



**YAZILIM PROJELERİNE YÖNELİK PROJE YÖNETİM YAZILIMI
GELİŞTİRİLMESİ**

Mert UYĞUR

Mücahid Bilal KESKİN

BM495 BİLGİSAYAR PROJESİ I

**GAZİ ÜNİVERSİTESİ
MÜHENDİSLİK FAKÜLTESİ**

ARALIK 2022

Danışman: Prof. Dr. Hacer KARACAN

Bilgisayar Mühendisliği, Gazi Üniversitesi

Bu çalışmanın, kapsam ve kalite olarak Dönem Projesi olduğunu onaylıyorum.

.....

Dönem Projesi Teslim Tarihi:/...../.....

Danışmanı tarafından kabul edilen bu çalışmanın Dönem Projesi olması için gerekli şartları yerine getirdiğini onaylıyorum.

ETİK BEYAN

Gazi Üniversitesi Fen Bilimleri Enstitüsü Tez Yazım Kurallarına uygun olarak hazırladığım bu tez çalışmada;

- Tez içinde sunduğum verileri, bilgileri ve dokümanları akademik ve etik kurallar çerçevesinde elde ettiğimi,
- Tüm bilgi, belge, değerlendirme ve sonuçları bilimsel etik ve ahlak kurallarına uygun olarak sunduğumu,
- Tez çalışmada yararlandığım eserlerin tümüne uygun atıfta bulunarak kaynak gösterdiğimi,
- Kullanılan verilerde herhangi bir değişiklik yapmadığımı,
- Bu tezde sunduğum çalışmanın özgün olduğunu,

bildirir, aksi bir durumda aleyhime doğabilecek tüm hak kayıplarını kabullendiğimi beyan ederim.

YAZILIM PROJELERINE YÖNELİK PROJE YÖNETİM YAZILIMI GELİŞTİRİLMESİ
(Bitirme Projesi)

Mert UYĞUR

Mücahid Bilal KESKİN

GAZİ ÜNİVERSİTESİ
MÜHENDİSLİK FAKÜLTESİ

Aralık 2022

ÖZET

Günümüzde kullanılan takım ile yazılım geliştirme araçları hep büyük ölçekli projeleri baz aldığından, orta ve küçük ölçekli projeler için bu yazılım geliştirme araçları kolaylıktan çok zorluk çıkarmaktadır. Bu sistemlerin karmaşıklığının azaltılması her ölçekte projeler için kullanışlı olmasını sağlayacaktır. Aynı zamanda, günümüzde yapay zekanın geldiği nokta düşünüldüğünde, bu sistemler yapay zekayı ya çok az kullanmakta, ya da hiç kullanmamaktadır. Yapay zekanın gücünün, proje geliştirme yazılımları için yeterince kullanılmamaktadır. Bu amaçla, bu eksikleri gidererek, daha çok orta ve küçük ölçekli projeleri hedef alarak bir sistem geliştirilecektir.

Bilim Kodu : 92408

Anahtar Kelimeler : Proje Yönetimi, Web Uygulaması, Yapay Zeka

Sayfa Adedi : 116

Danışman : Prof. Dr. Hacer Karacan

GAZİ UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
December 2022

ABSTRACT

Today, software development tools used with teams are based on large-scale projects, so they create more difficulties than convenience for medium and small scale projects. Reducing the complexity of these systems will make them useful for projects of all sizes. At the same time, given the current state of artificial intelligence, these systems either use very little or no artificial intelligence at all. The power of artificial intelligence is not sufficiently used for project development software. With this in mind, a system will be developed targeting more medium and small scale projects, addressing these deficiencies.

Science Code : 92408

Key Words : Project Management, Web Application, Artificial Intelligence

Page Number : 116

Supervisor : Prof. Dr. Hacer Karacan

TEŞEKKÜR

Dönemin başından itibaren emek verdiğimiz Bitirme Projesi'nin ilk dönemini tamamlamak üzereyiz. Bu bölümü bize yardımlarını esirgememiş ve süreç boyunca teşvik etmiş insanlara teşekkür etmek için kullanacağız.

Öncelikle danışmanlığımızı üstlenen, konu seçiminden araştırmanın yürütülmesine dek desteğini, fikirlerini ve zamanını esirgemeyen hocamız Prof. Dr. Hacer KARACAN' a danışmanlığı için çok teşekkür ediyoruz. Bunun yanında gereksinimlerin şekillenmesi için yaptığımız ankete katılan kişilere de teşekkürü borç biliriz.

İÇİNDEKİLER

	Sayfa
ETİK BEYAN	3
ÖZET	4
ABSTRACT	5
TEŞEKKÜR	6
İÇİNDEKİLER	7
ŞEKİLLERİN LİSTESİ	8
SİMGELER VE KISALTMALAR	9
1. GİRİŞ	10
2. LİTERATÜR TARAMASI	11
2.1 Scoro	12
2.2 Atlassian Jira	14
2.3 Proje Yönetiminde Yapay Zeka	15
3. GELİŞTİRİLEN YAKLAŞIM VE BULGULAR	15
3.1 Anketler	16
3.1.1 Geliştiriciler	16
3.1.2 Yöneticiler	17
3.1.3 Çıkarılan Sonuçlar	18
3.2 Gereksinimlerin Belirlenmesi	19
3.3 Uygulamanın İç Tasarımı	19
3.3.1 Demo	19
3.3.2 Gerçekleştirilen Tasarımlar	21
3.4 Kullanılan Araçlar ve Tercih Sebepleri	22
3.4.1 Sunucu Tarafında(Back-end) Kullanılan Araçlar	22
3.4.1.1 Veritabanı	22
3.4.1.2 API	23
3.4.2 Kullanıcı Tarafında(Front-end) Kullanılan Araçlar	23
4.SONUÇ VE ÖNERİLER	23

ŞEKİLLERİN LİSTESİ

Şekil	Sayfa
Şekil 2.1: Scoro <i>Projects</i> arayüzü.	12
Şekil 2.2: Scoro, proje sayfası.	13
Şekil 2.3: Scoro, proje görev yönetimi sayfası.	14
Şekil 2.4: Jira, Kanban tahtası.	15
Şekil 2.5: Jira, Scrum projesi.	15
Şekil 3.1: İlk sorunun cevap dağılımı.	16
Şekil 3.2: İkinci soru için verilen genel cevaplar.	17
Şekil 3.3: Platformların yapay zekadan faydalandığını düşünüyor musunuz? Sorusuna verilen cevaplar.	17
Şekil 3.4: Platform tercihi sorusunun cevapları.	18
Şekil 3.5: Platformların ihtiyaçlarını karşılayıp karşılamadığını sorduğumuzda aldığımız cevaplar.	18
Şekil 3.6: Yapay zeka sorusuna yöneticiler tarafından verilen cevaplar.	28
Şekil 3.7: Giriş yapma formu.	20
Şekil 3.8: Kayıt olma formu.	20
Şekil 3.9: Kayıt olduktan sonra maile gönderilen kodun girildiği kısım.	20
Şekil 3.10: Kayıt olma sayfası tasarımı.	21
Şekil 3.11: Giriş yapma sayfası tasarımı.	21
Şekil 3.12: Yan panel görünümü.	22

SİMGELER VE KISALTMALAR

Bu çalışmada kullanılmış simgeler ve kısaltmalar, açıklamaları ile birlikte aşağıda sunulmuştur.

Simgeler

Açıklamalar

MB

MegaByte

Kısaltmalar

Açıklamalar

SRS

Software Requirements Specification

SDD

Software Design Document

JS

Javascript

SPMP

Software Project Management Plan

AWS

Amazon Web Servisleri

1. GİRİŞ

Günümüzde kullanılan ana akım yazılım geliştirme sistemleri, her büyüklükte proje için tasarlanmamıştır. Bu nedenle orta ve küçük kapsamlı projeler için fazla karmaşık olabilmektedir. Ayrıca bu projeler yapay zekadan ya çok az faydalanmaktadır, ya da hiç faydalanmamaktadır. Bu nedenle gerçekleştirilecek olan sistem ile birlikte bu eksikler giderilmiş olacaktır. Bu proje için daha önceden planlamayı içeren SPMP(Software Project Management Plan), gereksinimlerin sunulup açıklandığı SRS(Software Requirement Specification), ve sistem hakkında daha detaylı bilgilerin yer aldığı SDD(System Description Document) hazırlanmıştır. Bu raporlarda sistemin fizibilitesi, gereksinimlerin belirlenmesi, ve doğrulanması gerçekleştirilmiştir.

Problem durumu / Konunun tanımı

Proje yönetim yazılımlarının orta ve küçük ölçekli proje geliştiren takımlar için yetersiz olması sonucunda böyle bir sisteme ihtiyaç duyulmuştur. Bu sistemlerin yapay zekayı neredeyse hiç kullanmıyor olması ve karmaşıklığı, bu projeye olan ihtiyacı artırmıştır.

Araştırmanın amacı

Sistemin amacı, orta ve küçük ölçekte projeler için daha çok kullanılabilir bir proje yönetim yazılımı ortaya koymaktır. Ayrıca bu sistem içerisinde yapay zekanın mümkün olduğu kadar fazla kullanılması, yapay zekanın gücü ile yazılım geliştirme dünyasını birleştirmek amaçlanmaktadır.

Projelerde zaman tahmini yeterince zor bir konu olduğundan, yapay zekanın daha çok bu kısımda kullanılması planlanmaktadır. Kişilerin daha önce yaptığı görevler ve görevlerin zorluklarına göre belirlenecek olan zaman sınırı, projelerin bitirilme süreleri için daha yakın tahminler yapılmasına olanak sağlayacaktır.

Araştırmanın önemi

Yazılım ile dönen dünyamızda, her gün farklı fikirler dünyayı değiştirebilmektedir. Yazılım ile yapılacak şeylerin neredeyse sınırı olmadığından, her gün küçük ve orta ölçekli yüzlerce proje oluşturulmaktadır. Fakat bu projelerin oluşturulduğu ortamlar, ana akım proje geliştirme araçları, bazen bu projeler için bir basamak olmak yerine engel olabilmektedir. Bu ana akım proje geliştirme araçlarının yetersiz olması, yeni bir takım yazılım geliştirme aracına ihtiyaç doğurmuştur.

Yapay zekanın kullanımının ana akım proje yönetimi yazılımı geliştirme sistemlerinde çok az olması ya da hiç olmaması, projelere bu konuda yardım etmemesi büyük bir eksiklikler. Bu eksikliğin giderilmesi ile projelerin geliştirme hızları, bitirme zamanı için tahminlerin daha doğru yapılması sağlanacaktır.

Varsayımlar / Sayıtlar

Günümüzdeki proje yönetimi yazılımı geliştirme araçları küçük ve orta ölçekli projeler için fazlaca karmaşıktır ve yardım etmek yerine engel oluşturmaktadır.

Günümüzdeki proje yönetimi yazılımı geliştirme araçları yapay zekadan yeterince faydalanmamaktadır. Yapay zeka ile geliştirilen inanılmaz ürünler göz önüne alındığında bu sistemlerin yapay zekayı yeterince kullanmadığı anlaşılmaktadır. Ayrıca bu sistemler projenin bitiş süresine yönelik tahminlerde bulunmamaktadır.

Sınırlılıklar

Proje için bir zaman sınırı vardır. Kısıtlı bir süre içerisinde bu projeyi geliştirmek zorunda olmak bir sınır olarak karşımıza çıkmaktadır.

2. LİTERATÜR TARAMASI

Her ne kadar piyasadaki en çok kullanılan proje yönetim yazılımı geliştirme araçları bazı açılardan gereksiz olsa da, bizim sistemimiz için birçok konuda fikir verebilir. Bu nedenle sistemi gerçekleştirmeden önce, en çok kullanılan araçlara yönelik bir araştırma yapmak kuşkusuz bizim sistemimiz için de kritik öneme sahip olacaktır. Bu amaçla, *Scoro* ve

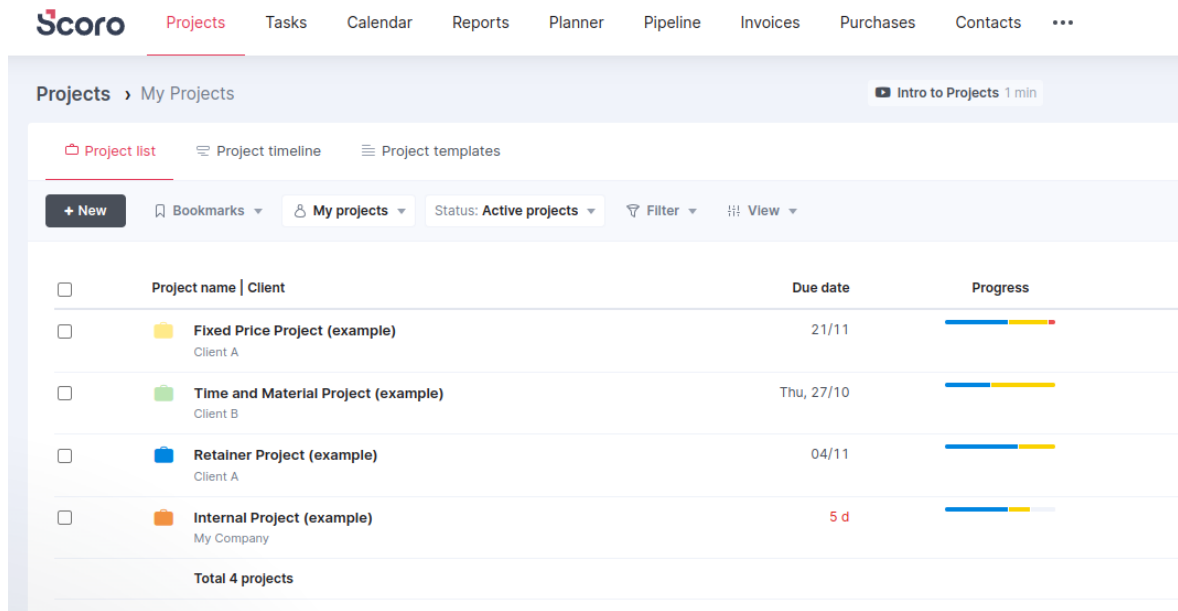
Atlassian araçlarını inceleyerek rapora ekledik.

2.1 Scoro

Yazılım projesi geliştirme araçlarının en önemlilerinden olan *Scoro*, proje yönetimine yönelik birçok yenilik getirmiştir. Daha çok büyük yazılım projelerini yönetmeyi hedefleyen *Scoro*, karmaşıklığı artırsa da birçok fonksiyonu bünyesinde barındıran, böylelikle müşteriye sunabileceği her şeyi sunmayı amaç edinmiş bir platform görünümü sunmaktadır.

Scoro'da ayırım *Projeler* kısmında başlar. Kullanıcıların projeleri olur ve her projenin kendi dünyası vardır. Projelerin içerisinde üye yönetimi, görev yönetimi, hedef belirleme gibi işlemler yapılabilmektedir.

Projeler kısmında her proje ile ilgili verilebilecek en önemli detayları verir *Scoro*. Böylelikle projelerin içine girmeden hepsi hakkında edinmesi iyi olabilecek bilgileri edinebilirsiniz.



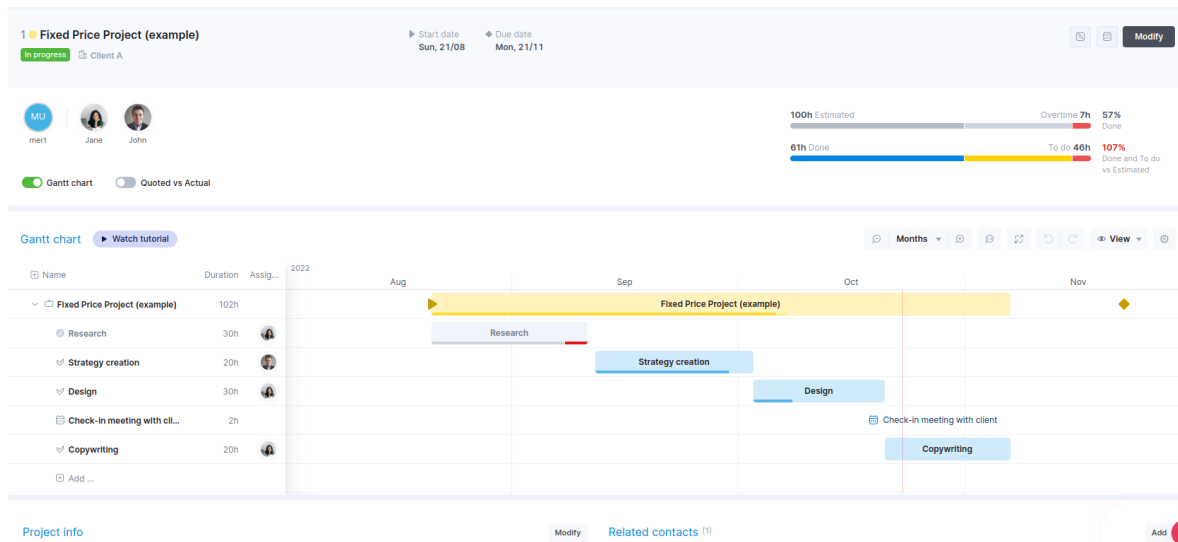
Şekil 2.1: Scoro *Projects* arayüzü.

Projenin tamamlanma yüzdesi dahil olmak üzere -muhtemelen görevler baz alınarak oluşturulmuş bir ilerleme barı- teslim tarihini de görebilmekteyiz. Scoro burada iyi bir iş çıkarmış diyebiliriz. Sadeliği ön planda tutmak bizim amaçlarımız arasında da yer aldığından, proje ile ilgili en önemli detaylar olan son teslim tarihi ve ilerleme barını

burada göstermek iyi bir tercih olmuş. Projenin ilerlemesini farklı bir şekilde göstermek bir seçenek olsa da bu nokta referans alacağımız noktalardan birisi olabilir.

Herhangi bir projenin ilk sayfasında ise, kullanıcının bilmesi gereken biraz daha detaylı bilgiler yer almakta. Bu sayfa çoğu yazılım geliştirme aracında olduğu gibi, tek sayfaya sığdırılmaya çalışılmış. Kullanıcıların proje ile ilgili bilmek istediği her şeyi tek bir sayfaya sığdırabilmek önemli bir kriter.

Burada doğru bilgileri seçmek işin kritik kısmı olacaktır. Müşteri bilgilerinden, zaman aşımına kadar birçok bilgi bu sayfada verilebilir. Ayrıca takım üyeleri, projenin gidişatı ile ilgili grafikler ve tablolar yerleştirilebilir. Bu sayfada olması gereken, Scoro’da görülmeyen bir diğer şey ise, müşteri yorumları kısmı olabilir.

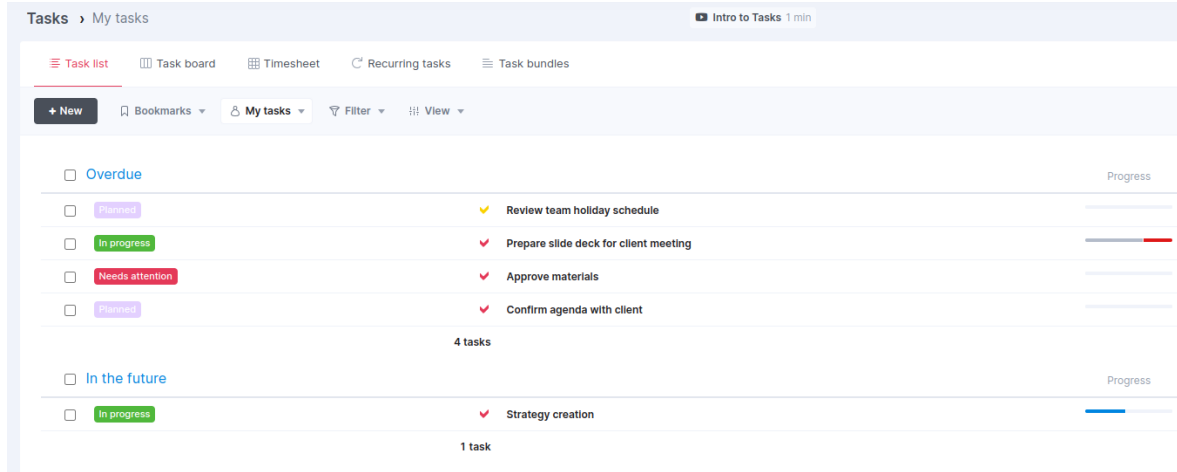


Şekil 2.2: Scoro, proje sayfası.

Müşteri ile iletişim kuran tek kişi proje yönetici olursa, kopukluklar meydana gelebilir. O kopukluğu gidermek için müşteriyi de sisteme giriş yapıp, sistemin içine dahil eden bir anlayış, bizim uygulamamız için önemli bir geliştirme olarak görülebilir. Müşterinin etkileşime geçtiği sayfalardan birisi de bu olabilir. Bazı kısımlar müşteriye gösterilmemek koşuluyla bu sayfaya giriş yaparsa, iletişimi de güçlendireceğimizi düşünüyoruz.

Diğer bir konu ise görev yönetimi. Görev yönetimi, şüphesiz, kullanışlı yapıldığı takdirde uygulamanın başarısını belirleyecek olan yegane faktörlerden biridir. Uygulamamız için üzerine çalışılacak en önemli konulardan biri olacaktır.

Görev yönetimi konusunda scoro oldukça sade bir tasarıma gitmiş. En önemli sayfalardan biri olan görev yönetimi, uygulamanın kullanışlılığını belirleyen sayfa olacaktır. Bu nedenle hem yöneticiler, hem de geliştiriciler için bazı kriterleri göz önüne alarak tasarlamalıyız.



Şekil 2.3: Scoro, proje görev yönetimi sayfası.

Herhangi bir görevin içinde ise, bu görev üzerine çalışan kişiler, açıklamalar, belirlenen son tarih ve bu görev ile ilgili işlemler bulunuyor. Görevlerin ayrı bir sayfada açılmasının bir sakıncası bulunmamakla birlikte, bu işlem çok sayfa sayfa geçişine sebep verebilir. Görevler sayfasında açılacak bir üst sayfa ile de yapılacak bir işlem, ayrı bir sayfaya alınarak detaylandırma tercih edilmiş. Biz de benzer bir yaklaşım sergileyeceğimizi söyleyebiliriz.

2.2 Atlassian Jira

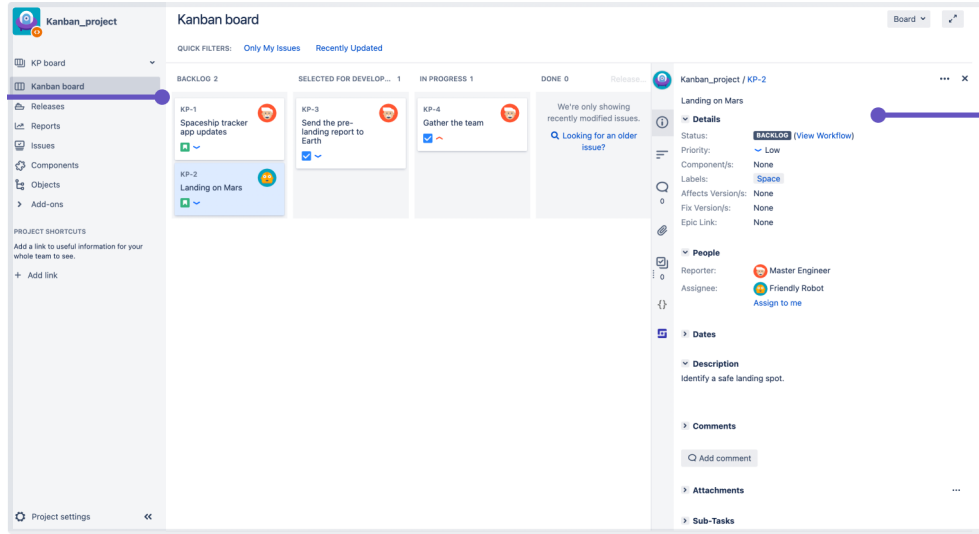
Atlassian, çevik geliştirme metodu yazılım şirketleri tarafından daha yaygın olarak kullanılmaya başlandıkça popülaritesini arttırdı ve yazılım şirketleri tarafından daha yaygın olarak kullanılmaya başlandı.

Atlassian Jira Software, sorun takibi ve proje yönetimi için kullanılan bir yazılımdır. Kullanım olarakta çevik geliştirme metodları kullanan yazılım geliştirici grupları veya organizasyonları hedef almıştır. [3]

Daha çok kurumsal şirketleri hedefleyen *Atlassian*, tüm ürünlerini bunu göz önüne alarak gerçekleştiriyor. Bu nedenle istenmeyen karmaşıklıklar ortaya çıkabiliyor. Bu karmaşıklığı yönetmek büyük şirketler için daha sonrasında çok fazla avantaj getiriyor olsa da, küçük ve orta ölçekli şirketler için çözülmesi gereken bir sorun olarak görünüyor.

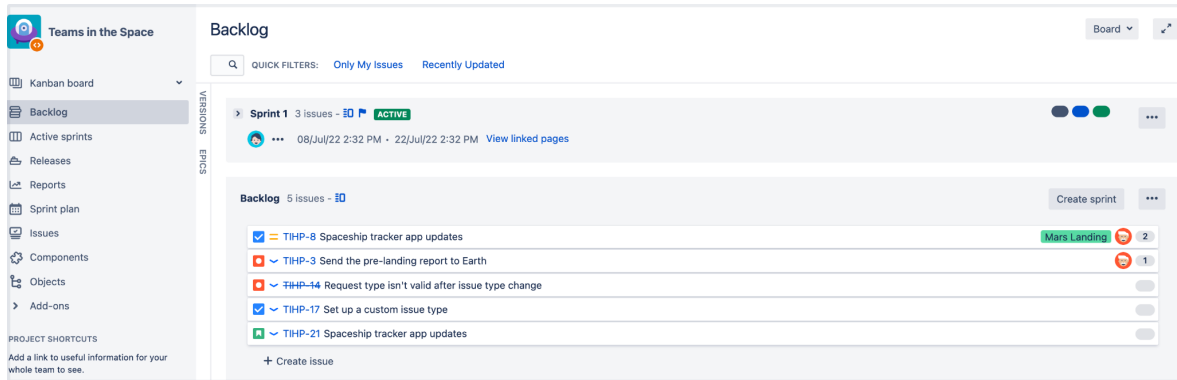
Jira yazılımında proje yönetimi platformu oluşturmak için hazır konfigürasyonu olan iki seçeneğiniz bulunur: Kanban ve Scrum. Bu iki geliştirme metodundan birini seçtiğinizde taslak olarak proje yönetim arayüzünü oluşturur. Bunun dışında geliştiriciler kendi ihtiyaçları doğrultusunda bu arayüzü düzenleyebilir.[4]

Kanban arayüzünde iş parçacıklarının gruplandırıldığı “Kanban board” adında bir sekme bulunur. Bu sekmede iş parçacıkları oluşturulabilir, bu iş parçacıklarına; ilgili kişi, etiket, yorum, öncelik gibi pek çok detay yerleştirilebilir. Bu iş parçacıkları ihtiyaç duyulduğunda diğer iş kümelerine yönlendirilebilir.



Şekil 2.4: Jira, Kanban tahtası.

Çevik geliştirme için kullanılan arayüzdür. Kanban projesinden farklı olarak Backlog için ayrı bir sekme ve “sprint”lerin yönetildiği ve görüntülendiği “Active Sprints” sekmesi vardır. Bu sekmede iş parçacıkları sprintlere ve kişilere atanır.



Şekil 2.5: Jira, Scrum projesi.

2.3 Proje Yönetiminde Yapay Zeka

Yapay zeka tabanlı araçlar, toplantı planlama, hatırlatıcılar, günlük güncellemeler ve diğer yönetim görevleri gibi işlevleri devralabilir. Bu, proje yöneticilerinin ve ekip üyelerinin daha üst düzey, karmaşık faaliyetleri planlayabilmesini sağlayacaktır.

Pulse of the Profession® anketine katılanların %80'inden fazlası, kuruluşlarının yapay zekadan faydalandığını bildiriyor. PMI'nın "AI Innovators: Cracking the Code on Project Performance" raporuna göre, önümüzdeki üç yıl içinde proje yöneticileri yapay zeka kullanarak yönettikleri projelerin %23'ten %37'ye çıkmasını bekliyor. Bunu düşünerek, yapay zekanın projenin olabildiğince her yerine yayılması gerektiği sonucunu çıkardık.

3. GELİŞTİRİLEN YAKLAŞIM VE BULGULAR

Günümüzde kullanılan ana akım birçok yazılım geliştirme aracını inceleyerek, birçok çıkarım yaptık. Bu incelemeler, çıkarımlar ve anketler bu kısımda paylaşılacaktır.

3.1 Anketler

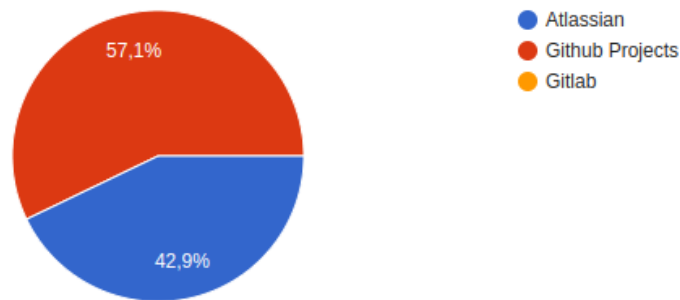
Projeye başlarken, işin içerisinde uzun süredir bulunan insanlardan fikir almak iyi bir şey olur diye düşünerek, biri yöneticiler, biri de geliştiriciler için cevaplandırılmak üzere iki adet anket hazırladık. Bu anketleri çevremizde bu pozisyonlarda çalışan insanlara dağıtarak cevaplar topladık.

3.1.1 Geliştiriciler

Geliştiricilere aşağıdaki soruları sorduk:

1. Bir ekip ile yazılım geliştirecek olsanız hangi platformu tercih edersiniz?
2. Kullandığınız platformun en sevdiğiniz özelliği ya da kullanmanızı sağlayan şey nedir?
3. Kullandığınız platformu yeterli ve kullanışlı buluyor musunuz? Eğer bulmuyorsanız hangi konuda?
4. Günümüz yazılım geliştirme programlarının yapay zekadan yeterince faydalandığını düşünüyor musunuz?

İlk soru olan platform tercihi için az bir farkla *Github Projects* seçildi. Bunu, Github'ın popülerliğine verebiliriz. Ankete katılan geliştiricilerin çoğunun yarı zamanlı çalışıyor olması ve *Atlassian* araçlarının genellikle kurumsal firmalarda kullanılıyor olması da bunun sebeplerinden biri olabilir.



Şekil 3.1: İlk sorunun cevap dağılımı.

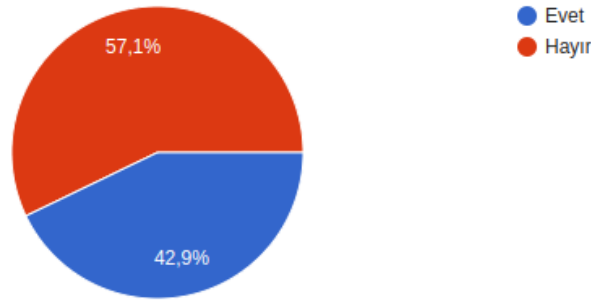
İkinci soru için ankete katılan geliştiricilerin birçoğu kullanılabilirliği ve erişebilirliği vurguladı. Kuşkusuz herkesin ilk tanıştığı git platformu olan Github için bu söylenenler doğru diyebiliriz.

Collaboration
Herkesin alışık olduğu bir platform olması
Ortak kod paylasimi
Dokümantasyon, kod ve issue'ları birbiriyle entegre biçimde çalıştırması.
Kolay kullanılabilir olması

Şekil 3.2: İkinci soru için verilen genel cevaplar.

Üçüncü soruda, katılımcılara kullandıkları platformun yeterli olup olmadığını sorduk. Ankete katılan katılımcıların %60'a yakını yeterli bulduğunu söylerken, %40'lık kısmı geliştirmeler yapılabileceğini vurguladı.

Dördüncü soruda, katılımcılara yapay zekanın bu platformlarda yeterince kullanılıp kullanılmadığını sorduk. Az da olsa çoğunluk yapay zekanın daha fazla kullanılabileceğini düşündüğünü belirtti.



Şekil 3.3: Platformların yapay zekadan faydalandığını düşünüyor musunuz? Sorusuna verilen cevaplar.

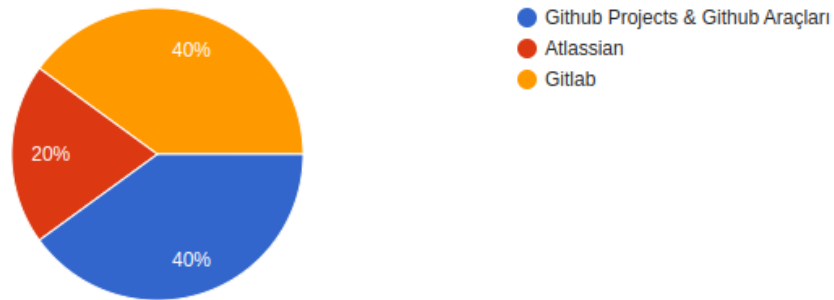
3.1.2 Yöneticiler

Yönetici konumunda çalışan insanlara gönderdiğimiz anketlerde ise farklılıklar gözümüze çarptı. Yöneticilere ise ankette şunları sorduk:

1. Orta ölçekli bir proje geliştirecek olsanız hangi platformu tercih edersiniz?
2. Kullandığınız proje yönetim yazılımlarının ihtiyaçlarınızı karşıladığını düşünüyor musunuz?
3. Günümüzdeki proje yönetim yazılımlarının yapay zekadan yeterince faydalandığını düşünüyor musunuz?
4. Eklemek istedikleriniz

Yöneticiler için farklı olan nokta, kullanılan platformlarda başladı. Geliştiriciler anketinde hiç kullanılmayan Gitlab, burada büyük çoğunluk yakaladı. Github yeniden, yöneticiler

tarafında da gayet popüler olan bir platform olarak gözümüze çarptı.



Şekil 3.4: Platform tercihi sorusunun cevapları.

İkinci soruda ihtiyaçlarını karşılayıp karşılamadığını sorduk.

Tam olarak hayır, günümüzde kullanılan çoğu araç orta ölçekli projeler için uygun değil. Basit projeler için çok karışık, büyük projeler için çok basitler.

Evet

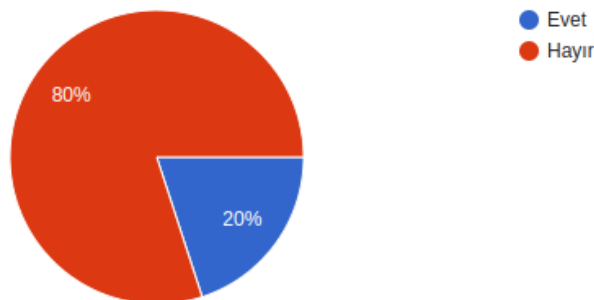
Hayır, yapay zeka'nın bu kadar gelişmesine rağmen bunun etkisi bu yazılımlarda hissedilmiyor.

Hayır, hepsi çok karmaşık. Bir proje için gerekli olan şeyler ortada. Buna göre platformların sadeleştirilmesi gerekir. Atlassian'da hiç kullanmadığım bir özellik yıllardır ana sayfamda.

Evet.

Şekil 3.5: Platformların ihtiyaçlarını karşılayıp karşılamadığını sorduğumuzda aldığımız cevaplar.

Yapay zeka konusunda ise, yöneticilerin çoğu bu gücün bu platformlarda yeterince kullanılmadığını düşünüyor.



Şekil 3.6: Yapay zeka sorusuna yöneticiler tarafından verilen cevaplar.

3.1.3 Çıkarılan Sonuçlar

Genele baktığımızda, geliştiricilerin çoğu github kullanıp, başka bir platforma ihtiyaç olmadığını düşünürken, yöneticiler için bu platformlara eklenmesi gereken şeyler var. Özellikle yapay zeka konusunda verilen cevaplar göz önüne alınırsa, platformların yapay

zeka konusunda geliřtirmesi gereken řeyler var diyebiliriz. Özellikle yöneticilerden biri yapay zeka konusunda řu eklemeyi yaptı: “Yapay zeka ile yapılan Chat GPT-3 botunu göz önüne alırsak, bu platformlar bu güçten hiç faydalanmıyormuř derim.” Bu nedenle yapay zeka konusuna projemizde daha çok önem gösterecek adımları da atma kararı aldık.

3.2 Gereksinimlerin Belirlenmesi

Yapılan literatür taraması ve anket sonucunda gereksinimleri belirlemeye başladık. Öncelikle literatür taraması sonucunda sistemin nasıl görüneceęi konusunda fikirlerimiz oluřmaya bařlamıřtı. Daha sonra anket sonuçlarını da göz önüne alarak sistemin kesinlikle sahip olması gereken birkaç maddeyi belirledik.

1. Orta ve küçük ölçekli projelerin yönetimini hedefleyen bir yapı kurmalıyız.
2. Olabildięinde Amazon Web Servisleri’nden faydalanmalıyız. Bu, geliřtirme sürecini oldukça hızlandıracak.
3. Yapay zekanın gücünü kullanabildięimiz her yerde kullanmalıyız. Dięer platformlarla aramızda fark yaratacak bu özellięe önem vermeliyiz.

Bu gereksinimlerden yola çıkarak SRS dokümanını hazırlayıp sunduk. SRS dokümanının içinde ayrıca sistemin sahip olacaęından emin olduęumuz fonksiyonları da kurgulayıp dokümana ekledik. Bu fonksiyonlar arasında sisteme kayıt olma, giriř yapma, organizasyon oluřturma, proje oluřturma ve görev oluřturup atama gibi fonksiyonlar vardı. Bunlar her yazılım geliřtirme projesinin sahip olduęu fonksiyonlar olduęundan, kurgulaması zor fonksiyonlar deęildi. Daha sonra yapay zeka işin içine girdięinde, hangi fonksiyonlarda bunu uygulayacaęımızı belirledik. Daha çok süre tahmin etmede kullanacaęımız fonksiyonların içerisinde yapay zekanın olması gerektięine karar verdik.

3.3 Uygulamanın İç Tasarımı

Gereksinimleri belirledikten sonra bu gereksinimleri hayata geçirmek için kullanacaęımız araçları tanımamız gerekiyordu. Bu sebeple kullanacaęımız Amazon Web Servisi olan *Amplify* hakkında daha detaylı bir araştırma sürecine girdik. Bu servisi kullanarak gereksinimleri nasıl gerçekleřtireceęimiz konusunda denemeler yaptık.

Yaptıęımız denemede AWS Amplify servisini kullanarak bir frontend uygulamasına(React.js) baęlandık ve giriř yapma ve kayıt olma fonksiyonlarını gerçekleřtirdik.

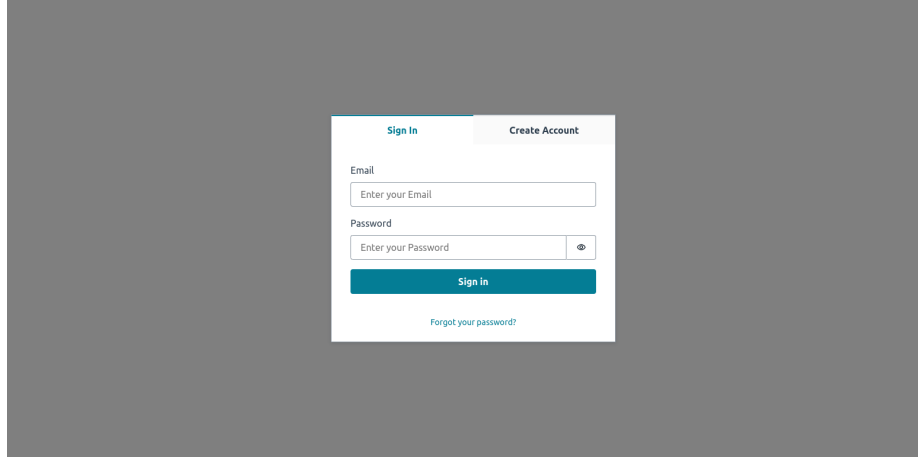
3.3.1 Demo

Kayıt olma ve giriř yapma fonksiyonlarını test etmek amacıyla AWS Amplify’ı bir frontend uygulamasına baęlayarak bir demo uygulaması geliřtirdik. Böylelikle kullanacaęımız servisin en azından bir fonksiyonunu istedięimiz gibi kullanabildięimizi gördük.

Öncelikle Amplify dokümantasyonunu takip ederek gerekli kütüphaneleri uygulamamıza ekledik. Daha sonra terminal üzerinden gerekli komutları çalıştırarak, Amplify’ın bize sunduęu yetkilendirme fonksiyonlarını uygulamamıza entegre ettik. Daha sonra

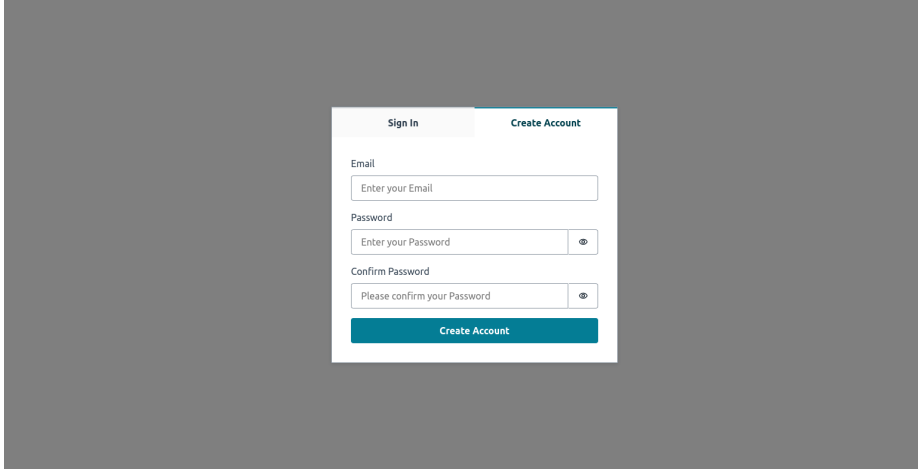
kütüphanenin içindeki fonksiyonları kullanarak frontend tarafında yapmak istediğimiz değişiklikleri yaptık ve fonksiyonları denemeye başladık.

Demo başarılı oldu ve yetkilendirme fonksiyonlarını başarılı bir şekilde çalıştırabildiğimizi gördük. Demo içerisinden görüntüleri de burada paylaşıyoruz:



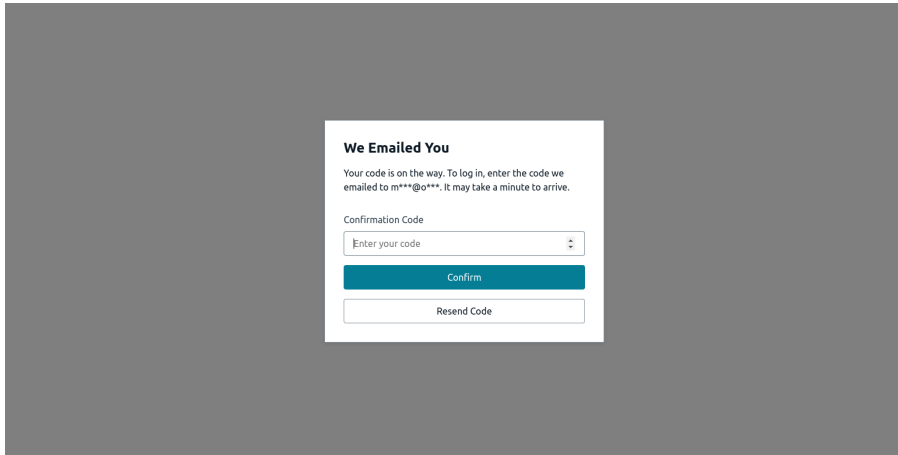
The image shows a 'Sign In' form with two tabs: 'Sign In' (active) and 'Create Account'. The form has two input fields: 'Email' with the placeholder 'Enter your Email' and 'Password' with the placeholder 'Enter your Password'. There is a 'Sign in' button and a link 'Forgot your password?' below it.

Şekil 3.7: Giriş yapma formu.



The image shows a 'Create Account' form with two tabs: 'Sign In' and 'Create Account' (active). The form has three input fields: 'Email' with the placeholder 'Enter your Email', 'Password' with the placeholder 'Enter your Password', and 'Confirm Password' with the placeholder 'Please confirm your Password'. There is a 'Create Account' button.

Şekil 3.8: Kayıt olma formu.



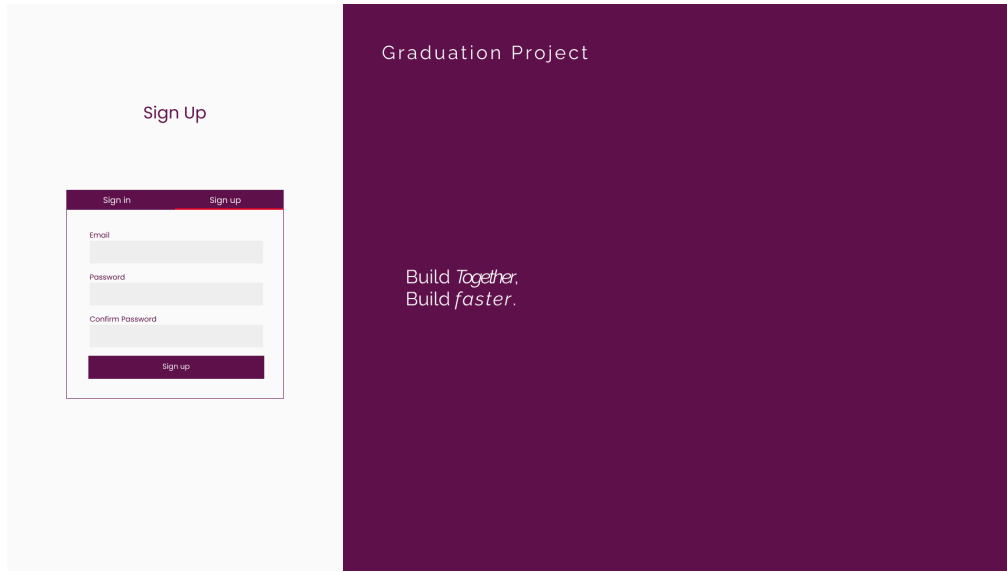
The image shows a 'We Emailed You' confirmation form. It has a title 'We Emailed You' and a message: 'Your code is on the way. To log in, enter the code we emailed to m***@o***. It may take a minute to arrive.' There is a 'Confirmation Code' input field with the placeholder 'Enter your code', a 'Confirm' button, and a 'Resend Code' button.

Şekil 3.9: Kayıt olduktan sonra maile gönderilen kodun girildiği kısım.

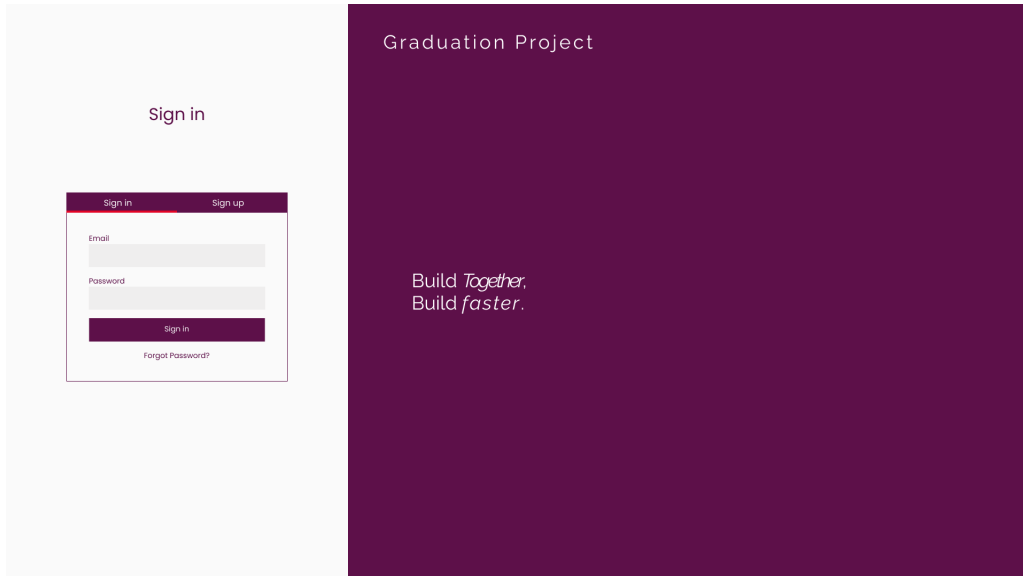
Görüntü olarak sistemin son halini teslim etmese de fonksiyonların çalıştığını görmemiz açısından önemli bir demo oldu. Böylelikle istediğimiz gibi AWS Servislerini uygulamamıza dahil etmeyi ve kullanmayı öğrenmiş olduk.

3.3.2 Gerçekleştirilen Tasarımlar

Uygulamanın yetkilendirme kısmı için arayüz tasarımı tamamlanmıştır. Aşağıda bu tasarım ile ilgili görselleri görebilirsiniz:



Şekil 3.10: Kayıt olma sayfası tasarımı.



Şekil 3.11: Giriş yapma sayfası tasarımı.

Tasarım kısmında ayrıca yan panelin nasıl görüneceği de belirlenmiştir.



Şekil 3.12: Yan panel görünümü.

Yan panelde projeler için ilerleme durumları görülmektedir. Ayrıca bu panel üzerinden insan kaynakları yönetimi ve organizasyon için gerekli ayarlamalar da yapılabilmektedir.

3.4 Kullanılan Araçlar ve Tercih Sebepleri

Uygulamanın geliştiriminde kullanılacak araçlar için hem sunucu tarafında hem kullanıcı tarafında gereksinimlerin değişimine ayak uydurabilmek için, ölçeklenebilir araçlar araştırılmış ve tercih edilmiştir.

3.4.1 Sunucu Tarafında(Back-end) Kullanılan Araçlar

Uygulamanın sunucu kısmı ölçeklenebilir ve kullanımının kolay olmasından dolayı AWS üzerine kurulmasına karar verilmiştir. Hem klasik web çözümlerine olan uyumluluğu hem de kendi çözümlerine sahip olması bize çok büyük kolaylıklar sağlamıştır. Sunucuda kullanılan bazı araçlar bu bölümün devamında detaylandırılmıştır.

3.4.1.1 Veritabanı

AWS üzerine sistemi kurmaya karar verdiğimizde iki veritabanı çeşidi arasında seçim yapmamız gerekti. Bunlardan birisi AWS RDS diğeri ise AWS DynamoDB veritabanı. DynamoDB veritabanının RDS e kıyasla diğeri servisler ile daha uyumlu olması ve diğeri sistemlere entegre etme kolaylığından ötürü tercih edilmiştir. Sunucu tarafında sıkça

kullandığımız işlem birimi olan AWS Lambda üzerinden veritabanı işlemlerinin DynamoDB üzerindeki işlemlerin kolayca yapılması buna bir örnektir.

3.4.1.2 API

API olarak AWS üzerinde 3 seçenek arasında kaldık ve araştırmalarımız sonucunda API'ın GraphQL olmasına karar verdik. Yenilikçi bir GraphQL olması ve uygulamanın kullanıcı tarafını kurduğumuz AWS Amplify servisi ile olan uyumluluğu ve kullanım kolaylıkları seçmemizde etkili oldu. Ayrıca daha önce bu tarz bir API tasarlamadığımız için bilgi dağarcığımıza bunuda eklemek, öğrenmek istedik.

3.4.2 Kullanıcı Tarafında(Front-end) Kullanılan Araçlar

Gereksinim araştırması yaptığımızda kullanıcı etkileşiminin yüksek olacağını belirledik. Bundan dolayı uygulamanın kullanıcı tarafında tek bir sayfa üzerinde dinamik objeler oluşturarak tekrar yüklenmeyen tek bir sayfa olmasına karar verdik. Bu verinin ışığında proje için uygun olan iki seçenek arasında kaldık. Bunlardan birisi Vue, diğeri ise React oldu. React üzerine bolca kaynak bulunması ve kendini ispatlamış olmasından ötürü React kütüphanesi ile geliştirmeye karar verdik

4.SONUÇ VE ÖNERİLER

Sonuç olarak platform, yapay zeka konusunda ve küçük ve orta ölçekli projeler konusunda gerekli görülmektedir. Ana akım yazılım geliştirme araçlarından farklı olarak yapay zeka üzerine yoğunlaşacak olan platform, proje tahminlerini daha gerçekçi kılmak adına birçok fonksiyon içerecektir. Bu fonksiyonlar Amazon tarafından desteklenen web servisleri tarafından sağlanacağından, en iyi pratiklerle gerçekleştirilmiş olacaktır.

Yapılabilirlik açısından herhangi bir engel görülmemekle birlikte, gerekli denemeler ve demo gerçekleştirilmiştir. Böylelikle sistemin fizibilitesi de test edilmiştir.



Gazili olmak ayrıcalıktır