

SMART HOSPITAL SYSTEM AKILLI HASTANE SİSTEMİ

Rabia Kuran^{1*}, Şaban Gülcü²

¹Bilgisayar Mühendisliği Bölümü / Mühendislik Fakültesi, Necmettin Erbakan Üniversitesi, Türkiye

²Bilgisayar Mühendisliği Bölümü / Mühendislik Fakültesi, Necmettin Erbakan Üniversitesi, Türkiye

*(rabiakuran34@gmail.com)

Özet – Hastanelerde yatış sürecinde çeşitli sorunlar ortaya çıkmaktadır. Hasta refakatçilerinin yanı sıra hastalar da bu süreçte çeşitli zorluklarla karşılaşabilmektedir. İletişim eksiklikleri, taleplerin etkili bir şekilde iletilmemesi ve ihtiyaçların karşılanmaması gibi konular hastanede yatış sürecindeki sorunları arttırmaktadır. Bu çalışma, hastanede kalma sürecinde karşılaşılan bu zorlukları azaltmayı hedeflemektedir. Hastaların mobil uygulamalar aracılığıyla ihtiyaçlarını iletebilmesi, taleplerin hızlı bir şekilde yönetilmesini sağlayabilir. Bu uygulama sayesinde hastalar, ihtiyaç duyduklarında hemşirelerle birebir iletişime geçmek zorunda kalmadan diğer hizmet birimleriyle iletişim kurabileceklerdir. Ayrıca, web tabanlı uygulama aracılığıyla hasta bilgileri, hastane durum bilgileri takip edilebilir ve hastanın durumuyla ilgili koordinasyon sağlanabilir. Hasta yatış süreçlerindeki sorunlara odaklanan bu çalışma, hastaların ve refakatçilerin deneyimini iyileştirerek sağlık sektöründe olumlu bir etki yaratmayı hedeflemektedir. Çalışmanın amacı, hastanede yatan hastaların veya refakatçilerin, mobil uygulama aracılığıyla o anki hemşire ve doktor bilgilerine erişim sağlamasıdır. Hastalar, odada buldukları sürece ihtiyaçları olduğunda uygulama üzerinden talepler oluşturabileceklerdir. Bu sayede hasta veya refakatçi, hastanenin odasından çıkmadan ihtiyaçlarını ve isteklerini iletebileceklerdir. Bu durum, hemşirelerin aracılık etmesini gerektirmeyecek ve refakatçilerin kısa süreli bile olsa hastanın yanından ayrılmamalarını sağlayacaktır. Ayrıca web tabanlı bir platformda, hangi hastanın hangi odada kaldığı ve hastanın refakatçi bilgileri gibi bilgiler yer alacaktır. Herhangi bir talep geldiğinde (kişisel, yiyecek-içecek veya acil durum talepleri gibi), ilgili hizmet biriminde çalışan kişiler, talepleri hastanın yatış odasının kat ve oda numarasıyla eşleştirerek teslim edebileceklerdir.

Anahtar Kelimeler – Hastane, Refakatçiler, Mobil uygulama, İletişim, Kolaylık, Talep yönetimi, Web uygulaması, Sağlık hizmetleri, Hastane hizmetleri.

I. GİRİŞ

Hastaların hastane ortamında desteklenmeleri amacıyla yakınlarının da yanlarında bulunmasına refakatçi uygulaması denilmektedir. Bu uygulama günümüzde pek çok kuruluştaki yaygın olarak sürdürülmektedir [1]. Sorumlu refakatçi bir hastanede, evdeki bakım ortamında veya ihtiyaç duyulan başka yerde hastalara yardımcı olmakla yükümlüdür. Hastanede refakatçisinin sorumlulukları arasında temel bakım ihtiyaçlarının yanı sıra gerektiğinde pansuman yapmak ve hastaların ilaç düzenini idare etmek de yer alır. Odaya gelen yemek servisini yemesini sağlamak ile, gerektiğinde hastaların beslenmesine yardımcı

olmak öğünler arasında su ve yardımcı gıdalarla destek olmak, hastaların yürümesi, döndürülmesi ve oturtulması gibi onların günlük yaşam aktivitelerini desteklemek belli başlı bazı görevleri arasındadır [2]. Bunca yapılacak görevin arasında bir refakatçinin oda içinde gerek hasta ile ilgili gerek kendi ile ilgili ihtiyacı olduğu zamanlarda, hastayı yalnız bırakıp bir hemşire ya da görevli bir personel aramak yerine bunu elinde bulunduğu bir telefon ile yapması büyük bir ihtiyaç haline gelmektedir.

Kullanıcı ilk olarak ister Android işletim sistemi ister ios işletim sistemi kullanıyor olsun

uygulamayı iki şekilde de kullanmış olduğu telefona indirebilir. Daha sonra kısa birkaç adım ile refakatçi kayıt işlemini tamamladıktan sonra uygulamayı rahatça kullanabilir.

Bu çalışma de öncelikle back-end tarafı java dilinde yazılıp postgresSQL tarafında tablolar ve tablo ilişkileri tamamlandı. Daha sonra web uygulaması için react typescript teknolojisi kullanılarak hastane de bulunan tüm ürünlerin listelenmesi, kaydedilmesi, güncellenmesi sağlandı. Böylece hastane içinde tüm ürünlerin durumları kontrol altına alınmış olundu. Refakatçilerin kullanacağı mobil uygulama ise react native teknolojisi kullanılarak yazıldı. Her iki uygulama içinde kullanıcı kayıt ve giriş işlemleri bulunmaktadır.

Refakatçi aktörlerin yaşam doyumu, bakım yükü ve tükenmişlik düzeylerini etkileyen faktörler açısından bir değerlendirme yapan makalenin araştırma sonucunda refakatçi bireyin yaşam doyumu %66,24; zaman-bağımlılık yükü %41,35; gelişimsel yük %25,05; fiziksel yük %26,45; sosyal yük %11,09 ve duygusal yük %13,72 düzeyinde bulunmuştur. Bakım verenler %47 düzeyinde tükenmişlik yaşamaktadırlar. Yaşam doyumu ve bakım verme yükü ile tükenmişlik arasında güçlü bir ilişki vardır. Bakım yükü arttıkça tükenmişlik artmakta; bakım yükü duygusal tükenmeye ($r=0.771$, $p<0.01$) ve duyarsızlaşmaya ($r=0.594$, $p<0.01$) neden olmaktadır. Çalışmanın ampirik bulguları, günümüzde hasta bakımı sorununun gittikçe artan bir eğilime sahip olduğunu göstermektedir. Bu nedenle hasta bakımına yönelik politikaların geliştirilmesi önemlidir. Özellikle hastaya refakat eden bireylerin fiziksel, psikolojik ve sosyal ihtiyaçlarını destekleyecek programların geliştirilmesi önem taşımaktadır [3]. Bu yüzden hastaya ve hasta yakınlarına kolaylık sağlayabilecek teknolojilerin sayısını arttırmamızda büyük bir fayda söz konusudur.

Bu çalışmada, hastane çalışanları ve yatışta bulunan hasta tarafı olmak üzere iki taraf bulunmaktadır. Çalışanlar, öncelikle hasta yatış yaparken o hastanın kişisel bilgileri, oda bilgileri, refakatçi bilgileri gibi gerekli tüm bilgilerini bu web uygulamasına kaydetmektedir. Hastalar ise uygulamayı indirip kayıt yaptıktan hemen sonra tüm isteklerini istediği bölüme ulaştırıp hemen

sonuç alabilmektedir. Ayrıca ulaşmak istediği herhangi bir sağlık personeline ya da hastane çalışanına ulaşabilir ve bu kişinin irtibat bilgilerine, meşguliyet durumlarını görebilir.

Sistem tasarımında güncel teknolojiler kullanılmıştır. Çalışma yapımı esnasında, veri tabanı olarak postgresSQL(pg Admin4) kullanılmıştır. Backend'i için eclipse uygulaması içinde java dili kullanılmıştır. Visual Studio Code uygulamasında web uygulaması için react typescript, mobil uygulama için ise react native kullanılmıştır.

Bu makale şu şekilde organize edilmiştir: Bölüm 1'de hastaların ve hasta refakatçilerinin önemi vurgulanmaktadır ve refakatçiler için önceki çalışmalar hakkında bilgiler vermektedir. Bölüm 2'de uygulamayı geliştirirken kullanılan teknolojilerden bahsedilmektedir. Bölüm 3'te bulgular, geliştirilen sistem ve ekran çıktıları sunulmaktadır. Bölüm4'te çalışmanın bilime katkısından bahsedilmektedir. Son olarak, Bölüm5'te makale sonuçlandırılmaktadır.

II. MATERYAL VE YÖNTEM

Günümüzde birçok hastane de yatış yapan ve refakatçi olan insanlara kolaylıklar sağlamak ve genel olarak refakatçilerin memnun olmadıkları gidişatın önüne geçmek ve bireylerin işlerini kolaylaştırmak için bu çalışma yapılmıştır.

A. Web Tabanlı Uygulama Nedir?

Web tabanlı uygulama, yalnızca ağ bağlantısı üzerinden HTTP protokolü kullanılarak erişilebilen, cihazın hafızası yerine internet tarayıcısı yardımıyla çalıştırılabilen uygulamalardır. Web tabanlı uygulamalar bilgisayarımızın depolama belleği üzerinde aktif olarak yer almaz ve internet tarayıcıları yardımıyla kullanılırlar. Buna karşın bazı web tabanlı uygulamalar istemci tabanlı olarak çalışabilir ve ufak bir kod parçanın bilgisayarınıza indirilmesi gerekebilir. Buna karşın asıl işlem sonucu tarafında gerçekleştirildiği için işlemlerin tümü yine sonucu tarafında tamamlanacaktır.

İnternet tarayıcı üzerinden kullanılan ve HTTP protokolü yardımıyla haberleşmeyi sağlayan yazılımların tümü web tabanlı uygulamalara birer örnektir. İnternet dünyasının gelişmesi ve Web 2.0

düzenine dâhil olmasıyla birlikte internette statik sayfa düzeni terk edilmiş ve web uygulamaları kullanılmaya başlanmıştır. Bu bağlamda, bugün internette kullandığımız birçok servisin bir web tabanlı uygulamaya veya bulut tabanlı uygulamaya örnek olabileceğini söyleyebiliriz.

B. PostgreSQL Nedir?

PostgreSQL, tüm dünyada popüler olan açık kaynak kodlu, platform bağımsız gelişmiş bir nesne ilişkisel (ORDBMS) veri tabanı yönetim sistemidir. Yüksek performanslı, kararlı ve güvenilirdir. Modern kurumsal veri tabanı kabiliyet ve özelliklerine sahiptir [4].

PostgreSQL'de güvenlik, yalnızca bir özellik değildir, temel yapılarından biridir. Her minör ve majör sürümü yeni güvenlik özellikleri ve güncellemeleri ile gelir. Kimlik doğrulama, yetkilendirme, denetim, veri güvenliği, veri şifreleme, satır (row) seviyesinde güvenlik gibi birçok güvenlik yapısı vardır [4].

Çok zengin SQL yapılarını destekler. İlişkilendirilmiş ve ilgisiz alt sorguları destekler. Ortak tablo ifadesini (CTE), window fonksiyonlarını ve özyinelemeli sorguları destekler. PostgreSQL'e her sürümde yeni SQL özellikleri eklenmektedir. PostgreSQL kolayca genişletilebilecek şekilde tasarlanmıştır. Veritabanına yüklenen uzantılar, yerleşik özellikler gibi çalışır. İlk kurulumla gelen eklentileri contrib dizini altında görülür [4].

C. React Nedir?

Genellikle React olarak anılan React JS, Facebook ve açık kaynaklı bir geliştirici topluluğu tarafından yürütülen JavaScript tabanlı bir UI geliştirme kütüphanesidir [5].

React TypeScript, JavaScript tabanlı React kütüphanesinin TypeScript ile kullanılmasını sağlayan bir geliştirme ortamıdır. TypeScript, JavaScript'e bir süper set olarak eklenen bir programlama dilidir ve statik tür denetimi sunar. TypeScript, JavaScript kodunun güvenliğini, okunabilirliğini ve sürdürülebilirliğini artıran ek özellikler sağlar.

React JavaScript ile geliştirme yaparken, genellikle dinamik tür çıkarımına dayalı olarak çalışır. Bu, kodun çalışma zamanında türleri

belirlemesini ve uygulamasını sağlar. TypeScript ise statik tür denetimi yaparak, kodun derleme aşamasında hataları yakalar ve tür güvenliğini sağlar. Bu, hataların erken tespit edilmesini, daha sağlam bir kod yazılmasını ve kodun daha kolay anlaşılmasını sağlar. React TypeScript, JavaScript'e kıyasla bazı avantajlara sahip olduğu için bu çalışmada React TypeScript kullanıldı.

D. Java Nedir?

Referanslar Java, basit ve anlaşılabilir sözdizimiyle geliştiricilerin yazılım uygulamaları oluşturmasını sağlar. Nesne yönelimli programlama paradigmalarına dayanarak, bir uygulamayı küçük bileşenlere, yani nesnelere böler ve bu nesnelere birbirleriyle iletişim kurmasını sağlar. Java'nın diğer bir önemli özelliği ise platform bağımsızlık sağlamasıdır. Java kodu, Java Sanal Makinesi (JVM) tarafından yorumlanır ve herhangi bir işletim sistemi üzerinde çalışabilir.

Java'nın kullanım alanlarından biri de back-end geliştirme olarak bilinir. Back-end, bir web uygulamasının sunucu tarafıyla ilgilenir ve veri tabanı işlemleri, iş mantığı ve veri işleme gibi görevleri üstlenir.

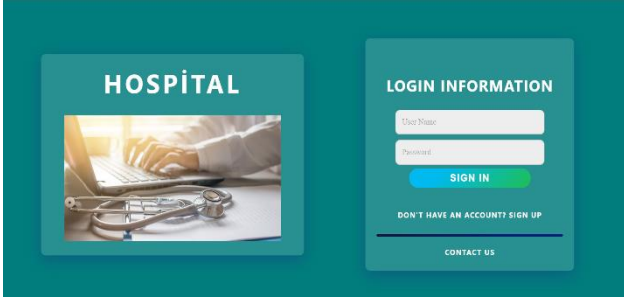
Swagger UI ile de oluşturulan API'lar ile ilgili bilgilerin görselleştirilmesi ve otomatik belgelendirme oluşturabilmesine yarayan yardımcı bir arayüzdür. Bu arayüz sayesinde web api çalışmamızda hangi resource'lara sahip olduğumuzu ve bu resource'larla ilgili hangi eylemleri yapabileceğimizle ilgili bir belgelendirme oluşturmuş oluruz [6].

III. BULGULAR VE GELİŞTİRİLEN SİSTEM

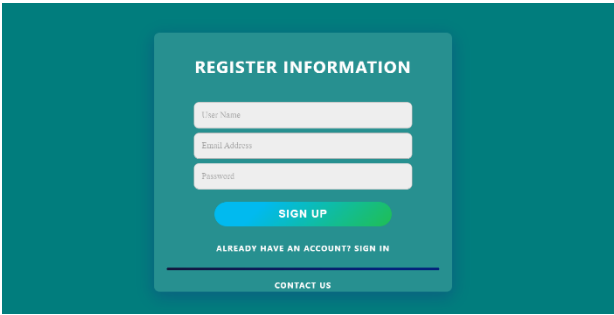
Çalışmamızda web tabanlı akıllı hastane uygulamasını ve mobil akıllı hastane uygulamasını geliştirmek için yukarıda bahsedilen programlama dilleri ve teknolojiler kullanılmıştır. Her iki uygulamada da birer kullanıcı girişi sayfası bulunmaktadır. Web için kullanıcılar çalışanlardır, mobil için ise kullanıcılar hastalar veya hasta yakınlarıdır.

Şekil 1'de çalışanların giriş yapabildiği ekran gösterilmektedir ve bu ekranda görüldüğü gibi çalışanın kaydı yok ise Şekil 2'de görülen ekrana yönlendirme ya da yetkililere ulaşmak istiyorsa

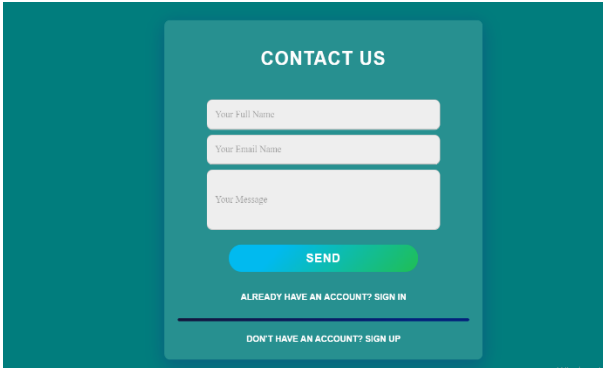
Şekil 3'teki ekrana yönlendirme yapılabilmesi için iki tane link yönlendirilmesi mevcuttur.



Şekil 1. Çalışan Giriş Ekranı



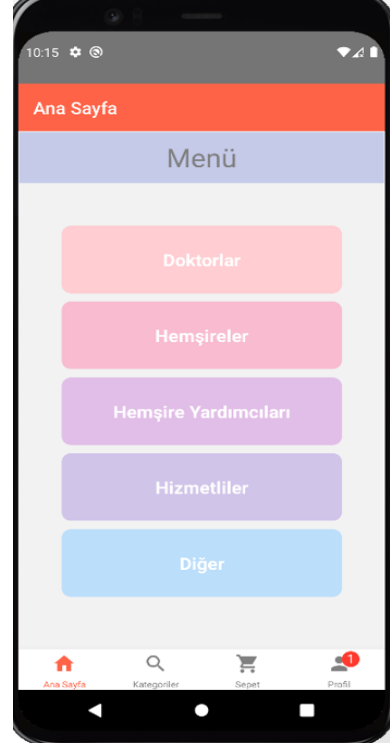
Şekil 2. Çalışan Kayıt Ekranı



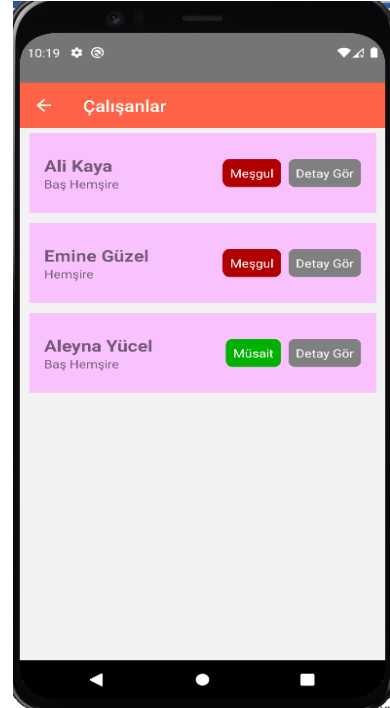
Şekil 3. Yetkililere Ulaşma Ekranı

Mobil tarafında ise Şekil 4'te hastanın ya da hasta refakatçisinin uygulamaya giriş yaptıktan sonraki karşısına çıkan ekran gösterilmektedir. Bu ekranda çalışanların bölümleri kullanıcılar için kullanım rahatlığı olması için kategorilere ayrılmıştır.

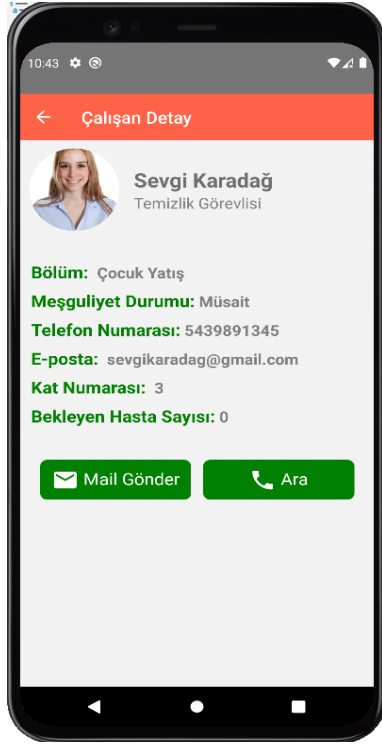
Burada herhangi bir kategoriye giriş yapıldığında Şekil 5'teki ekran açılır ve burada o bölümde çalışan tüm çalışanlar listelenir ve çalışanların meşgulliyet durumu görünür.



Şekil 4. Anasayfa



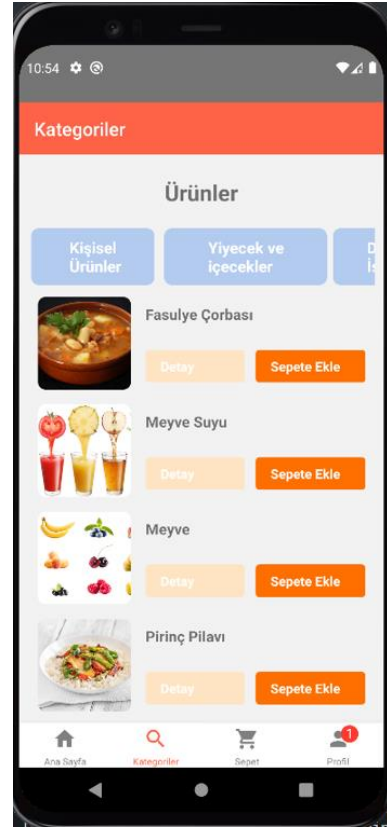
Şekil 5. Çalışanlar Ekranı



Şekil 6. Çalışan Detay Ekranı

Kullanıcı çalışan ile ilgili detay bilgi görmek isterse detay butonuna basıp Şekil 6'daki ekrana ulaşabilir. Kullanıcı bu ekrandan çalışanın iletişim bilgilerine, meşguliyet durumuna çalıştığı bölüme ve kata ulaşabilir. Burada bulunan mail gönder ve ara butonları da kullanıcıyı direkt mail ve arama bölümlerine yönlendirir böylece rahatlıkla ulaşmak istediği çalışana ulaşılabilir bir ortam sunulmuştur.

Ürünler sayfasında ise Şekil 7'de görüldüğü gibi üst kısmında kişisel ürünler, yiyecek ve içecek ürünler, diğer şeklinde kategorilere ayırma işlemi gerçekleştirildi. Burada sepete ekleme işlemleri ve ürün detayını görme gibi işlemler yapılabilir. Detay kısmına tıkladığında her ürün için Şekil 8'deki gibi bir ekran açılır. Burada ürüne ait açıklamalar bulunur. Kullanıcı istediği ürünü favorilere ekleyebilir.



Şekil 7. Kategoriler

Sepetin boş hali Şekil 9'da gösterilmiştir. Burada kullanıcıya sepetin boş olduğu ve eklemek isterse kategori sayfasından ekleyebileceği bir yönlendirme yapılmıştır. Sepete ürünler eklendikten sonra sepet sayfasına geçildiğinde Şekil 10'da görülen butonlar sayesinde ürün sayısında arttırma, azaltma, silme işlemleri rahatlıkla yapılabilir. Ürün seçimleri tamamlandıktan sonra ise 'Onayla' butonuna basılarak Şekil 11'de görüldüğü gibi istek gönderilir ve sepet boşaltılmış olur. Buradan gönderilen isteklerin tümü web uygulamasının ekranında görülür.



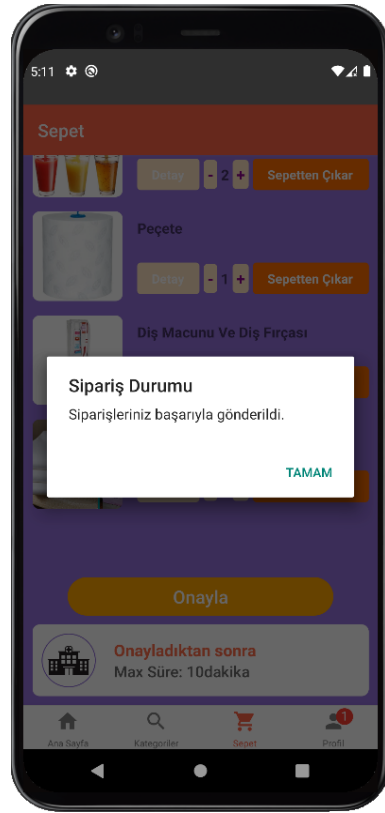
Şekil 8. Ürün Detay Ekranı



Şekil 10. Dolu Sepet



Şekil 9. Boş Sepet



Şekil 11. Sepet Onaylama

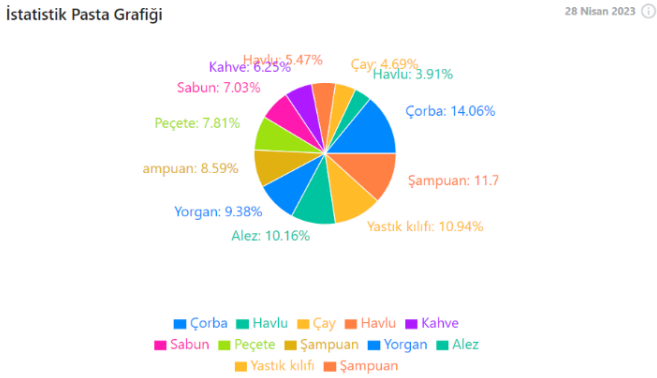
Web uygulamasında kullanıcıların tüm istekleri Şekil 12'de görüldüğü gibi çalışanların

ekranlarında görünür. Talep edilen ürünler hasta odalarına ulaştıktan sonra çalışanlar tablonun durum kısmında bulunan onaylanmadı yazısını onaylandı olarak günceller ve onaylanan ürün sayısı mevcut ürün sayısından azalarak veri tabanında güncellenir.

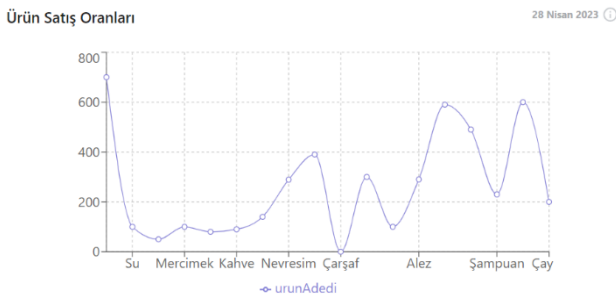
Bildirimler	Oda Numarası	Ürün Id	Ürün Kategorisi	Ürün Adı	Ürün Adedi	Tarih	Notifications	Durum	Ürün Onaylama
	A-308	113	Yemek Bölümü	Maye	1	06.06.2023 13:37:27	Not Eklemedi	Onaylandı	
	A-308	114	Yemek Bölümü	Maye Sıvı	1	06.06.2023 13:37:24	Not Eklemedi	Onaylandı	
	A-308	112	Yemek Bölümü	Pirinç Pilavı	3	06.06.2023 13:37:28	Not Eklemedi	Onaylandı	
	A-308	113	Yemek Bölümü	Maye	1	06.06.2023 13:37:27	Not Eklemedi	Onaylandı	
	A-308	114	Yemek Bölümü	Maye Sıvı	1	06.06.2023 13:37:24	Not Eklemedi	Onaylandı	

Şekil 12. Gelen İstekler Sayfası

Ayrıca, web uygulamasında ürünler için Şekil 13, Şekil 14'te görüldüğü gibi çeşitli grafikler ve ürünlerin, hastaların eklenebileceği, bilgilerinin güncellenebileceği, silineceği Şekil 15, Şekil 16'da görüldüğü gibi ekranlar bulunmaktadır.



Şekil 13. Ürünler İçin Pasta Grafiği



Şekil 14. Ürün Sipariş Grafiği

Hasta Bilgileri	
Hasta Adı Soyadı *	Giriş Tarihi *
Doktor Adı Soyadı *	Refakatçi Adı Soyadı *
Oda Numarası *	Telefon *
Doğum Tarihi *	Yatış Sebebi *

Şekil 15. Hasta Kayıt

Edit Product		
Ürün Kategorisi : Kışlet Bölümü	Ürün Id : 202	Ürün Adı: Peçete
Ürün Bilgi: Ürün Bilgi	Ürün Adedi : 700	Ürün Tarihi : 23 Şubat 2023
Ürün Resmi : https://www.sarfinarket.com.tr/tork-matic-hareketli-havlu-advanced-150-m-68-fiyati-fotoselli-makine-havlulari-tork-250067-18170-38-K.jpg		
Ürün Kategorisi : Yemek Bölümü	Ürün Id : 136	Ürün Adı: Su
Ürün Kategorisi : Yemek Bölümü	Ürün Id : 138	Ürün Adı: Çorba
Ürün Kategorisi : Yemek Bölümü	Ürün Id : 139	Ürün Adı: Yemek
Ürün Kategorisi : Yemek Bölümü	Ürün Id : 140	Ürün Adı: Tarhana
Ürün Kategorisi : Yemek Bölümü	Ürün Id : 141	Ürün Adı: Mercimek

Şekil 16. Ürün Ekleme Güncelleme Silme

IV. TARTIŞMA

Bu çalışma, hastane yatış sürecinde ortaya çıkan zorlukları azaltmayı hedefleyen bir fikir üzerine odaklanmaktadır. Hastalar ve refakatçiler, iletişim eksikliği, taleplerin etkili bir şekilde iletilmemesi ve ihtiyaçların karşılanmaması gibi sorunlarla karşılaşmaktadır. Bu çalışma, mobil uygulamaların hastane süreçlerinde kullanımının hastaların deneyimini iyileştirebileceğini araştırmaktadır.

Mobil uygulamalar aracılığıyla hastalar, ihtiyaçlarını hızlı bir şekilde iletebilme imkânı bulacaklardır. Bu uygulama sayesinde hastalar, hemşirelerle birebir iletişim kurmadan diğer hizmet birimleriyle iletişim kurabileceklerdir.

Bu çalışma, hasta yatış sürecindeki sorunlara odaklanarak hastaların ve refakatçilerin deneyimini iyileştirme ve sağlık sektöründe olumlu bir etki yaratma amacını taşımaktadır. Mobil uygulamaların kullanımı, hastaların ihtiyaçlarını etkili bir şekilde iletmelerini ve taleplerin daha hızlı bir şekilde yönetilmesini sağlayarak hastane süreçlerindeki verimliliği artırabilir.

V. SONUÇLAR

Çalışmamızda hastaların ve hasta yakınlarının hastane ile kolayca iletişime geçmesi, isteklerini rahatça oluşturabilmesi, çalışanların ise hastaların isteklerine daha hızlı yanıt verebilmesi, tüm

hastaların ve ürünlerin kontrolünü rahatça yapabilmesi amaçlanmıştır. Bunun için hem web tabanlı uygulama hem de mobil uygulama geliştirilmiştir. İnsanların hastanede yatış şartları hem hasta için hem de yanında kalan kişi için psikolojik durumlarına büyük etki etmektedir. Ayrıca, böyle kolaylıklar sağlayan bir hastane ortamı bulunulan ortamın kalitesin önemli ölçüde arttırır. Bundan dolayı, çalışmamız insanların yaşam kaliteleri ve sağlıkları açısından önem arz eden bir çalışmadır.

Geliştirilen uygulamamızda, hastaların ve hasta yakınlarının yatış yapacakları süre zarfında hastanede daha rahat şekilde kalacakları öngörülmektedir. Bundan dolayı, çalışmamıza konu olan hastalar ve hasta yakınları ile çalışanlar arasında iletişimin daha iyiye gitmesi ve iletişim kopukluğunda azalma olması beklenmektedir.

KAYNAKLAR

- [1] [1] Sharp T (1990). Relatives involvement in caring for the elderly mentally ill following long- term hospitalization. Journal of Advanced Nursing 15:67-73.
- [2] [2] <https://www.gaiadanismanlik.com/blog/gaiadanismanlik-2.html>
- [3] [3] https://www.researchgate.net/publication/326299169_Refakatci_Aktorlerin_Yasam_Doyumu_Bakim_Yuku_Ve_Tukenmislik_Duzeylerini_Etkiley
- [4] [4] <chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/https://dijitalakademi.bilgem.tubitak.gov.tr/wp-content/uploads/2019/09/postSQL-Katalog.pdf>
- [5] [5] <https://www.hosting.com.tr/blog/react-nedir/>
- [6] [6] <https://academy.patika.dev/courses/net-core/1-swagger-ui-nedir-nas%C4%B1-kullan%C4%B1%C4%B1rA>. Karnik, “Performance of TCP congestion control with rate feedback: TCP/ABR and rate adaptive TCP/IP,” M. Eng. thesis, Indian Institute of Science, Bangalore, India, Jan. 1999.