmak için kullanılır. Web sayfalarının yanı sıra, web tabanlı uygulamalar, RESTful servisler, API'ler ve diğer internet tabanlı hizmetler de HTTP üzerinden çalışır. HTTP, belirli bir format ve sözdizimine sahip belirli bir dizi mesaj türü kullanır. Örneğin, bir HTTP isteği bir başlık satırı, bir istek satırı ve isteğin

gövdesi olmak üzere üç ana bölümden oluşur. Benzer şekilde, bir HTTP yanıtı da bir başlık satırı, bir durum satırı ve yanıt gövdesi içerir.

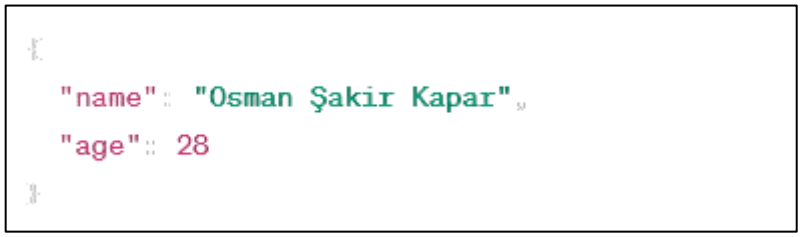
3.1.22. RESTful API

RESTful API (Representational State Transfer Application Programming Interface), bir yazılım arayüzüdür ve web servislerini tasarlamak için kullanılan bir web uygulama programlama arayüzüdür. REST (Representational State Transfer), web hizmetleri arasında veri alışverişi yapmak için bir mimaridir ve bu mimari, web hizmetleri arasında veri alışverişini kolaylaştırır. Bir sunucu ile istemciler arasındaki veri iletişimini yönetir ve istemciler, bir RESTful API aracılığıyla sunucudan kaynakları (verileri) talep eder veya kaynakları sunucuya gönderir. RESTful API, HTTP protokolü üzerinde çalışır ve HTTP methodlarını kullanarak kaynakları işler.

RESTful API, kaynakları benzersiz bir URI (Uniform Resource Identifier) adresiyle tanımlar. URI adresleri, sunucuda kaynakları belirtir ve istemciler, bu adresleri kullanarak sunucudan kaynakları talep eder. RESTful API, HTTP methodlarını kullanarak kaynaklar üzerinde işlemler gerçekleştirir. Örneğin, bir istemci, HTTP GET methodunu kullanarak sunucudan bir kaynağı alabilir, HTTP POST methodunu kullanarak sunucuya yeni bir kaynak ekleyebilir veya HTTP DELETE methodunu kullanarak bir kaynağı silebilir.

3.1.23. JSON

JSON (JavaScript Object Notation), insanlar ve bilgisayarlar arasında veri alışverişini kolaylaştıran bir veri biçimidir. JSON, verileri açıklayan bir metin formatıdır ve genellikle web uygulamalarında ve API'lerde kullanılır. JSON, birkaç temel veri türünü destekler, bunlar arasında metin (string), sayılar (numbers), boolean, nesneler (objects), diziler (arrays) ve null bulunur. JSON, XML gibi diğer veri biçimlerine göre daha az karmaşıktır ve daha okunaklıdır. JSON, verileri anahtar-değer çiftleri şeklinde depolar. Anahtar-değer çiftleri, JSON'da "anahtar": "değer" şeklinde ifade edilir.



Şekil 3.4. Örnek bir JSON gösterimi

JSON, web uygulamaları ve API'ler tarafından sıklıkla kullanılır, çünkü JSON verileri, farklı dillerde yazılmış uygulamalar arasında kolayca paylaşılabilir. Ayrıca, JSON verileri, web tarayıcıları tarafından da kolayca işlenebilir, bu nedenle dinamik web sayfaları oluşturmak için de kullanılır.

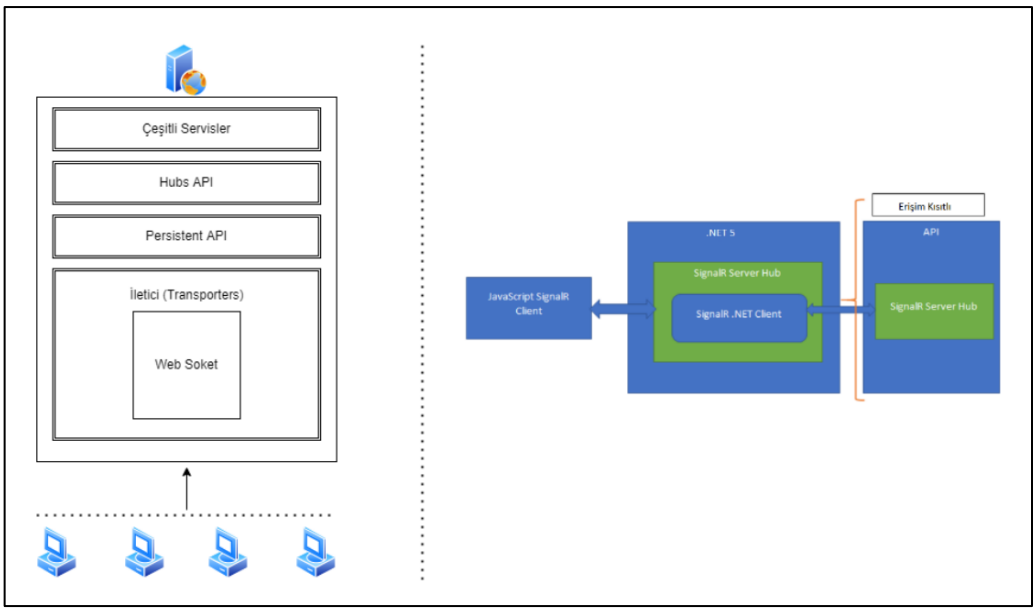
TeleNöroRehab uygulama içerisinde yer alan karşılıklı anlık konuşma uygulaması TCP protokülü üzerine inşa edilmiş Web Soket üzerinden anlık veri haberleşmesi sağlanmaktadır. .NET üzerinden SignalR kütüphanesi ile Web Soket mimarisi oluşturulmuştur.

3.1.24. Web Soket

Web Soket (Web Socket), iki yönlü iletişim sağlayan bir iletişim protokolüdür. Web Soket, web sayfalarının gerçek zamanlı ve sürekli olarak güncellenen verileri almasına ve göndermesine olanak tanır. Web Soket protokolü, bir web sunucusu ve bir istemci arasında bir bağlantı kurar. Bu bağlantı, sürekli açık kalır ve web sayfası tarafından kullanılabilecek bir "soket" sağlar. Bu soket, iki yönlü iletişim için kullanılabilir, yani sunucu, verileri web sayfasına gönderebilir ve web sayfası, sunucuya veri gönderebilir.

3.1.25. SignalR

SignalR, Microsoft tarafından geliştirilmiş bir .NET kütüphanesidir. SignalR, gerçek zamanlı web uygulamaları için kullanılır ve Web soket protokolünün yanı sıra, diğer mekanizmalarla da uyumlu olarak çalışabilir.

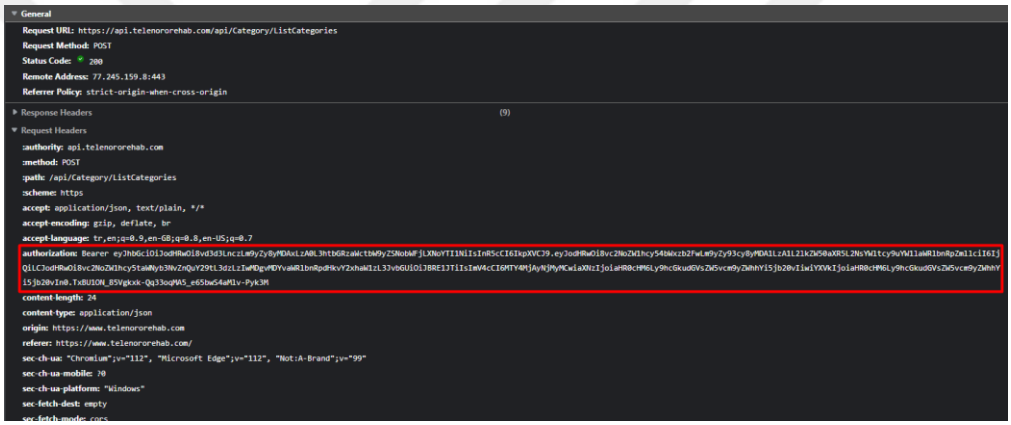


Şekil 3.5. SignalR çalışma prensibi

Veri haberleşme teknolojileri günümüzde hızlı bir şekilde gelişip kullanıcılara daha optimize ve güvenlikli ortam sunmayı amaçlamaktadır. TeleNöroRehab projesinde kullanıcıların verilerinin güvenliğini sağlamak amacı ile hem RESTful servislerinde hem de SignalR ile sunulan web sokette kimlik doğrulamak için "Bearer Authentication" ve "JWT" ile sağlanmaktadır.

3.1.26. Bearer Authentication

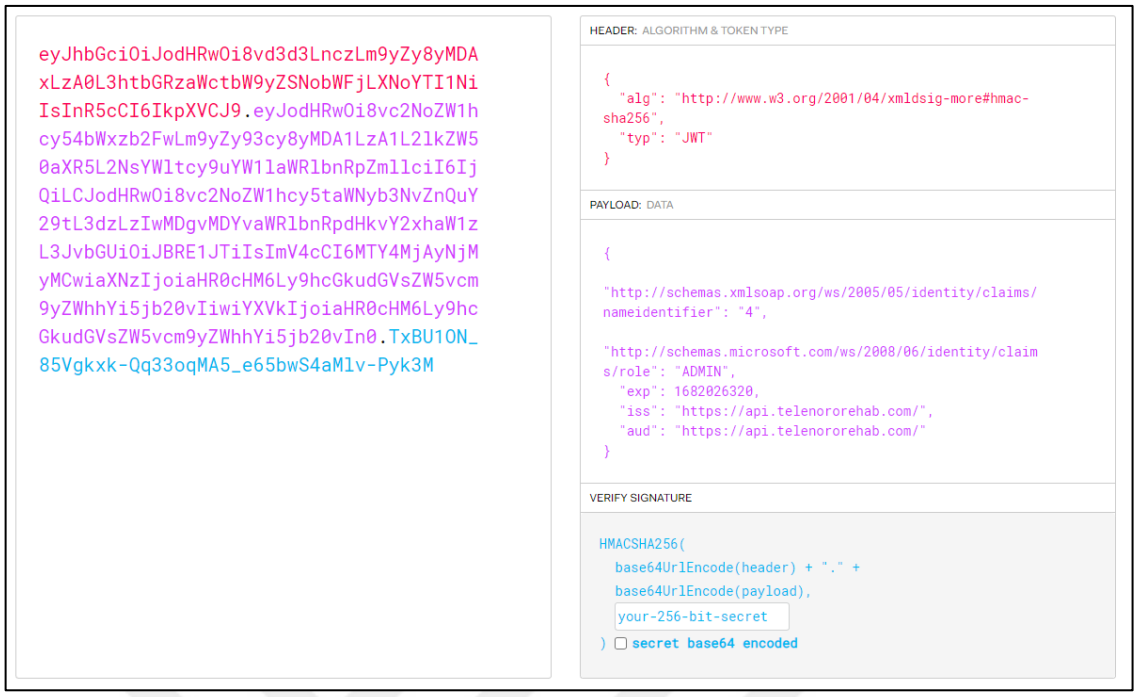
Bearer kimlik doğrulaması (ya da token bazlı doğrulama), Bearer olarak adlandırılan kimlik doğrulama alanları içeren bir HTTP kimlik doğrulama şemasıdır. Belirli bir kaynağa veya URL'ye erişime izin veren kimlikler genellikle bir oturum açma isteğine yanıt olarak sunucu tarafından oluşturulan şifreli bir dize veridir.



Şekil 3.6. Bearer Authentication kullanılan bir HTTP method Header örneği

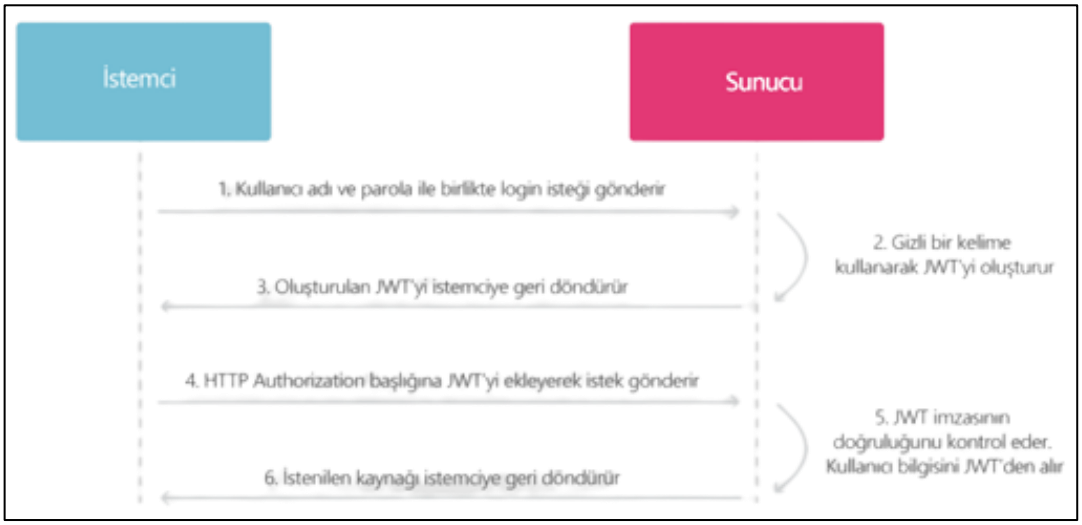
3.1.27. JWT Token

JWT (JSON Web Token), web uygulamaları arasında güvenli veri alışverişi sağlamak için kullanılan bir kimlik doğrulama ve yetkilendirme mekanizmasıdır. JWT, kullanıcının kimliği ve belirli bir web sitesindeki veya uygulamadaki ayrıcalıkları gibi bilgileri içeren bir JSON nesnesi içerir. JWT, üç bölümden oluşur: başlık (header), pay yükü (payload) ve imza (signature). Başlık, JWT'nin kendisi hakkında bilgi sağlayan bir JSON nesnesidir. Pay yükü, JWT'nin taşıdığı bilgileri içerir. İmza, JWT'nin doğrulanabilirliğini sağlamak için kullanılır.



Şekil 3.7. JWT token ve içerisinde yer alan veri gösterimi

JWT, bir web uygulamasında kullanıcı oturum açma işlemi için kullanılabilir. Kullanıcı başarılı bir şekilde kimlik doğrulama işleminden geçtikten sonra, sunucu JWT oluşturur ve bu JWT'yi kullanıcıya verir. Kullanıcı, bu JWT'yi kullanarak uygulama içindeki işlemlerini yapabilir.



Şekil 3.8. JWT çalışma akışı gösterimi

JWT'nin en büyük avantajlarından biri, sunucunun oturum bilgilerini saklamak zorunda kalmamasıdır. Bu, sunucunun ölçeklenebilirliğini artırır ve daha iyi bir performans sağlar.

Ayrıca, JWT'lerin şifrelenmesi ve doğrulanması için birçok kütüphane ve araç mevcuttur, bu nedenle uygulama geliştiricilerinin bunları kullanması daha kolaydır.

3.1.28. Bootstrap

Bootstrap, web uygulamaları geliştirmek için kullanılan açık kaynaklı bir CSS ve JavaScript kütüphanesidir. Twitter tarafından geliştirilmiştir ve ilk olarak 2011 yılında piyasaya sürülmüştür. Bootstrap, web siteleri veya uygulamaları geliştirmek için hızlı ve kolay bir yöntem sunar. Kütüphane, önceden tasarlanmış bileşenleri ve stilleri içerir, böylece geliştiriciler, web sayfalarını veya uygulamaları oluşturmak için bu bileşenleri kullanabilirler. Bootstrap, web sayfalarının duyarlı tasarımını destekler, bu nedenle sayfalar farklı cihazlarda (masaüstü, tablet, mobil) görüntülendiğinde uygun şekilde ölçeklenebilir.

3.1.29. ImageSharp

ImageSharp, .NET platformu üzerinde çalışan ve görüntü işleme işlevselliği sağlayan açık kaynaklı bir kütüphanedir. Bu kütüphane, GDI+ gibi eski teknolojiler yerine daha modern bir görüntü işleme motoru kullanır. Böylece daha yüksek performans, daha iyi ölçeklenebilirlik ve daha yüksek kaliteli görüntü işleme işlevselliği sunar.

3.2. Entegrasyon ve Mimari

TeleNöroRehab, Bölüm 3.1'de belirtilen teknolojileri ve araçları kullanarak 3 farklı web yazılım uygulaması ortaya çıkartmaktadır, bu uygulamalar şunlardır:

1. Kurumsal Web Sitesi
2. Kullanıcı Arayüz Web Uygulaması
3. Sunucu Tarafında Çalışan Algoritmik Program (Web Servis)

3.2.1. Kurumsal Web Sitesi

Kurumsal web sitesi, bir şirketin ya da projenin dijital varlığı ve çevrimiçi varoluşudur. Bu nedenle, günümüzün dijital çağında bir kurumsal web sitesi, projenin başarısı için hayati önem taşımaktadır. Kurumsal web sitesi, TeleNöroRehab'ın kimliğini ve imajını yansıtır. Projenin paydaşları, hizmetleri, proje hakkında bilgilendirme, önceki çalışmalar ve iletişim hakkında bilgi verir. Bu nedenle, web sitesi, kullanıcılar için önemli bir ilk izlenim oluşturur.

TeleNöroRehab projesinin web sitesi, internet üzerinden hizmet veren web tabanlı statik bir web sitesidir. Bu statik web sitesi Grunt ve Node.js teknolojilerini kullanarak HTML, CSS ve

JavaScript araçları ile geliştirilmiş olup HTML dosyaları olarak çıktı alınıp IIS üzerinde internete sunulmuştur.

Bu önyüz uygulaması “package.json” üzerinden NPM paketlerinin yüklendiği bir Node.js alt yapısını kullanan statik bir uygulamadır.



```
{
  "name": "Tele-Neuro",
  "version": "1.0.0",
  "description": "Tele Neuro Home Web Page",
  "devDependencies": {
    "grunt": "^1.4.1",
    "grunt-bake": "^1.9.0",
    "grunt-contrib-clean": "^2.0.0",
    "grunt-contrib-copy": "^1.0.0",
    "grunt-contrib-watch": "^1.1.0"
  }
}
```

Şekil 3.9. Web site NPM paketleri için kullanılan package.json

Şekil 3.9. da görüldüğü gibi NPM paketlerinden Grunt adlı JavaScript görev çalıştırısınını ve ona alt yapısını kullanan yardımcı eklentileri yüklenerek geliştirme sırasında yardımcı araçlar ile daha hızlı ve optimize geliştirmeye olmuştur.

Grunt eklentilerinden;

Bake: Bileşen oluşturma ve statik HTML web siteleri için daha hızlı ve verimli oluşturulmasını sağlar.

- Clean: Belirli dosyaları yada klasörleri temizlemek amacı ile kullanılır.
- Copy: Belirli dosyaları yada klasörleri belirli konumlara kopyalama amacı ile kullanılır.
- Watch: Dosya değişikliğini otomatik algılayıp belirli işlemleri tetikleme için kullanılır.

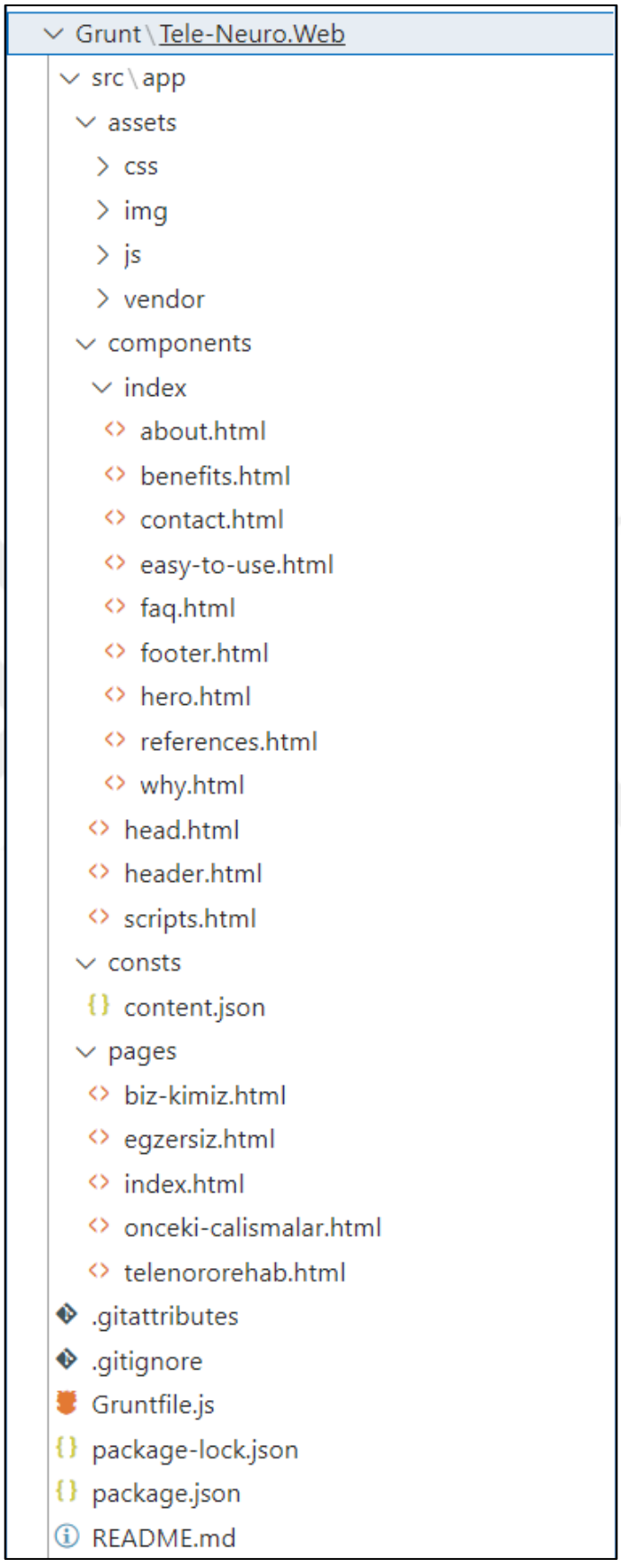
Grunt görev çalıştırıcısının ekletisi olan “grunt-bake” ile projeyi küçük parçalara bölerek tekrar kullanabilir bileşenler oluşturarak geliştirilmiştir. Şekil 3.11. bileşenler ve onları üstü olan sayfaları ve projenin genel olarak dosya mimarisini göstermektedir. Şekil 3.11. görüldüğü gibi “components” klasörünün içerisinde belirli görevlere göre bölünmüş

bileşenleri “grunt-bake” kullanarak üst sayfa olan “page”ler içerisinde bileşen olarak kullanılması mümkün kılınmıştır.

Şekil 3.10. bir üst sayfa içerisinde “grunt-bake” yardımcı aracını kullanarak alt bileşenlerin belirli bir işaretleme yardımı ile kullanımı gösterilmiştir.

```
<!DOCTYPE html>
<html lang="en">

<head>
    <!--(bake ../components/head.html)-->
</head>

<body>

    <!--(bake ../components/header.html)-->
    <!--(bake ../components/index/hero.html)-->

    <main id="main">
        <!--(bake ../components/index/about.html)-->

        <!--(bake ../components/index/easy-to-use.html)-->

        <!--(bake ../components/index/benefits.html)-->

        <!--(bake ../components/index/contact.html)-->
    </main>

    <!--(bake ../components/index/footer.html)-->

    <a href="#" class="back-to-top d-flex align-items-center justify-content-center">
        <i class="bi bi-arrow-up-short"></i>
    </a>

    <!--(bake ../components/scripts.html)-->

</body>

</html>
```

Şekil 3.10. Örnek Grunt Bake kullanımı

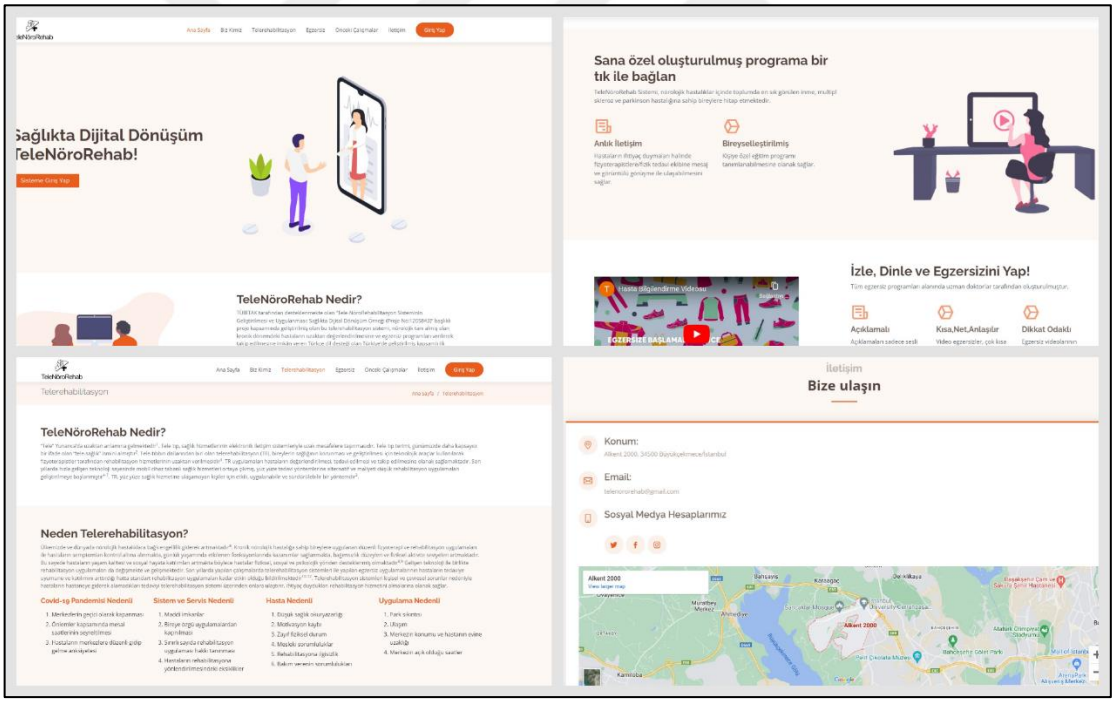


Şekil 3.11. Web site proje dosyalama gösterimi

Şekil 3.11. de gösterilen dosya mimarisi içerisinde yer alan klasörler;

- Assets: Genel resim, CSS, JavaScript, Video ya da müzik gibi kullanılan içerikleri saklar
- Components: “grunt-bake” ile oluşturulan alt bileşenleri içerir.
- Consts: Proje içerisinde dil dosyaları, ayar dosyaları gibi sabit içerikleri tutar.
- Pages: Alt bileşenleri kullanarak oluşturulan üst sayfaların içeriklerini kapsar.
- Package.json: NPM ve Node.js yönetim dosyası
- Gruntfile.js: Grunt görev çalıştırıcısının ayarlarını kapsar. (EK-1 detaylı kaynak kodları)

Web Sitesi, TeleNöroRehab projesinin kullanıcı ilk karşılayan ve bilgilendiren bölümüdür. Statik bir web sitesi olduğu için çok fazla değişikliğe gerek duymaz. Şekil 3.12. de birkaç ekran görüntüsü resmedilmektedir.



Şekil 3.12. Web sitesi için ekran görüntüleri

3.2.2. Kullanıcı Arayüzü Web Uygulaması

Kullanıcı arayüzü TeleNöruRehab’ı hasta ve uzman kullanıcılar için hazırlanmış ana uygulamasıdır. Bu uygulama üzerinden hastalar, uzman tarafından hazırlanan uzaktan tele-rehabilitasyon sistemi almaları için birçok özellik sunar. Hastalar, tele olarak rehabilitasyon hizmeti alabilir, uzman fizyoterapistler ile mesaj ya da görüntülü olarak iletişime geçebilir,

kendilerine özgü hazırlanmış video ya da broşür tabanlı eğitimler topluluğu ile tedavi hizmeti alabilir. Uzman kullanıcılar ise çeşitli hastalıklara yönelik video ya da broşür yükleyebilir, hastalara özgü programlar oluşturabilir, hastalar ile bire bir iletişim geçip hastaların etkinlikleri ile ilgili verilere erişebilir, analiz edebilir, raporlamalar yapabilirler.

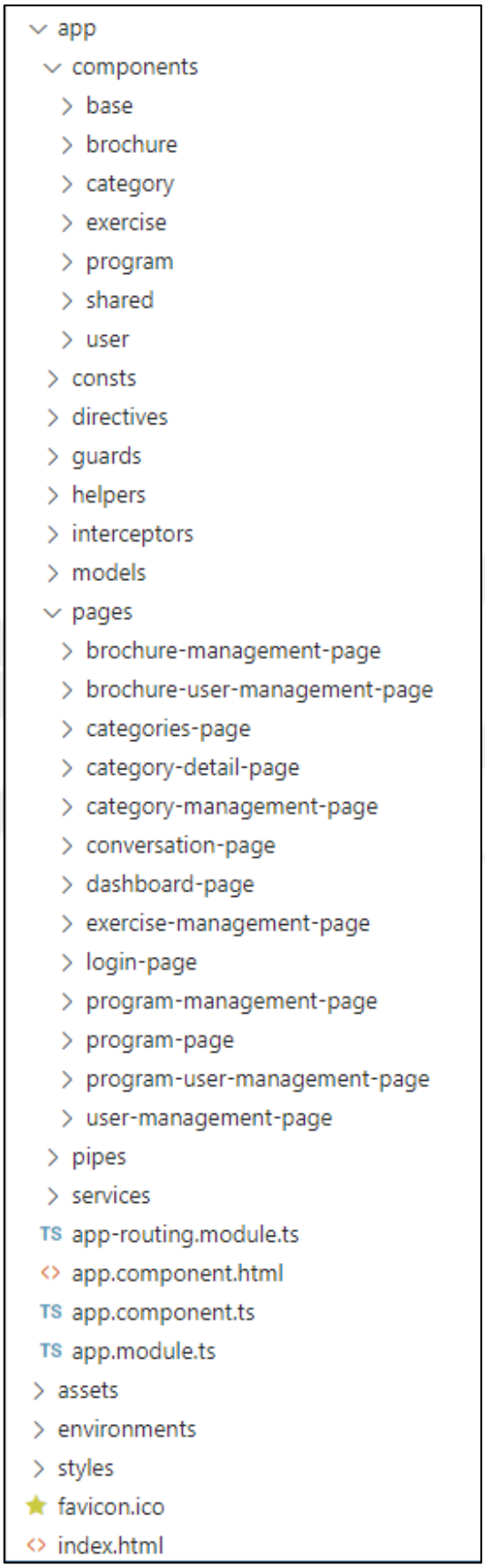
Bu uygulama SPA mimari altyapısında Angular yazılım çerçevesinde HTML, CSS, JavaScript, TypeScript ve diğer yardımcı teknolojiler kullanılarak geliştirilmiştir. Angular, bileşen tabanlı bir web uygulama çerçevesi olduğundan TeleNöroRehab üzerinden birçok bölüm bileşen haline getirilip tekrar kullanabilir hale getirilip kod tekrarlarından uzak durularak geliştirilmiştir.

Şekil 3.13. kullanıcıları karşılayan kullanıcı arayüz uygulamasının Angular tabanlı dosya mimarisi özet olarak gösterilmektedir. Dosyalama stratejileri bir projenin sürdürülebilirliğini ve geliştirilebilişini önemli ölçüde etkiler.

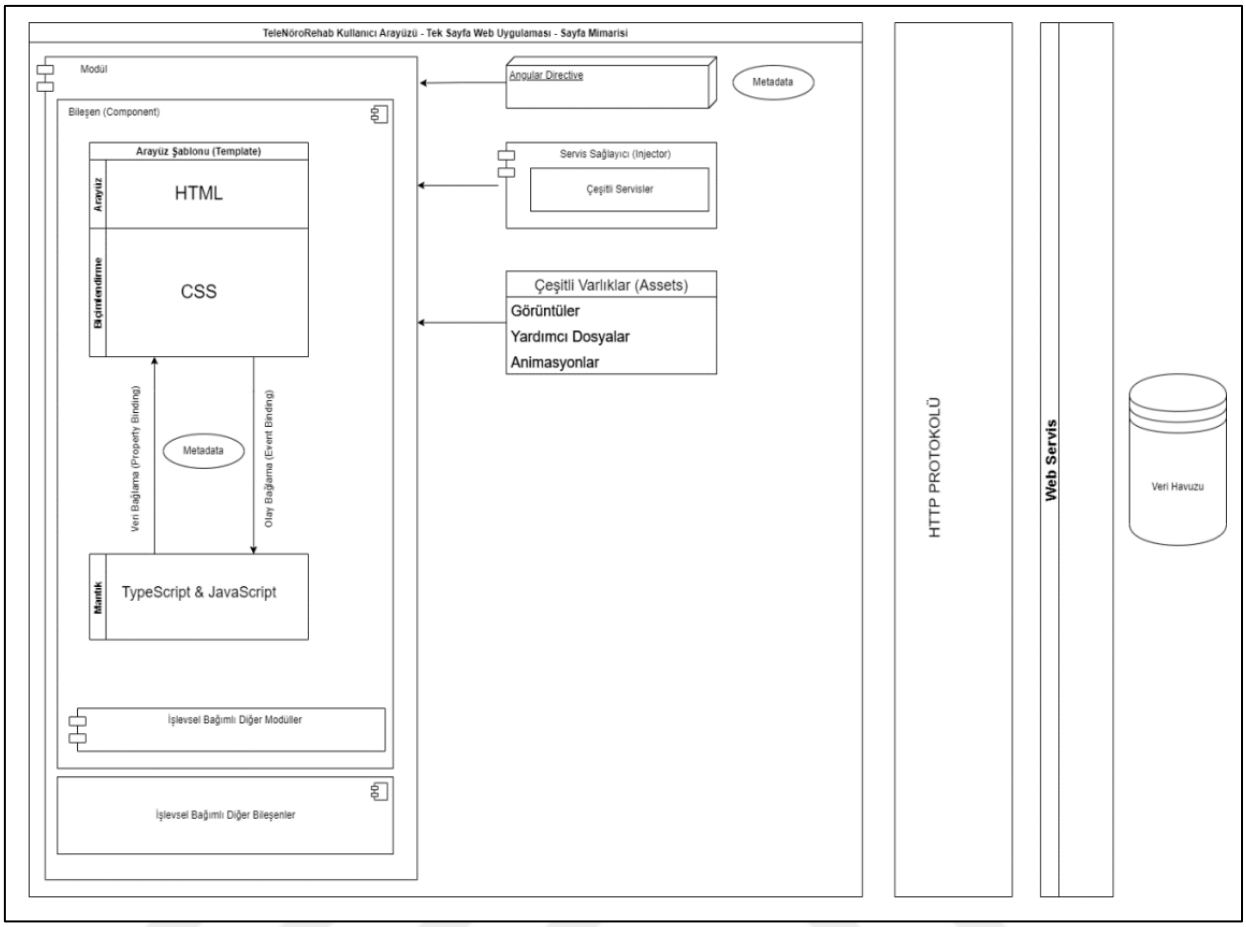
Bu dosyala stratejisindeki;

- Components: Bir sayfayı oluşturan bileşenleri sayfa bazında ya da paylaşımlı bazında dosyalar.
- Consts: Uygulama kullanılan sabit ve değişmeyen değerleri tutar.
- Directives: Angular Directive, Angular bileşenlerine ve öğelerine ek özellikler kazandırmak için kullanılan önemli bir Angular özelliğidir ve bu dosya Directive'leri bir arada dosyalayıp modüler halen getiren bölümdür. (EK-2 de bir Angular Directive'nin TeleNöroRehab projesinde kullanım örneği gösterilmektedir.)
- Guards: TeleNöroRehab projesinde ki sayfaların yada bileşenlerin güvenlik, kimlik doğrulama ve yetkinlendirme işlemlerinin gerçekleştirilmesine yardımcı olan Angular çerçevesi altında ki Guard'ları tutar.
- Helpers: Yardımcı modül, sınıf ve kaynak kodları dosyalamak için kullanılır.
- Interceptors: Bir Interceptor HTTP isteklerini ele almak ve işlemek için kullanılır. TeleNöroRehab projesinde kullanılan JWT Token ile kimlik doğrulamaya yardımcı olması için kullanılır. Bu dosya bölümünde HTTP Header bölümüne token ekleyen ve yetkisiz erişim ya da hatalı kimlik doğrulama durumunda hata ve giriş sayfasına yönlendiren Interceotor'leri tutar.

- Models: Uygulama üzerinden verileri modellemek ya da web servis üzerinden gelen verileri haritalayıp modelleyen veri modellerini saklar.
- Pages: Angular bileşenlerini kullanarak oluşturulmuş web sayfalarının Angular şablonlarını saklar.
- Pipes: Angular Pipe, verilerini yönetmenin ve görüntülemenin birçok yönünü kontrol edilmesini sağlar. Pipes'lar, verileri düzenlemek, sıralamak, filtrelemek, biçimlendirmek, birleştirmek, küçültmek, büyütmek gibi birçok işlemi gerçekleştirebilirler.
- Services: Bu bölüm Angular yazılım çerçevesi üzerinden sunucu tarafında çalışan program ile HTTP servisleri üzerinden veri alışverişini sağlayan servisleri barındırır.
- Assets: Uygulamada kullanılan içeriklerin saklandığı ve sunulduğu bölümdür.
- Environments: Uygulama kullanılan uygulamanın yayınladığı çerçevelere göre değişkenlik gösteren verileri kapsar.
- Styles: Uygulamada kullanılan SASS, SCSS ve CSS kodlarını saklar.



Şekil 3.13. TeleNöroRehab Angular projesi için dosyalama gösterimi



Şekil 3.14. Bir arayüz sayfasının çalışma prensibi

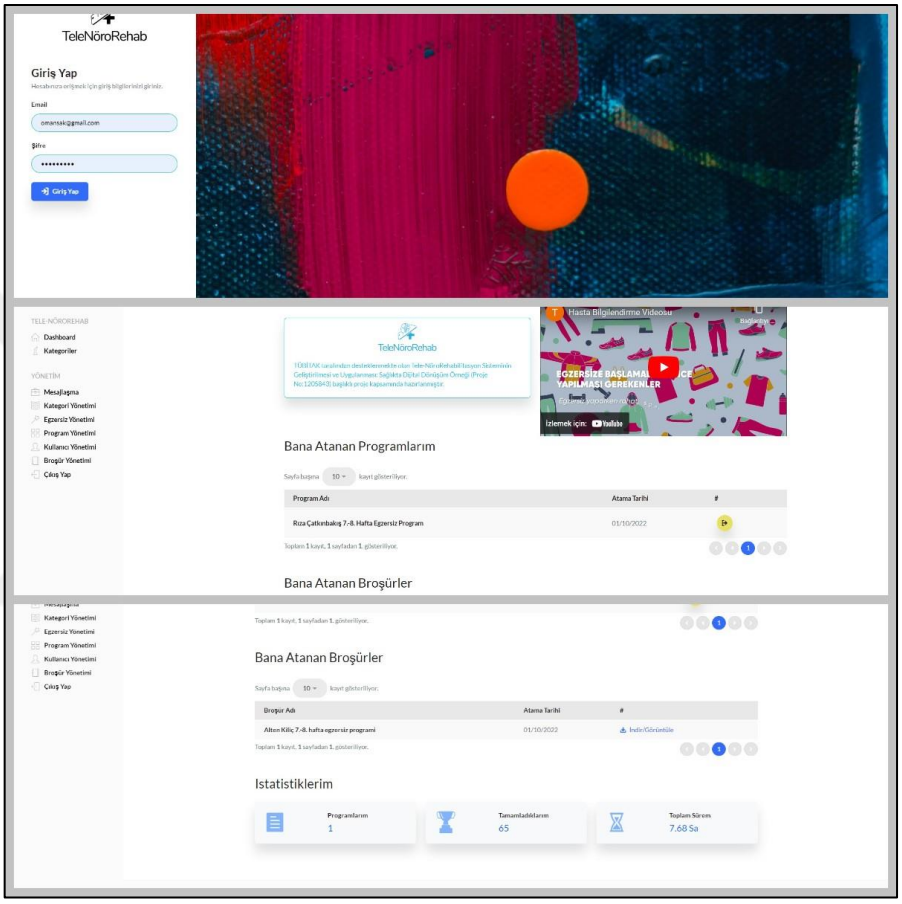
Şekil 3.14. gösterildiği gibi, TeleNöroRehab üzerindeki bir sayfa Angular ile SPA mimarisi üzerine kuruludur. Bu, uygulamanın tüm sayfalarının tek bir HTML sayfasında yüklendiği anlamına gelir ve bu sayede kullanıcı arayüzünde daha hızlı ve etkili bir deneyim sunulur. Angular uygulamaları, modüler bir yapıya sahiptir ve uygulamanızı küçük ve yeniden kullanılabilir parçalara ayırmanıza olanak sağlar. Bu, uygulamanızın yönetimi ve bakımı için daha kolay ve daha etkili bir yol sağlar. Angular oluşturulan bileşenler, modüller ve sayfalar Router aracılığıyla yönlendirilir ve sayfalar arasında geçiş yapılmasına olanak tanır.

Kullanıcı arayüzü bileşen tabanlı bir mimariye sahip olup her sayfa veya bileşen, kendi HTML şablonunu, CSS dosyasını ve TypeScript kodunu içerir. Bileşenler, Angular Directive'leri, Pipes, Services ve diğer öğeleri kullanarak birbirleriyle iletişim kurulmasına olanak.

Kullanıcı arayüzü, verileri HTTP istekleri aracılığıyla sunucu üzerinden çalışan programdan alır ve verileri RESTful API'ler kullanarak yönetir. Veriler, bileşenler arasında paylaşılabilir

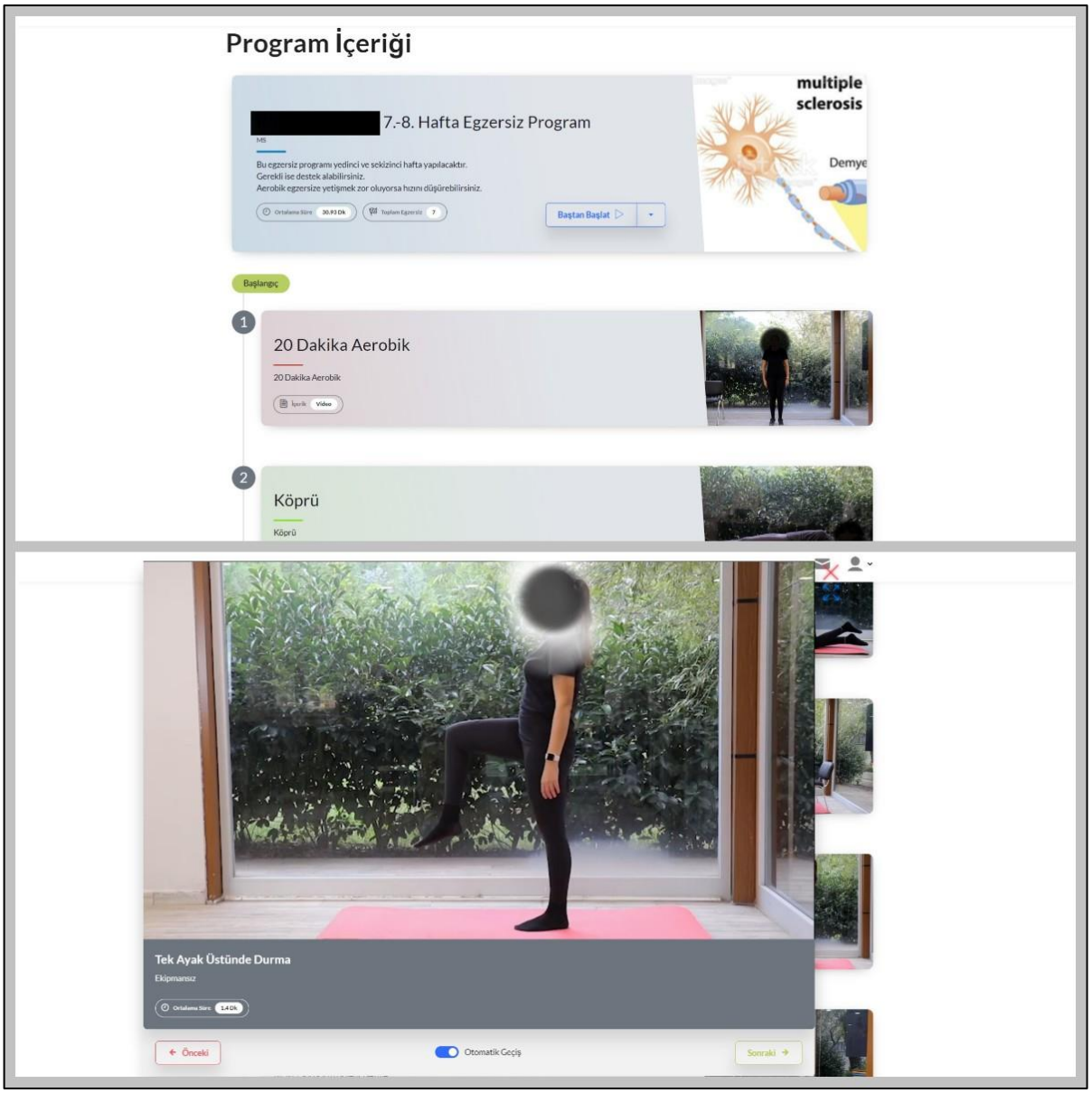
ve Service'ler aracılığıyla yönetilebilir. Angular, veri bağlama özelliği sayesinde bileşenler ve şablonlar verilerle kolayca senkronize edebilir.

TeleNöroRehab kullanıcı arayüzü her yaşta hastaların dinamik ve özelleştirilmiş olarak rehabilitasyon hizmeti alabilmelerini ve uzmanların hastaları program oluşturup takip etmelerini sağlamak amacı ile geliştirilmiştir.



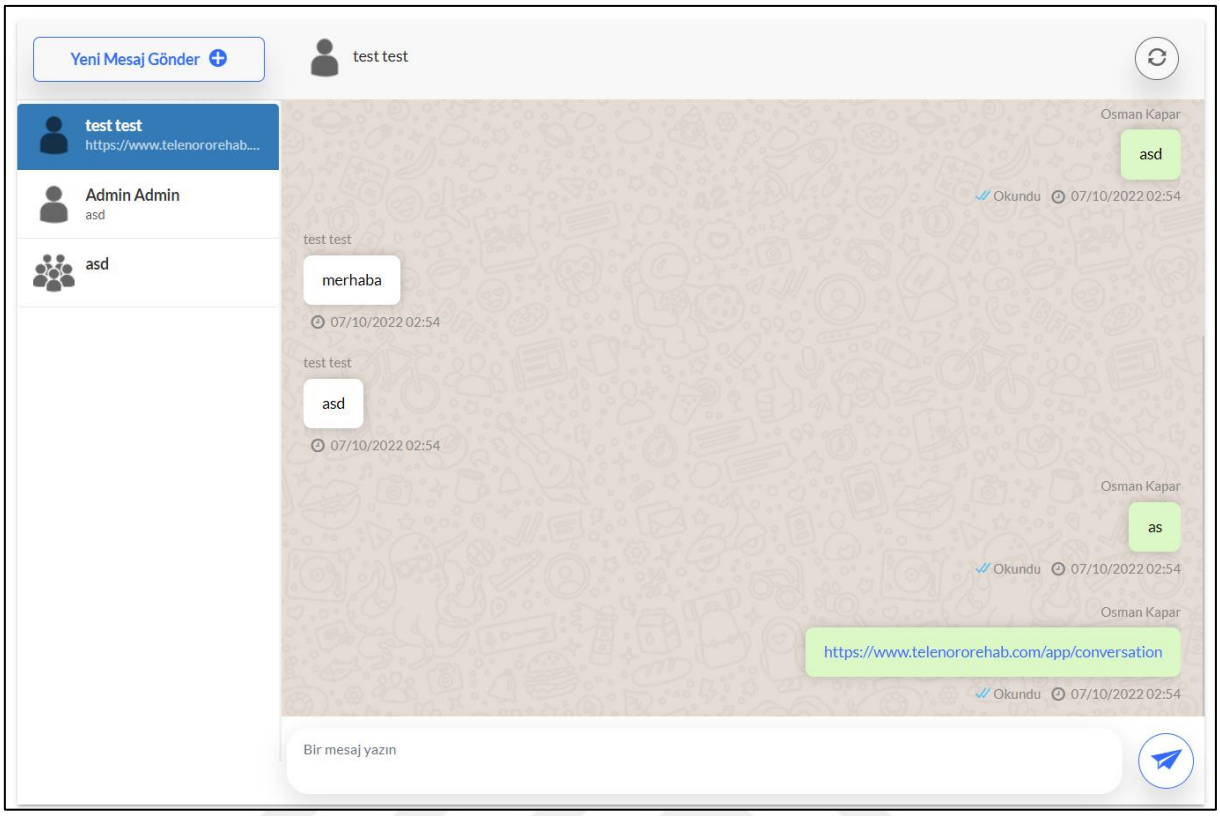
Şekil 3.15. Kullanıcı arayüzünde giriş ve anasayfa ekran görüntüleri

Şekilde 3.15 deki ekran görüntülerinde gösterildiği gibi hastalar, uzmanlar tarafından oluşturulan Email ve Şifre ile sisteme giriş yaparak anasayfaya aktarılır. Anasayfa üzerinde kullanıcılar sistem hakkında bilgilendirme içerikleri, kendisine atanan özelleştirilmiş programlar, broşürler ve istatistiklerini görüntüleyebildiği bir sayfa sunar. Kullanıcılar bu sayfa üzerinden hızlı tele-rehabilitasyon hizmetine başlayabilirler.



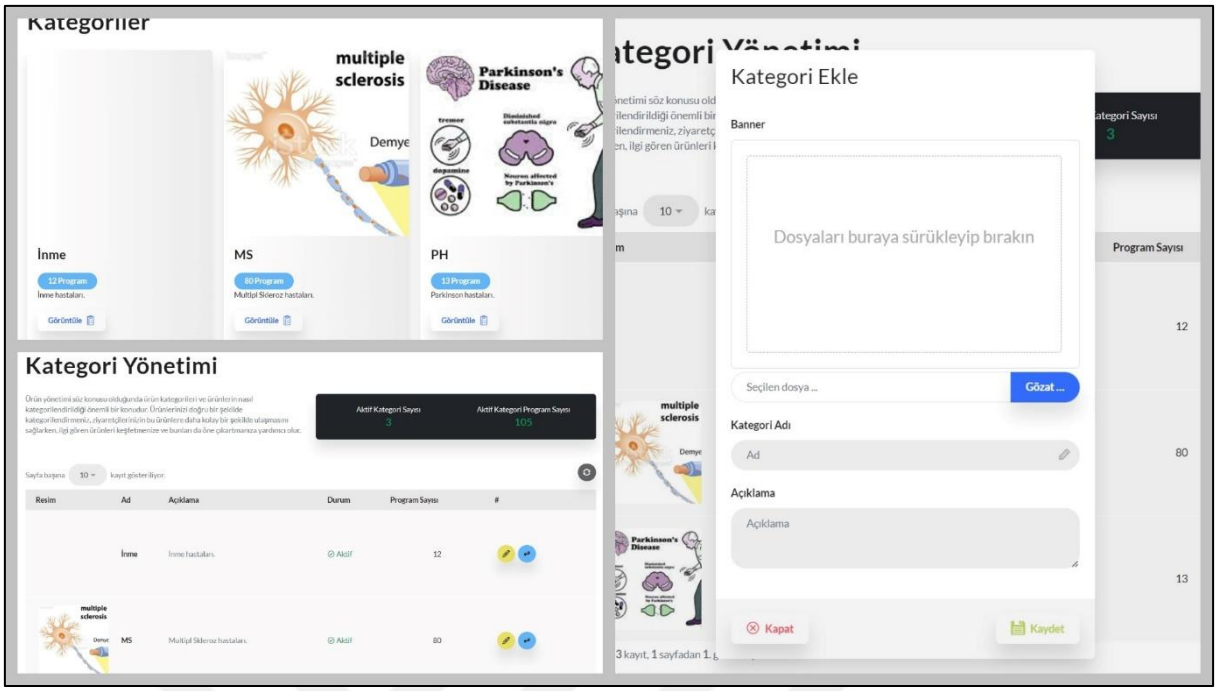
Şekil 3.16. Egzersiz programı görüntüleme sayfası için ekran görüntüleri

Şekil 3.16 da gösterilen ekran görüntüsü bir hasta için uzmanlar tarafından egzersiz havuzundan oluşturulan hastaya özgü bir egzersiz programının web üzerinden aktarıldığı video olarak hizmet alacağı sayfayı göstermektedir. Kullanıcılar kendilerine atanan programı Şekil 3.16 gösterilen sayfa üzerinden programın hazırlanış sırasına göre izleyebilir hastaların kendi kişisel ortamlarında ya da evlerinde tele-rehabilitasyon hizmetini sağlar.



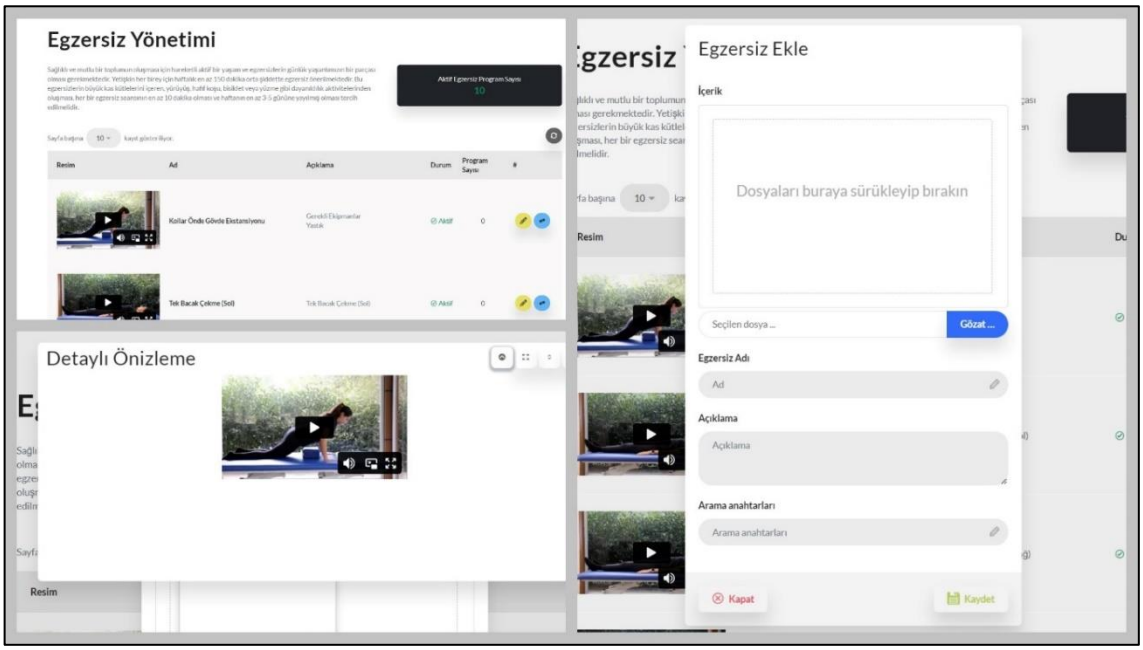
Şekil 3.17. Mesajlaşma arayüzü için ekran görüntüsü

Şekilde 3.17 de gösterilen ekran görüntüsü hastaların ve uzmanların kullanıcılar arasında anlık olarak mesajlaşmasına imkân veren bir sayfadır. Bu sayfa üzerinden uzmanlar, hastalar ile bire bir ya da grup olarak iletişime geçebilir ve istedikleri takdir de görüntülü konuşmak için bir aracı hizmet kullanarak birbirlerine web sitesi linki gönderebilirler. Bu arayüz Bölüm 3.1.25 de tanımlanan SignalR teknolojisini kullanarak web soket ile sunucu ve kullanıcılar arasında anlık veri alışverişini sağlar. EK-3 de bir Angular servisi içerisinde SignalR altyapısını kullanarak web soket üzerinden sunucu ile anlık olarak bağlantıların oluşturulup kontrol edildiği ve gelen verilerin işlendiği bir kaynak kod gösterilmektedir.



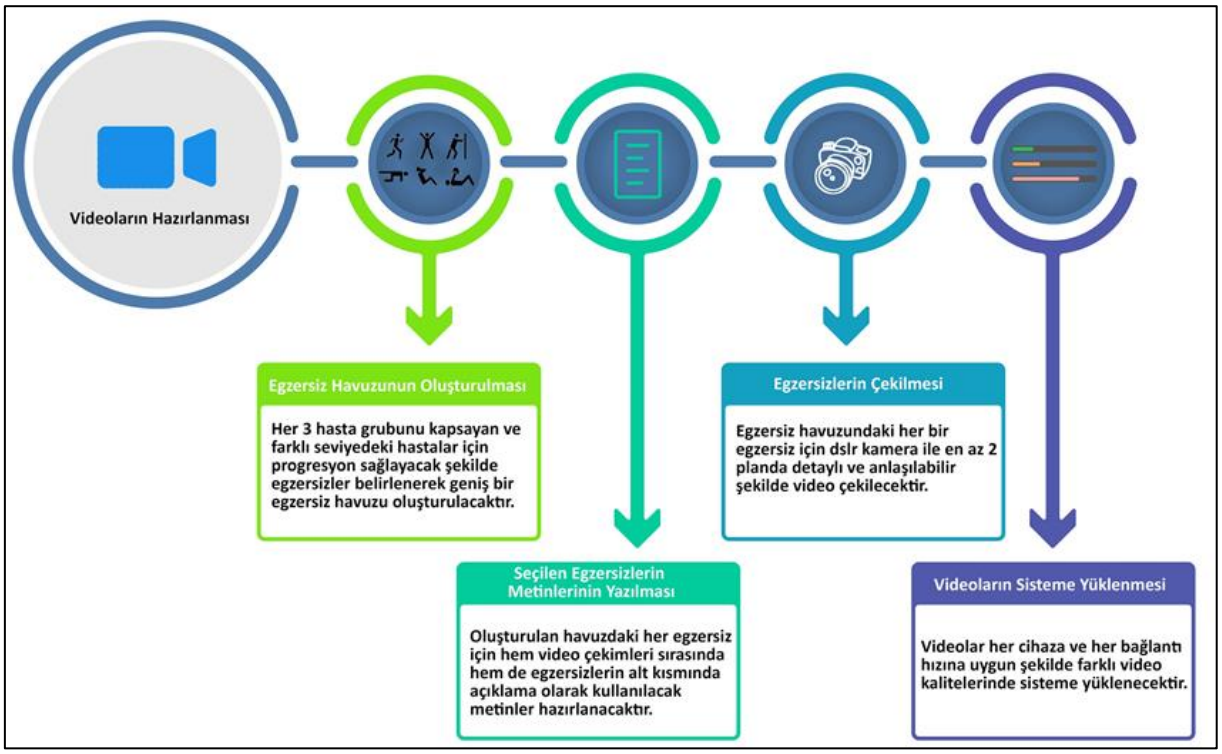
Şekil 3.18. Kategori yönetimi sayfası için ekran görüntüleri

Şekil 3.18 programların belirli bir kategorileri ayırmak ve düzenlemek için uzmanlar tarafından yönetilen kategori yönetim ekran görüntülerini göstermektedir. Hastalar için oluşturulmuş programlar belirli bir kategoriye atanarak kategorik ve düzenli bir program listesi oluşturmayı sağlar.

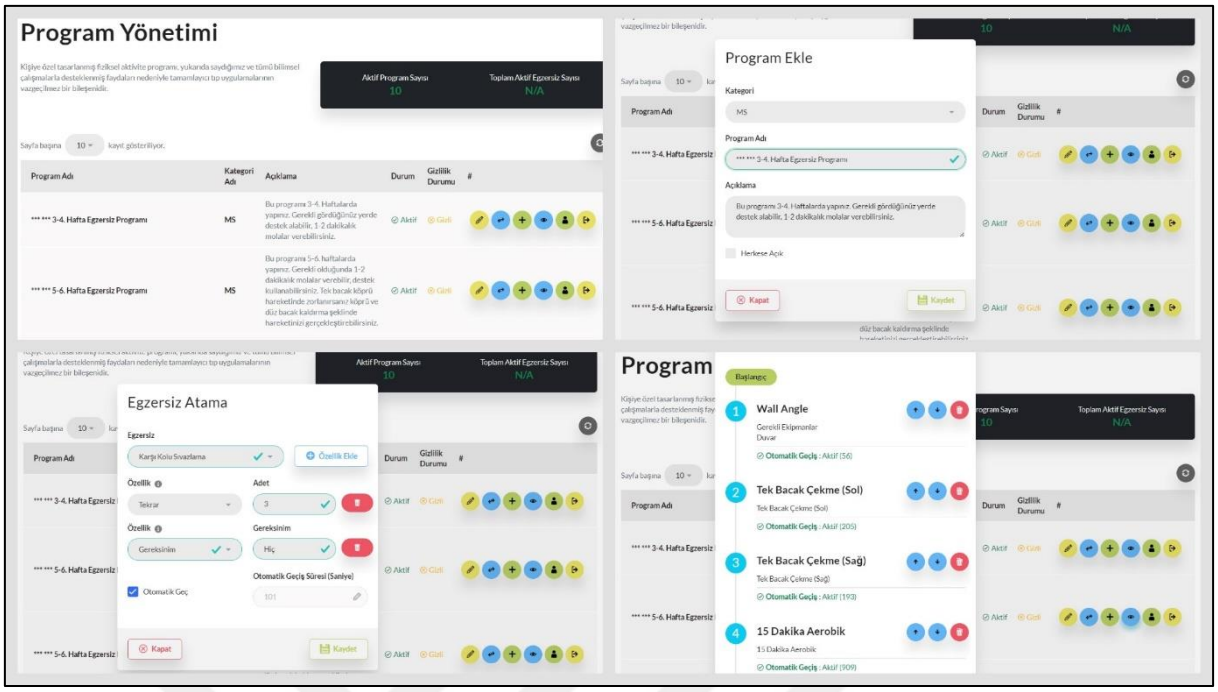


Şekil 3.19. Egzersiz Yönetimi sayfası için ekran görüntüleri

Şekil 3.19 da gösterilen ekran görüntüleri uzmanlar tarafından oluşturup yönetilen egzersiz listesini göstermektedir. Sistem üzerinden yetkili kullanıcılar ve uzman terapistler ile Şekil 3.20 gösterilen aşamalar ile kaydedilen videoların ve resimlerin sisteme yüklenip düzenlenebildiği bir arayüzdür. Bu arayüzde üzerinden eklenen egzersizler daha sonra hastalara özgü olarak oluşturulacak bir program için kullanılmaktadır.



Şekil 3.20. Sistemde yer alan egzersiz videolarının hazırlanma aşamaları



Şekil 3.21. Program yönetimi sayfası için ekran görüntüleri

Şekil 3.21 sistem üzerinde yetkili kullanıcıların hastalara özgü olarak oluşturulacak, egzersiz havuzdan seçilen programların yönetildiği ekran görüntülerini göstermektedir. Yetkili kullanıcılar program yönetimi ekranları üzerinden çeşitli programlar oluşturabilir, egzersiz atamaları ve egzersizlere özgü özellikleri ayarlayabilir, programdaki egzersizlerin düzenini oluşturabilir ve son olarak oluşturulan programın bir kullanıcıya atamasını gerçekleştirebilir. Program yönetim ekranlarının dinamik ve düzenlenebilir olması sebebi ile kullanıcılara devamlı ve sürdürülebilir olarak hastalığının durumuna göre farklı programlar atayıp hastalara bire bir tedavi imkânı sunmaktadır.

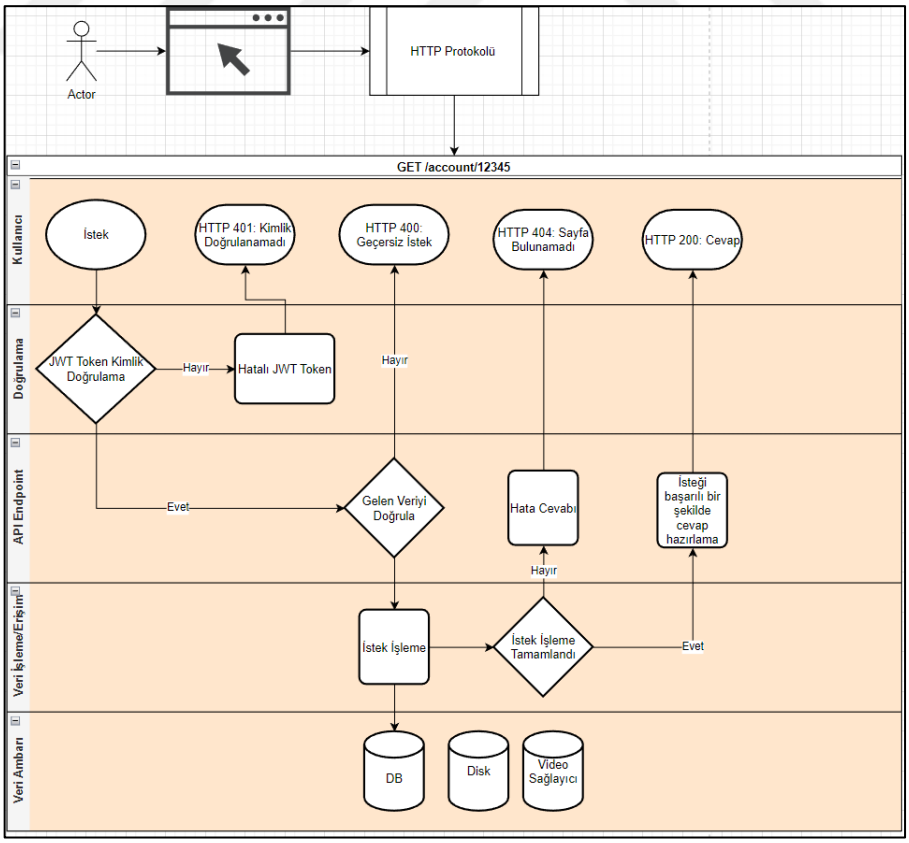
TeleNöroRehab sisteminin kullanıcı arayüzü tarayıcı tabanında çalışan bir web uygulaması olmasından dolayı internete erişimi olan her cihazdan kolayca kullanılabilecek şekilde Bölüm 3.1.28 de bahsedilen Bootstrap stil aracı ile geliştirilmiş olup her yaşta kullanıcının ister bilgisayar ister mobil cihazlarda kullanabilmeleri ve uzaktan tedavi hizmeti alabilmeleri amacı ile yola çıkılıp geliştirilmiştir.

3.2.3. Sunucu Tarafında Çalışan Algoritmik Program (Web Servis)

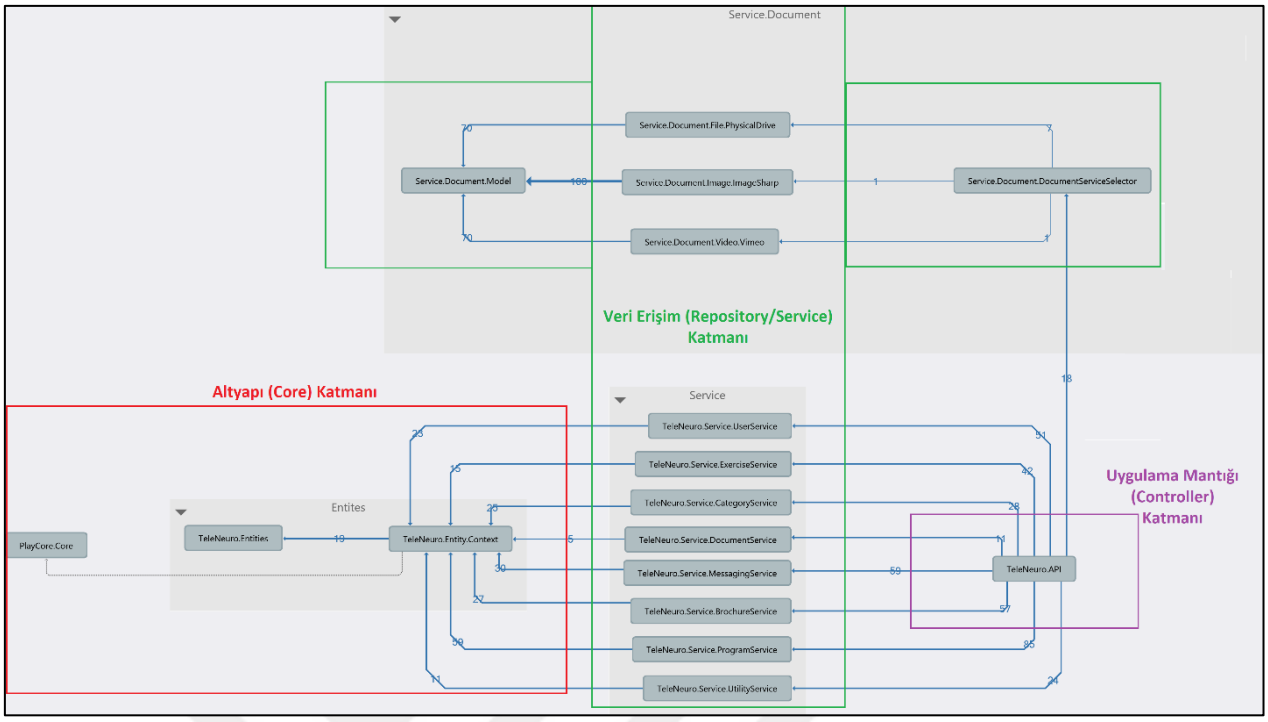
Web servis programı, uygulamanın istemci tarafı ile sunucu tarafı arasındaki iletişimi sağlayarak, uygulamanın işlevselliğini mümkün kılan, RESTful API gibi modern web servisleri kullanarak, uygulama tarafından sağlanan verilerin güvenli bir şekilde alışverişi

sağlayan bölümdür. Web servis programı, uygulamanın ölçeklenebilirliğini artırarak, kullanıcıların daha hızlı ve güvenilir bir deneyim elde etmesini sağlar. TeleNöroRehab uygulamasının sunucu tarafında çalışan verileri ve haberleşmeyi sağlayan RESTful API web servisi sağlayıcısı olarak görev yapan bir web servis programı bulunmaktadır. Bu program kullanıcının tarayıcısında çalışan Angular çerçevesi ile programlanmış olan kullanıcı arayüzü uygulamasının ihtiyacı olan verileri ve haberleşmeyi sağlayan sunucu tabanlı bir algoritmik program olup .NET çerçevesinde C# programlama dili kullanılarak geliştirilmiş bir programdır. Bu program Soğan (Onion) Mimariye uygun olarak geliştirilmiştir.

Kullanıcı arayüzünden sunucuya gönderilen bir istek ilk olarak IIS tarafından karşılanır ve kontrol edilir daha sonra IIS tarafından karşılanan istek sunucu tarafından çalışan programa aktarılır. Aktarılan istek .NET çerçevesinde çalışan uygulamaya aktarılır ve uygulama gelen isteği kimlik doğrulama, veri modeli doğrulama gibi işlemlere sokarak istediğin doğruluğunu işler. Doğrulanan istek yapılacak işleme göre görevlendirilmiş servisler aracılığı ile işlenip cevap üretilir. Doğrulanmayan istekler ise hata durumuna göre cevap üretir. Şekil 3.23 bir kullanıcı tarayıcısı tarafından oluşan isteğin işleme aşamalarının özeti akışını göstermektedir.



Şekil 3.22. İstek işleme akışı için çalışma prensibi



Şekil 3.23. Web Servis Katman mimarisi

Şekil 3.23 de gösterildiği gibi ana olarak farklı görevlere sahip birbirine sadece alt üst ilişkisi ile bağlı 3 katmandan oluşmaktadır;

1. Altyapı (Core) Katmanı
2. Veri Erişim (Repository/Service) Katmanı
3. Uygulama Mantığı (Controller) Katmanı

Altyapı (Core) Katmanın, web servis uygulamasının ana ve en alt katmanıdır. Bu katman web servis içerisinde verilerin servis edilmesi, veri tabanına erişim, kimlik ve güvenlik doğrulama gibi web servisin ihtiyaç duyduğu ve ana görevleri üstlenmektedir.

Bu katman içerisinde aşağıdaki işlemler sağlanmaktadır:

- Entity Framework çerçevesini kullanarak veri tabanına bağlanmak için gereken temel ve en alt düzey bağlantı modülleri
- Veri tabanı tablo modelleri
- Rollendirme görevleri için oluşturulmuş C# modülleri
- RESTFul API leri oluşturmak için temel “Controller” modülleri
- SignalR ile web soket bağlantısı sağlamak için sağlanan temel “Hub” adındaki arayüzler (Inteface)

- Arka Plan (Thread/Task) işlemlerini çalıştırmak için gereken modüller
- Veri doğrulama, veri şifreleme, kimlik ve güvenlik doğrulamaları için gereken JWT Token modülleri ve projenin tamamında kullanılabilen bazı yardımcı modüller kapsamaktadır.

Veri Erişim (Repository/Service) Katmanı, proje içerisinde Altyapı (Core) Katmanını kullanarak veri işleme görevilerini üstelenmektedir. Bu katman üzerinden veri tabanı üzerinden ihtiyaca göre verilerin elde edilmesi ve düzenlenmesi sağlanmasının haricinde proje içerisinde her türlü işlemlerin sağlandığı katmandır. Bu katman içerisinde dış servis bağlantıları, fiziksel disk kayıtları, resim ve video işleme algoritmaları gibi tüm işlemler gerçekleşmektedir. Bu katman kendi içerisinde alt bölümlere ayrılarak bileşen mimarisine dayanan her farklı işlem için bir modül oluşturularak geliştirilmiştir. Her işlem için farklı bir modülün oluşturulması modüller arasında bağımlılığı azaltarak geliştirme sürecinde gereksiz tekrarlardan, gereksiz hafıza ve disk kullanımlar gibi birçok işlemi optimize etmeyi sağlamaktadır.

Modüler programlama, büyük ve karmaşık yazılım projelerinde kullanılan bir tekniktir ve önemli avantajlar sağlar. Bu teknik, bir projeyi daha küçük ve daha yönetilebilir parçalara ayırarak, her bir parçanın bağımsız olarak geliştirilmesine, test edilmesine ve sürdürülmesine olanak tanır. Böylece, proje daha kolay yönetilir hale gelir ve hataların izlenmesi ve düzeltilmesi daha kolay olur. Modüler programlama ayrıca, yeniden kullanılabilir kodların oluşturulmasına olanak tanır. Bir kez yazılan ve test edilen bir modül, farklı projelerde veya farklı parçalarda kullanılabilir. Bu, yazılım geliştirme sürecini hızlandırır ve daha az kod yazarak daha fazla iş yapmanızı sağlar (Littlewood, 1979).

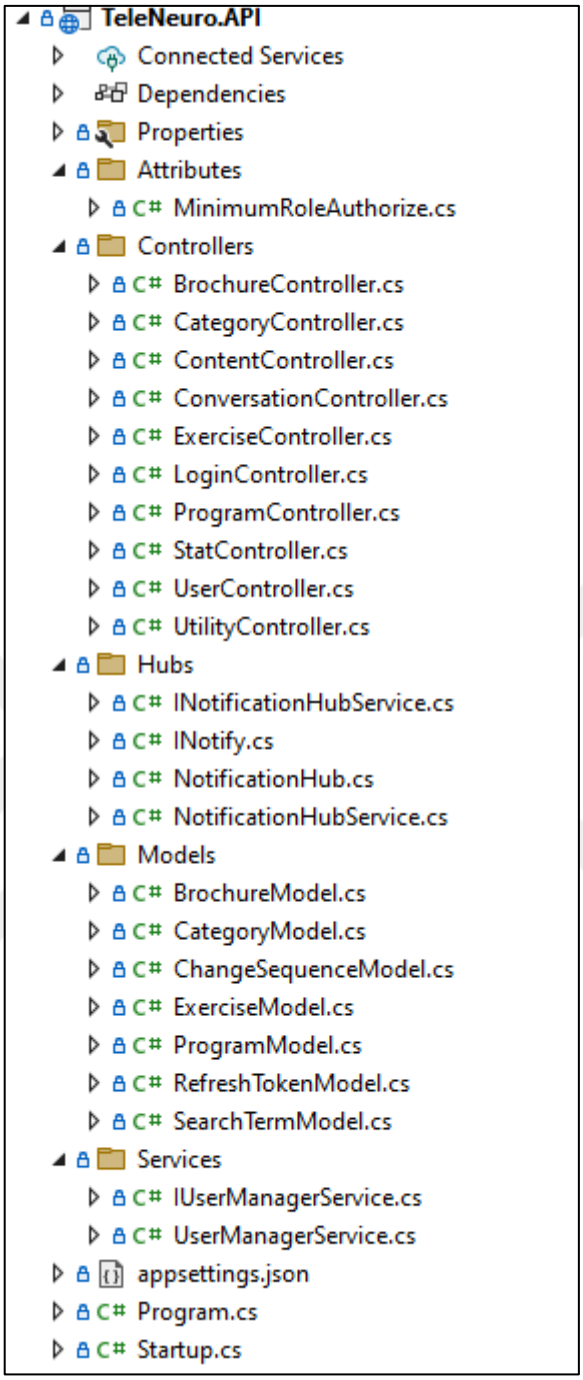
Veri Erişim (Repository/Service) Katmanı içerisinde yer alan resim, video ve belge gibi içeriklerin işlenmesi ve depolanmasından sorumlu modüller bulunmaktadır. Bu modüller içerik türüne göre farklılaşarak EK-4'te gösterilen arayüze uygun olarak geliştirilmiştir.

Resim veri türleri için Bölüm 3.1.29 üzerinde tanımı yapılan ImageSharp .NET kütüphanesi kullanılarak resimlerin depolanması ve kullanıcı arayüzüne isteklerin uygunluğuna göre boyutlandırılıp işlenmesi için kullanılmaktadır. Resimler, ImageSharp üzerinden işlendikten sonra fiziksel disk üzerinden saklanıp önbellek (cache) kullanımı için hazır hale getirilmektedir.

Video veri türleri dış video sağlayıcı olan Vimeo üzerinden saklanıp işlenmektedir. Web servis depolanan video içeriklerini HTTP yöntemlerini kullanarak Vimeo'ya aktarır ve karşılığında elde edilen videoya özgü bir kimlik ile tekrardan erişim imkânı oluşur. Video depolama için Vimeo kullanmamızın sebebi Türkiye içerisinde hizmet vermesi, diğer video sağlayıcılara göre fiyatlandırma politikalarının uygun olması, video oynatıcısının gelişmiş bir dinamik yapıya sahip olması ve en önemli özellik olarak depoladığımız videoların halk açık olmadan gizli olarak farklı çözünürlüklerde sunulmasıdır.

Video ve resim türlü haricindeki diğer tüm içerikler herhangi bir veri işlemine uğramadan fiziksel disk üzerinde depolanmaktadır.

Uygulama Mantığı (Controller) Katmanı diğer katmanlar üzerinden hazırlanan verilerin sunulması için iş mantığının ve işlemlerin kontrolü sağlayan katmandır. Bu katman diğer tüm katmanların birleşiminden doğan sunucu tarafından çalışacak ana programın kendisidir.



Şekil 3.24. Web Servis iş mantığı katmanı için dosyalama gösterimi

Şekil 3.23, iş mantığı katmanın içerisinde yer alan dosyaları ve bölümleri göstermektedir. Bu katman içerisindeki “Program.cs” ve “Startup.cs” sunucu tarafında çalışacak programın ana kodlarını ve işlem tanımlarını kapsamaktadır. “appsettings.json” dosyası uygulama içerisinde kullanılan sabit verileri tutmaktadır. Attributes klasörü katmanı içerisinde kullanılan yardımcı modülleri, Controller klasörü iş mantığı kapsamında RESTful API ile servis edilen kontrolcü yapıları, Hub klasörü SignalR kullanılarak oluşturulmuş web soket bağlantılarını, Models klasörü katman içerisinde gereken veri modellerini, Services katmanı sadece katman içerisinde kullanılan modülleri kapsamaktadır.

Bu bölümde bir isteğin sunucu tarafında çalışan bir program ile nasıl islendiğini ve geliştirme süreçlerinde nasıl bir yol izlendiği aktarılmaktadır. Web servis, TeleNöroRehab uygulama geliştirme sürecinde verimliliği ve esnekliği artıran önemli bir programdır. Web servis sayesinde, farklı platformlar arasında veri paylaşımı ve iletişim kolaylaşırken, ölçeklenebilir ve dağıtık uygulamalar geliştirme olasılığı da artar. Bu nedenle, geliştirilen uygulama web aracılığı ile birçok platformda farklı hizmetler ile servis edilebilinir.

3.2.4. Veri Tabanı Tasarımı

TeleNöroRehab, ilişkisel verileri bir SQL veri tabanı üzerinde saklamaktadır ve kullanılan SQL, Microsoft tarafından geliştirilen MsSQL’dir. Uygulama üzerinden kullanılan veri tabanı tasarımı EK-5 üzerinde gösterilmektedir. Veri tabanı tasarımı yapılırken ilişkisel veri tabanı modellemesine uygun olarak tasarlanmıştır.

İlişkisel veri tabanı, verilerin mantıksal olarak ilişkili olduğu bir yapıda depolandığı, verilerin kategorilendirildiği ve bu kategoriler arasındaki ilişkilerin belirtildiği bir veri tabanı türüdür. İlişkisel veri tabanları, en popüler veri tabanı türlerinden biridir ve özellikle işletmelerde yaygın olarak kullanılmaktadır.

İlişkisel veri tabanları, verilerin depolanması, güncellenmesi, silinmesi ve aranması için kullanılan bir dizi standart SQL (Structured Query Language) sorguları kullanarak verilere erişim sağlarlar. İlişkisel veri tabanları ayrıca, verilerin bütünlüğünü korumak için bir dizi kısıtlama (constraint) sağlarlar. Bu kısıtlamalar, verilerin doğru şekilde depolanmasını ve ilişkilerin korunmasını sağlar.

4. BULGULAR

Oluşturulan sistem Kasım 2021 tarihinde erişime açılmış olup Mart 2022 tarihinde hastaların erişimine açılmıştır. Mart 2022 tarihinden itibaren Mayıs 2023 tarihine kadar sisteme toplamda 134 kullanıcı kaydedilmiştir. Bu kullanıcıların 69'u kadın, 63 erkekten oluşmaktadır (2 kullanıcı kimliği belirtilmemiştir). Sistem üzerinden elde dilen verilen Mayıs 2023 tarihine kadar elde edilen verilerdir, sistem aktif kullanılmakta olup gün geçtikçe veriler artmaktadır. Tablo 1, TeleNöroRehab sistemi üzerinden genel verilerin özetini göstermektedir.

Tanım	**Sayı**
Kullanıcı Sayısı	134
Editor/Uzman Sayısı	8
Hasta Sayısı	126
Egzersiz Sayısı	356
Özelleştirilmiş Program Sayısı	260
Özelleştirilmiş Broşür Sayısı	147
Kategori Sayısı	3
Doküman Sayısı	554
Toplam Rol Sayısı	4
Başlatılan Sohbet Sayısı	88
Toplam Sohbet Mesaj Sayısı	359
Toplam Atanmış Program/Broşür Sayısı	410
Toplam Atanmış Program Sayısı	260
Toplam Atanmış Broşür Sayısı	150
Toplam Sisteme Giriş Sayısı	8055

Tablo 1. TeleNöroRehab genel sayısal verileri

Sistem üzerinde uzaktan rehabilitasyon hizmeti alacak hastalar iki farklı şekilde hizmet almaktadır. İlk grup video içerikleri ile hizmet alırken diğer grup ise broşür üzerinde hizmet almaktadır. Video içerik grubunda hizmet alan kullanıcı sayısı 87, broşür grubunda hizmet alan kullanıcı sayısı 42'dir.

Sistem üzerinde toplamda 147 adet özelleştirilmiş broşür 42 hastaya 150 farklı atama yapılarak uzaktan broşür ile rehabilitasyon hizmeti verilmektedir. Hastalar kendilerine atanan broşürleri %100 oranında indirmişlerdir ve toplamda 320 kere bu broşürleri indirmişlerdir.

Video içerik grubunda hizmet alacak hastalar için geniş bir egzersiz video havuzu oluşturulmaktadır. Bu havuz içerisinde toplam 359 video egzersizi bulunmaktadır. Bu egzersizlerin en uzunu 20 dakika (1219 saniye), en kısası 5 saniye ve ortalama 88 saniye uzunluğundadır. Egzersiz havuzu içerisinde hastalara özgü olarak toplamda 260 adet program oluşturulmuştur. Bu programların en uzunu 68 dakika (4129 saniye), en kısası 109 saniye ve ortalama 33 dakika (2012 saniye) oluşmaktadır.

Sistem üzerinden özelleştirilmiş olarak 260 adet program 259 atama ile 87 hastaya atanmıştır. Atanan tüm programlar tüm hastalar tarafından açılıp başlatılmış olarak görülmektedir ama 259 atamanın sadece 174 tanesini programın sonuna kadar tamamen izleyerek tamamlamıştır, bu da gösteriyor ki kullanıcılar %100 olarak programa katılıp %67 olarak programı tamamen izleme oranına sahiptirler. Tamamlama oranı kadınlar arasında %67,7 erkekler arasında ise %65,6'dır.

Video içerik grubu kullanıcıları toplamda sistem üzerinde 562 saat (2,025,670 saniye) video izleme süresi geçirmişlerdir. Bu sürenin %54,37 erkekler, % 43,63 kadarını ise kadınlar geçirmektedirler. Sistem üzerinden kullanıcı tarafından en fazla video izleme süresi 100 saat (362,230 saniye), en az 420 saniye ve ortalama 7 saattir (25,640 saniye).

Bu tez kapsamında yapılan sistemi kullanan Yazgan ve arkadaşlarının 2022 yılında yaptığı çalışmasında 48 MS hastalığı grubu içerisinde 29'u kadın 19 erkek olmak üzere yaş ortalamaları 42,85 yıl, 21 PH hastalığı grubu içerisinde 9'u kadın 12 erkek olmak üzere yaş ortalamaları 64,09 yıl, 11 inme hastalığı gurubunda 8'si kadın 4'ü erkek olup yaş ortalaması 49,50 yıl olan 3 farklı grup içerisinde sistemin kullanılabilirliği sırasıyla %81,14, PH %61,68, inme %82,04 ve tüm hastalar için %76,18 oran öne sürülmüştür.

5. TARTIŞMA

Bu çalışma kapsamında yapılmış sistemin Mart 2022 ve Mayıs 2023 tarihleri arasında elde edilen bulgular göz önüne alındığında web tabanlı bir tele-rehabilitasyon sisteminin hayata geçirilmesi ve kullanılabilirliği açısından önemli veriler sunmaktadır. Elde edilen verilere göre sistemi kullanan tüm kullanıcılar sisteme hem broşür hem de video içeriği eğitimleri yoluyla tamamen katılım göstermiştir. Ortalama 7 saatlik video izleme süresi ile %67 program tamamlama oranı bir web tabanlı uzaktan tedavi sisteminin kullanabilirliğini göstermiştir.

Sistemi kullanabilirliğini gösteren Yazgan ve arkadaşlarının 2022 yılında yaptıkları çalışmada, çalışmaya katılan orta yaş grubu üzerinden tüm katılımcılar için %76,18 sistemin kullanılabilirliğini göstermiştir.

Bu sistemi kullanıp yapılan diğer çalışmalardan Vural ve arkadaşlarının 2022 yılında "Multipl Skleroz Hastalarında İki Farklı Telerehabilitasyon Yönteminin Fiziksel Parametreler Üzerine Etkisinin Karşılaştırılması" isimli yaptığı çalışmada, Yılmaz ve arkadaşlarının 2022 yılında yaptığı "Parkinson Hastalarında Web Tabanlı Tele-Rehabilitasyon Uygulamasının Fiziksel Parametreler Üzerine Etkisinin İncelenmesi" çalışmasında ve Yazgan ve arkadaşlarının 2022 yılında "Developing a Web-based Tele-NeuroRehabilitation System and Implementation in Multiple Sclerosis Patients" isimli çalışması gösteriyor ki web tabanlı bir tele rahabilitasyon sisteminin İnme, Parkinson ve MS hastalıkları için fiziksel parametreler üzerinde yararlı sonuçlar ürettiği bir yöntem olduğu gösterilmiştir.

Bu tez kapsamında yapılan sistemi kullanan bu ve diğer çalışmalar web tabanlı, mobil ve bilgisayarlar da kolayca erişebilen bir sistemin bu alanda yapılan diğer video konferans ve oyun tabanlı sistemleri kullanan çalışmaya göre daha fazla kullanılabilirlik ve izlenme oranına sahip olduğu ortaya çıkarmıştır. Uzaktan tedavi amaçlayan tele-rehabilitasyon sistemleri hastaların kişisel ve çevresel sebeplerle fizyoterapi ve rehabilitasyon hizmetlerine katılım zorluğu yaşayan bireylerin tedavisini ve değerlendirilmesini amaçlamaktadır. Bu amaç ile yapılan çalışmalar belirli ortama ya da bireye yönelik değil de genel olarak bireylerin katılmasını mümkün kılan sistemlerin geliştirilmesi önemlidir.

6. SONUÇ VE ÖNERİLER

Bu çalışmada geliştirilen sistem, kişisel ve çevresel sebepler ile fizyoterapi ve rehabilitasyon hizmetlerine erişim zorluğu yaşayan kronik nörolojik hastalıklara sahip bireylerin, hastalık özelliklerine göre değerlendirme, takip ve ihtiyaçlarına yönelik hazırlanan özelleştirilmiş broşür ve video egzersiz programları ile hastane ortamı dışında kişisel alanlarında kolayca uzaktan tedavi uygulamalarına katılım sağlayabileceği Türkiye'de geliştirilmiş Türkçe alt yapı destekli bir tele-rehabilitasyon sistemidir. Tele-rehabilitasyon alanında ülkemizde yapılan çalışmaların sayıları fazla olmasına rağmen Türkçe alt yapı ile geniş ve özelleştirilmiş programlar sunan dinamik olarak genel kullanıma uygun geniş kapsamlı bir sistem ülkemizde örnekleri oldukça sınırlıdır. Bu çalışma bu alanda yapılan Türkçe alt yapı sunan geniş ilk sistemdir. Bu çalışma ve bu sistem yerli ve yerelleştirilmiş çalışmalara öncü ve basamak olacak bir model çalışma olması amaçlanmaktadır.

Bu tezde günümüzde güncel ve yaygın olarak kullanılan teknolojiler ve araçlar ile web tabanlı sistemin internet tarayıcısının desteklediği (bilgisayar, telefon, tablet vb.) tüm cihazlarda birçok mimari kullanarak nasıl geliştirdiğini göstermektedir. Bu sistem içerisindeki 3 farklı uygulama yer almaktadır. Bu uygulamaların sırası ile kurumsal web site, kullanıcı arayüzü ve web servistir. Her uygulama kendi alanına uygun olan teknoloji ve araçlar ile geliştirilmiştir. Kullanılan ana teknolojiler sırası ile Grunt, Angular ve .NET Web API (RESTful). Geliştirilen sistem web tabanlı olduğu için her uygulama kullanıcı verilerinin ve bilgilerinin önemi açısında güvenlik ve yetkilendirme konularında özenli çalışma yapılmıştır. Kullanıcı arayüzü ve web servis arasındaki haberleşmeyi sağlayan iki farklı yöntem olan HTTP ve Web Soket yöntemlerinde JWT Token ile kimlik doğrulama ve yetkilendirme sağlanmıştır. Sistem üzerinde elde edilen bilgi ve belgeler SQL ortamında veri tabanında, resimler ve benzer belgeler fiziksel disk ortamında, video içerikleri ise Vimeo sağlayıcısında saklanmaktadır. Bu sistem ileriye yönelik geliştirme ve bakımlar göz önüne alındığında Soğan (Katmanlı) Mimari kullanılarak geliştirilmiştir.

Bu çalışmada geliştirilen sistemi kullanan her birey için kişisel demografik bilgiler ve bireyin özelleştirilmiş broşür veya egzersiz programlarını kullanırken harcadığı süre, kullandığı içerik, katıldığı ve bitirdiği program gibi sistem yaptığı birçok hareketi kullanıcı gizliliği esas

alınarak sayısal veriler toplanıp araştırmalar için veri bankası oluşturulmuştur. Oluşturulan bu veri bankası sistem kullandıkça daha fazla artacağı gibi ileride yapılacak çalışmalar için daha kesin sonuçlar üreteceği hedeflenmektedir. Oluşturulacak veri havuzu ile beraber gelecekte yapay zeka yordamı ve makine öğrenmesi ile rehabilitasyon tedavilerinde kişiye özgü olarak tedavilerin daha sürdürülebilirliği ve kullanılabilirliği için çalışmalar yapılması planlanmaktadır.

Oluşturulan sistem 2 farklı tele-rehabilitasyon hizmeti vermektedir. İlk hizmet özelleştirilmiş broşürle ile hastalara hizmet vermektedir. Toplamda 147 özelleştirilmiş broşür 42 kullanıcıya 150 farklı atama ile %100 katılım oranı ile tele-rehabilitasyon hizmeti vermiştir. Sistemin sağladığı diğer tele-rehabilitasyon video egzersizleri ile sağlanmaktadır. Sistem üzerinde 359 video egzersizi içerisinde toplamda 260 özelleştirilmiş program 87 kullanıcıya 259 atama yapılmıştır. Kullanıcılar %100 oranında programlara katılım göstererek toplamda 562 saat video izleme süresine sahiptirler. Kullanıcıların %67'si kendilerine atanan programı tamamen en az bir kere tamamlamışlardır. Kullanıcılar ortalama olarak kendilerine atanan video egzersiz programlarını 7 saat izleme süresi geçirmişlerdir. Mart 2022 ve Mayıs 2023 tarihleri arasında elde edilen bu bulgular web tabanlı bir sistemin kullanılabilirliğini açıkça göstermektedir.

Bu tez kapsamında geliştirilen sistem ile yapılan diğer çalışmalarda İnme, Parkinson ve MS hastalıkları için fiziksel parametreler üzerinde olumlu etkilere sahip sonuçlar ürettiği gösterilmiştir.

Bu çalışma içerisinde geliştirilen sistem ve bu sistemi kullanılarak yapılan diğer çalışmalar, nörolojik hastaların kullanımına yönelik olarak geliştirilen kolayca erişe bilinen web tabanlı bir tele-rehabilitasyon sisteminin yüksek kullanılabilirlik düzeyi ile rehabilitasyon hizmetini sağladığı görülmüştür. Bu çalışma, yakın gelecekte ülkemizde geliştirilip dünya geneline yaygınlaşabilecek çoklu dil ve çoklu hastalık grupları destekli sistemlerin geliştirilmesi için bir temel model oluşturacağından, ülkemiz için büyük bir önem arz etmektedir. Son yıllarda yaşanan olaylar uzaktan eğitim, uzaktan çalışma gibi uzaktan sağlık hizmetleri hem ülke hem her vatandaş için olumlu sonuçlar ile yaşam kalitesini artıracaktır. Bu çalışmada geliştirilen sistem her ne kadar dinamik olsa da içerisindeki özelleştirilmiş programlar sadece 3 hastalık grubuna yönelik olarak hazırlanmıştır. Geliştirilen sisteme uygun çalışmalar ve uzmanlar ile

uzaktan tedavi sağlanabilecek diğer hastalık grupları içinde hazırlanan özelleştirilmiş programlar dahil edilerek daha geniş kapsamlı bir tele-sağlık hizmetine dönüştürüle bilinir.

KAYNAKLAR

Agostini, M., Moja, L., Banzi, R., Pisotti, V., Tonin, P., Venneri, A., Turolla, A., 2015, Telerehabilitation and recovery of motor function: a systematic review and meta-analysis, Journal of Telemedicine and Telecare, 21(4), 202-213.

Akıncı, B., Zenginler, Y. 2015, Tele-rehabilitasyon, Turkiye Klinikleri J Physiother Rehabil Special Topics, 1(1), 14-21.

Cason, J., 2007, A Pilot Telerehabilitation Program: Delivering Early Intervention Services to Rural Families, Int J Telerehabil, 2009, 1(1), 29–38.

Ceylan, A., 2021, COVID-19 Süreci ve Fizyoterapide Telerehabilitasyon Uygulamaları: Derleme, Adnan Menderes Üniversitesi Sağlık Bilimleri Fakültesi Dergisi, 5(3), 617-627.

Divyani, G., Rajinder, K. 2020, Teleneurorehabilitation for Parkinson's Disease: A Panacea for the Times to Come?, Annals of Indian Academy of Neurology, 23(5).

Dodakian, L., McKenzie A., Le V., See, J., Pearson-Fuhrhop, K., Quinlan, E., Zhou1, R., Augsberger, R., Tran, X., Friedman, N., Reinkensmeyer, D., Cramer, S., 2017, A Home-Based Telerehabilitation Program for Patients With Stroke, Neurorehabilitation and Neural Repair 2017, 31(10-11), 923–933.

Fallon, K. 2020, Exercise in the time of COVID-19, Australian Journal of General Practice, 49.

Ferraris, C., Ronga, I., Pratola, R., Coppo, G., Bosso, T., Falco, S., Amprimo, G., Pettiti, G., Priore, S., Priano, L., Mauro, A., Desideri, D., 2022, Usability of the REHOME Solution for the Telerehabilitation in Neurological Diseases: Preliminary Results on Motor and Cognitive Platforms, Sensors 2022, 22(23), 9467.

Greenwood, N., Mackenzıe, A., Cloud, G. C., & Wılson, N., 2008, Informal carers of stroke survivors–factors influencing carers: A systematic review of quantitative studies. Disability and Rehabilitation, 30(18), 1329 – 1349.

Ilse, I. B., Feys, H., De Wıt, L., Putman, K., & Weerdt, W. D., 2008, Stroke caregivers' strain: prevalence and determinants in the first six months after stroke. Disability and Rehabilitation, 30(7), 523 – 530.

Isernia, S., Pagliari, C., Jonsdottir, J., Castiglioni, C., Gindri, P., Gramigna, C., Baglio, F. 2019, Efficiency and Patient-Reported Outcome Measures from clinic to home: The Human Empowerment Aging and Disability program for digital-health rehabilitation, Frontiers in Neurology, 10, 1206.

Kalyon, T. Telerahabilitasyon, Fiziksel Tıp 1999, 2(2), 53-58.

Knepley, K. D., Mao, J. Z., Wieczorek, P., Okoye, F. O., Jain, A. P., Harel, N. Y. 2020, Impact of Telerehabilitation for Stroke-Related Deficits, Telemedicine and e-Health.

Licher, S., Darweesh, K. L. S., Wolters, F. J., Fani, L., Heshmatollah, A., Mutlu, U., Koudstaal, P. J., Heeringa, J., Leening, M. J. G., Ikram, M. K., Ikram, M. A., 2019, Lifetime risk of common neurological diseases in the elderly population, J Neurol Neurosurg Psychiatry 2019, 90, 148–156.

Littlewood, B., 1979, Software Reliability Model for Modular Program Structure, IEEE Transactions on Reliability, R-28 (3), 241-246.

Lundström, E., Terent, A., & Borg, J., 2008, Prevalence of disabling spasticity 1 year after first-ever stroke. European Journal of Neurology, 15(6), 533-539.

Melnick, S., 2016, Physiotherapy and clinical approaches–An overview, European Journal of Physical Education and Sport Science, 1(1).

Mutluay F. K., 2006, Multipl Skleroz Rehabilitasyonu. Türk Nöroloji Dergisi 2006, 12(2), 134-143.

Ortiz-Gutiérrez, R., Cano-de-la-Cuerda, R., Galán-del-Río, F., Alguacil-Diego, I., Palacios-Ceña, D., Miangolarra-Page, J., 2013, A Telerehabilitation Program Improves Postural Control in Multiple Sclerosis Patients: A Spanish Preliminary Study, Int. J. Environ. Res. Public Health 2013, 10(11), 5697-5710.

Öksüz N., Öztürk Ş., Doğu O., 2022, Parkinson Hastalığı Tanı ve Tedavisinde Yeni Ufuklar, Arch Neuropsychiatry 2022, 59(1), 36−41.

Öztürk, S., Aytaç G., Kızılay, F., Multipl Skleroz, Akd Tıp D / Akd Med J, 2017, 3, 137-147.

Rosen, M.J., 2004. Telerehabilitation. Telemedicine Journal and e-Health, 10(2), 115–117.

Schreıner, A. S., Morımoto, T., Arai, Y., & Zarıt, S., 2006, Assessing family caregiver's mental health using a statisticallyderived cut-off score for the Zarit Burden Interview. Aging & Mental Health, 10(2), 107–111.

Seidler, K. J., Duncan, R. P., McNeely, M. E., Hackney, M. E., Earhart, G. M. 2017, Feasibility and preliminary efficacy of a telerehabilitation approach to group adapted tango instruction for people with Parkinson disease, Journal of telemedicine and telecare, 23(8), 740-746.

Şener, İ., Erekdağ, A., Alpözgen, A., 2021, Yüz Yüze ve Tele-Değerlendirme Yöntemleri Kullanılarak Spastik Tip Serebral Palsili Çocukların Alt Ekstremite Fonksiyonel Performans Değerlemelerinin Karşılaştırılması: Olgu Serisi, Gazi Sağlık Bilimleri Dergisi 2021, 6(3).

Topçu, S., Bölüktaş, R. P., İnmeli Hastalarda Yaşam Kalitesi ve Sosyal Desteğin Yaşam Kalitesine Etkisinin İncelenmesi 2012, Yeni Tıp Dergisi 2012, 29(3), 159-164.

Ulicny, G., Temkin, A., Veskmarovich, S. Telemedicine: Does It Have a Place in Rehab? Team RehabReport 1996, 7, 19- 22

Vellata, C., Belli S., Balsamo, F., Giordano, A., Colombo, R., Maggioni, G., 2021, Effectiveness of Telerehabilitation on Motor Impairments, Non-motor Symptoms and Compliance in Patients With Parkinson's Disease: A Systematic Review, Front. Neurol., 2021 Sec. Neurorehabilitation, 12.

Vural, P., Yazgan Zenginler, Y., Tarakcı, E., Hüseyinsinoğlu, B. E., İsenkul, M. E., Kürtüncü, M., 2022, Multipl Skleroz Hastalarında İki Farklı Telerehabilitasyon Yönteminin Fiziksel Parametreler Üzerine Etkisinin Karşılaştırılması, 58. Ulusal Nöroloji Kongresi 2022, 132.

Waters, CH., Çev: Büyükkal B., 2000, Parkinson Hastalığının Tanı ve Tedavisi, Turgut Yayıncılık ve Tic. A.Ş., İstanbul.

Yazgan Zenginler, Y., Tarakcı, E., Hüseyinsinoğlu, B. E., İsenkul, M. E., Kürtüncü, M., 2023, İnme, Multipl Skleroz ve Parkinson Hastalarına Yönelik Telerehabilitasyon Sistemi Geliştirilmesi ve Kullanılabilirliğinin İncelenmesi, 58. Ulusal Nöroloji Kongresi 2022, 138-140.

Yıldırım, Ş. 2021, Telerehabilitasyon, Türkiye Klinikleri.

Yılmaz, E., Yazgan Zenginler, Y., Tarakcı, E., Hüseyinsinoğlu, B. E., İsenkul, M. E., Kürtüncü, M., 2022, Parkinson Hastalarında Web Tabanlı Tele-Rehabilitasyon Uygulamasının Fiziksel Parametreler Üzerine Etkisinin İncelenmesi, 58. Ulusal Nöroloji Kongresi 2022, 91-92.

Zenginler Yazgan, Y., Vural, P., Tarakcı, E., Huseyinsinoglu, E., B., Ormen, R., Kapar, O. S., Isenkul, M. E., Kurtuncu, M., 2022, Developing a Web-based Tele-NeuroRehabilitation System and Implementation in Multiple Sclerosis Patients, 38th Congress of the European Committee for Treatment and Research in Multiple Sclerosis, 2060.

EKLER

EK-1 Grunt Yönetim Dosyası

```
var DIST_FOLDER = () => {
  return "dist";
}

//Grunt
module.exports = function (grunt) {
  // Grunt Config
  var initGruntConfig = () => {
    return {
      bake: {
        options: {
          content: function () {
            return grunt.file.readJSON("src/app/consts/content.json");
          }
        },
        build: {
          files: [{
              expand: true,
              cwd: "src/app/pages/",
              src: "*.html",
              dest: DIST_FOLDER(),
            }
          ],
        },
      },
      clean: {
        dist: ["dist"]
      },
      copy: {
        main: {
          files: [{
              expand: true,
              cwd: 'src/app/assets/',
              src: '**',
              dest: DIST_FOLDER() + '/assets/'
            }
          ],
        },
      },
      watch: {
        copy: {
          files: ["src/app/**/*.*", 'Gruntfile.js', 'consts/*.*'],
          tasks: ["copy", "bake"]
        },
        livereload: {
          files: ["dist/**/*.*"],
          livereload: true
        },
        options: {
          livereload: {
            host: 'localhost',
            port: 35729
          }
        }
      }
    }
  };

  // Init Message
  grunt.registerTask("init", function () {
    console.log("----------> Debug Start <----------");
    console.log("file://" + require('path').resolve('dist/index.html'));
    console.log("----------> Debug Start <----------");
  });

  // Build
  grunt.registerTask("build", function () {
    grunt.task.run("clean", "copy", "bake");
  });

  // Default Grunt
  grunt.initConfig(initGruntConfig());
  grunt.loadNpmTasks("grunt-bake");
  grunt.loadNpmTasks('grunt-contrib-copy');
  grunt.loadNpmTasks('grunt-contrib-watch');
  grunt.loadNpmTasks('grunt-contrib-clean');

  //Default grunt tasks
  grunt.registerTask("default", ["clean:dist", "copy", "bake", "init", "watch"]);
};
```

EK-2 Angular Directive Örneği

```typescript
import { Directive, ElementRef, Input, OnChanges, Renderer2, SimpleChanges } from "@angular/core";

@Directive({
    selector: '[cardLoader]'
})
export class CardLoaderDirective {
    private alreadyDisabledElements: any[] = [];
    private isLoading: boolean = false;
    private buttonSpinner: string = '<span class="spinner-border spinner-border-sm ms-2" role="status" aria-hidden="true"></span>';
    constructor(
        private _renderer: Renderer2,
        private _elementRef: ElementRef
    ) { }
    start() {
        this.isLoading = true;
        this.disable();
        this._renderer.addClass(this._elementRef.nativeElement, 'card-loader');
        this._renderer.addClass(this._elementRef.nativeElement, 'overlay');
        this._renderer.addClass(this._elementRef.nativeElement, 'overlay-block');
    }

    stop() {
        this.isLoading = false;
        this.active();
        this._renderer.removeClass(this._elementRef.nativeElement, 'card-loader');
        this._renderer.removeClass(this._elementRef.nativeElement, 'overlay');
        this._renderer.removeClass(this._elementRef.nativeElement, 'overlay-block');
    }

    private disable() {
        this.alreadyDisabledElements = [];
        var nodes = this._elementRef.nativeElement.getElementsByTagName('*');
        for (var i = 0; i < nodes.length; i++) {
            if (nodes[i].disabled) {
                this.alreadyDisabledElements.push(nodes[i]);
            }
            else {
                nodes[i].disabled = true;
            }
            if (nodes[i].tagName == "BUTTON") {
                $(nodes[i]).append(this.buttonSpinner);
            }
        }
    }
    private active() {
        var nodes = this._elementRef.nativeElement.getElementsByTagName('*');
        for (var i = 0; i < nodes.length; i++) {
            if (!this.alreadyDisabledElements.includes(nodes[i])) {
                nodes[i].disabled = false;
            }
            if (nodes[i].tagName == "BUTTON") {
                $(nodes[i]).children(".spinner-border").remove();
            }
        }
        this.alreadyDisabledElements = [];
    }
}
```

```html
<div class="modal-content" cardLoader>
</div>
```

EK-3 Angular-SignalR Servis Kaynak Kodu

```typescript
import { Injectable } from "@angular/core";
import * as signalR from "@microsoft/signalR";
import { Subject } from "rxjs";
import { ConversationMessage } from "src/app/models/conversation/conversation-message";
import { environment } from "src/environments/environment";
import { AuthenticationService } from "../authentication/authentication-service";

@Injectable()
export class NotificationHubService {
    private _messageSubject: Subject<ConversationMessage | null>;
    private _conversationReadSubject: Subject<{ UserId: number, ConversationId: number } | null>;
    private _hubConnection: signalR.HubConnection;
    private retryCount = 0;

    constructor(private _authenticationService: AuthenticationService) {
        this._messageSubject = new Subject<ConversationMessage | null>();
        this._conversationReadSubject = new Subject<{ UserId: number, ConversationId: number } | null>();
        this.initConnection();
    }

    public newMessageObservable() {
        return this._messageSubject.asObservable();
    }

    public conversationReadObservable() {
        return this._conversationReadSubject.asObservable();
    }

    public async startConnection() {
        try {
            await this._hubConnection.start();
        } catch (err) {
            if (this.retryCount < 5) {
                setTimeout(async () => {
                    this.retryCount++;
                    await this.startConnection();
                }, 5000);
            }
        }
    }

    private initConnection() {
        this._hubConnection = new signalR.HubConnectionBuilder()
            .withUrl(`${environment.request.host}${environment.request.hubEndpoint}${environment.request.endPoints.notificationHub}`, {
                accessTokenFactory: () => this._authenticationService.getUser()?.Token ?? ""
            })
            .configureLogging(signalR.LogLevel.Critical)
            .withAutomaticReconnect([0, 3000, 5000, 10000, 15000, 30000])
            .build();

        this._hubConnection.onreconnecting((error) => {
            console.error(`Connection lost due to error "${error}". Reconnecting.`);
        });

        this._hubConnection.onreconnected(() => {
            console.info('Connection reestablished. Connected.');
        });

        this._hubConnection.on("NotifyNewMessage", (i) => {
            if (i) {
                this._messageSubject.next(new ConversationMessage().mapModel(i));
            }
        });

        this._hubConnection.on("NotifyReadConversation", (i, j) => {
            if (i) {
                this._conversationReadSubject.next({ UserId: i, ConversationId: j })
            }
        });
    }
}
```

EK-4 Doküman İşleme Arayüzü (Interface)

```
6 references | 0 changes | 0 authors, 0 changes
public interface IDocumentService
{
    3 references | 0 changes | 0 authors, 0 changes
    Task<DocumentResult> SaveAsync(string filePath, string fileName = null, string mimeType = null, Action<DocumentResult> completed = null);
    5 references | 0 changes | 0 authors, 0 changes
    Task<DocumentResult> SaveAsync(byte[] data, string fileName = null, string mimeType = null, Action<DocumentResult> completed = null);
    6 references | 0 changes | 0 authors, 0 changes
    Task<DocumentResult> SaveAsync(Stream stream, string fileName = null, string mimeType = null, Action<DocumentResult> completed = null);
    3 references | 0 changes | 0 authors, 0 changes
    Task<DocumentResult> SaveAsync(Uri uri, string fileName = null, string mimeType = null, Action<DocumentResult> completed = null);
}
```

EK-5 Veri Tabanı Tasarımı

