

T.C.
SAKARYA ÜNİVERSİTESİ
BİLGİSAYAR VE BİLİŞİM BİLİMLERİ FAKÜLTESİ

BSM 498 BİTİRME ÇALIŞMASI

YENİ NESİL İSTİHDAM UYGULAMASI

G181210554 – Maftun HASHİMLİ
G181210020 - Berkay ÇETE

Fakülte Anabilim Dalı : BİLGİSAYAR MÜHENDİSLİĞİ
Tez Danışmanı : Prof. Dr. İbrahim Özçelik
Dr. Öğr. Üyesi Murat İSKEFİYELİ

2021-2022 Yaz Dönemi

T.C.
SAKARYA ÜNİVERSİTESİ
BİLGİSAYAR VE BİLİŞİM BİLİMLERİ FAKÜLTESİ

YENİ NESİL İSTİHDAM UYGULAMASI

BSM 498 - BİTİRME ÇALIŞMASI

G181210554 – Maftun HASHİMLİ
G181210020 - Berkay ÇETE

Fakülte Anabilim Dalı : BİLGİSAYAR MÜHENDİSLİĞİ

Bu tez .. / .. / ... tarihinde aşağıdaki jüri tarafından oybirliği / oyçokluğu ile kabul edilmiştir.

.....
Jüri Başkanı

.....
Üye

.....
Üye

ÖNSÖZ

Günümüzde insanların hayatlarını sürdürebilmesi için en önemli faktörlerden birisi para kazanmaktır. Para kazanabilmek için insanların iş bulması gerekmektedir. Fakat şu anki koşullarda insanların iş bulması, bulduğu işte hak ettiği parayı kazanması oldukça zorlaşmıştır. Aynı zamanda firmaların da kendilerine uygun çalışanları bulması çalışan insan çokluğu sebebiyle oldukça güçleşmiştir.

Bu sebeple iş arayışında olan insanların iş bulmalarını, çalışan arayan iş verenlerin çalışan bulmalarını kolaylaştırmak ve tek bir çatı altında toplamak adına bu proje hayata geçirilmiştir.

Proje kullanıcılar için kendilerine en uygun işe ulaşmalarını ayrıca iş verenlerin de firmalarına en uygun çalışanı bulmalarını hedeflemektedir.

İÇİNDEKİLER

ÖNSÖZ.....	iii
İÇİNDEKİLER.....	iv
ŞEKİLLER LİSTESİ.....	vii
TABLolar LİSTESİ.....	ix
ÖZET.....	10
BÖLÜM 1.	
GİRİŞ.....	11
1.1. Proje tasarım.....	11
1.2. Proje amacı.....	12
1.3. Proje kapsamı.....	12
1.4. Proje hedefleri.....	13
BÖLÜM 2.	
PROJE ANALİZ SONUÇLARI.....	14
2.1. Proje kartı.....	14
2.2. Toplum Analiz Sonuçları.....	15
2.3. Proje maliyet analizi.....	20
2.4. Risk listesi ve olasılık etki matrisi.....	21
2.4.1. Risk listesi.....	21
2.4.2. Olasılık etki matrisi.....	22
2.5. Proje başarı kriterleri.....	23

BÖLÜM 3.

İŞ AKIŞI	25
3.1. Roller.....	25
3.2. Şirketler.....	25
3.3. İş arayışında olan kullanıcılar.....	25
3.4. Arama ve filtreleme.....	26
3.5. Eşleştirme ve öneri.....	26
3.6. İlan.....	26

BÖLÜM 4.

PROJE TASARIM DÖKÜMANTASYONU.....	27
4.1. Uygulama Sayfa Tasarımları.....	27
4.1.1. Ana Sayfa Ekranı.....	27
4.1.2. Kayıt Olma Ekranı.....	28
4.1.3. Giriş Yapma Ekranı.....	29
4.1.4. İlan Listeleme Ekranı.....	30
4.1.5. İlan Detay Ekranı.....	31
4.1.6. İlan Oluşturma Ekranı.....	32
4.1.7. Profil Ekranı.....	33
4.1.8. Şifre Değiştirme Ekranı.....	34
4.1.9. Admin Giriş Ekranı.....	34
4.1.10. Ek Ekran Görüntüleri.....	35
4.2. Veritabanı İlişkisel Gösterim Diyagramı.....	39
4.3. Akış Şemaları.....	40
4.3.1. İlana Başvurma İşlemi.....	40
4.3.2. Firmaların İlan Oluşturma İşlemi.....	42
4.3.3. Kullanıcıların Bilgilerini Güncelleme İşlemi.....	43
4.3.4. Firmaların Bilgilerini Güncelleme İşlemi.....	44
4.4. Sistem Diyagramı.....	45
4.5. Kaynak Kodları.....	46

BÖLÜM 5.

KULLANILAN TEKNOLOJİLER.....	50
5.1. Java.....	50
5.2. React.....	50
5.3. Spring Boot.....	51
5.4. JWT.....	51
5.5. JavaScript.....	52
5.6. Axios.....	52
5.7. Jenkins.....	53
5.8. Bootstrap.....	53
5.9. CSS.....	53
5.10. GIT.....	54
5.11. PostgreSQL.....	54
5.12. Postman.....	54

BÖLÜM 6.

KULLANILAN ALGORİTMALAR.....	55
6.1. Apriori Algoritması.....	55
6.1.1. Apriori Algoritması Nedir.....	55
6.1.2. Algoritmanın Çalışması.....	55
6.1.3. Algoritmanın Projede Kullanımı.....	57
6.2. Naive Bayes Algoritması.....	58
6.2.1. Naive Bayes Algoritması Nedir.....	58
6.2.2. Algoritmanın Çalışması.....	58
6.2.3. Algoritmanın Projede Kullanımı.....	59

KAYNAKLAR.....	60
----------------	----

ÖZGEÇMİŞ.....	62
---------------	----

BSM 498 BİTİRME ÇALIŞMASI DEĞERLENDİRME VE SÖZLÜ SINAV TUTANAĞI.....	63
---	----

ŞEKİLLER LİSTESİ

Şekil 2.1.	Proje Kartı.....	14
Şekil 2.2	Katılım Sağlayanların Cinsiyet Dağılımı.....	15
Şekil 2.3	Katılım Sağlayanların Yaş Dağılımı.....	16
Şekil 2.4	Katılım Sağlayanların Öğrenim Durumu.....	16
Şekil 2.5	Katılım Sağlayanların Gelecek Kaygı Durumu.....	17
Şekil 2.6	Katılım Sağlayanların İş Bulma Düşüncesi.....	17
Şekil 2.7	Katılım Sağlayanların Kariyer Sitesi Kullanımı.....	18
Şekil 2.8	Katılım Sağlayanların Site İlanlarına Güveni.....	18
Şekil 2.9	Katılım Sağlayanların Alternatif İsteği.....	19
Şekil 4.1	Ana Sayfa Ekranı.....	27
Şekil 4.2	Kayıt Ekranı.....	28
Şekil 4.3	Giriş Ekranı.....	29
Şekil 4.4	İlan Listeleme Ekranı.....	30
Şekil 4.5	İlan Detay Ekranı.....	31
Şekil 4.6	İlan Oluşturma Ekranı.....	32
Şekil 4.7	İlan Oluşturma Ekranı Devamı.....	32
Şekil 4.8	Profil Ekranı.....	33
Şekil 4.9	Profil Ekranı Devamı.....	33
Şekil 4.10	Şifre Değiştirme Ekranı.....	34
Şekil 4.11	Admin Giriş Ekranı.....	34
Şekil 4.12	Admin İş Tipi.....	35
Şekil 4.13	Admin Etiketleri.....	35
Şekil 4.14	Admin Yorumları.....	35
Şekil 4.15	Moderatör Ekranı.....	36
Şekil 4.16	Kullanıcı Özgeçmişi.....	36
Şekil 4.17	Etiket Filtresi.....	36
Şekil 4.18	Naive Bayes Kullanımı.....	37
Şekil 4.19	Naive Bayes Kullanımı Devamı.....	37
Şekil 4.20	İş Arama.....	37
Şekil 4.21	Kabul Edilen İşler.....	38

Şekil 4.22	Veritabanı İlişkisel Gösterim Diyagramı.....	39
Şekil 4.23	İlana Başvurma Akış Diyagramı.....	41
Şekil 4.24	İlan Oluşturma Akış Diyagramı.....	42
Şekil 4.25	Kullanıcı Bilgilerini Güncelleme Diyagramı.....	43
Şekil 4.26	Firma Bilgilerini Güncelleme Diyagramı.....	44
Şekil 4.27	Sistem Diyagramı.....	45
Şekil 4.28	Kayıt Olma Metodu.....	46
Şekil 4.29	Servis Katmanı Metodu.....	46
Şekil 4.30	Kontrol Katman Metodu.....	47
Şekil 4.31	Kayıt İşlemi Gösterimi.....	47
Şekil 4.32	Veritabanına Kayıt Eklenmesi.....	48
Şekil 4.33	Oturum Süresi Dolumunda Oluşan Cevap.....	48
Şekil 4.34	Yetkisiz İşlem Cevabı.....	48
Şekil 4.35	Başarılı Doğrulama İşlemi.....	49
Şekil 4.36	Başarılı Kayıta Güncelleme İşlemi.....	49

TABLÖLAR LİSTESİ

Tablo 2.2.	Proje Maliyet Tablosu.....	20
Tablo 2.3.1.	Risk Listesi.....	21
Tablo 2.3.2.	Olasılık Etki Matrisi.....	22
Tablo 2.4.	Proje Başarı Kriterleri.....	23
Tablo 2.4.	Proje Başarı Kriterleri (devamı).....	24

ÖZET

Anahtar kelimeler: web platformu, istihdam, iş, kariyer, çalışan

Web platformu olarak geliştirilmesi planlanan sisteme kullanıcılar kendilerine ait hesaplarını oluşturarak giriş yapabilecekler. Kullanıcılar bu oluşturdukları hesaplarında kendilerine ait bilgileri dolduracak ve iş arayışına çıkacaklardır. Sistemde gezinerek kendilerine uygun ilanları favorilerine ekleyebilir veya direkt olarak açık ilanlara başvurarak sistemi kullanabileceklerdir.

Sisteme katılabilmek için iş verenler de iş veren hesaplarını oluşturacak ve firmalarına ait bilgileri dolduracaklardır. Ardından ihtiyaç duydukları çalışanı bulabilmek adına kendi iş ilanlarını oluşturup sistemde paylaşabileceklerdir.

Uygulamanın gerçekleşmesi için ReactJS, Java, RestAPI, Redux gibi teknolojiler kullanılırken aynı zamanda proje Apriori Algoritması kullanılarak desteklenmiştir.

BÖLÜM 1. GİRİŞ

Günümüzde iş arayışı ve uygun kalifiye çalışan edinme durumu önemli ve dikkat gerektiren bir konudur. Bu kapsamda çalışanların uygun iş fırsatlarını değerlendirmesi, işverenlerin kendilerine uygun çalışanlara ulaşmasına yönelik bir proje geliştirilmesi amaçlanmıştır.

1.1. Proje Tasarımı

Proje tasarımında kullanıcıların deneyimlerine önem verilecektir. Kullanıcıların sisteme kolay bir şekilde uyum sağlayıp kullanıcı dostu olması hedeflenmektedir. Böylece sistem yeterli kullanıcı sayısında daha kolay ulaşabilecektir.

Tasarımda her iki kullanıcı tipinin sistemden en fazla verim alacağı şekilde proje gerçekleştikten sonra gerekli iyileştirme ve geliştirmeler yapılacak olup sistem çalıştıktan sonra da destek devam edecektir. Böylece sistemde bulunan kullanıcıların sistemi kullanma süresinin uzaması planlanarak sistemin süreklilik kıstasına dayandırılması istenmektedir.

Sistem hayata geçtikten sonra sistem yönetici hesapları olup sistemdeki hatalı ve kasten kötü kullanımların takibi yapılacaktır. Bunun sayesinde sistem kullanıcı deneyimini düşürmeden kullanıcılara temiz bir kullanım kolaylığı sağlayacaktır. Proje bu sayede dinamik bir yapıya kavuşacak ve gelişim sağlayacaktır.

1.2. Proje Amacı

Sistemde bulunan iki tip kullanıcının sistemden en fazla verimi alması amaçlanmaktadır. Sistemdeki ilk kullanıcı tipi olan işverenler kendi çalışma bünyelerine gerekli olan çalışanları bulmak amacıyla iş arayan kullanıcı tipinde olanları bulabilir ve onlara iş ilanı açabilmektedir.

Bir diğer kullanıcı tipi olan çalışanlar ise sistemde bulunan işverenlerin hesaplarında bulunan bilgilerden faydalanarak kendilerine uygun işverenleri bularak onlarla etkileşime geçebilir. Ayrıca işverenlerin açmış olduğu ilanları görüntüleyerek kendilerine en uygun olan iş fırsatını bularak bu ilana başvurarak işverenlerle etkileşime geçebilir.

Sistemin gelir kalemi işveren ve çalışanların açtığı ilanların sisteme ödeme yaparak açılması kurgusu üzerine kurulmuştur.

1.3. Projenin Kapsamı

İşverenler sistem sayesinde kendi iş bünyelerine uygun çalışanı kolayca bulup kendi iş gelişimine fayda sağlayacaktır. Bu sayede iş arayışında olan çalışanları kolayca bulup onlarla iletişime geçebilir veya uygun çalışanların kendilerine ulaşması için ilanını açabilmektedir.

İş arayışında olan çalışanlar kendi kariyerlerine, yetkinliklerine uygun olan iş ilanları üzerinden görüntüleyerek daha doğru ve emin bir kariyer adımı atmış olacaktır.

Sistem hem çalışanın hem de işverenin yanında olmaktadır. Bu kapsamda sistemin daha yaygın bir kullanıcı kitlesine ulaşması beklenmektedir. Bu kapsamda sistem sayesinde iki tip kullanıcı da en hızlı, kolay, sürdürülebilir, güvenilir ve doğru tercihlerde bulunmuş olup maksimum fayda sağlanması beklenir.

1.4. Projenin Hedefleri

Projenin gerçekleştirmek istediği hedefler aşağıdaki gibidir:

- Çalışan tipindeki kullanıcıların işverenlerle kolay şekilde etkileşime geçebilmesi.
- İşveren tipindeki kullanıcıların çalışanlarla kolay bir şekilde etkileşime geçebilmesi.
- İşverenlerin açtığı ilanlarla kendilerine en uygun çalışanı hızlı bir şekilde bulabilmesi.
- İş arayışında olan ve kariyer adımı atmak isteyen kullanıcıların kendi yetkinliklerine göre ilanları görerek kendilerine en uygun işe ulaşabilmeleri.
- İş arayışında olmasa dahi çalışan kullanıcıların işverenlerden etkileşime geçilmesiyle kariyer gelişimini yapabilmesi.

BÖLÜM 2. PROJE ANALİZ SONUÇLARI

Projenin geliştirme adımları düşünülerek bir plan ortaya koyulmuş, hedef süre ve maliyet analizi yapılmıştır. Proje baştan sonuna kadar iki kişi ile birlikte yürütülecektir.

2.1. Proje Kartı

Proje adı, hedef tarihler, temel bilgiler, proje yaklaşım ve sonuçlarını içeren bilgi karta eklenmiştir. Bunun yanı sıra proje amaçları, görev paylaşımı, proje yürütenlerin sorumlulukları da karta eklenerek açıklanmıştır.

PROJE KARTI			
Proje Adı: Yeni Nesil İstihdam Platformu			
Proje Başlangıç Tarihi:		07-07-2022	Proje Bitiş Tarihi:
			27-08-2022
Proje Yöneticisi		Berkay Çete, berkay.cete@ogr.sakarya.edu.tr, Maftun Hashimli, maftun.hashimli@ogr.sakarya.edu.tr	
Projenin Amaçları: Günümüzde pandeminin de etkisiyle iş bulmanın daha da zorlaştığı bugünlerde iş arayışında olan insanların kendilerine en uygun iş bulma sürecini en kısa ve en doğru şekilde gerçekleştirebilmesi, çalışan arayışında olan işverenlerin kendilerine uygun çalışanı en kısa sürede bulabilmelerine olanak sağlayacak bir platform geliştirmek amaçlanmaktadır. Projede yazılım, donanım, servis, hosting, domain maliyetleri için öngörülen bütçe 1000\$ ve çalışan maliyetleri için öngörülen bütçe 0\$'dır.			
Yaklaşım ve Sonuçlar: <ul style="list-style-type: none">• Projenin önyüzünün tasarlanması ve örnek dizaynların yapılması.• Veritabanı tasarımının yapılması.• Önyüz ve arkayüzün eşzamanlı gerçekleşmesi.• Test verilerinin girişi.• Gerekli takdirde veritabanı optimizasyonunun yapılması.• Projenin canlı ortamda yayına alınması.			
Görev ve Sorumluluklar			
İSİM	GÖREV	SORUMLULUK	
Maftun Hashimli	Proje yöneticisi, yazılım geliştiricisi, veritabanı tasarımcısı	Proje fikrinin planlanması ve yönetilmesi, ihtiyaç duyulan teknolojilerin belirlenmesi, gerçekleştirme sürecinde ortaya çıkan aksaklıkların giderilmesi, veritabanı tasarımının yapılması ve optimizasyonunun sağlanması, kurtarma ve felaket senaryolarına hazırlık yapılması.	
Berkay Çete	Bilişim ve teknoloji yöneticisi, yazılım geliştiricisi, kullanıcı deneyimi ve arayüzü tasarımcısı	Projenin gerçekleştirilmesi, arayüz ve tasarım gereksinimlerinin sağlanması, yapılan iyileştirme ve değişikliklerin kullanıcıya nasıl yansıdığını izlemek ve değerlendirmek, arayüzü kullanıcı dostu hale getirmek.	
İmzalar			
Maftun Berkay			

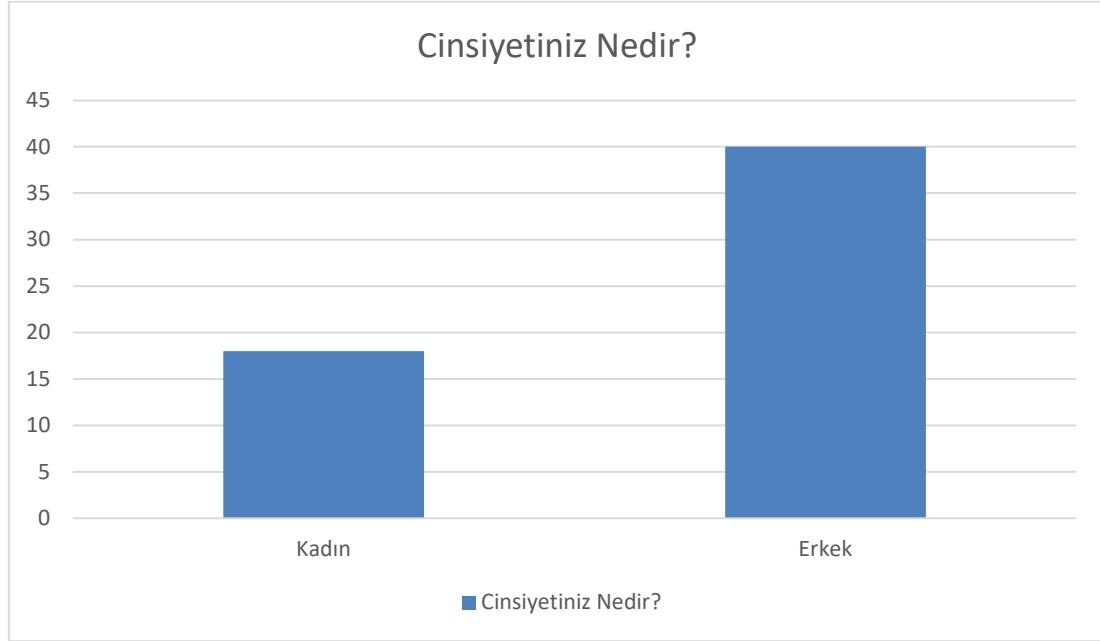
Şekil 2.1 Proje Kartı

2.2. Toplum Analiz Sonuçları

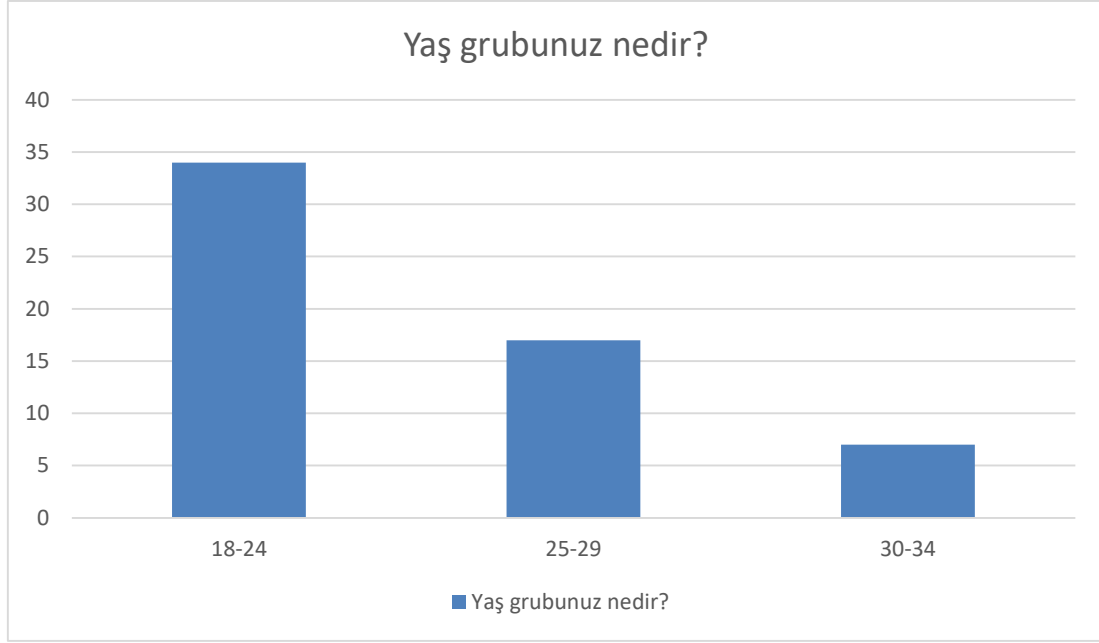
Geliştirecek olduğumuz projenin hitap ettiğini düşündüğümüz kullanıcı kitlesine iş arama, kariyer siteleri ile ilgili bazı soruları içeren bir anket oluşturduk.

Söz konusu anket ile kullanıcıların iş bulma süreçlerinde ne kadar hızlı yol alacaklarını, öğrenim durumlarını, iş ararken kullandıkları sitelerden memnuniyetlerini, sitelerde bulunan ilanlarla ilgili düşüncelerini öğrenebilme imkanına sahip olduk.

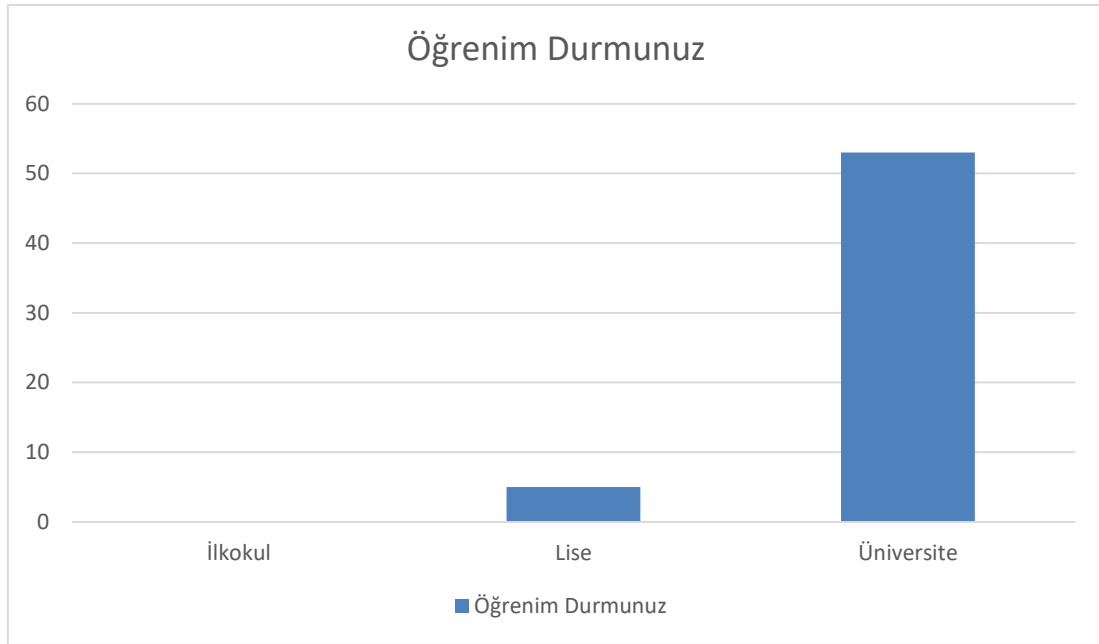
Ankete farklı yaş ve cinsiyet gruplarından 58 kişi katılım göstermiştir.



Şekil 2.2. Katılım sağlayanların cinsiyet dağılımı



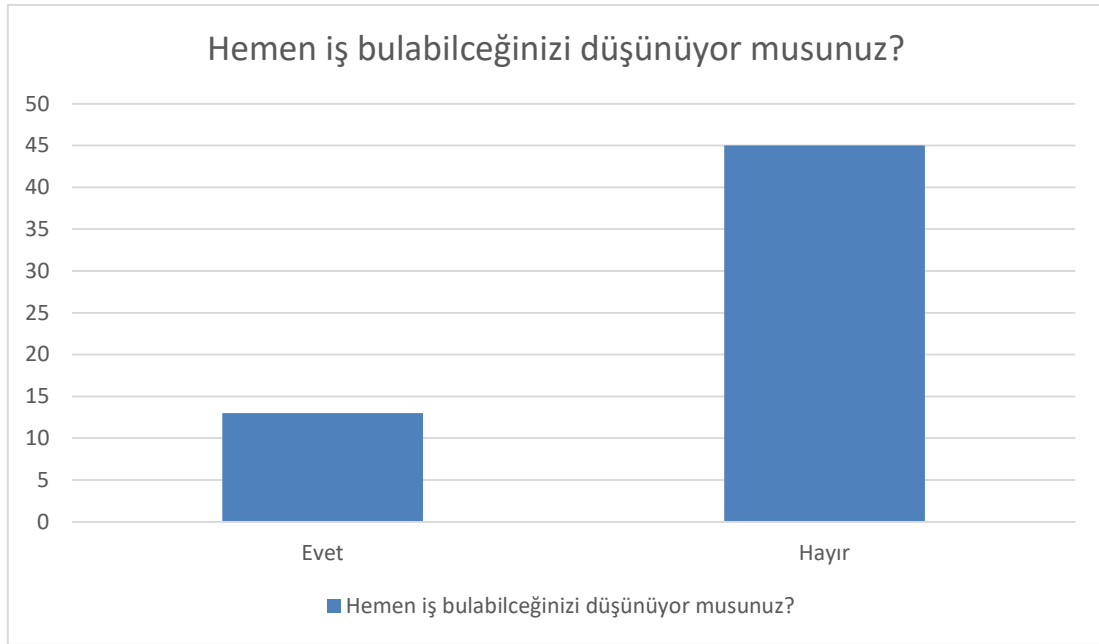
Şekil 2.3. Katılım sağlayanların yaş dağılımı



Şekil 2.4. Katılım sağlayanların öğrenim durumu



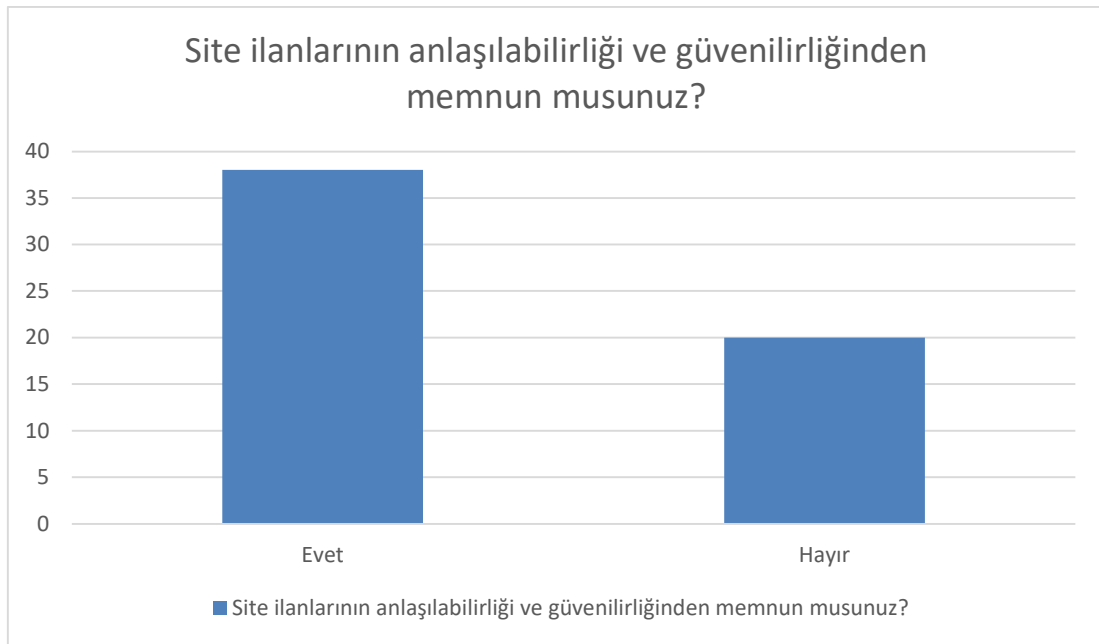
Şekil 2.5. Katılım sağlayanların gelecek kaygı durumu



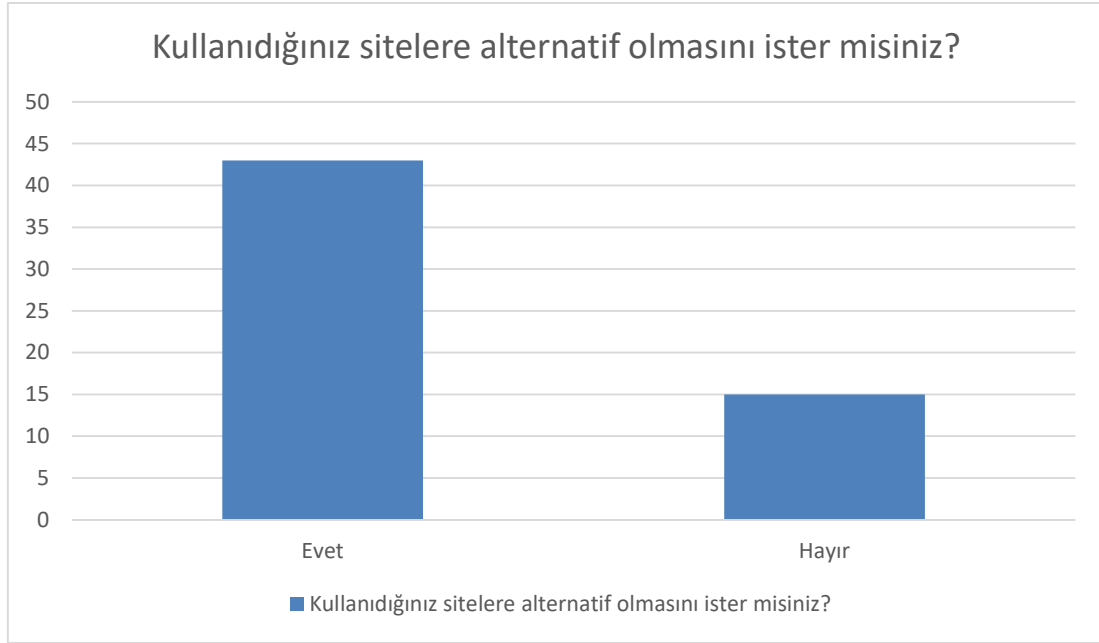
Şekil 2.6. Katılım sağlayanların iş bulma düşüncesi



Şekil 2.7. Katılım sağlayanların kariyer sitesi kullanımı



Şekil 2.8. Katılım sağlayanların site ilanlarına güveni



Şekil 2.9. Katılım sağlayanların alternatif isteği

Anket sonuçları incelendiğinde ankete farklı yaş, cinsiyet ve öğrenim durumu gruplarından insanların katılım gösterdiği görülmektedir. Ankete erkeklerin daha çok katılım gösterdiği ve katılımcılarının yüksek çoğunluğunun öğrenim durumunun üniversite olduğu gözlenmektedir. Ayrıca katılımcıların yaş grubunun oldukça genç olduğu görülmektedir.

Katılımcılar genel olarak iş bulmalarının zor olduğunu ve gelecek kaygısını taşıdıklarını söylemektedir. Günümüzde bir çok insan gelecek kaygısı ve iş bulma zorluklarından yakınmaktadır, anket sonuçlarıyla da bu tekrar gözükmektedir.

İnsanların çok büyük çoğunluğu iş aramak için kariyer sitelerini kullandıklarını belirtmelerine rağmen hala azımsanmayacak bir grup ilanlardan memnun olmadıklarını dile getirmekte ve alternatif site olmasını istemektedir. Bu sebeple projenin en önemli kısmının ilanların güvenilirliğinin oluşturduğunu ve buna yoğunlaşılması gerektiği belirlenmiştir.

2.3. Proje Maliyet Analizi

Proje başında öngörülen maliyet bilgisi hesaplanmıştır. Mevcutta var olan donanım ve iş gücü maliyetleri dahil edilmemiştir. Proje gerçekleşme sürecinde gerekli durumlar oluşursa ihtiyaç duyulan eğitimlerin alınabilmesi için bütçe eklenmiştir. Projenin gerekliliklerinden olan yazılım, hosting ve domain maliyetleri için ihtiyaç duyulan bütçeler ayrılmıştır. Yukarıdaki bilgiler maliyet analizi tablosuna işlenmiş olup, toplam proje maliyeti de belirlenmiştir.

Bütçe kategorisi	Tahmini maliyet	Açıklama
İş gücü	0₺	1 proje yöneticisi/yazılım geliştiricisi, 1 yazılım geliştiricisi/bilişim ve teknoloji yöneticisi.
Donanım	0₺	Elde bulunan donanımlar ile geliştirme yapılması planlandığından bir maliyet bulunmamaktadır.
Yazılım	300₺	Olası yazılım lisans ücretleri için belirlenen bütçe miktarı.
Domain	120₺	Domain hizmetinin alınması için öngörülen bütçe miktarı.
Hosting	400₺	Hosting hizmeti için belirlenen bütçe miktarı.
Eğitim	100₺	Proje geliştirme sürecinde ihtiyaç durumunda eğitim alınabilmesi için ayrılan bütçe.
Toplam	920₺	

Tablo 2.2 Proje Maliyet Tablosu

2.4. Risk Listesi ve Olasılık Etki Matrisi

Günümüzde giderek yaygınlaşan yazılım projelerinde oldukça fazla ve önemli hatalar meydana gelmektedir. Bu hatalar geliştirilen yazılımı olumsuz etkilemekle birlikte, projenin hedeflenen zamanda bitirilememesine, daha fazla iş gücü ve maddi ihtiyaçlar doğurmaktadır. Dolayısıyla bu durumu en aza indirmek için projedeki olası riskler belirlenmiş ve gerekli planlanmalar yapılmıştır.

2.4.1.Risk Listesi

Risk No	Risk Açıklaması
1	Projenin öngörülen süreçten daha uzun sürmesi.
2	Muadil platformların fazla sayıda olması.
3	UI/UX tasarımlarının kullanıcılara hitap etmemesi veya anlaşılabilirliğinin düşük olması.
4	Uygulamada optimizasyon sorunlarının olması, öngörülen talepten daha fazla talep olması durumunda gerekli altyapının sağlanamaması.
5	Piyasa analizinin yeterli seviyede yapılamaması.
6	İşveren ve çalışan kullanıcılar arasındaki dengenin sağlanamaması.
7	Proje için öngörülen bütçe planlamasının beklenen düzeyi aşması ve yeterli bütçe sağlanamaması.
8	Önceden belirlenen teknolojilerin proje gerçekleştirme sürecinde yetersiz kalması.
9	Uygulamada güvenlik açıklarının ortaya çıkması.

Tablo 2.3.1 Risk Listesi

2.4.2. Olasılık Etki Matrisi

		OLASILIK ↓		
		DÜŞÜK	ORTA	YÜKSEK
DÜŞÜK	7	1	5	
ORTA	9	3	2	
YÜKSEK	8	6	4	
	ETKİ →			

Tablo 2.3.2 Olasılık Etki Matrisi

2.5. PROJE BAŞARI KRİTERLERİ

Tablo 2.4 Proje Başarı Kriterleri

İp N o	İş Paketi Adı	Başarı Ölçütü(%)	Proje Başarısındaki Önemi(%)
1	Proje Analizinin Yapılması	80%	15%
1.1	Proje isterlerinin belirlenmesi		3%
1.2	Proje ihtiyaçlarının belirlenmesi		2%
1.3	Proje paydaş analizi		1%
1.4	Proje süre ve maliyet analizi		3%
1.5	Proje risklerinin belirlenmesi		2%
1.6	Proje teknolojilerinin belirlenmesi		3%
1.7	Proje başarımlarının belirlenmesi		1%
2	Proje Tasarımının Yapılması	100 %	20%
2.1	Veritabanı tasarımının yapılması		7%
2.2	UI/UX tasarımlarının yapılması		8%

Tablo 2.4 Proje Başarı Kriterleri (devamı)

2.3	Web servis tasarımının yapılması		5%
3	Projenin Teknik Olarak Gerçeklenmesi	100%	45%
3.1	Veritabanı tasarımının gerçekleşmesi		7%
3.2	Kullanıcıların giriş ve kayıt sayfalarının gerçekleşmesi		8%
3.3	Admin panelinin gerçekleşmesi		9%
3.4	Ana sayfanın gerçekleşmesi		10%
3.5	Kullanıcı profil sayfalarının gerçekleşmesi		6%
3.6	Gerekli diğer sayfaların gerçekleşmesi		5%
4	Projenin Testi ve Hata Düzeltmeleri	75%	7%
4.1	Proje testlerinin yapılması		4%
4.2	Hataların giderilmesi		3%
5	Projenin Canlı Ortamda Yayınlanması	100%	15%

BÖLÜM 3. İŞ AKIŞI

3.1. Roller

Kullanıcılar 5 tipe ayrılacaktır. Çalışan rolündeki kullanıcılar ve iş arayışında olan job seeker rolündeki kullanıcılar, ilanlar üzerinde içerikleri denetleyen, düzenleme ve onay yetkisine sahip editörler, ilanlar üzerinde silme, düzenleme ve onaylama yetkisine sahip moderatörler. Moderatörler aynı şekilde şirketler üzerinde bilgilerin değişmesi durumunda bilgilerin siteye yansımaları için onay verme yetkisine sahiptir. Adminler moderatör ve editörlerin yetkilerinin yanı sıra moderatörü, editörü şirketi ve kullanıcıyı banlama yetkisine sahiptir.

3.2. Şirketler

Şirketler 2 tipe ayrılacaktır; onaylı ve onaysız şirketler. Onaylı şirketler gerçekten şirket olduklarını onaylayan belgelerle birlikte moderatörden onay aldıktan sonra bunu belirten imge profil ve ilanlarında gözükecektir. Şikâyet et butonu sayesinde kullanıcılar şüpheli gördükleri şirketleri moderatöre bildirebilecektir.

Onaylı şirketlerin şüphe takibi 50 şikâyet, onaysız şirketlerinki 10 şikâyet sonrası başlatılacaktır. Bunu moderatör sistem panel ekranından takip edebilecektir.

3.3. İş Arayışında Olan Kullanıcılar

Öncelikle giriş ekranında cv'lerini oluşturduktan sonra sisteme giriş yapıp kendilerine önerilen iş ilanlarına başvuru yapabilecekler. İsterlerse şirket araması yapıp iş başvurusunda bulunabilirler. Geri dönüş geldiği takdirde bilgilendirme ekranından bunu görebilecekler. İletişim bildirimlerinin yanı sıra profil bölümünde bulunan iletişim bilgileri yardımıyla gerçekleştirilecektir.

3.4. Arama ve Filtreleme

İs arayışında olan kullanıcılar ilanları içeriği, belirlenen ücret, gerekli is tecrübesi, bölgesi, uzaktan veya ofis olmasına göre filtreleyebilecektir. İsim ve içeriğe göre arama yapabilecektir. Performans iyileştirilmesi açısından algolia kullanılacaktır.

3.5. Eşleştirme ve Öneri

CV doldurulurken kullanıcıdan alınan bilgiler doğrultusunda (is tecrübesi, alandaki hakimiyeti, daha önce hangi şirketlerde tecrübesinin olduğu gibi vs...) açılan ilanlarla karşılaştıracak ve önerilen ilanları kullanıcının önüne getirecektir. Örneğin: C#, tecrübe 3 yıl, daha önce rest ve soap api tecrübesi var, integration ve unit test tecrübesine sahip bir kullanıcı bu veya buna yakın kriterler istenen isleri önerilenler kısmında görebilecektir.

3.6. İlan

İlanlar, şirketler tarafından oluşturulduktan sonra onaya gönderilecektir. Moderatör veya editör onayından sonra ilan aktif hale gelecektir. İlan sahibi isterse, yalnızca önerilen adaylar kutucuğunu işaretleyerek bu ilana sadece belirli kullanıcıların başvurmasını sağlayabilecektir.

Kullanıcı ilanı sonrası için kaydedebilir, şirketi gösterme kutucuğunu işaretleyerek o şirketi bir daha görmeyebilir.

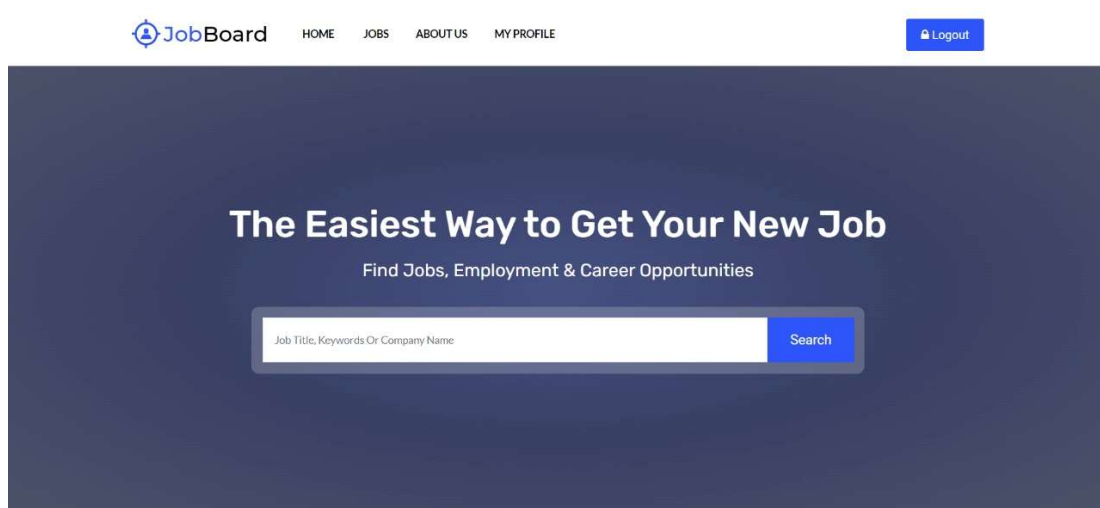
BÖLÜM 4. PROJE TASARIM DÖKÜMANTASYONU

Ekran görüntüleri son haliyle biçimlendirilmemiş olup Figma uygulamasında hazırlanmıştır. Hazırlanan bu prototiplere bağlı kalınarak proje uygulamasının gerçekleştirilmesi hedeflenmektedir.

Gerekli durumlarda ilgili tasarımlarda gidişata göre gerekli görülen parçalar eklenebilir veya istenmeyen tasarımlar çıkarılarak göze hitap eden kullanıcı dostu bir uygulama yapılması beklenmektedir.

4.1. Uygulama Sayfa Tasarımları

4.1.1. Ana Sayfa Ekranı



Şekil 4.1 Ana Sayfa Ekranı

Yukarıda gösterilen ekran tasarımı kullanıcı ilk karşılayan ekran olarak eklemiştir. Kullanıcı burada kendine uygun olan işi bulmak için ilgili arama çubuğunu kullanabilir, dilerse aşağıda bulunan kategoriler sekmesinden herhangi bir alanda keşif yapabilir.

4.1.2 Kayıt Olma Ekranı

Kullanıcı eğer uygulamaya kayıt olmak isterse aşağıdaki ekranla karşılaşacaktır. Bu ekranda kullanıcı mail adresi ile sisteme kayıt olabilir. Ayrıca mail adresiyle kayıt olmak istemezse ‘Linkedin’, ‘Google’, ‘Facebook’ gibi platform üyelikleriyle sisteme kayıt olabilmektedir.

JobBoard HOME JOBS ABOUT US MY PROFILE POST A JOB Sign In Sign Up

Register
Home > Register

PERSONAL INFORMATION
If you have an account with us, please log in.

Email *
Your Email Address

First Name *
First Name

Last Name *
Last Name

Password *
Type Password

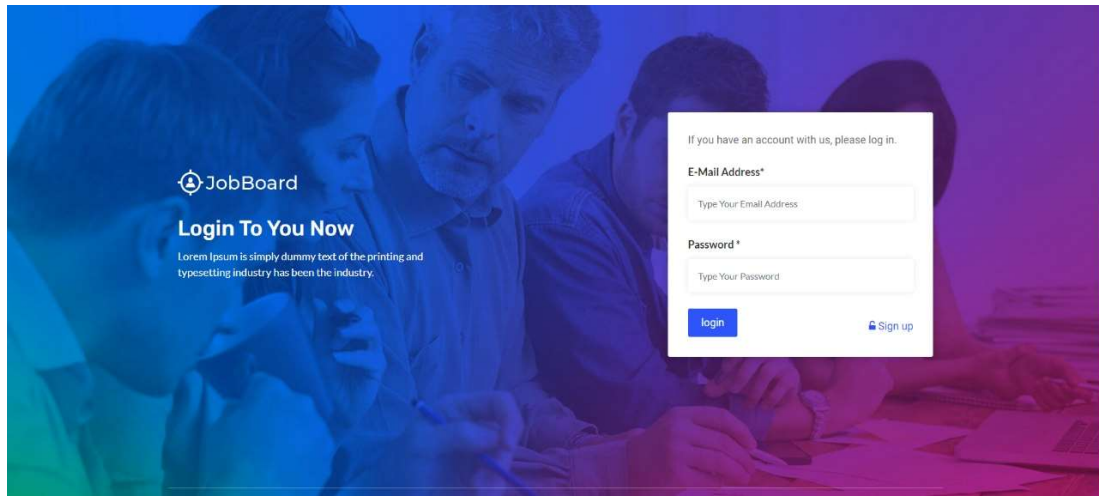
Phone *
Your Phone Number

User Or Company *
Employee
Company
Freelancer

Şekil 4.2 Kayıt Ekranı

4.1.3. Giriş Yapma Ekranı

Sisteme üye olan kullanıcılar uygulamaya giriş yapmak istediklerinde aşağıdaki ekranla karşılaşacaklardır. Bu ekranda kayıt oldukları mail adresi ve şifreleriyle giriş yapabilirler. Ayrıca diğer servislerle üye olan kullanıcılar servis butonlarını kullanarak servis üyelikleriyle uygulamaya giriş yapabilmektedir.



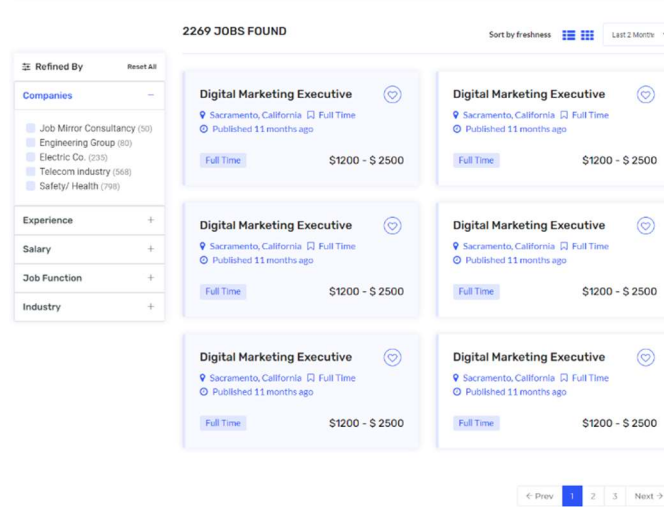
Şekil 4.3 Giriş Ekranı

4.1.4. İlan Listeleme Ekranı

Kullanıcılar uygulamaya girip arama çubuğunu veya kategorileri kullanarak arama yaptıklarında aşağıdaki ekranla karşılaşacaklardır. Bu ekranda sol tarafta kullanıcının daha detaylı arama yapabilmesini sağlamak amacıyla bir filtreleme kısmı bulunmaktadır.

Ekranın kalan kısmında ise kullanıcının aramasına uygun olan ilanların listelendiği bölüm bulunmaktadır. Buradaki her ilanda işin başlığı, konumu, ücreti, favorilere ekleme butonu gibi temel özellikler yer almaktadır.

Kullanıcı buradaki favori butonunu kullanarak ilgili ilana daha kolay ulaşmak için profillerinden ulaşabileceklerdir.



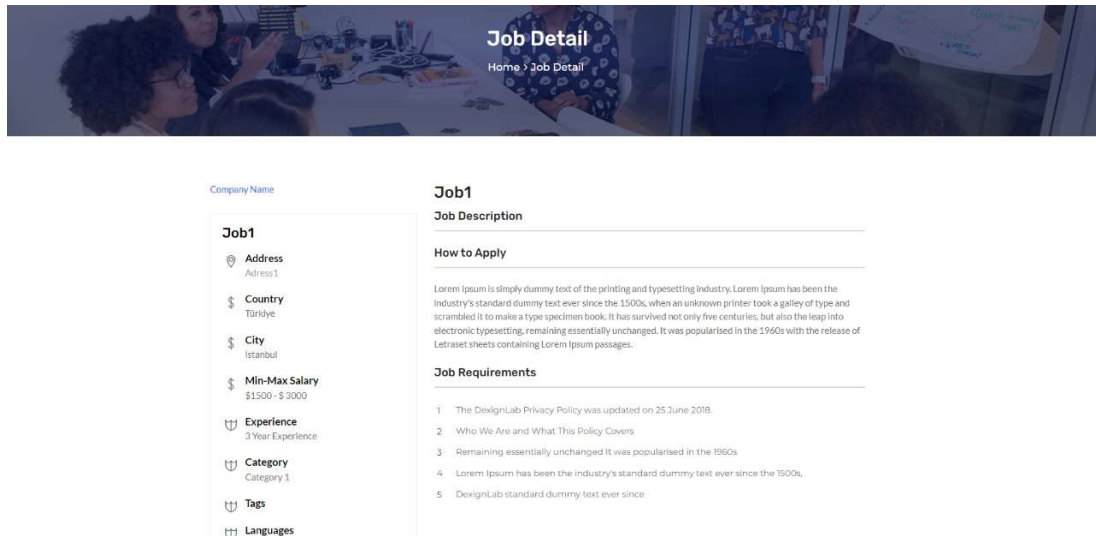
Şekil 4.4 İlan Listeleme Ekranı

4.1.5. İlan Detay Ekranı

Kullanıcılar aramalarında buldukları ilanlara tıkladıklarında aşağıdaki detay sayfasına yönlendirileceklerdir. Bu ekran ilgili ilanın detaylarını içeren, kullanıcı bilgilendiren ekran olması hedeflenmektedir.

Ekranın sol kısmında adres, ücret, tecrübe alanları yer alırken sağ tarafta ilanın başlığı, ilan sahibi firmanın çalışan adaylardan beklediği özellikler, işin detayları, işverenin kriterleri gibi kısımlar bulunmaktadır.

Burada hem işverenin, aday çalışana istediklerini doğru aktarması hem de iş arayışında olan kullanıcının başvurmayı düşündüğü işin detaylarını öğrenip kendine en uygun kararı vermesi amaçlanmaktadır.



Şekil 4.5 İlan Detay Ekranı

4.1.6. İlan Oluşturma Ekranı

İş veren kullanıcı tipindeki kullanıcılar sitede ilan açmak istediklerinde şekilde gösterilen ekranla karşılaşacaklardır.

The screenshot shows the 'POST A JOB' form on the JobBoard website. The form is divided into two main sections: a left sidebar and a main content area. The sidebar contains a user profile icon, a 'Post A Job' button, and links for 'Company Profile', 'Manage Jobs', 'Company Jobs', and 'Company See Comments'. The main content area is titled 'POST A JOB' and includes a 'Back' button. It contains several input fields: 'Job Title' (placeholder: Enter Job Title), 'Address' (placeholder: address), 'Add Tag' (a list of tags: Tag1, Tag2, Tag3, Tag4), 'Add Language' (a list of languages: Turkish, English, Russian, AddedName), 'Job Type' (placeholder: Type1), 'Experience' (placeholder: 1 Years), 'Categories' (placeholder: Category1), and 'Working Type' (placeholder: Full Time). There are also star ratings and a '4' rating visible in the sidebar.

Şekil 4.6 İlan Oluşturma Ekranı

The continuation of the 'POST A JOB' form shows the following fields: 'Minimum Salary (\$):' with a placeholder 'e.g. 10000', 'Maximum Salary (\$):' with a placeholder 'e.g. 20000', 'Country' with a placeholder 'e.g. 20000', 'City' with a placeholder 'London', 'Description' (a large text area), 'Upload File' (a button with an upload icon), and a final 'Upload' button.

Şekil 4.7 İlan Oluşturma Ekranı Devamı

4.1.7. Profil Ekranı

Sistemde bulunan kullanıcıların profil ekranları şekilde gösterildiği gibi olacaktır.

JobBoard HOME JOBS ABOUT US COMPANY PROFILE POST A JOBS Logout

COMPANY PROFILE Back

Company Name: company

Your Email: company@gmail.com

Website: random.com

Founded Date: 2022

City: İstanbul

Country: Türkiye

Phone: 213123213

About:

Şekil 4.8 Profil Ekranı

Website

random.com

Founded Date

2022

City

İstanbul

Country

Türkiye

Phone

213123213

About:

12312321324234

Update Setting

Şekil 4.9 Profil Ekranı Devamı

4.1.8. Şifre Değiştirme Ekranı

Siteye kayıtlı olan kullanıcılar diledikleri herhangi bir zamanda kayıtlı hesaplarının şifrelerini değiştirebilmektedir. İlgili değiştirme ekranı şekilde gösterilmiştir.

CHANGE PASSWORD [Back](#)

Old Password

New Password

Confirm New Password

[Update Password](#)

Şekil 4.10 Şifre Değiştirme Ekranı

4.1.9. Admin Giriş Ekranı

Sistemde admin yetkisi bulunmayan kullanıcılar admin olarak giriş yaptıklarında hata almaktadır. Bu giriş ekranında yalnızca admin yetkisine sahip kullanıcılar giriş yapabilmektedir.

ADMIN DASHBOARD

User Disabled

E-Mail Address*

asdfasfd@gmail

You are not authorized to login

Password *

[login](#) [Forgot Password](#)

Şekil 4.11 Admin Giriş Ekranı

4.1.10 Ek Ekran Görüntüleri

JobBoard LANGUAGES TAGS JOB TYPES COMPLAINTS COMMENTS USERS Logout

CREATE

pgsqlAdmin_PostgreSQL

Name

On Local DELETE

Remote DELETE

Şekil 12 Admin İş Tipi

JobBoard LANGUAGES TAGS JOB TYPES COMPLAINTS COMMENTS USERS Logout

CREATE

Name

Tag1 DELETE

Tag2 DELETE

Tag3 DELETE

Şekil 4.13 Admin Etiketi

JobBoard LANGUAGES TAGS JOB TYPES COMPLAINTS COMMENTS USERS Logout

comment	point	From user id	accepted	rejected
Example Comment	5	2		ACCEPT REJECT
Example Comment	5	2		ACCEPT REJECT
Example Comment	5	2		ACCEPT REJECT
Example Comment	5	2		ACCEPT REJECT
Example Comment	3	2		ACCEPT REJECT
Example Comment	3	2		ACCEPT REJECT
Example Comment	3	2		ACCEPT REJECT
Example Comment	5	2	accepted	REJECT
Example Comment	5	2		ACCEPT REJECT
Example Comment Updated	5	2		ACCEPT REJECT
Example Comment	5	2		ACCEPT REJECT
Example Comment	5	2		ACCEPT REJECT
Example Comment	5	2		ACCEPT REJECT
Example Comment	5	2		ACCEPT REJECT

Şekil 4.14 Admin Yorumları

JobBoard					Logout	
comment	point	From user id	accepted	rejected		
Example Comment	5	2			ACCEPT	REJECT
Example Comment	5	2			ACCEPT	REJECT
Example Comment	5	2			ACCEPT	REJECT
Example Comment	5	2			ACCEPT	REJECT
Example Comment	3	2			ACCEPT	REJECT
Example Comment	3	2			ACCEPT	REJECT
Example Comment	3	2			ACCEPT	REJECT
Example Comment	5	2	accepted		REJECT	
Example Comment	5	2			ACCEPT	REJECT
Example Comment Updated	5	2			ACCEPT	REJECT
Example Comment	5	2			ACCEPT	REJECT
Example Comment	5	2			ACCEPT	REJECT
Example Comment	5	2			ACCEPT	REJECT
Example Comment	5	2			ACCEPT	REJECT

Şekil 1.15 Moderatör Ekranı

Resume Headline
Key Skills
Employment
Education
IT Skills
Projects
Profile Summary
Accomplishments
Desired Career Profile
Personal Details
Attach Resume

Resume Headline

Job board currently living in USA

Key Skills

JavaScript CSS HTML Bootstrap Web Designing Photoshop

Employment

Junior Software DeveloperEdit

W3schools

Oct 2015 to Present (3 years 4 months)

Available to join in 1 Month

Junior Software Developer

Education

Mention your employment details including your current and previous company work experience

London - 12th

2017

London - 10th

2015

Add Doctorate/PhD

Add Masters/Post-Graduation

Şekil 4.16 Kullanıcı Özgeçmişi

JobBoard
HOME JOBS ABOUT US COMPANY PROFILE POST A JOB
Logout

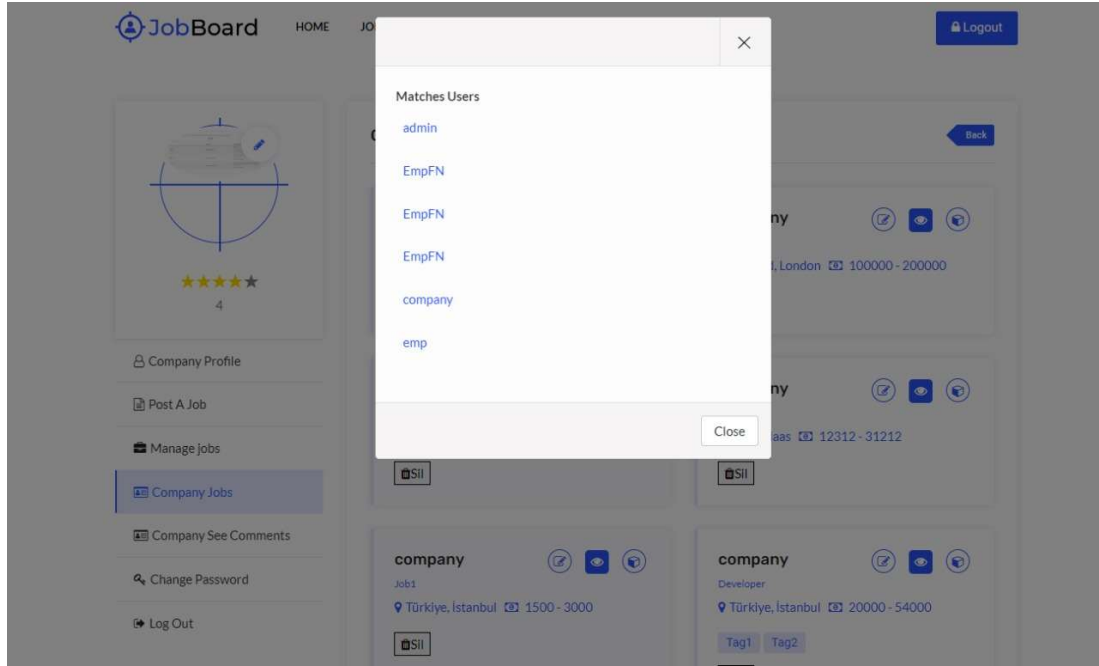
Browse Job Filter Grid
Home > Browse Job Filter Grid

Job Title, Keywords, or Phrase
Find Job

1 JOBS FOUND

Jobupd
Turkey, Istanbul Full-Time
Full-Time \$3000 - \$5000

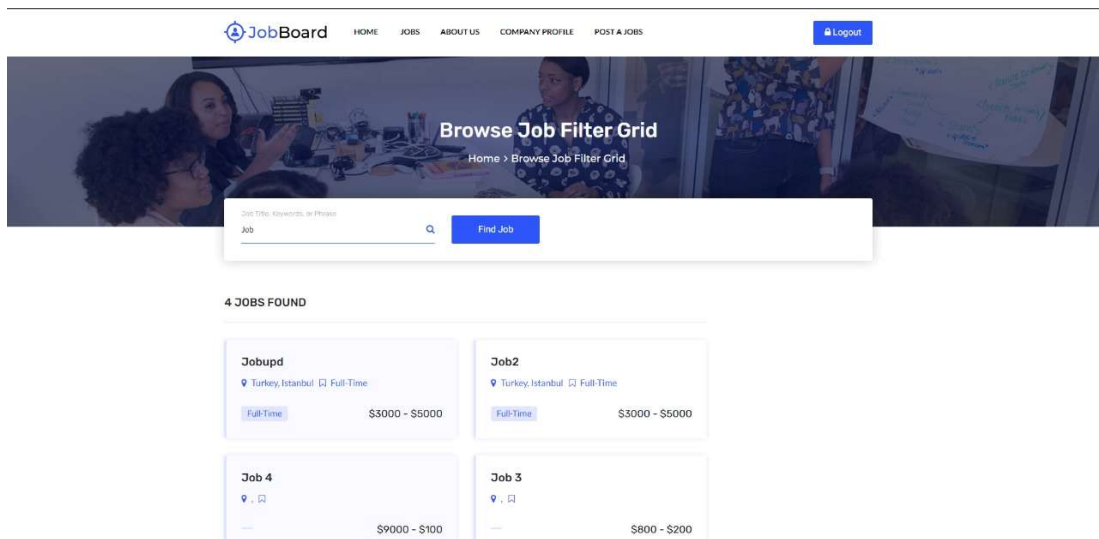
Şekil 4.17 Etiket Filtresi




Şekil 4.18 Naive Bayes Kullanımı



Şekil 4.19 Naive Bayes Kullanımı Devamı



Şekil 4.20 İş Arama



David Martin
Web developer

- Profile
- My Resume
- Saved Jobs
- Applied Jobs**
- Change Password
- Log Out

PHP Web Developer

@company-name Sacramento, California 25,000

PHP Angular Bootstrap Wordpress

Posted: 2 day ago [Apply Job](#)

Software Developer

@company-name Sacramento, California 25,000

PHP Angular Bootstrap Wordpress

Posted: 2 day ago [Apply Job](#)

Branch Credit Manager

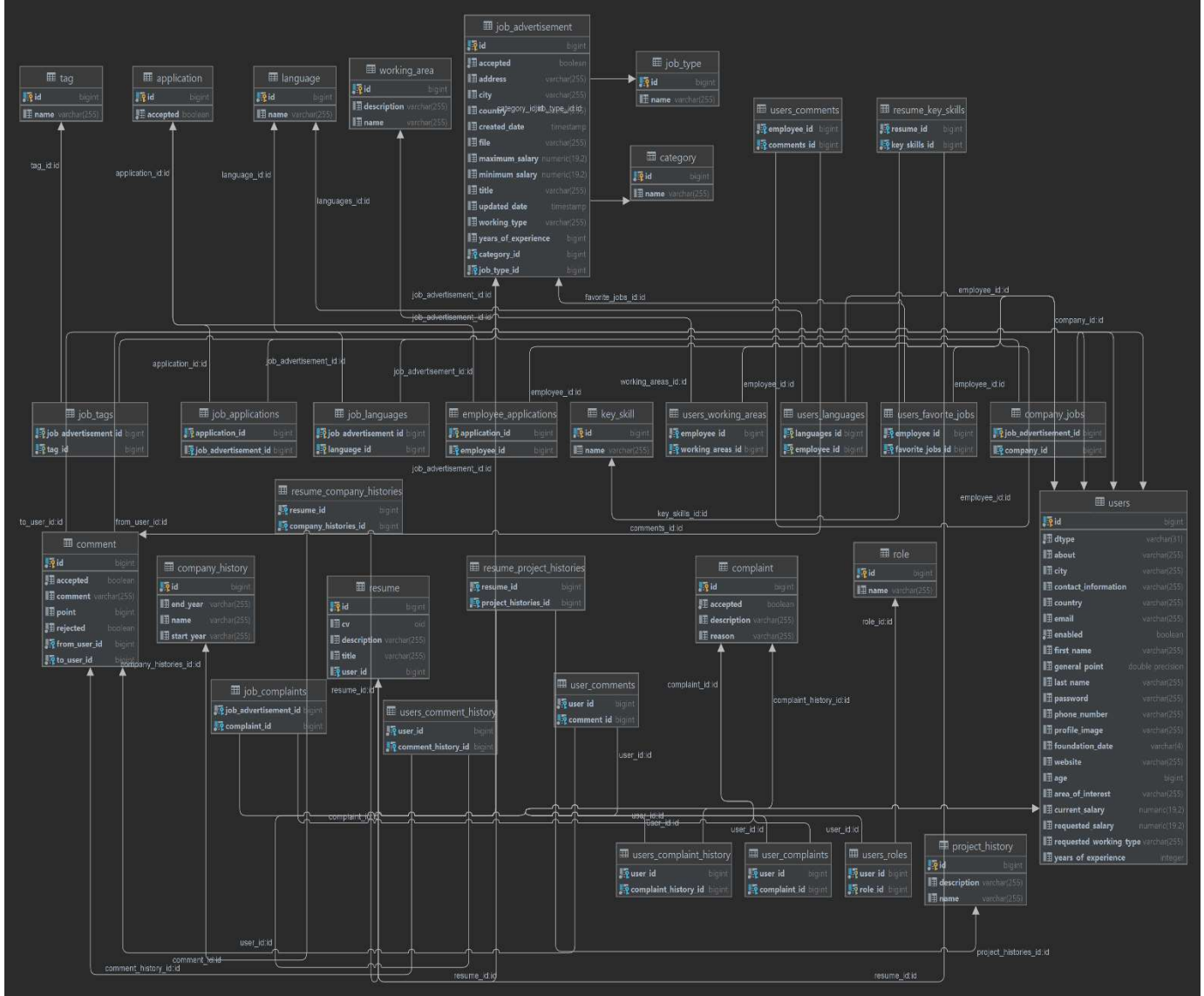
@company-name Sacramento, California 25,000

PHP Angular Bootstrap Wordpress

Posted: 2 day ago [Apply Job](#)

Şekil 4.21 Kabul Edilen İşler

4.2. Veritabanı İlişkisel Gösterim Diyagramı



Şekil 4.22 Veritabanı İlişkisel Gösterim Diyagramı

4.3. Akış Şemaları

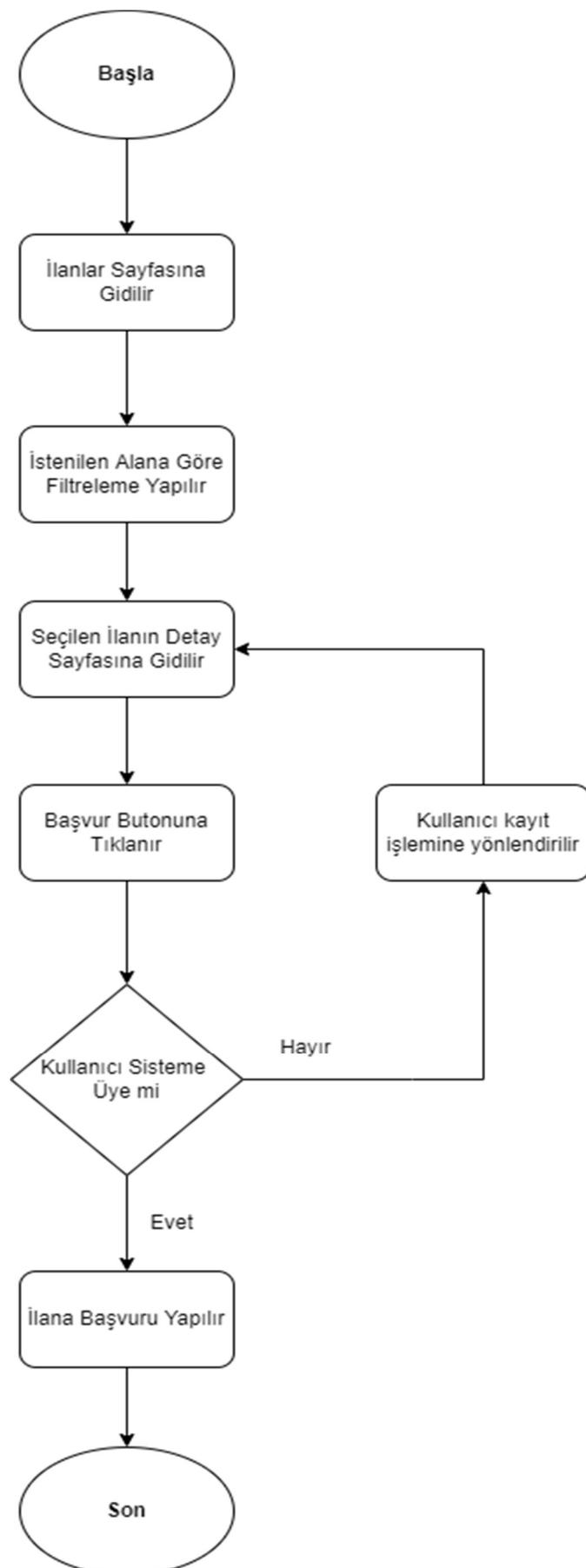
Akış şeması; bilgisayar yazılımlarının işlem basamaklarının geometrik şekillerle ifade edildiği şemalardır. Akış şemaları bir işin tamamlanması sırasında uygulanan adımların daha iyi anlaşılabilir olmasını sağlayan görsellerdir. Programın akış yönü ve işlemler arası bağlantıyı çubuklar ve oklar gösterir. Bazı işlemler için genel şekiller veya o işleme ait özel şekiller kullanılabilir.

Akış şemaları iletişimi kolaylaştırır. Etkili analizi sağlamaya fırsat tanır. Akış diyagramı yardımı ile, sorun daha etkin bir şekilde analiz edilebilir. Program ile ilgili bir plan elde edilmiş olur. Verimli kodlama sağlar. Sistem analizi ve program geliştirme aşaması sırasında bir rehber olarak hareket. Hata ayıklama sürecine yardımcı olur.

4.3.1. İlan Başvurma İşlemi

Hali hazırda sistemde firmalar tarafından bulunan ilanlara kullanıcılar tarafından başvurulması işlemidir. Kullanıcılar kayıt olmadan sistemde gezinme hakkına sahiptir ancak herhangi bir ilana başvurabilmesi için kullanıcıdan sisteme kayıt olması beklenmektedir.

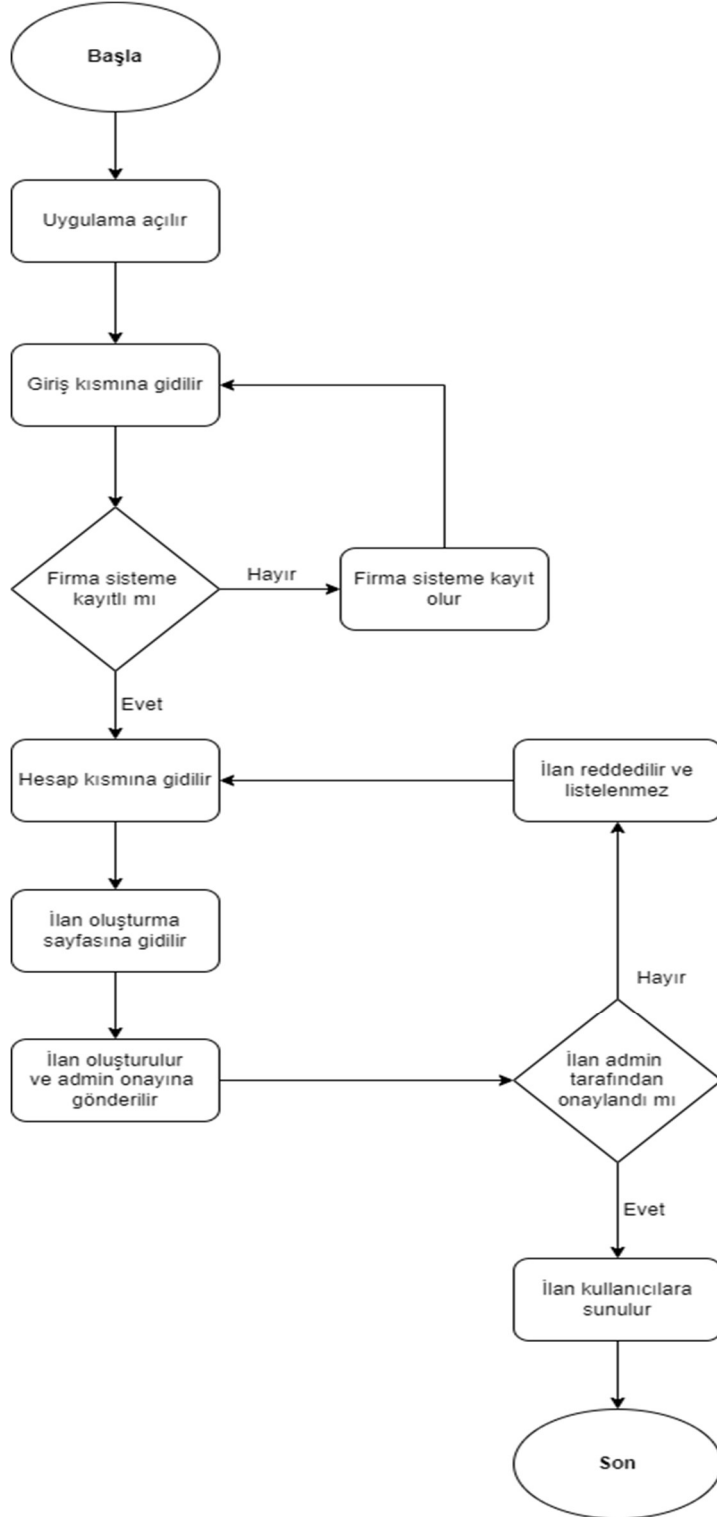
Aşağıda gösterilen akış şemasında bir kullanıcının ilana başvurma eylemini gösteren, adımları açıklayan bir tasarım yapılmıştır.



Şekil 4.23 İlana Başvurma Akış Diyagramı

4.3.2. Firmaların İlan Oluşturma İşlemi

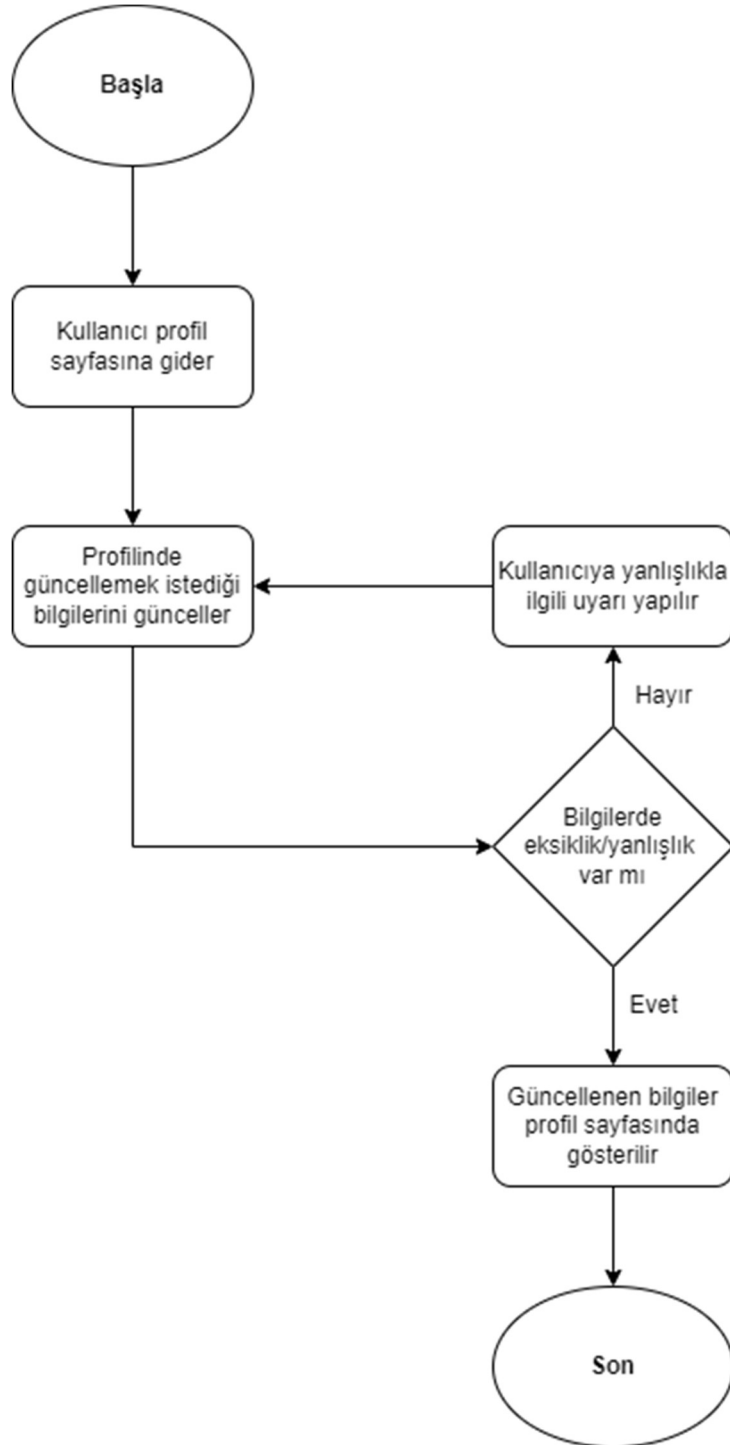
Firmaların sistemde bulunabilmesi için öncelikle sisteme kayıt olması beklenmektedir. Ardından dilediği şekilde kendileri için ilan açabilme hakkına sahiptirler. Aşağıda bir firmanın sistemde ilan açma adımları bir akış şemasıyla gösterilmiştir.



Şekil 4.24 İlan Oluşturma Akış Diyagramı

4.3.3. Kullanıcıların Bilgilerini Güncelleme İşlemi

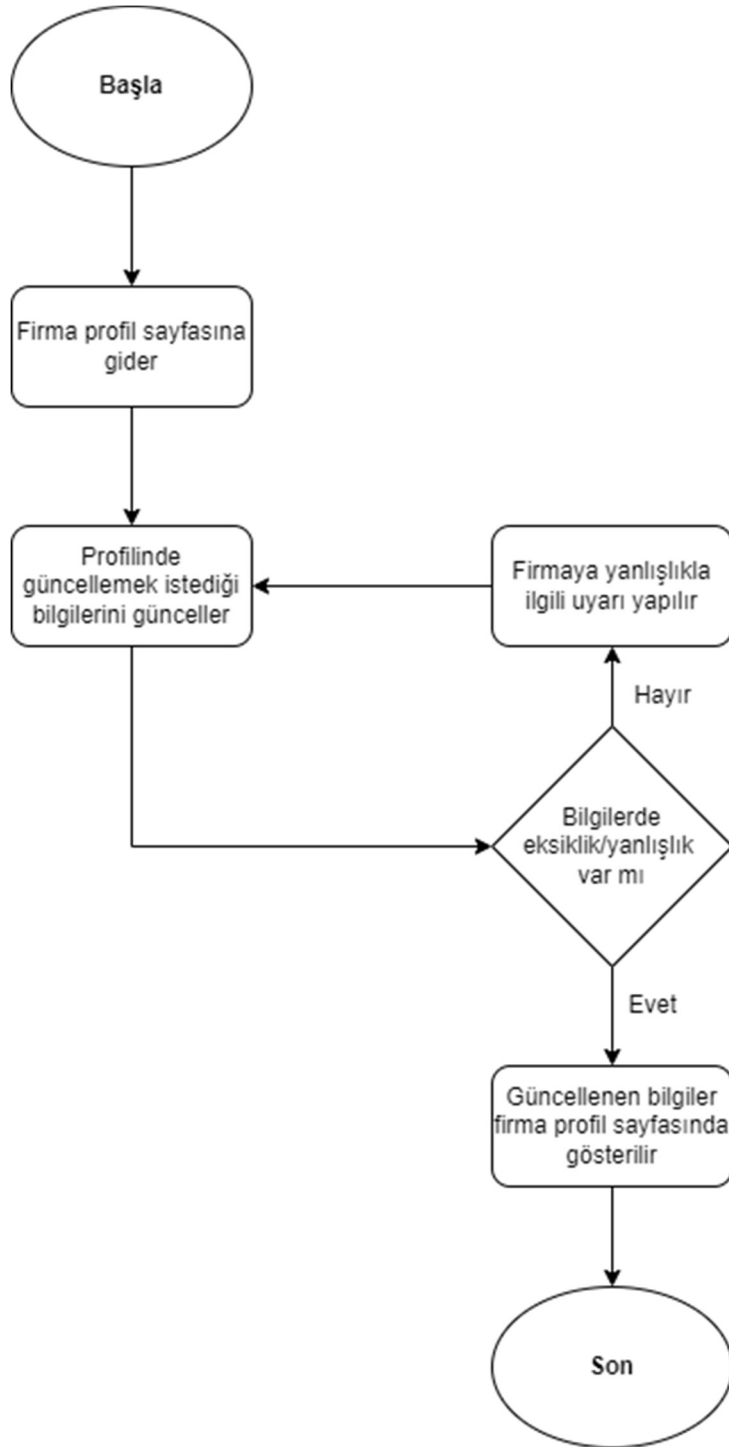
Kullanıcılar diledikleri zaman profillerindeki bilgileri düzenleyebilir, değiştirebilirler. Bu güncelleme işlemi akış şemasıyla gösterilmiştir.



Şekil 4.25 Kullanıcı Bilgilerini Güncelleme Diyagramı

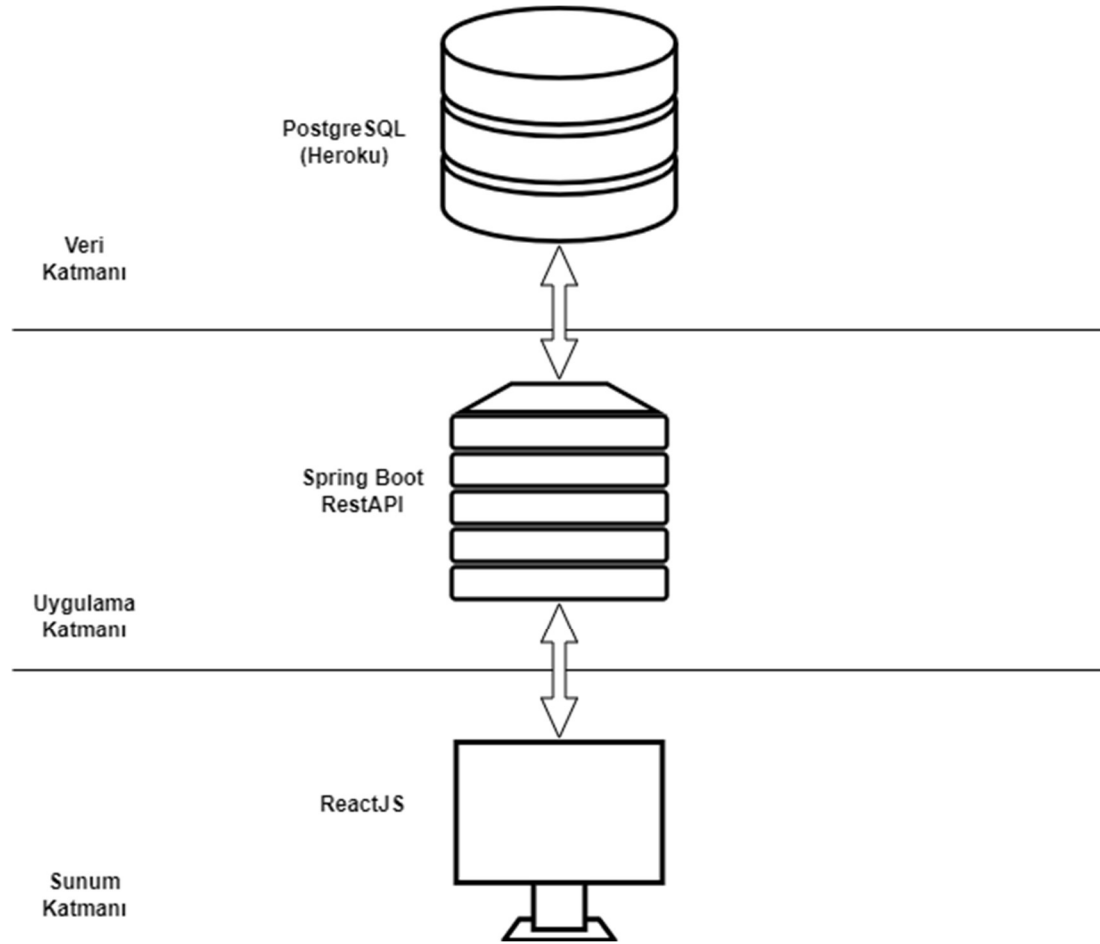
4.3.4. Firmaların Bilgilerini Güncelleme İşlemi

Sistemde kayıtlı bulunan firmalar profillerinde gösterilen bilgileri değiştirme hakkına sahiptir. İstedikleri zaman bu işlemi gerçekleştirebilirler. Söz konusu işlem akış şemasıyla gösterilmiştir.



Şekil 4.26 Firma Bilgilerini Güncelleme Diyagramı

4.4. Sistem Diyagramı



Şekil 4.27 Sistem Diyagramı

Sistem veri katmanı, sunum katmanı ve bu iki katmanı birbirine bağlayan uygulama katmanından oluşmaktadır. Sunum katmanı veri alışverişi için RestAPI ile iletişim kuruyor. Bu servislerle sunum katmanı için gerekli veriler işlenmektedir.

4.5. Kaynak Kodları

```
@Override
public void registerUser(@NotNull SignUpRequest request) {
    if (existByEmail(request.getEmail())) {
        throw new UserAlreadyExistsException("There is an account with that email address: " + request.getEmail());
    }
    isValidRegisterRole(request.getRole());
    String requestRole = StringUtils.isEmpty(request.getRole()) ? RoleEnum.EMPLOYEE.code() : request.getRole();
    User user;
    if (RoleEnum.EMPLOYEE.code().equals(requestRole)) {
        user = new Employee();
    } else if (RoleEnum.COMPANY.code().equals(requestRole)) {
        user = new Company();
    } else {
        user = new User();
    }
    user.setEmail(request.getEmail());
    user.setEnabled(true);
    user.setFirstName(request.getFirstName());
    user.setPhoneNumber(request.getPhoneNumber());
    user.setPassword(passwordEncoder.encode(request.getPassword()));
    user.setEnabled(false);
    Role role = roleRepository.findByName(requestRole).orElseThrow(() -> new RoleNotFoundException("This role no found. -> " + RoleEnum.EMPL
    Set<Role> roles = new HashSet<>();
    roles.add(role);
    user.setRoles(roles);
    userRepository.save(user);
}
```

Şekil 4.28 Kayıt Olma Metodu

Sistemde “user” ve “company” şeklinde iki kullanıcı bulunacak. Kullanıcılar kayıt olduklarında onay işlemine düşeceklerdir. Moderatör veya admin yetkisine sahip kullanıcı işlemlerini onaylarsa sisteme giriş yapabileceklerdir. Yukarıda servis katmanında kullanıcının kayıt olmasını sağlayan metot verilmiştir.

```
new *
@Override
public void verifyUser(Long userId) {
    User user = userRepository.findById(userId).orElseThrow(() -> {
        throw new UserNotFoundException("User not found");
    });
    user.setEnabled(true);
    save(user);
}
```

Şekil 4.29 Servis Katmanı Metodu

Şekildeki görselde kullanıcının doğrulanmasını sağlayan servis katmanı metodu yer almaktadır.

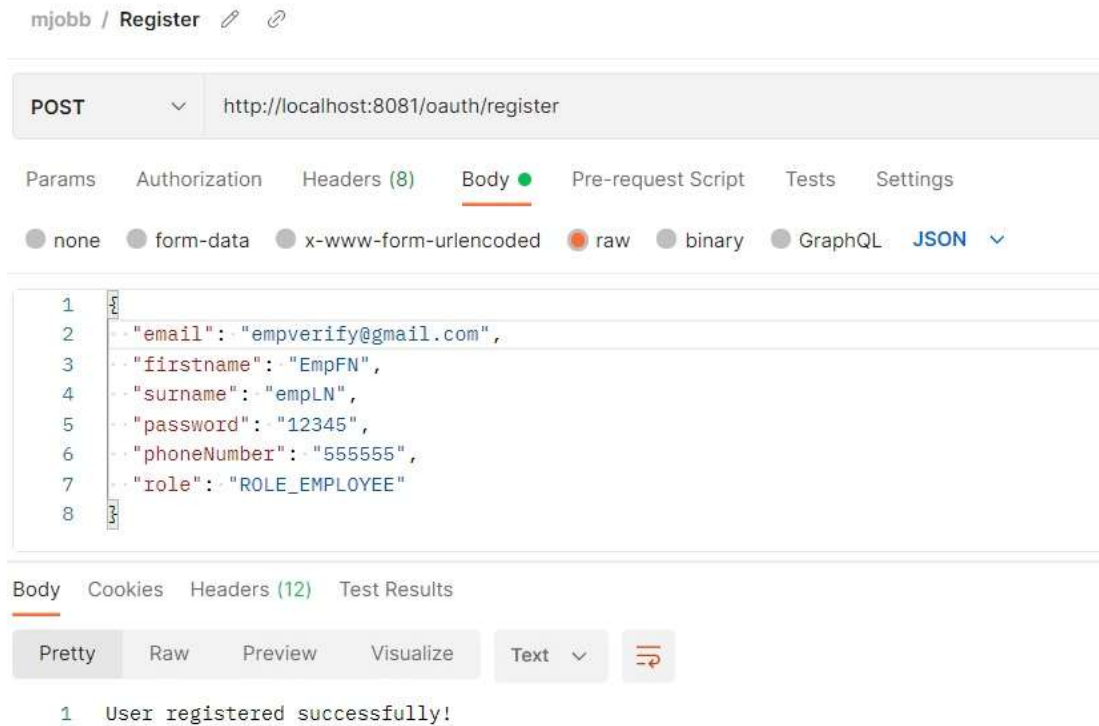
```

new *
@PutMapping("user/{userId}/verify")
@Secured({"ROLE_ADMIN", "ROLE_MODERATOR"})
public ResponseEntity<User> verifyUser(@PathVariable Long userId) {
    userService.verifyUser(userId);
    return ResponseEntity.ok(userService.getUserById(userId));
}

```

Şekil 4.30 Kontrol Katman Metodu

Api endpoint ile role bazlı güvenliği sağlayan controller katmanı metodu şekilde gösterilmiştir.



Şekil 4.31 Kayıt İşlemi Gösterimi

Kayıt işleminin arka planda gösterimi şekilde verilmiştir.

dtype	id	about	city	contact_information	country	email	enablec
character varying (31)	[PK] bigint	character varying (255)	character varying (255)	character varying (255)	character varying (255)	character varying (255)	boolean
1 Employee	1	[null]	İstanbul	55555555	Turkey	admin@gmail.com	true
2 Employee	2	Employee about section222	Ankara	33333333	Turkey	empfromlocal@gmail.com	true
3 Company	3	[null]	[null]	[null]	[null]	company@gmail.com	true
4 Employee	4	[null]	[null]	[null]	[null]	empfromlocal2@gmail.com	true
5 Employee	5	[null]	[null]	[null]	[null]	empfromlocal6@gmail.com	true
6 Employee	6	[null]	[null]	[null]	[null]	empfromlocal7@gmail.com	true
7 Employee	7	[null]	[null]	[null]	[null]	empfromlocal321@gmail.com	true
8 Employee	8	[null]	[null]	[null]	[null]	empverify@gmail.com	false

Şekil 4.32 Veritabanına Kayıt Eklenmesi

Kayıt işlemi gerçekleştirilen kullanıcın veri tabanında eklenmesi işlemi gösterilmiştir.

PUT http://localhost:8081/api/v1.0/admin/user/8/verify

Params Authorization Headers (8) Body Pre-request Script Tests Settings

Type Bearer Token

The authorization header will be automatically generated when you send the request. Learn more about [authorization](#)

Token ef3c88e8-a239-4a4e-9df6-979bda4f8615

Body Cookies Headers (15) Test Results Status: 401 Unauthorized Time: 25 ms

Pretty Raw Preview Visualize JSON

```

1 {
2   "error": "invalid_token",
3   "error_description": "Invalid access token: ef3c88e8-a239-4a4e-9df6-979bda4f8615"
4 }
```

Şekil 4.33 Oturum Süresi Dolumunda Oluşan Cevap

Eğer ki admin veya moderatörün oturum süresi dolduğunda doğrulama işlemi yapmaya kalkarsa dönmesi beklenen cevap şekilde gösterilmiştir.

PUT http://localhost:8081/api/v1.0/admin/user/8/verify Send

Params Authorization Headers (8) Body Pre-request Script Tests Settings Cookies

Type Bearer Token

The authorization header will be automatically generated when you send the request. Learn more about [authorization](#)

Token 8ec38897-a13b-4455-88f6-ab9d0963dd69

Body Cookies Headers (17) Test Results Status: 403 Forbidden Time: 39 ms Size: 629 B Save Response

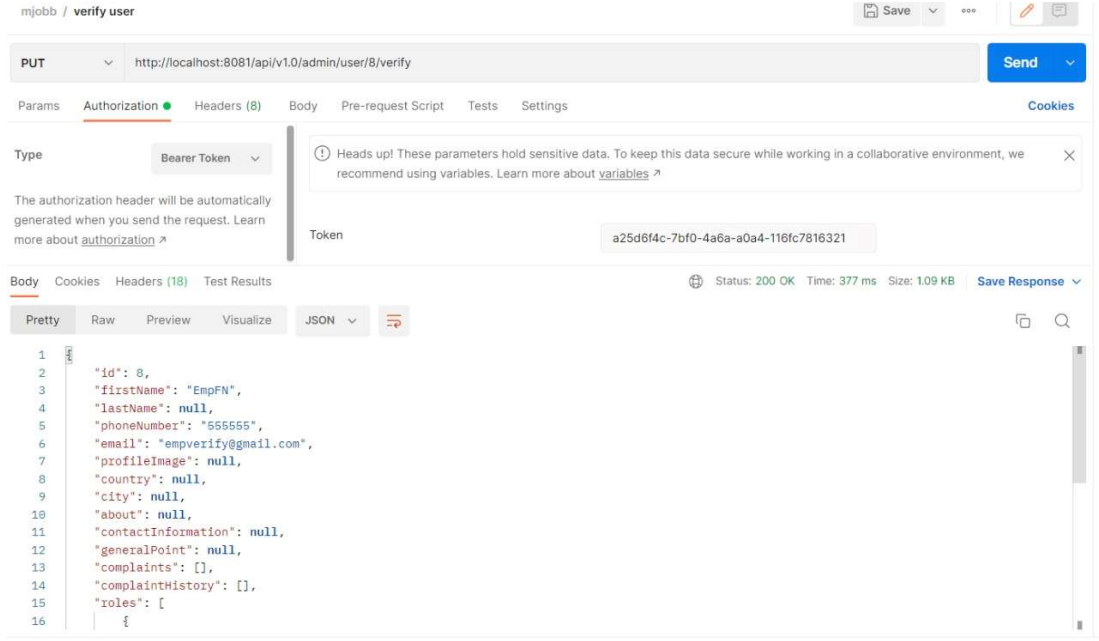
Pretty Raw Preview Visualize JSON

```

1 {
2   "error": "access_denied",
3   "error_description": "Access is denied"
4 }
```

Şekil 4.34 Yetkisiz İşlem Cevabı

Eğer ki admin veya moderatör yetkisi bulunmayan bir kullanıcı doğrulama işlemi yapmaya çalışır ise dönmesi beklenen cevap şekilde gösterilmiştir.



Şekil 4.35 Başarılı Doğrulama İşlemi

Doğrulama işlemi başarıyla gerçekleştiğinde yukarıdaki şekilde belirtilen işlemin gerçekleşmesi beklenmektedir.

8	Employee	8	[null]	[null]	[null]	[null]	empverify@gmail.com	true
---	----------	---	--------	--------	--------	--------	---------------------	------

Şekil 4.36 Başarılı Kayıta Güncelleme İşlemi

Kayıt işlemi tamamen başarılı bir şekilde sonuçlanırsa veri tabanı şekilde gösterildiği gibi güncellenmektedir.

BÖLÜM 5. KULLANILAN TEKNOLOJİLER

5.1. Java

Java, 1995 yılında Sun Microsystems tarafından yayınlanmış bir hesaplama platformu ve programlama dilidir. Basit başlangıçlar ile ortaya çıkmıştır ve birçok hizmet ve uygulamanın oluşturulduğu güvenilir platformu sağladığı için bugünün dijital dünyasında en büyük paya sahiptir. Gelecekte kullanmak üzere tasarlanmış yeni, inovatif ürünler ve dijital hizmetler de Java'yı temel almaya devam ediyor.

Bu dil C ve C++'dan birçok sözdizim türetmesine rağmen bu türevler daha basit nesne modeli ve daha az düşük seviye olanaklar içerir. Java uygulamaları bilgisayar mimarisine bağlı olmadan herhangi bir Java Sanal Makinesi üzerinde çalışabilen tipik bytecode'dur.

Java'nın sık kullanılan sloganlarından biri olan, çevirisi "bir defa yaz, her yerde çalıştır" olan "write once, run anywhere - WORA", Java'nın; derlenmiş Java kodunun, Java'yı destekleyen bütün platformlarda tekrar derlenmeye ihtiyacı olmadan çalışabileceğini ima eder. 2016 yılında bildirilen 9 milyon geliştiricisi ile, özellikle istemci sunucu web uygulamaları için olmak üzere, kullanımda olan en popüler programlama dillerinden birisidir.[1]

5.2. React

React.js, özellikle tek sayfalı uygulamalar için kullanıcı arayüzleri oluşturmak için kullanılan açık kaynaklı bir JavaScript kütüphanesidir. Web ve mobil uygulamalar için görünüm katmanını işlemek için kullanılır. React ayrıca yeniden kullanılabilir UI bileşenleri oluşturmamıza da olanak tanır.[2]

React, tek yönlü veri bağlama kullanır ve Flux adlı bir uygulama mimarisi, bileşenlerin tek bir kontrol noktası olan gönderici aracılığıyla veri akışını kontrol eder. Büyük ReactJS uygulamalarının bağımsız bileşenlerinde hata ayıklamak daha kolaydır.

5.3. Spring Boot

Spring framework ile geliştirilen, güçlü varsayılan-otomatik ayarları sayesinde kolay ve hızlı Spring tabanlı uygulama yapımı için geliştirilmiş bir projedir.

Springboot Java programlama dilinde geliştirilen ve Java dilinde güçlü MicroService yazılmasına olanak tanıyan bir frameworktür. Springboot sayesinde backend uygulama geliştirirken veritabanı işlemlerinizi Java ile Language Integrated olarak yazabilir ve Springboot tarafından bu işlemler gerçekleştirebilirsiniz. Aynı zamanda client tarafındaki uygulamalar içinde servis uçları sağlayarak veri girdi-çıktısı oluşturulabilir.[3]

5.4. JWT

JSON Web Token (JWT), tarafların birbirleri arasındaki veri alışverişini ve bunun doğrulamasını sağlayan JSON tabanlı RFC 7519'de tanımlanmış açık bir standarttır. Örneğin bir sunucu, kullanıcının yönetici ayrıcalıklarına sahip olduğunu belirten bir anahtar (token) oluşturabilir ve bunu kullanıcıya gönderebilir. Kullanıcı daha sonra bu anahtar ile kendisine tanımlanmış olan yönetici yetkisini bir istemcide kullanabilir ve bütün taraflar tarafından yetkisi doğrulanabilir.[4]

JSON Web Token'ları genellikle üç bölümden oluşur; header (başlık), payload (yük) ve signature (imza). Header, oluşturulacak imzada hangi algoritmanın kullanılacağını belirtir.

5.5. JavaScript

JavaScript (genellikle JS olarak kısaltılır), HTML ve CSS ile birlikte World Wide Web'in temel teknolojilerinden biri olan programlama dilidir . Web sitelerinin %97'sinden fazlası, web sayfası hareketleri için istemci tarafında JavaScript kullanırlar ve kullanılan kodlar genellikle üçüncü taraf kitaplıkları içerir. Tüm büyük web tarayıcılarında, kaynak kodunu kullanıcıların cihazlarında yürütebilmek için özel bir JavaScript motoru bulunur.

JavaScript, ECMAScript standardına uyan, genellikle eş zamanlı olarak derlenmiş, üst düzey bir dildir. Dinamik yazma, prototip tabanlı obje yönelimi ve birinci sınıf işlemlere sahiptir. Olay odaklı, işlevsel ve zorunlu programlama stillerini destekleyen çoklu paradigmadır . Metin, tarihler, düzenli ifadeler, standart veri yapıları ve Belge Obje Modeli (DOM) ile çalışmak için uygulama programlama arayüzlerine (API'ler) sahiptir.

ECMAScript standardı, ağ oluşturma, depolama veya grafik olanakları gibi herhangi bir giriş/çıkış (I/O) içermez. Pratikte, web tarayıcısı veya diğer çalıştırma ortamları, I/O için JavaScript API'leri sağlarlar.

JavaScript motorları başlangıçta yalnızca web tarayıcılarında kullanılıyordu, ancak günümüzde bazı sunucuların ve çeşitli uygulamaların da temel bileşenleridir. Bu kullanım için en popüler çalışma zamanı sistemi Node.js'dir .

Java ve JavaScript'in ad, sözdizimi ve ilgili standart kitaplıklar açısından benzerlikleri olsa da, iki dil birbirinden farklıdır ve tasarım açısından da büyük farklılıklar gösterir.[5]

5.6. Axios

Axios XMLHttpRequest hizmeti temel alan hafif bir HTTP istemcisidir . Getirme API'sına benzer ve HTTP isteklerini gerçekleştirmek için kullanılır.

Axios, hem tarayıcıda hem de Node.js ortamında çalışan söz tabanlı bir HTTP istemcisidir. XMLHttpRequests ve node http arabirimi ile ilgili tek bir API sağlar. Bunun yanı sıra, ES6 sözdizimi için bir çoklu dolgu kullanarak istekleri gönderir.[6]

5.7. Jenkins

Jenkins, Sürekli Entegrasyon (CI / Continuous Integration) ve Sürekli Dağıtım (CD / Continuous Delivery) amacıyla Java ile yazılmış açık kaynaklı bir otomasyon aracıdır. Jenkins, yazılım projelerinizi sürekli olarak oluşturmak ve test etmek için kullanılır, bu da geliştiricilerin değişiklikleri projeye entegre etmesini ve kullanıcıların yeni bir sürüm elde etmesini kolaylaştırır. Ayrıca, çok sayıda test ve dağıtım teknolojisiyle bütünleşerek yazılımınızı sürekli olarak teslim etmenize olanak tanır.[7]

Jenkins ile kuruluşlar, otomasyon yoluyla yazılım geliştirme sürecini hızlandırabilir. Jenkins, derleme, belgeleme, test etme, paketleme, aşama, devreye alma, statik analiz ve çok daha fazlası dahil olmak üzere her türlü geliştirme yaşam döngüsü sürecini entegre eder.

5.8. Bootstrap

Bootstrap, HTML, CSS ve JavaScript ile yazılmış, kullanışlı, yeniden kullanılabilir kod bitlerinin dev bir koleksiyonudur. Aynı zamanda geliştiricilerin ve tasarımcıların hızlı bir şekilde tamamen duyarlı web siteleri oluşturmasını sağlayan bir ön uç geliştirme framework'udur.[8]

5.9. CSS

CSS, web sayfalarını sunulabilir hale getirme sürecini basitleştirmeyi amaçlayan basit bir tasarım dilidir.[9]

5.10. GIT

Git, yazılım geliřtirmede kullanılan bir sürüm kontrol sistemidir.[10]

5.11. PostgreSql

PostgreSQL, hem ilişkisel (SQL) hem de ilişkisel olmayan (JSON) sorgulamayı destekleyen açık kaynaklı, kurumsal sınıf ve gelişmiş bir nesne-ilişkisel veri tabanı sistemidir. SQL dilini, karmaşık veri iş yüklerini yüksek verimlilikle işlemesini sağlayan ek özelliklerle kullanır ve genişletir.[11]

5.12. Postman

Postman, API'leri oluşturmaya, test etmeye ve değiřtirmeye yardımcı olan bir API (Uygulama Programlama Arayüzü) geliştirme aracıdır.

HTTP isteklerini GET, POST, PUT, PATCH gibi komutla ile kullanma, oluşturulan ortamları daha sonra kullanmak için kaydetme, API'leri çeşitli yazılım dilleri ile kullanmak için koda dönüřtirme yeteneklerine sahiptir. API ise yazılım uygulamalarının API çağrıları aracılığıyla birbirleriyle iletişim kurmasını sağlayan Uygulama Programlama Arayüzü anlamına gelir.[12]

Projeye ait Postman Collection bağlantısı aşağıdadır.

<https://www.getpostman.com/collections/2bb4c988481a57b389e5>

BÖLÜM 6. KULLANILAN ALGORİTMALAR

6.1. Apriori Algoritması

6.1.1. Apriori Algoritması Nedir

Apriori Algoritmasının ismi, bilgileri bir önceki adımdan aldığı için “prior” anlamında Apriori’dir . Bu algoritma temelinde iteratif (tekrarlayan) bir niteliğe sahiptir ve hareket bilgileri içeren veri tabanlarında sık geçen öge kümelerinin keşfedilmesinde kullanılır.

Apriori algoritmasına göre, eğer k-öge kümesi (k adet elemana sahip öge kümesi) minimum destek değerini sağlıyorsa, bu kümenin alt kümeleri de minimum destek ölçütünü sağlar.

Birliktelik kuralı analizi, tüm sık geçen öğelerin bulunması ve sık geçen bu öğelerden güçlü birliktelik kurallarının üretilmesi olmak üzere iki aşamalıdır.

Birliktelik kuralının ilk aşaması için kullanılan Apriori Algoritması, sık geçen öğeler madenciliğinde kullanılan en popüler ve klasik algoritmadır. Bu algoritmada özellikler ve veri, Boolean ilişki kuralları ile değerlendirilir. Apriori algoritması, Birliktelik kuralları çıkarmak için en çok kullanılan algoritmadır.

6.1.2. Algoritmanın Çalışması

Apriori algoritması minimum Destek (Support) ve minimum Güven (Confidence) parametrelerini alarak çalışmaktadır.[13]

Destek ve Güven parametreleri ne kadar büyükse birliktelik kurallarının da o kadar güçlü olduğunu ifade eder. Parametre başlangıç değerlerinin çok büyük olmasının bazı kuralların gözden kaçmasına, çok küçük olmasının ise bulunmak istenen örüntüden uzaklaşılmasına sebebiyet vereceğini unutmamak gerekir. Apriori algoritmasının

varsayılan Destek değerinin %10 ve Güven değerinin ise %80 olduğunu ve eğer parametre değeri verilememiş ise bu değerler ile kurallar oluşacaktır.[14]

Destek (Support) Değeri: Bir ilişkinin tüm alışverişler içinde hangi oranda tekrarlandığını belirtir.

$$\text{Destek (X} \rightarrow \text{Y)} = \text{Frekans(X, Y)} / N$$

N: Toplam alışverişlerin sayısı

Güven (Confidence) Değeri: X ürününü alan müşterilerin Y ürününü alma olasılığını ortaya belirtir

$$\text{Güven (X} \rightarrow \text{Y)} = \text{Frekans(X, Y)} / \text{Frekans(X)}$$

- Birliktelik Kuralları Analizi yapılabilmesi için minimum Destek ve minimum Güven parametre değerleri belirlenir. Uygulamadan elde edilen sonuçlar bu parametre değerlerine eşit ya da büyük olacaktır. Yani algoritmaya başlamadan kendiniz için destek ve güven değerleri belirlemeniz gerekiyor. Belirttiğimiz gibi değerler ne kadar yüksek olursa kurallar da bir o kadar güçlü olacaktır.
- Veri seti taranarak her bir ürün için tekil olarak frekans sayıları yani Destek değerleri belirlenir ve Minimum Destek parametre değerinden küçük değerlere sahip olan ürünler çözümlemeden çıkarılır.
- Ardından veri setindeki elemanlar ikişerli olarak gruplanarak frekans değerleri elde edilir. Minimum Destek değerinden küçük olan ikili gruplar çözümlemeden çıkarılır.
- Yeni tablo oluşturulduktan sonra 2. Adım içerisinde yapılanlar yeni tabloda tekrar edilir. Yalnız bu sefer frekans değeri, her iki ürünün aynı anda var olma durumunu belirtmektedir.

Aynı şekilde üçlü, dörtlü vb. gruplandırmalar yaparak gruplara ait frekans bilgileri ile Minimum Destek değeri karşılaştırılır. Uygun olmayanlar çözümlemeden çıkarılır. Bu işlem Support değerinin altında ürün grubu kalmayana kadar devam eder.

Yukarıdaki işlemler bittikten sonra elde kalan Support değerini karşılayan eleman grupları için yukarıda belirtilen formüle göre Confidence(güven) değeri hesaplanır.

Güven değeri başında belirlenen değerleri karşılayan eleman grupları kurallar haline gelir. Böylece çıkan sonuca göre hangi elemanların birlikte kullanıldığı gözlemlenmiş olur.[15]

6.1.3. Algoritmanın Projede Kullanımı

Yukarıda da açıklandığı üzere Apriori Algoritması veri kümeleri veya veriler arasındaki ilişkiyi ortaya çıkarmaya yaramaktadır. Burada amaç kullanıcıların kendilerine en uygun öneriyi bulmasıdır.

Kullanıcı kendisine uygun öneriler sunulduğunda hem gereksinimlerini daha hızlı ve kolay karşılayabilmektedir hem de kullanıcının sistemde kalmasını sağlamaktadır. Bu sayede kullanıcıyla birlikte aynı zamanda sistem de kazançlı durumda olmaktadır.

Projede bu algoritmanın kullanılmasının ana sebebi kullanıcılar ilanları incelerken sayfaların aşağısında onlara başka ilanların önerilmesidir. Bu önerilen ilanların rastgele şekilde sunulması yerine arayışta olan kişiye en uygun ve doğru ilanların sunulması amaçlanmıştır. Kişiye en uygun ilanlar, kişinin daha öncesinde yaptığı arama ve gezinme esnasında yaptığı işlemlerle sağlanmaktadır.

Bu sayede projeyi kullanan kullanıcıların uygulamadan memnun ve ihtiyaçlarını karşılayarak ayrılabilmeleri aynı zamanda sistemin de kullanılabilirliğinin artırılması ve kullanıcıların uygulamada kalması hedeflenmektedir.

6.2. Naive Bayes Algoritması

6.2.1. Naive Bayes Algoritması Nedir

Naive Bayes sınıflandırıcı, örüntü tanıma problemine ilk bakışta oldukça kısıtlayıcı görülen bir önerme ile kullanılabilen olasılıksal bir yaklaşımdır. Bu önerme, örüntü tanımada kullanılacak her bir tanımlayıcı öznelik ya da parametrenin istatistik açıdan bağımsız olması gerekliliğidir. Her ne kadar bu önerme Naive Bayes sınıflandırıcının kullanım alanını kısıtlasa da istatistik bağımsızlık koşulu esnetilerek kullanıldığında da daha karmaşık yapay sinir ağları gibi metotlarla karşılaştırabilir sonuçlar vermektedir. Bir Naive Bayes sınıflandırıcı, her özneliğin birbirinden koşulsal bağımsız olduğu ve öğrenilmek istenen kavramın tüm bu özneliklere koşulsal bağlı olduğu bir Bayes ağı olarak da düşünülebilir.

Naive Bayes sınıflandırıcısı adını İngiliz matematikçi Thomas Bayes'ten almaktadır.[16]

6.2.2. Algoritmanın Çalışması

Naive Bayes sınıflandırmasında sisteme belirli bir oranda öğretilmiş veri sunulur (Örn: 100 adet). Öğretim için sunulan verilerin mutlaka bir sınıfı/kategorisi bulunmalıdır. Öğretilmiş veriler üzerinde yapılan olasılık işlemleri ile, sisteme sunulan yeni test verileri, daha önce elde edilmiş olasılık değerlerine göre işletilir ve verilen test verisinin hangi kategoride olduğu tespit edilmeye çalışılır. Elbette öğretilmiş veri sayısı ne kadar çok ise, test verisinin gerçek kategorisini tespit etmek o kadar kesin olabilmektedir.

Naive Bayes sınıflandırma yönteminin birçok kullanım alanı bulunabilir fakat, burada neyin sınıflandırıldığından çok nasıl sınıflandırıldığı önemli. Yani öğretilecek veriler binary veya text veriler olabilir, burada veri tipinden ve ne olduğundan ziyade, bu veriler arasında nasıl bir oransal ilişki kurduğumuz önem kazanıyor.[17]

6.2.3. Algoritmanın Projede Kullanımı

Yukarıda da açıklandığı üzere Naive Bayes Algoritması bir çeşit sınıflandırma algoritmasıdır. Amaç elde bulunan verilerin uygunluk ölçütü en yüksek olacak şekilde sınıflandırılması üzerine kuruludur.

Algoritmanın projede kullanılma amacı Apriori Algoritması'nın tersine iş verenler için kolaylık sağlamaktır. Bunu yapmamızın temel sebebi projede aynı zamanda işverenlerin çalışan arayışında bulunabilme imkanının olmasıdır. Algoritma sayesinde bu sefer iş verenler aradıkları çalışanları uygun bir sınıflandırma içerisinde ulaşabileceklerdir. İşverenler ilanlarına en uygun görülen çalışanları bu algoritmanın çalışması sayesinde listeleme imkanına sahip olacaklardır.

Bu sayede işveren tipindeki kullanıcıların da işlemlerinden olumlu şekilde ayrılmaları, ayrıca bu işe alım sürecinin onlar için de hızlandırılacağını düşünmekteyiz. Dolayısıyla işveren kullanıcıların sistemi kullanmaları için bir sebep olacak ve kullandıkları için memnun bir şekilde ayrılacaklardır.

KAYNAKLAR

- [1] WİKİPEDIA, “Java”, [https://tr.wikipedia.org/wiki/Java_\(programlama_dili\)](https://tr.wikipedia.org/wiki/Java_(programlama_dili)) (17 Mart 2020)
- [2] Alastyr, “React Nedir? Nasıl Çalışır”, <https://www.alastyr.com/blog/react-nedir-react-nasil-calisir> (3 Şubat 2021)
- [3] Microsoft, “Spring Boot Nedir?”, <https://azure.microsoft.com/tr-tr/resources/cloud-computing-dictionary/what-is-java-spring-boot> (17 Mayıs 2021)
- [4] WİKİPEDIA, “JSON Web Token”, https://tr.wikipedia.org/wiki/JSON_Web_Token, (23 Mayıs 2022)
- [5] Aws.Amazon, “What is JavaScript”, <https://aws.amazon.com/tr/what-is/javascript/>, (6 Şubat 2022)
- [6] KARABAY A, “Axios Nedir ?”, <https://www.karabayyazilim.com/blog/javascript/axios-nedir-2020-05-21-115817>, (21 Mayıs 2020)
- [7] Kerteriz, “Jenkins nedir ve ci/cd süreç örneği”, <https://kerteriz.net/jenkins-nedir-kurulumu-ve-ci-cd-surec-ornegi>, (19 Eylül 2021)
- [8] İnnova, “Bootstrap Nedir, Ne İşe Yarar?” <https://www.innova.com.tr/tr/blog/dijital-donusum-blog/bootstrap-nedir-ne-ise-yarar> (25 Ocak 2022)
- [9] Wikipedia, “CSS”, <https://tr.wikipedia.org/wiki/CSS> (4 Haziran 2022)
- [10] Coderspace, “Git Nedir, Nasıl Kullanılır?”, <https://coderspace.io/blog/git-nedir-nasil-kullanilir> (9 Haziran 2021)

- [11] WİKİPEDIA, “PostgreSql”, <https://tr.wikipedia.org/wiki/PostgreSQL> (20 Haziran 2022)
- [12] İlaydadastan, “Postman’a Giriş”, <https://ilaydadastan.com/postman-giri%C5%9F-postman-nedir-ve-ne-i%C3%A7in-kullan%C4%B1%C4%B1r-76e42d7f80a2>, (20 Haziran 2021)
- [13] Veribilimiokulu, “Birliktelik Kuralları Analizi”, <https://www.veribilimiokulu.com/associationrulesanalysis>, (4 Eylül 2018)
- [14] Industnovation, “Apriori Algoritması Nedir?”, <https://industnovation.com/apriori-algoritmasi-nedir>, (20 Ocak 2021)
- [15] Budak V. Ö. , Kartal E. , Gülseçen S. Site-içi Aramalar ve Apriori Algoritması Kullanılarak Web Sitesi Ziyaretçilerinin İhtiyaç Tespitine Yönelik Bir Örnek Olay İncelemesi. Bilişim Teknolojileri Dergisi. 2018; 11(2): 211-222.
- [16] WİKİPEDIA, “Naive Bayes Sınıflandırıcı”, https://tr.wikipedia.org/wiki/Naive_Bayes_s%C4%B1n%C4%B1fland%C4%B1r%C4%B1c%C4%B1s%C4%B1, (19 Nisan 2021)
- [17] Miracozturk, “Naive Bayes Algoritması”, <https://miracozturk.com/python-ile-siniflandirma-analizleri-naive-bayes-algoritmasi/>, (30 Ocak 2022)

ÖZGEÇMİŞ

Maftun Hashimli, 29.11.2001 tarihinde Azerbaycan'ın Cebrayıl şehrinde doğdu. Çocukluğunu Bakü'de geçirdi. Eğitim hayatına 20 numaralı okulda başladı. İlköğrenimini burada tamamladı. Eğitimine 5. sınıftan itibaren Bakü Atatürk Lisesi'nde devam etti. Ortaokulu ve liseyi burada bitirdikten sonra 2018 yılında Sakarya Üniversitesi Bilgisayar Mühendisliği Bölümü'ne kabul edilmiştir. Üniversite eğitimine Sakarya Üniversitesi'nde devam etmekte olup 4. sınıfa devam etmektedir. Ocak 2020'den Şubat 2020'ye kadar Telecom Invest Company'de. Software Development stajı yaptı. Ekim 2020'den Temmuz 2021'e kadar BITE Consulting Services Ltd. Şirketinin Bakü ofisinde yazılım geliştirici olarak çalıştıktan sonra Epigra Bilgi Tek. ve İlt. Hız. Ltd. Şti. İstanbul ofisinde Backend Developer olarak çalışmaktadır.

Berkay Çete, 18.02.2000 tarihinde Afyonkarahisar ilinde doğdu. Çocukluğunu Trabzon'da geçirdi. İlköğrenimine Trabzon Cumhuriyet İlkokulu'nda tamamladı. Ortaöğrenimini Ankara Cumhuriyet Ortaokulu'nda tamamladı. Lise eğitimine Afyonkarahisar Ali Çağlar Anadolu Lisesi'nde başladı ve bitirdi. Lise mezuniyetinin ardından Sakarya Üniversitesi Bilgisayar ve Bilişim Bilimleri Fakültesi Bilgisayar Mühendisliği Bölümü'ne yerleşti. Üniversite öğrenimini halen Sakarya Üniversitesi'nde Bilgisayar Mühendisliği 4. Sınıf öğrencisi olarak sürdürmektedir.

BSM 498 BİTİRME ÇALIŞMASI

DEĞERLENDİRME VE SÖZLÜ SINAV TUTANAĞI

KONU :

ÖĞRENCİLER (Öğrenci No/AD/SOYAD):

Değerlendirme Konusu	İstenenler	Not Aralığı	Not
Yazılı Çalışma			
Çalışma klavuza uygun olarak hazırlanmış mı?	x	0-5	
Teknik Yönden			
Problem tanımı yapılmış mı?	x	0-5	
Geliştirilecek yazılımın/donanımın mimarisini içeren blok şeması (yazılımlar için veri akış şeması (dfd) da olabilir) çizilerek açıklanmış mı?			
Blok şemadaki birimler arasındaki bilgi akışına ait model/gösterim var mı?			
Yazılımın gereksinim listesi oluşturulmuş mu?			
Kullanılan/kullanılması düşünülen araçlar/teknolojiler anlatılmış mı?			
Donanımların programlanması/konfigürasyonu için yazılım gereksinimleri belirtilmiş mi?			
UML ile modelleme yapılmış mı?			
Veritabanları kullanılmış ise kavramsal model çıkarılmış mı? (Varlık ilişki modeli, noSQL kavramsal modelleri v.b.)			
Projeye yönelik iş-zaman çizelgesi çıkarılarak maliyet analizi yapılmış mı?			
Donanım bileşenlerinin maliyet analizi (prototip-adetli seri üretim vb.) çıkarılmış mı?			
Donanım için gerekli enerji analizi (minimum-uyku-aktif-maksimum) yapılmış mı?			
Grup çalışmalarında grup üyelerinin görev tanımları verilmiş mi (iş-zaman çizelgesinde belirtilebilir)?			
Sürüm denetim sistemi (Version Control System; Git, Subversion v.s.) kullanılmış mı?			
Sistemin genel testi için uygulanan metotlar ve iyileştirme süreçlerinin dökümü verilmiş mi?			
Yazılımın sızma testi yapılmış mı?			
Performans testi yapılmış mı?			
Tasarımın uygulamasında ortaya çıkan uyumsuzluklar ve aksaklıklar belirtilerek çözüm yöntemleri tartışılmış mı?			
Yapılan işlerin zorluk derecesi?	x	0-25	
Sözlü Sınav			
Yapılan sunum başarılı mı?	x	0-5	
Soruları yanıtlama yetkinliği?	x	0-20	
Devam Durumu			
Öğrenci dönem içerisindeki raporlarını düzenli olarak hazırladı mı?	x	0-5	
Diğer Maddeler			
Toplam			

DANIŞMAN (JÜRİ ADINA):

DANIŞMAN İMZASI: