

**ÇANKIRI KARATEKİN ÜNİVERSİTESİ  
MÜHENDİSLİK FAKÜLTESİ  
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**



**PROJENİN KONUSU**

**.NET Core 6 Tabanlı Bir API ile Avukat Soru-Cevap Sistemi ve Otomatik Cevaplama  
Özelliğı Geliştirme**

**BİTİRME PROJESİ**

**ÇAĞRI ŞENTÜRK  
EMRE UĞUZ  
HAKAN ÖZDEMİR**

## ÖNSÖZ

Sevgili Okuyucular,

Bu önsözde, sizi .Net Core 6 tabanlı bir API kullanarak geliştirdiğimiz avukat soru-cevap sistemi projesine davet etmek istiyoruz. Bu proje, hukuk alanında yapay zekanın kullanımını ele almakta ve avukatların iş yükünü azaltmayı hedeflemektedir.

Günümüzde, hukuk alanında hızla ilerleyen gelişmeler yaşanmaktadır. Bu gelişmeler arasında yapay zeka teknolojilerinin hukuk sistemine entegre edilmesi büyük bir önem taşımaktadır. Bu projede, yapay zeka tekniklerini kullanarak avukatların iş verimliliğini artırmayı ve aynı zamanda kullanıcıların hukuki destek hizmetine daha hızlı erişimini sağlamayı amaçladık.

Projemizde, kullanıcıların sorularını iletebilecekleri bir web arayüzü tasarladık. Kullanıcıların soruları, özel bir Python kodu aracılığıyla daha önce sorulan sorularla karşılaştırılmaktadır. Eğer soru benzerlik oranı yüzde 80 ve üzerindeyse, sistem otomatik olarak kullanıcıya daha önce verilen cevabı iletmektedir. Bu sayede, avukatların aynı sorulara tekrar tekrar cevap verme ihtiyacı ortadan kalkmakta ve kullanıcılar daha hızlı bir şekilde hukuki bilgiye ulaşabilmektedir.

Bu proje aynı zamanda bir lisans tezi olarak hazırlanmaktadır. Çalışmamızda, .Net Core 6 API, Python kodlama ve Case-Based Reasoning gibi yöntemlerden yararlanarak bir soru-cevap sistemi oluşturulması üzerine odaklandık. Tez çalışmamızda, projenin materyal, metot ve sonuçlarını ayrıntılı bir şekilde ele alacağız.

Bu projede emeği geçen danışmanlarımız ve destekçilerimize en içten teşekkürlerimizi sunmak istiyoruz. Ayrıca, projemizin gerçekleştirilmesinde bize yardımcı olan tüm arkadaşlarımıza da teşekkür etmek istiyoruz.

Umarız bu çalışmamız, hukuk alanında yapay zeka teknolojilerinin kullanımına ilişkin farkındalığı artıracak ve gelecekte daha ileri çalışmalara ilham verecektir.

Saygılarımızla,

ÇAĞRI ŞENTÜRK,  
EMRE UĞUZ,  
HAKAN ÖZDEMİR

Çankırı 2023

## İÇİNDEKİLER

	Sayfa No
ÖNSÖZ.....	II
İÇİNDEKİLER.....	III
ÖZET.....	IV
1. GENEL BİLGİLER.....	1,3
1.1. Giriş.....	1
1.2. Amaç ve Hedefler.....	1,2
1.3. Literatüre Katkı.....	2
1.4. Tez Çalışması Akış Şeması	3
2. LİTERATÜR TARAMASI	4,8
2.1. Genel Çalışma alanlarında yapay zeka kullanımı.....	4,5
2.2. Hukuk alanında yapay zeka kullanımı ve web çalışmaları.....	5,6,7
3. MATERYAL VE METOT	8,16
3.1. Soru Sorulması.....	9
3.2. Retrive .....	9
3.2.1.Python ile yeni gelen soruyu önce ki soru ile kıyaslama .....	10,11
3.3. Reuse .....	11
3.3.1. Benzerlik oranı en yüksek olan soru belirlenir .....	11
3.4. Revise .....	11
3.4.1 Benzerlik Oranı en yüksek çıkan sorunun cevabının yeni sorulan soruya cevap olarak gönderilmesi .....	11
3.5. Veritabanı .....	12
3.5.1. Veritabanı tasarımı ve uygulaması.....	12
3.5.2 Veritabanı Tasarımı .....	13,14
3.6. Kullanılan Teknolojiler .....	14,15,16
4. ARAYÜZ TASARIMI	16,22
4.1. Anasayfa tasarımı.....	16
4.2. Kayıt olma sayfası tasarımı.....	17
4.3.Giriş yapma sayfası tasarımı.....	18
4.4 Soru sorma ekranı tasarımı.....	19
4.5. Soru detay ekranı tasarımı.....	20
4.6. Avukat soru listesi sayfası tasarımı .....	21
4.7. Avukat cevap sayfası tasarımı .....	22
5. BULGULAR VE TARTIŞMA	22,23,24
5. SONUÇ	24,25
6. KAYNAKLAR.....	26

## ÖZET

Günümüzde hukuk alanında sürekli olarak yeni gelişmeler yaşanmaktadır ve bu gelişmelerle birlikte yapay zekanın hukuk sistemine entegre edilmesi konusu uzun bir süredir tartışılmaktadır. Bu çalışma, avukatların iş yükünü azaltmayı ve aynı zamanda kullanıcıların avukatlara daha hızlı bir şekilde erişim sağlamayı hedeflemektedir. Bu amaçla, bir web arayüzü tasarlanmış ve bu arayüz sayesinde kullanıcılar ile avukatlar bir araya getirilmiştir.

Bir avukatın günlük olarak karşılaştığı soru sayısı oldukça fazladır ve bu soruların bir kısmı birbirine oldukça benzerdir. Bu çalışmada, .Net Core 6 tabanlı bir API kullanılarak bir avukat soru-cevap sistemi geliştirilmiştir. Ayrıca, Python ile sorulan soruların benzerlik oranı %80'in üzerinde ise avukatın daha önce verdiği cevabı otomatik olarak kullanıcıya göndermektedir. Bu yöntemle, avukatların iş yükü azalırken kullanıcılar da çok daha hızlı bir şekilde cevap alabilmektedir.

Bu çalışma, avukatların zamanını daha etkin bir şekilde kullanmasını sağlarken aynı zamanda hukuk hizmetlerine erişimi kolaylaştırmaktadır. Yapay zeka tabanlı bu soru-cevap sistemi, hukuk alanında teknolojinin gücünü kullanarak daha verimli bir iletişim ve hızlı çözümler sunmayı hedeflemektedir.

## 1. GENEL BİLGİLER

### 1.1. Giriş

Günümüzde, avukatlar gün içerisinde birçok soru ile karşılaşır. Bu soruların cevaplandırılması ise belli bir zaman alarak avukatlara ayrı bir iş yükü olarak döner. Aynı zamanda bu durum müşteri memnuniyeti konusunda da olumsuz etkiler bırakabilir. Bu nedenle avukatlar, doğru ve hızlı şekilde yanıt vermek için yeni çözümler aramaktadır.

Bu ihtiyacı karşılamak üzere geliştirilen çalışmamız, avukatların sorulan sorulara daha hızlı cevap vermesini sağlayan ve benzer sorularla sürekli uğraşmalarını gerektirmeyen bir API tasarladık. Bu API, kullanıcıdan gelen soruları önceden sorulan sorular ile kıyaslayarak, benzerlik oranı %80 ise önceden sorulan soruya verilen cevabı otomatik olarak yeni kullanıcıya gönderiyor.

Bu proje, avukatların zamanını verimli bir şekilde kullanmalarına yardımcı olmak ve müşteri memnuniyetini artırmak amacıyla geliştirilmiştir. Çalışma aynı zamanda otomatik olarak yapılabilecek bu işleri manuel şekilde iş yükü olmaktan çıkarmayı hedeflemektedir.

Projenin kullanılan teknolojileri arasında yapay zeka ve makine öğrenimi gibi gelişmiş teknolojiler de yer almaktadır. Yapay zeka alanı, Dartmouth College'daki bir konferansta 1956'ya kadar resmi olarak kuruldu [1]. Bu sayede, proje gelecekte daha doğru ve hızlı bir soru-cevap sistemi oluşturma potansiyeli taşımaktadır.

Çalışmanın ana bileşenleri API ve makine öğrenmesi ile soruları karşılaştıran Python kodudur. API kullanıcıdan gelen soruları alır daha sonra Python kodu sorulan bu soruyu önce ki sorular ile kıyaslar. Eğer %80 benzerlik oranı var ise benzer soruya verilen cevabı otomatik olarak kullanıcıya gönderir. Eğer benzerlik oranı yeterli değilse soru avukata ulaşır ve avukat kullanıcıya cevap gönderir. Bu sayede hem kullanıcı için hem de avukatlar için faydalı bir uygulama olmuş oluyor.

Projenin gelecekteki çalışmaları arasında yapay zeka ve makine öğrenimi gibi daha gelişmiş teknolojilerin kullanılması yer almaktadır. Bu sayede, daha doğru ve hızlı bir soru-cevap sistemi oluşturma potansiyeli taşınmaktadır. Ayrıca, projenin diğer meslek gruplarına da kullanılması da düşünülebilir. Bu tez çalışması, proje geliştirme sürecinin ayrıntılı bir incelemesini sunarak, benzer projelerin geliştirilmesine katkı sağlamayı hedeflemektedir.

### 1.2. Amaç ve Hedefler

Bu çalışma, hukuk alanında yapay zeka kullanarak insanların avukatlara olan erişimini hızlandırmayı ve aynı zamanda avukatların iş yüklerini azaltmayı amaçlamaktadır. Bu amaç doğrultusunda, proje aşağıdaki hedefleri hedeflemektedir:

- **Hukuk alanında yapay zeka kullanarak insanların avukatlara olan erişimini hızlandırarak kolaylaştırmak**
- 
- **Avukatların iş yükünü azaltarak hem daha hızlı hem de daha verimli şekilde müşterilere cevap verebilmelerini sağlamak.**
- 
- **Bu çalışmanın yaygınlaşması ile insanların hukuki konularda daha çok bilgilendirilmesini sağlamak.**
- 
- **Otomatik cevaplama sistemi ile çalışmanın verimliliğini ve müşteri memnuniyetini arttırmak.**
- 
- **Gelecekte ki çalışmalara bir temel oluşturmak ve hukuk alanında yapılacak yeni yapay zeka projelerine örnek teşkil etmek.**

Bu hedefler, projenin geliştirme sürecinde rehberlik edecek ve projenin amaçlarına ulaşılmasını sağlayacaktır.

### **1.3.Literatüre Katkı**

Bu çalışmanın literatüre katkısı, hukuk alanında yapay zeka ve web uygulamalarının fırsatlarını ve zorluklarını ele almasıdır. Çalışmada kullanılan algoritmalar, müşterilerin hukuki sorularını hızlı ve doğru şekilde yanıtlamak için tasarlanmıştır. Bu tür bir uygulama avukatların iş yükünü azaltıp aynı zamanda müşterilerin hukuki alanda bilgilenmelerini sağlamaktadır.

Projenin literatüre katkısı, hukuk alanında yapay zeka kullanımının geliştirilmesine yönelik öneriler içermesidir. Projede kullanılan yöntemlerin daha da geliştirilmesi, hukuk alanında yapay zeka kullanımının daha etkili ve doğru sonuçlar vermesini sağlayabilir. Bu da, hukuk alanında yapay zeka kullanımının daha yaygın bir şekilde kullanılmasına olanak tanıyabilir.

Bu çalışmanın ayrıca, bir başka literatüre katkısı, yapay zeka ve makine öğrenmesinin hukuk alanında sağlayabileceği potansiyel faydaları göz önüne sermesidir. Özellikle büyük veri kümelerinin analizi ve hukuki süreçlerin giderek otomatikleştirilmesi gibi alanlarda bu tip çalışmaların kullanılması, hukuk sisteminin daha verimli çalışmasına öncülük edebilir.

Ancak unutulmamalıdır ki, yapay zeka kullanımının hukuk sistemi üzerinde olumsuz etkileri de olabilir. Yapay zekanın karar verme süreçlerinde hata yapabileceği ve bu hataların insanlara zarar verebileceğini unutmamak gerekiyor. Bu sebeple yapay zeka uygulamalarının hukuk sistemi üzerinde etkileri çok titizlikle incelenmeli ve gerekli önlemlerde buna göre

alınmalıdır.

Sonuç olarak, bu proje, hukuk alanında makine öğrenmesi kullanımı ve web uygulaması yapılması konusunda mevcut literatüre bir katkı sağlamaktadır. Aynı zamanda bu çalışma hukuk alanı dışında da kullanılmaya açık bir çalışmadır. Bu açıdan bakıldığında birden fazla katkısı olduğu söylenebilir. Kullanılan yöntemler ve makine öğrenmesi kısmı geliştirilerek çok daha iyi ve farklı sektörlerde kullanılabilir şekilde geliştirilmesi mümkün olan bir çalışmadır.

#### 1.4. Tez Çalışması Akış Şeması



*Şekil 1:Tez çalışması akış şeması*

## 2. LİTERATÜR TARAMASI

### 2.1. Genel Çalışma alanlarında yapay zeka kullanımı:

Müşteri Hizmetleri Yönetiminde Yapay Zeka Temelli Chatbot Geliştirilmesi [2] isimli çalışmada gerçek bir işletmenin çağrı merkezi işlemlerini yönetmek için makine öğrenmesi kullanarak farklı NLP teknikleri ile bir chatbot tasarlanmıştır. Bu chatbotu tasarlamakta ki amaç ise çalışanların iş yükünü azaltmak ve kullanıcıların firma, ürün v.s hakkında ki sorulara karşı bir çalışan kullanmadan otomatik cevaplar vermek. Bu sayede firma hem çalışanın iş yükünü azaltıp hemde hızlı bir şekilde sorulara dönüş sağlayabilecektir. Chatbotun çalışma mantığı ise şöyle, kullanıcılar sorular sorarak bu chatbota girdiler oluşturuyor. Chatbot ise yanıt olarak daha önce veri setinde oluşturulmuş olan cevabı otomatik olarak kullanıcıya gönderiyor. Bu çalışmada veri seti için Samsun'da bir teknoloji şirketinin sıkça sorulan sorular kısmında ki sorular ve cevaplar, çağrı merkezine gelen sorular ve cevapları, whatsapp müşteri hizmetleri kısmına gelen sorular olmak üzere 1000 adet veri kullanılıyor. Bu çalışmada ayrıca soruları sınıflandırmak için BERT modeli kullanılıyor. Kullanıcıdan alınan veriler ile BERT modeli eğitildikten sonra ise kullanıcıdan gelen sorunun veri setinde hangi sınıf ile eşleştiği tespit ediliyor. Modele gelen girdiler ile yapılan tahminlerle yüzdesel olarak hangi sınıfa en yakın ise o sınıftan seçilen cevap kullanıcıya gönderiliyor. Chatbot, gerçek insanlar ile denendiğinde ise whatsapp üzerinden gelen 3640 adet soruya cevap veriyor. Gelen soruların %77'sine doğru cevap veriyor. Denemenin son gününde ise doğruluk oranı %85 'lere kadar çıkıyor. Bunun sebebi ise veri setinin her geçen gün gelen sorular ile büyümesi. Veri seti büyüdükçe doğru cevap verme olasılığı artıyor. Projenin dezavantajları ise şu şekilde, bu çalışmada kullanılan BERT modeli basit sınıflandırmalar için uygun olsa da ayrıntılı sınıflandırma için yeterli değildir. Spesifik sorularda bu yüzden yanlış sınıflandırma yapılabilir. Aynı zamanda burada yapılan işlemde soruları karşılaştırıp ona göre bir cevap verilmiyor. Sadece sınıflandırma yapıp o sınıfa hangi cevap verildiyse o cevap veriliyor. Cevaplar ise çok ayrıntılı olmadığı için yeterince tatmin edici değil. Aynı zamanda başka bir firmada kullanılmak istendiğinde bu çalışmanın o firmaya entegre edilmesi zaman alır çünkü her firma için yeniden veri seti oluşturulması gerekiyor.

Bir başka çalışma olan “Bir Soru Cevaplama Sistemi: BayBilmiş” [3] isimli çalışmada Türkçe dili için çalışan bir soru cevaplama sistemi geliştirilmiş. Çalışmanın çıkış noktası arama motorlarına bir alternatif olmak. Çalışmanın iddiasına göre arama motorlarına sorulan belli kalıpta sorular için kullanıcının önüne sayfalarca web site çıkarmak yerine kullanıcının sorduğu soruya net bir cevap çıkarmak, hem daha efektif hem de daha kullanışlı. Örneğin bir kullanıcı "Türkiye'nin başkenti neresidir?" diye bir arama yaptığında, önüne onlarca site değil, sadece cevap olan "Ankara" yazısı çıkacaktır. Bu çalışma Türkçe dili için bir ilk olma özelliği taşıyor. Sistemin çalışma mantığı ise şöyle, Kullanıcının sorduğu soruyu sistem sınıflandırıyor ve çözümlüyor. Daha sonra ise bu soruyu arama motoru sorgularına dönüştürüyor. Arama motorundan gelen sonuç sayfalarında ki cümleler kurulan bir filtreden geçiriliyor ve cevaplar puanlandırılıyor. Puan olarak en yüksek çıkan 5 cevap kullanıcının önüne çıkıyor. Sorulan soruyu anlamak için içinde ki soru kelimeleri inceleniyor. Sistemde ise 53 adet soru kelimesi bulunuyor. Sorulan soruları çözümlmek için “Bir Türkçe Metindeki



Kelimelerin Cümle İçindeki Durumlarına Bakılarak Eklerine Ayrılması” [4] isimli çalışmadan referans alınmış. Kullanıcıya verilecek cevap konusunda ise iki yöntem kullanılmış. Birincisinde, arama motorunda çıkan sayfaların özetlerine göre bir sonuç çıkarmak. Diğer ise sayfaların içine girerek daha ayrıntılı bir çözümleme ile cevap göndermek. İlk yöntem hızlı ancak doğruluk oranı %43,9 olarak belirlenmiş. İkinci yöntem ise çok daha yavaş ancak doğruluk oranı %43,9+%16,6 olarak ölçülmüş. Buradan çıkan sonuç ise iki yönteminde dezavantajları var. Hızlı sonuç kullanıcı için önemli ancak, verilen cevabın doğruluk oranı çok düşük. Bu durumda ilk yöntem yeterince kullanıcı dostu olamamış. İkinci yöntem ise oldukça yavaş ve doğruluk oranı ilk yönteme göre fazla olsa da çok yeterli görünmüyor. Çalışmanın arayüzü ve verdiği cevap örneklerine baktığımızda, belli oranda işe yaradığı görülüyor. Ancak çok net cevaplar yerine yarıda kesilmiş cümlelerde sonuç olarak çıkıyor. Bu durum düzeltildiği takdirde çalışma çok daha faydalı bir hal alacaktır. Bu çalışmayı değerlendirmek için ise ortak soru bankasından (TREC) 524 adet soru sisteme sorulmuş. İlk yöntemle soruların %43,9'una doğru cevap verilmiş. Diğer yöntemde ise soruların %60'ına doğru cevap verilmiş. Çalışmada doğruluk oranları istenenin altında kalmış.

Bir başka çalışması olan Sıradan Sıraya LSTM Modeli ile Telegram Bot Uygulaması [5] isimli çalışmada LSTM ve sıradan sıraya modeli kullanılarak telegramda çalışan bir chatbot geliştirilmiş. Tasarlanan chatbotun kullanım amacı iş görüşmeleri öncesi adaylar için bir değerlendirme yapmak. Bu çalışmada RNN mimarisi kullanılmıştır. LSTM sayesinde bir sonraki konuşmayı tahmin etmek kolaylaşmıştır. Çünkü LSTM konuşma geçmişine dönmeyi sağlamıştır. Çalışmada veri kümesi olarak chatterbot isimli veri kümesi kullanılmıştır. Bu veri kümesinde 1984 adet soru ve cevap verisi vardır. Pythonwhere kullanarak telegram ile entegrasyon sağlanmıştır. Çalışmada doğruluk oranı ise %79 olarak hesaplanmıştır.

## **2.2. Hukuk alanında yapay zeka kullanımı ve web çalışmaları:**

Hukuk alanında da yapay zeka kullanımı yıllar geçtikçe yaygınlaşmaya başlıyor. Bu alanda çeşitli çalışmalar ülkemizde de devam ediyor. Örneğin “Yargıtay Kararlarının Suç Türlerine Göre Makine Öğrenmesi Yöntemleri İle Sınıflandırılması” [6] isimli çalışmada FastText yöntemi kullanılmış.2020 yılına ait ve kara bağlanan 531 adet karar metinlerini suç türlerine göre sınıflandırılmış, eğitim ve test grubu olarak ayrılmış. FastText'in kendi optimizasyon seçenekleri ile test edildiğinde sınıflandırma konusunda %44.7 doğruluk oranına ulaşılmış. Çalışmada amaç avukatların emsal karar bulma konusunda zorlanmalarını önlemek. Bir davanın emsal kararına ulaşmak bir avukat için zor bir durum. Birden fazla emsal karar olabiliyor ve bu durum bir iş yükü oluşturuyor. Çalışmada amaç emsal kararları makine öğrenmesi ile bulup avukatın iş yükünü azaltmak. Model olarak kullanılan FastText'in sağladığı avantaj veri setinde ki noktalama işaretleri, boşluklar gibi nedenlerin sonuca etki etmesini engellemesidir. Veri setinin eğitilmesi konusunda ise 3 adet test grubu oluşturulmuş. Tüm kombinasyonlar için 3 test grubunun üzerinde çalışılmış. Bu testlerin ardından emsal kararlara ulaşması açısından elde edilen doğruluk oranı %44.7 olarak hesaplanmış. Bu oran oldukça düşük bir orandır.

Bir başka çalışmada, otomatik özetleme için atanmış anahtar kelimeler kullanılmıştır ve bu nedenle anahtar kelime ataması çalışmanın önemli bir parçasını oluşturmaktadır. Anahtar kelimeler, metinlerdeki bilgilere erişimi kolaylaştırmak, metinleri içerdikleri içeriğe göre

sınıflandırmak ve istenen bilgilere erişmek için kullanılan tanımlayıcı sözcük veya ifadelerdir (Liu et al., 2009 [7]). Anahtar kelime atama sürecini otomatikleştirme fikri, doğrudan anahtar kelimeleri veya bilgi çıkarma yöntemiyle elde edilen bilgileri içeren metni belgeleme mantığına dayanmaktadır. Turney'in 1992 yılındaki çalışmasında, metinlerin yaklaşık %75'inin atanması gereken anahtar kelimeleri içerdiği gösterilmiştir, bu da çoğu kelimenin metin üzerinde işlem yapılarak çıkarılabileceğini göstermektedir (Turney, 1992 [8]).

Anahtar kelime ataması aynı zamanda metin kategorizasyonu olarak da adlandırılır. Bu yöntemde, önceden oluşturulmuş bir anahtar kelime listesindeki kelimeler metinlerle ilişkilendirilir. İlk olarak, bir anahtar kelime havuzu oluşturulur. Ardından, metinler, kelime yapılarının bu anahtar kelime havuzundaki kelimelere olan yakınlığına göre sınıflandırılır (Uzun, 2005 [9]). Bu süreçte Naive Bayes, Logistic Regression Model, Support Vector Machine gibi teknikler birçok çalışmada kullanılmıştır (Joachims, 1998 [10]; Ikonomakis et al., 2005 [11]; Aseervatham, 2011 [12])."

Bir başka çalışma olan "Emsal Hukuk Dokümanlarının Otomatik Belirlenmesi" [13] isimli çalışma ise benzer mantıkla yapılmıştır. Çalışmada UYAP bilgi bankası üzerinden bir veri seti oluşturulmuştur. Dava dosyalarının incelenmesi ile elde edilen dava şablonları ile girdi ve çıktılar sağlamak için LSTM modeli ile 3 farklı sistem oluşturulmuş. Çıkan metinler farklı BERT modellerinden elde edilen temsil vektörleri ile test verileri için dökümanlar elde edilmiştir. Veri setinde 97485 adet döküman bulunuyormuş. Elde edilen veriler ön işleme tabii tutulmuş. Belirlenen dosyaları avukatlara da sorarak bir test grubu daha oluşturulmuş. Çıkan sonuca göre avukatlar arasında dahi emsal kararlara karar verme konusunda çelişkiler ortaya çıkmış. Bu durum çalışılan konunun büyük bir problem olduğunu gösteriyor.

Bir başka çalışmada ise İsviçre'de ki yasama metinlerinin sınıflandırılması için yapılmıştır. Çalışmanın amacı ise yasama metinlerinin yazım formatını tespitini otomatik olarak yapmaktır. Çalışma iki basamaktan oluşmaktadır. İlk adımda metinde ön işleme yapılır. Tokenization, text segmentation gibi işlemler bu bölümde yapılır. Buradan gelen çıktı hata modelleme için kullanılır. Bu sayede hatalı formatlar otomatik olarak tespit edilir. Bu çalışma sayesinde federal büro çalışanlarının iş yükü azaltılıp metin denetimi otomatik hale getirilmiştir.(Höfler ve Kyoko,2012)[14]

Yapay Zeka 'nın hukuk alanındaki kullanımıyla ilgili bir çalışma, 2018 yılında ODTÜ Teknokent'te faaliyet gösteren Kodex Bilişim adlı bir şirketin geliştirdiği ARYA adlı uygulama üzerinde gerçekleştirilmiştir.[15]ARYA, Türkçe karar metinlerini yapay zekanın doğal dil işleme teknolojisiyle sayısal verilere dönüştürerek ve çeşitli istatistiksel yöntemlerle sonuçları tahmin ederek geliştirilmiştir. ODTÜ Teknokent tarafından yayınlanan bir haberde ARYA ile ilgili olarak, "Firma, bu sistemi geliştirirken Adalet Bakanlığı'nın elinde bulunan 1,5 milyon veriden sadece bir kısmını analiz edebilen yazılım, şu anda yalnızca boşanma davalarıyla ilgilenmektedir. Yazılım, hakimlere yardımcı olmak amacıyla tasarlanmış olup onların işini elinden almayı hedeflememektedir. Ayrıca şirket, hukukçuların kullanabileceği yerli yapay zeka teknolojisiyle donatılmış bir arama motoru ve hukuki sorunları tartışabilecekleri yerli bir sosyal medya platformu projesi üzerinde de çalışmaktadır" ifadelerine yer verilmiştir. Bu bilgilere göre, ARYA uygulamasının Türkçe karar metinleri üzerinde çalıştığı ve tüm kararları kapsamadığı, yalnızca boşanma davalarına odaklandığı

anlaşılmaktadır. Ayrıca, aynı şirketin hukukçuların kullanımı için yapay zeka teknolojisi kullanan bir arama motoru üzerinde çalıştığı da vurgulanmaktadır. Bu projeye ilişkin olarak, İstanbul, Ankara ve İzmir Baroları tarafından düzenlenen "Yapay Zeka Çağında Hukuk 2019" başlıklı çalıştayın sonuç raporunda, ARYA uygulamasının Yargıtay kararlarını Onama, Bozma ve Düzelterek Onama şeklinde sınıflandırdığı ve bunu %90 doğruluk oranıyla gerçekleştirdiği belirtilmiş, ayrıca projenin TÜBİTAK Bireysel Genç Girişimcileri Destekleme Programı tarafından desteklendiği ifade edilmiştir."

Lame (2005), Fransızca hukuki metinler üzerinde yaptığı çalışmada, hukuki metinlerdeki hukuki kavramların, metin içindeki konumlarından yola çıkarak sözdizimsel analiz ile Fransızca için bir hukuki terim ontolojisi oluşturmuştur. Bu çalışma, sözdizimsel analizle tespit edilen kelimelerin kullanım sıklığı ve tf-idf (Term Frequency-Inverse Document Frequency) değerlerini kullanarak hukuki terim olma durumunu incelemiştir. Ayrıca, çalışma hukuki terimler arasındaki ontolojik ilişkileri (anlamsal ilişkileri) belirlemeyi amaçlamıştır." (Lame, 2005)[16]

Doğal dil işleme teknikleri, araştırma alanı olarak yaygınlaşmasıyla birlikte farklı disiplinlerde hızla doğal dil işleme uygulamalarının arttığı bir döneme girmiştir. Hukuk alanında da bu çalışmalar hızla yaygınlaşmaktadır ve farklı dillerdeki hukuk metinleri üzerinde çalışmalar yürütülmektedir (Lenci et al., 2007). Bu çalışmalar için öncelikle bir hukuk ontolojisi/leksikonu oluşturulur. Oluşturulan ontolojiler, verilerin düzenlenmesi ve yapılandırılması, muhakeme ve problem çözümü, semantik indeksleme ve arama, semantik uyum ve ilişki, ve kapsam çıkarımı gibi beş temel problem için veri sağlamaktadır (Breuker et al., 2004). [17]

## **LİTERATÜR TARAMASI SONUCU**

Literatür taraması, bir araştırma çalışması veya tez yazımı sürecinde oldukça önemli bir adımdır. Bu adım, mevcut bilimsel yayınlardan ve akademik kaynaklardan ilgili konuyla ilgili bilgi edinmeyi ve mevcut bilgi birikimini anlamayı sağlar. Literatür taraması yaparak, daha önce yapılan çalışmaları, bulguları, yöntemleri ve açıklamaları inceleyebiliriz.

Literatür taraması, bir araştırma çalışmasının temelini oluşturur ve araştırma sorularına yanıtlar bulmayı amaçlar. Bu aşama, araştırmacının konuyla ilgili güncel bilgileri öğrenmesini, araştırma açığı ve boşluklarını belirlemesini, var olan teorileri ve yaklaşımları analiz etmesini ve kendi çalışmasını bu temel üzerine inşa etmesini sağlar.

Yaptığımız bu literatür taramasında ... adet çalışma detaylıca incelenmiştir. Bu çalışmaları ise iki gruba ayırdık. Bu gruplar, hukuk için olanlar ve hukuk dışı sektörde olanlar.

Hukuk dışı sektörlerde yapay zeka kullanımıyla ilgili projelerde genel olarak, müşteri hizmetleri kısmında yapay zeka kullanılmış. Ancak çoğu çalışmada, hazır cevaplar üstünden gidilmiş. Kullanılan BERT modeli yerine durum tabanlı çıkarsama yöntemi kullanılsaydı çalışmalar çok daha verimli olabilirdi. Çalışmalar incelendiğinde, başarı oranı da

beklenenden çok alt seviyede. Başarı oranı düştüğü için kullanıcı için kullanışlı olma ihtimali azalıyor. Aynı zamanda, başarı oranının düşmesi, iş yükünü azaltmak yerine arttırabilir. Yapay zeka ile verilen cevabı beğenmeyen kullanıcılar müşteri hizmetlerine kendileri ulaşmak isteyecektir. Bu durumda ekstra bir iş yükü ortaya çıkacaktır. Bundan dolayı yöntem olarak durum tabanlı çıkarsama kullanılsaydı çalışmalar çok daha başarılı sonuçlar alabilirdi.

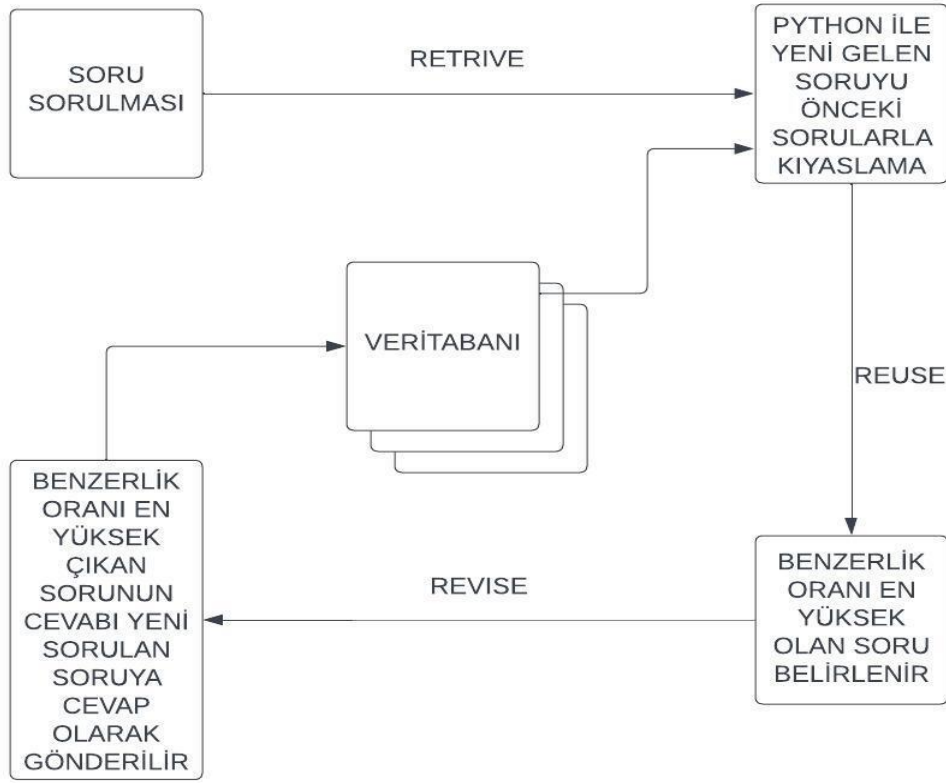
Hukuk sektöründe yapay zeka kullanımıyla ilgili yaptığımız literatür taramasında ise , kendi çalışmamıza benzer çalışma sayısının ülkemizde çok az olduğu ortaya çıkmıştır. Yapay zeka gelişen bir alan ve bu alanın hukuk sektörüne entegre edilmesi için çok daha fazla çalışma yapılmalıdır. Yapılan çalışmaların çoğu , yapay zeka ile sınıflandırma yapmak üzerine. Benzer dava dosyalarını eşleştirmek gibi çalışmalar mevcut. Ancak bizim çalışmamızda olan duruma yakın bir çalışma bulunmuyor.

Sonuç olarak, yaptığımız çalışma hukuk sektörü için değerli bir çalışmadır. Hukuk alanında yapay zeka kullanımı konusunda daha çok çalışmalar yapılmalıdır. Hukuk dışında ki sektörlerde çalışmamıza benzer çalışmalar yapılmış olsa da genellikle BERT modeli üstünde durulmuş ve durum tabanlı çıkarsama kullanılmamıştır. Genellikle ise başarı oranı oldukça düşük kalmıştır. Ancak durum tabanlı çıkarsama yöntemi tercih edilseydi bu çalışmalar çok daha başarılı olabilirlerdi. Çalışmamız, hem iş yükünü azaltırken hem de kullanıcı için verimli ve tutarlı cevaplar göndermeyi amaçlıyor. Ancak incelediğimiz çalışmalarda görülen durum tutarlılık konusunda yeterli başarı sağlanamamıştır. Çalışmamızda, soru ve cevap sayısı arttıkça, başarı oranı da bir o kadar artacaktır. Durum tabanlı çıkarsama yöntemi kullandığımız için çalışmamızda, sorulan soru sayısı arttıkça gönderilen cevap kısmında tutarlılığı o kadar artacaktır. Ancak literatür taramasında incelenen çalışmalarda böyle bir durum söz konusu değildir.

### **3. MATERYAL VE METOT**

Bu çalışmanın temel materyali, .Net Core 6 ile geliştirilen API ve Python kodundan oluşmaktadır. .Net Core 6 API, kullanıcıların soru sorması, cevap alması ve soruların kaydedilmesi gibi işlevleri gerçekleştiren temel platformdur. Python kodu ise soruların benzerlik oranını hesaplamak ve önceden verilen cevabı kullanıcıya iletmek için kullanılır.

Bu çalışmada kullanılan yöntem, Case-Based Reasoning (CBR) olarak adlandırılan bir yaklaşıma dayanmaktadır. CBR, bir problemi çözmek için önceden benzer problemlere dayanarak bir çözüm geliştirmeyi hedefleyen bir yapay zeka yöntemidir. Bu projede, önceden sorulan sorular benzerlik analizi için bir "case base" oluşturur ve yeni sorulara bu case base üzerinden çözümler sunulur.



Şekil 2: Case-Base Reasoning Diagramı

### 3.1 SORU SORULMASI

Çoğu sektörle müşteri ile iletişim çok önemlidir. Gün içinde müşterilerin sorularına cevap vermek tüm sektörler için çok değerlidir. Hukukta bu sektörlerden birisidir. Avukatlar, günde onlarca belki de yüzlerce soru cevaplamak durumunda kalıyor. Müşteri ile iletişim hukuk sektöründe de önemli bir faktör olduğundan, bu tür bir sistem hukuk sektöründe faydalı olabilir. Sorulara hızlı ve doğru cevaplar vermek, müşteri memnuniyetini artırır. Çalışmamızda, kullanıcı sistemde ki istediği avukata seçtiği kategoride soru sorabiliyor. Sorulan soruların tamamı veri tabanı olarak kullandığımız Mssql veri tabanına kaydediliyor. Sorulan sorular; soruyu soran kullanıcı, sorunun tamamı, sorunun sorulduğu avukat bilgileri ile veri tabanına kaydediliyor.

### 3.2. RETRIVE

Çalışmanın bu aşamasında, kullanıcının sorduğu yeni soru sisteme girilir. Python kodu, yeni soruyu daha önceki sorularla karşılaştırır ve benzer soruları bulur. Benzer soruların cevaplarına bu aşamada ulaşılır.

### 3.2.1. PYTHON İLE YENİ GELEN SORUYU ÖNCE Kİ SORU İLE KIYASLAMA

Python, makine öğrenmesi ve yapay zeka alanında en çok kullanılan programlama dillerinden biridir. Python, birçok açık kaynaklı kütüphane ve araç içermesi sebebiyle makine öğrenmesi uygulamalarında oldukça yaygın olarak kullanılmaktadır. Örneğin, Scikit-Learn, TensorFlow gibi kütüphaneler, Python dili ile yazılmıştır ve makine öğrenmesi algoritmalarının hızlı ve verimli bir şekilde uygulanmasını sağlar.

Python dilinin makine öğrenmesi alanında kullanımı, genellikle veri analizi ve veri işleme süreçlerine dayanır. Verilerin okunması, temizlenmesi, ön işleme ve özellik çıkarımı gibi işlemler, Python dilinde bulunan Numpy, Pandas, Scipy, Matplotlib, Seaborn ve Statsmodels gibi kütüphaneler ile yapılabilir.

Python'da bulunan Scikit-Learn gibi kütüphaneler, sınıflandırma, regresyon, kümeleme, boyut indirgeme gibi birçok makine öğrenmesi algoritmasını içermektedir. Ayrıca, Deep Learning alanında kullanılan TensorFlow, Keras ve PyTorch gibi kütüphaneler, yapay sinir ağları ve derin öğrenme modellerinin kolayca oluşturulmasına olanak sağlar.

Python'un makine öğrenmesinde kullanımı, hem araştırmacılar hem de uygulama geliştiricileri tarafından yaygın olarak kabul edilir. Python, kolay öğrenilebilir olması ve çok sayıda kütüphane ve araç sunması nedeniyle, makine öğrenmesi ve yapay zeka alanındaki gelişmelerin hızlanmasına katkıda bulunmuştur.

Yaptığımız çalışmada, Python ile gelen soruları veritabanında önceden bulunan sorular ile anlamsal olarak kıyaslıyoruz. Bu çalışmada, Scikit-learn kütüphanesindeki "TfidfVectorizer" ve "cosine\_similarity" fonksiyonları kullanılmıştır.

Scikit-learn kütüphanesindeki TfidfVectorizer ve cosine\_similarity fonksiyonları, bir metin belgesi kümesindeki metin belgelerin vektörlerine dönüştürülmesi ve bu metin belgeleri arasındaki benzerliğin ölçülmesi için çoğunlukla kullanılır. TfidfVectorizer fonksiyonu, bir metin belgesinin kelime sıklıklarını TF-IDF (term frequency-inverse document frequency) değerlerine dönüştürerek, her bir metin belgesinin bir vektöre dönüştürülmesine olanak sağlar. Bu işlemin neticesinde, her bir kelime metin belgesinin vektöründe bir boyut olarak temsil edilir ve her metin belgesi, içerdiği tüm kelimelerin bir kombinasyonunu içeren bir vektör haline gelir. cosine\_similarity fonksiyonu ise, iki metin belgesinin vektörleri arasındaki kosinüs benzerliğini hesaplar ve bu sayede bu metin belgelerinin ne kadar benzer olduğunun ölçülmesine olanak sağlar. Benzerlik ölçümü sonucu, iki metin belgesinin arasındaki açığa bağlıdır ve 1'e yaklaştıkça iki metin belgesi arasındaki benzerlik artar. Bu yöntem, özellikle belirli bir konu veya anahtar kelimeyle ilişkili metin belgeleri bir metin belgesi kümesinden ayıklamak için kullanılabilir.

Bu çalışmada kullandığımız Python kodu, veri tabanından alınan soruları "TfidfVectorizer" kullanarak vektörlere dönüştürür. Daha sonra ise, verilen bir soru için benzer soruları "cosine\_similarity" fonksiyonunu kullanarak bulur.

Çalışmada kullanılan Python kodu, argüman olarak bir soru alır ve “find\_similar\_questions” fonksiyonunu çağırarak benzer soruları bulur.” find\_similar\_questions” fonksiyonu, verilen soruya en çok benzeyen soruyu ve benzerlik derecesini bulur ve bu sonucu bir dizide depolar. Sonra, JSON formatında bir sonuç dizisi oluşturur ve sonucu bulur.

### **3.3. REUSE**

Çalışmanın bu aşamasında, benzer soruların cevaplarından en uygun olanı bulunur, seçilir. Seçim işlemi, benzerlik oranına bakılarak yapılır.

#### **3.3.1. BENZERLİK ORANI EN YÜKSEK OLAN SORU BELİRLENİR**

Çalışmada, “Treshold” değeri ise 0.8 olarak belirlendi. Bu sayede, bir soru sorulduğunda ,kullanıcı soruyu gönderdiği anda Python kodu devreye giriyor ve sorulan soruyu veri tabanında ki diğer sorular ile anlamsal olarak kıyaslıyor. Bu kıyaslama sonucunda yüzde 80 ve üzeri benzerlik bulunan sorular belirleniyor. Bu belirlenen sorular arasından ise benzerlik oranı en yüksek çıkan soru seçiliyor. Seçilen sorunun cevabı ise otomatik olarak soruyu soran kullanıcıya gönderiliyor. Bu sayede hem kullanıcı tarafında müşteri memnuniyeti artıyor. Hem de avukat tarafında iş yükü azalıyor.

### **3.4. REVISE**

Bu aşamada, kullanıcının aldığı cevapla ilgili geri bildirimleri toplamak ve saklamak gerekir. Geri bildirimler, cevap kalitesini iyileştirmek ve sistemdeki soruları geliştirmek için kullanılabilir.

#### **3.4.1 BENZERLİK ORANI EN YÜKSEK ÇIKAN SORUNUN CEVABININ YENİ SORULAN SORUYA CEVAP OLARAK GÖNDERİLMESİ**

Çalışmamızda, kullanıcı sorusunu gönderdiğinde öncelikle Python ile yapılan benzerlik oranı hesaplama işleminden sonra ise, benzerlik oranı yüzde 80’in üzerinde çıkan tüm sorular arasında bir kıyaslama daha yapılır. Benzerlik oranı en yüksek olan soru belirlenir . Daha sonra ise belirlenen bu sorunun cevabı Mssql veri tabanından API tarafından çekilir. Bu cevap, yeni kullanıcıya gönderilir. Çalışmamızda ki amaç, gönderilen bu cevabın yeni kullanıcıyı tatmin edecek seviyede olmasıdır.

### 3.5. VERİTABANI

Çalışmanın verileri, Microsoft SQL Server kullanılarak bir veri tabanında güvenli bir şekilde tutulmaktadır. SQL Server, Microsoft tarafından geliştirilen ve yaygın olarak kullanılan bir ilişkisel veri tabanı yönetim sistemidir. Verilerin bu güçlü ve güvenilir platformda saklanması, çalışmanın veri bütünlüğünü ve güvenliğini sağlamak açısından önemli bir rol oynamaktadır.

Çalışmada, veri tabanı yönetimi için Entity Framework Core kullanılmıştır. Entity Framework Core, .NET Core için açık kaynaklı bir ORM (Object-Relational Mapping) aracıdır. Bu araç, veri tabanı işlemlerinin daha kolay ve verimli hale getirilmesine olanak sağlamaktadır. Veri tabanı ile etkileşim gerektiren işlemler, örneğin; GET, PUT, POST, Delete v.s, Entity Framework Core'un sağladığı zengin API'leri kullanılarak gerçekleştirilmektedir. Bu sayede, veri tabanı sorgularını doğrudan SQL dilinde yazmak yerine, nesne odaklı bir yaklaşım benimsenerek kod tekrarı azaltılmış ve geliştirme süreci hızlandırılmıştır.

Kullanıcıların soruları, veri tabanında saklanır ve ilgili avukatlara yönlendirilir. Bu sayede, kullanıcılar sorularını sisteme iletebilir ve uzman avukatlar tarafından cevaplandırılmasını sağlayabilirler. Veri tabanında saklanan sorular, kullanıcıların güvenliği ve gizliliği ön planda tutularak korunmaktadır. Kullanıcıların kimlik doğrulama işlemleri JWT (JSON Web Token) kullanılarak sağlanmaktadır. Bu sayede, her kullanıcıya benzersiz bir kimlik belirtilir ve yalnızca kendi sorularını görebilir ve cevaplarını alabilirler. Bu güvenlik önlemi, kullanıcı verilerinin yetkisiz erişime karşı korunmasını sağlar.

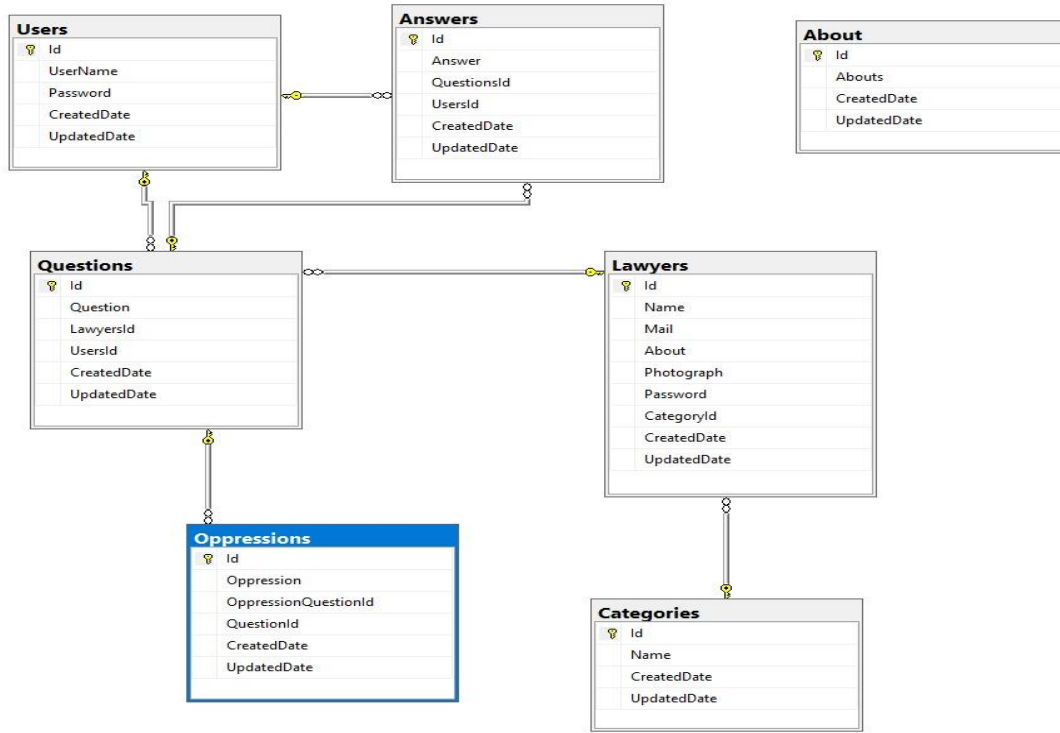
Veriler, Microsoft SQL Server kullanılarak güvenli bir şekilde saklanır ve veri tabanına erişim sınırlandırılır. Veri tabanına erişim izinleri, kullanıcı rolleri ve yetkilendirmeleriyle kontrol edilir, böylece sadece yetkilendirilmiş kullanıcılar veri tabanına erişebilir. Bu da veri tabanının güvenliğini sağlayarak verilerin gizliliğini ve bütünlüğünü korur.

#### 3.5.1. VERİTABANI TASARIMI VE UYGULAMASI

Veri tabanı tasarımı, projenin gereksinimlerine ve kullanıcıların ihtiyaçlarına uygun şekilde yapılmıştır. Veri tabanı uygulaması ise MSSQL veri tabanı yönetim sistemi kullanılarak gerçekleştirilmiştir. Çalışmada tablolar arası bağlantı birden çok kez kullanılmıştır. Kullanıcı ile soru, soru ile cevap, avukat ile kategori, kullanıcı ile cevap, avukat ile kategori ve soru ile benzerlik oranı tablosu arasında bağlantılar vardır.



### 3.5.2 Veritabanı Tasarımı



Şekil 4:ER diagramı (entity relationship diagram)

Veri tabanında ki Users tablosu, uygulamayı kullanacak olan kullanıcıların bilgilerini tutar. Bu bilgiler; kullanıcı adı, şifresi, kayıt olduğu tarih ve güncelleme yapılan tarihleri kapsamaktadır. Bu bilgiler her kullanıcı için kendine özeldir. Id değeri ise primary key olduğu için her kullanıcı için sadece bir kere kullanılabilir.

Veri tabanında ki Lawyers tablosu, uygulamayı kullanacak olan avukatların bilgilerini tutar. Bu bilgiler; avukatın adı, mail adresi, hakkında bilgisi, fotoğrafı, şifresi, kayıtlı olduğu kategori, kayıt tarihi ve güncelleme tarihi şeklindedir. Bu tabloda avukatlar kategori tablosu ile bire-çok bağlantı kurmuştur. Yani bir kullanıcı bir kategoriye bağlı iken, bir kategoride birden fazla kullanıcı olabilecektir.

Veri tabanında ki kategori tablosu ise, avukatların kayıtlı olduğu kategorileri ve o kategorilere ait olan Id'leri tutmaktadır.

Veri tabanında ki Question tablosunda ise, Soru, soruyu soran kullanıcının Id'si ve sorunun sorulduğu avukatın Id'si tutulmaktadır. Bu tabloda Users tablosu ile soru arasında bire-çok bağlantı bulunmaktadır. Yani, bir kullanıcı birden fazla soru sorabilirken, bir soru sadece bir kullanıcı id'si ile bağlantılı olacaktır. Aynı şekilde, Question tablosu Lawyers tablosu ile bire-çok bağlantı bulunmaktadır. Yani, Bir soru bir avukata giderken, bir avukat birden fazla soruya cevap verebilmektedir.

Veri tabanında ki Answers tablosunda ise;cevap, cevaplanan sorunun id'si, soruyu soran kullanıcının id'si,cevabın veri tabanına kaydedilme tarihi ve güncelleme tarihi tutulur. Bu tabloda, Question tablosu ile bire-bir bağlantı vardır. Yani,Her sorunun bir cevabı vardır. Her cevabında bir sorusu vardır. Users tablosu ile ise bire-çok bağlantı vardır. Her cevabın eşleştiği bir kullanıcı varken, bir kullanıcı birden fazla soru sorup bunun cevabını almış olabilir.

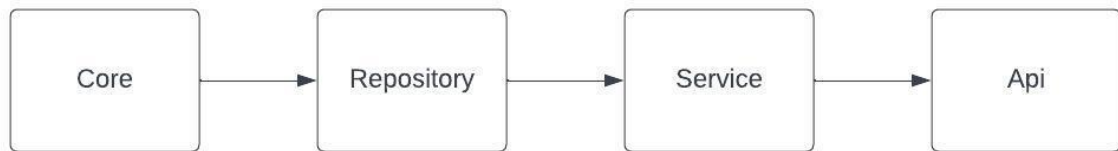
Veri tabanında ki en önemli tablolardan biri olan Oprresions tablosu, Python kodu ile yapılan benzerlik oranıyla alakalı bir tablodur. Python kodu ile yapılan benzerlik oranı hesaplama işlemi sonucunda elde edilen sonuç bu tabloya kaydedilir. Bu tabloda sorulan soru ve benzediği soru tutulur. Aynı zamanda benzerlik oranı da bu tabloda kaydedilir.Bu sayede sorulan soru hangi soruya yüzde kaç benziyor buradan anlaşılabilir. Bu tablonun bir diğer önemi ise dolaylı yoldan bağlantılı olduğu answers tablosundan cevaba ulaşmamıza yardımcı olmasıdır. Opperrassions tablosu Questions tablosu ile bire-bir bağlantıya sahiptir. Bu durum bize dolaylı yoldan Answers tablosuna da ulaşmamıza olanak sağlıyor. Bu yüzden benzerlik oranına göre cevabı kullanıcıya göndermek için bu tablodan oldukça yararlandık.

Veri tabanı tasarımı, normalizasyon prensiplerine uygun olarak yapılmıştır. Veritabanı tabloları arasında ilişkiler kullanılarak veri bütünlüğü sağlanmıştır.

### 3.6. Kullanılan Teknolojiler

Çalışmanın API kısmı, C# programlama dili kullanılarak .NET Core 6.0 teknolojisi ile yazılmıştır. .NET Core, açık kaynaklı bir platformdur ve farklı işletim sistemleri üzerinde çalışabilen uygulamalar geliştirilmesine olanak tanır. C# programlama dili ise, nesne yönelimli bir dil olup, güçlü bir tip kontrolüne sahiptir ve kolay anlaşılabilir bir sözdizimine sahiptir.

API kısmı bu çalışmanın en önemli noktalarından biri konumunda. Bu çalışmada .NET CORE 6.0 ile backend kısmı yapılırken katmanlı mimari yapısı benimsenmiştir.



Şekil 3:Çalışmada ki katmanlı mimari yapısı

Katmanlı mimari, bir yazılım uygulamasının bileşenlerini mantıksal olarak ayıran ve her bir bileşeni belirli bir sorumlulukla donatan bir tasarım yaklaşımıdır. Kullanım amacı, uygulamanın iş mantığını, veri erişim kısmını ve kullanıcı arayüzü tarafını birbirinden ayrı ve

modüler katmanlara ayırarak, kodun sürdürülebilirliğini, ölçeklenebilirliğini ve yeniden kullanılabilirliğini arttırmaktır.

Katmanlı mimari bu çalışmada 4 temel katmandan oluşmaktadır:

### **1.Core katmanı:**

Bu katmanda, projenin model kısmı tutulur. Çalışmada Code-first yaklaşımı kullanılmıştır. Bu yaklaşım sayesinde veri tabanına tabloları kaydetmeden önce Model tarafında sanal tablolar oluşturularak önce kodu yazıp daha sonra bu tablolar Repository katmanında veri tabanına aktarılıyor. Ayrıca Core katmanında, Dto'larda tutulur. DTO'lar ilgilendiğimiz bir tabloda sadece istediğimiz veriyi çekmemize olanak sağlar. Örneğin bir tabloda sadece isim kısmını almak istersek bir DTO ile bunu yapabiliriz.

### **2.Repository katmanı:**

Bu katman, çalışmada oluşturulan model kısmının veri tabanına aktarılması ve bundan sonra ki işlemlerde kod kısmının ve veri tabanının senkronize çalışması için kullanıldı. Migration işlemi, veri tabanında ki tablolar ile ilgili bir değişiklik yapıldığında bunu veri tabanına yansıtmamız için bize olanak sağladı. Aynı zamanda bu katmanda, çalışma için gerekli olan konfigürasyonlar da yapıldı. Bu konfigürasyonlara örnek olarak, girilen bazı verilerin 500 harf uzunluğunu geçmemesi gibi konfigürasyonlar bu katmanda yapıldı.

### **4.Service katmanı:**

Bu katmanda, çalışma için gerekli olan servisler yazıldı. Örnek olarak, "CategoryService", kategori işlemlerini yönetmek için kullanılan bir hizmet sınıfı yazıldı. Bu servis, ilgili bağımlılıkları ekleyerek, veri tabanı işlemlerini gerçekleştirmek, haritalama işlemleri yapmak ve ilgili DTO nesneleri döndürmek için kullanıldı. Bu katmanda, aynı zamanda gerekli olan mapping işlemleri de bu katmanda yapıldı. Bu katmanda, AutoMapper kütüphanesini kullanarak veri transfer nesneleri (DTO - Data Transfer Object) ile veri tabanı nesneleri (Entity) arasında dönüşümün yapıldığı bir haritalama profili tanımlamaktadır.

### **5.Api katmanı:**

Bu katman, kullanıcının göreceği arayüz ile back end tarafı arasında ki bağlantıyı sağlayacağımız katmandır. Bu katmanda, gerekli olan controllerlar yazıldı. Controller sınıfı, API isteklerini karşılar, iş mantığını yönetir ve uygun yanıtları döndürerek API 'nin doğru çalışmasını sağlar. Bu adımlar, çok katmanlı mimari yapısına uygun olarak, iş mantığına ait görevleri ayırtmak, modülerlik sağlamak ve kodun sürdürülebilirliğini arttırmak için kullanılır.

Proje içerisinde Python kodu da kullanılmıştır. Python, açık kaynaklı bir programlama dili olup, kolay okunabilir ve yazılabilir bir sözdizimine sahiptir. Bu projede, Python kodu önceden verilmiş cevapları kontrol eden bir sistem için kullanılmıştır. Python kodu, proje içerisinde ayrı bir modül olarak yer almaktadır.

#### 4. ARAYÜZ TASARIMI

Çalışmamızda, arayüz tasarımı gerçekten çok önemli bir konu durumunda. Tasarımda dikkat ettiğimiz noktalar, Kullanıcının kolay erişim sağlaması, kullanışlı ve anlaşılır olması. Çalışmamızın arayüz tasarımı için Angular framework'ü kullandık.

Angular, modern ve güçlü bir framework 'dür. Web uygulamaları geliştirmek için kullanılan Angular, özellikle büyük ve karışık projelerde etkili bir şekilde kullanılabilir. Angular, tek sayfalık uygulamaları (Single-Page Applications - SPA) geliştirmek için ideal bir framework 'tür.

##### 4.1. Anasayfa tasarımı

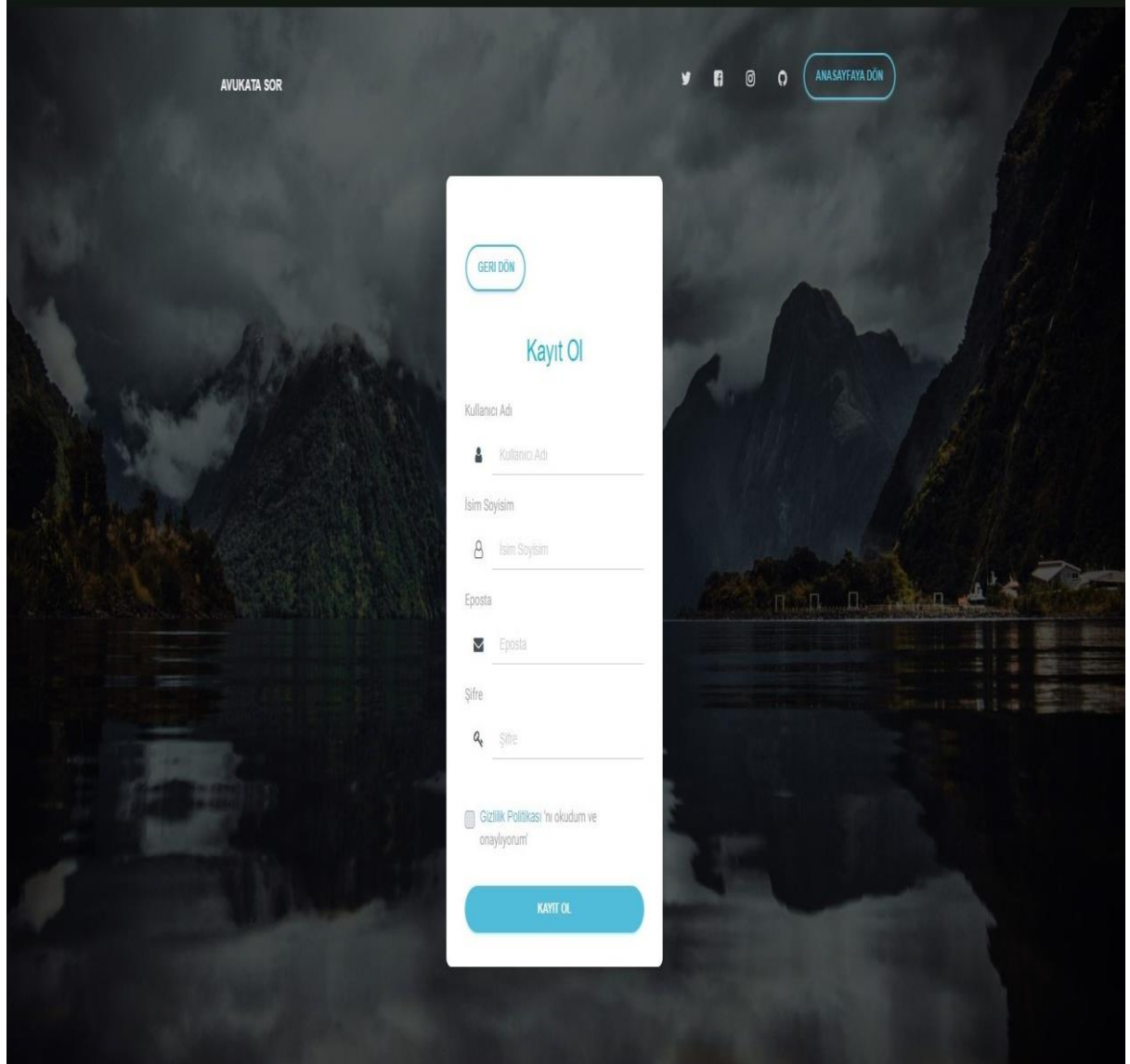
Çalışmamızda ki anasayfa tasarımında, hukuk alanı ile ilgili bir arka plan, giriş sayfasına yönlendirebilecek bir buton bulunmaktadır. Anasayfa kısmı, kullanıcının ilk göreceği sayfa olduğundan tasarımının basit, zarif ve anlaşılır olması çok önemlidir.



Şekil 4:Anasayfa tasarımı

## 4.2.Kayıt olma sayfası tasarımı

Çalışmamızda ki kayıt olma sayfası, uygulamaya yeni bir kullanıcı eklemek için kullanılıyor. Buradan kayıt olup giriş sayfasına yönlendirme yapılıyor. Kayıt olan kullanıcı veri tabanına müşteri olarak kaydediliyor.



AVUKATA SOR

ANASAYFA DÖN

GERİ DÖN

Kayıt Ol

Kullanıcı Adı

Kullanıcı Adı

İsim Soyisim

İsim Soyisim

Eposta

Eposta

Şifre

Şifre

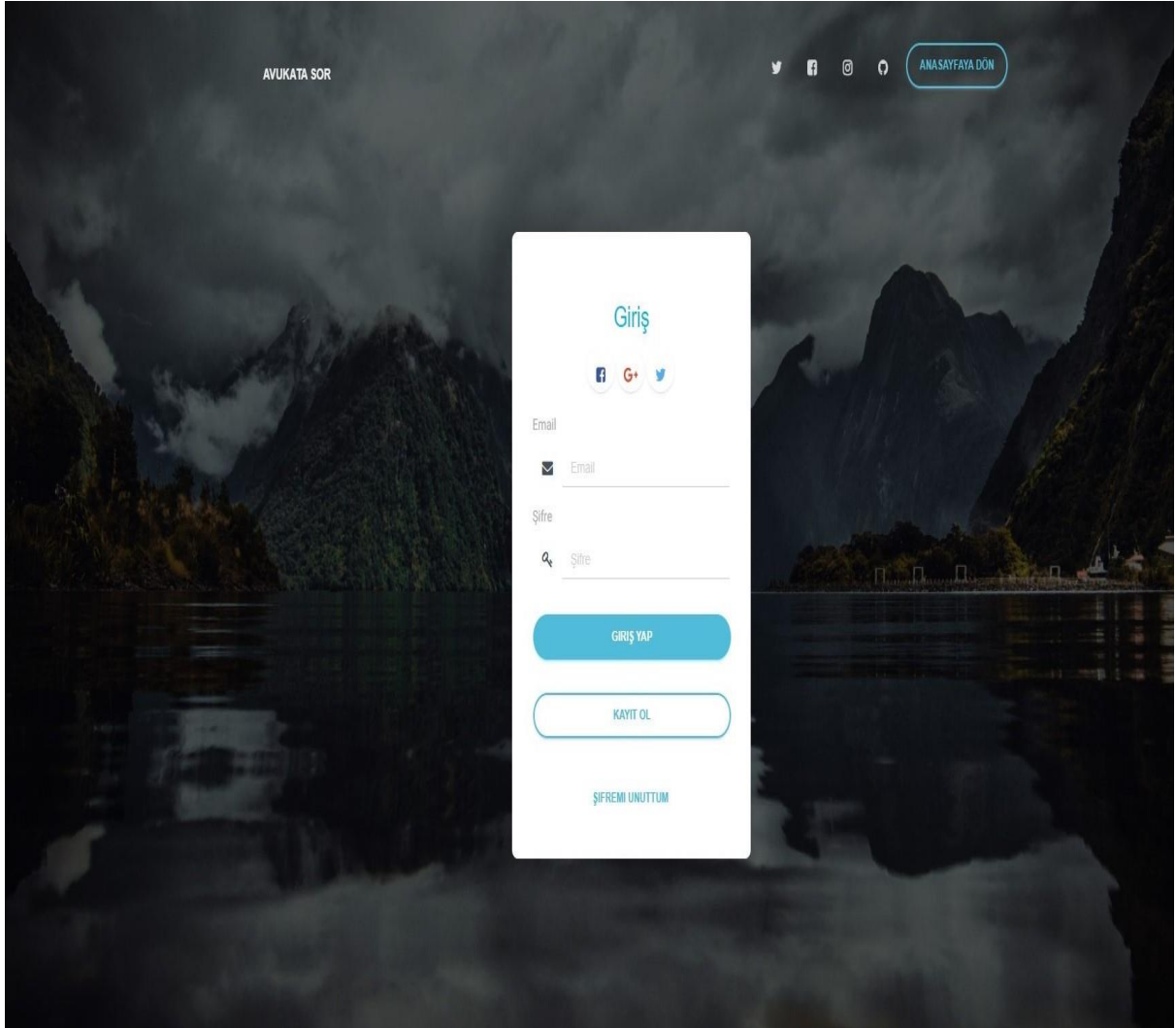
☐ Gizlilik Politikası' nı okudum ve onaylıyorum

KAYIT OL

Şekil 5:Kayıt olma sayfası tasarımı

### 4.3. Giriş yapma sayfası tasarımı

Çalışmamızda ki giriş yapma ekranı tasarımında, kayıt olan kullanıcının maili ve şifresini kullanıyoruz. Mail ve şifre veri tabanında ki müşteri ile eşleşir ise otomatik olarak müşteri anasayfasına yönlendirme yapılıyor. Ancak, şifre veya mail yanlış olursa giriş yapma sayfası tekrar yükleniyor ve girişe izin verilmiyor.



Şekil 6:Giriş yapma sayfası tasarımı

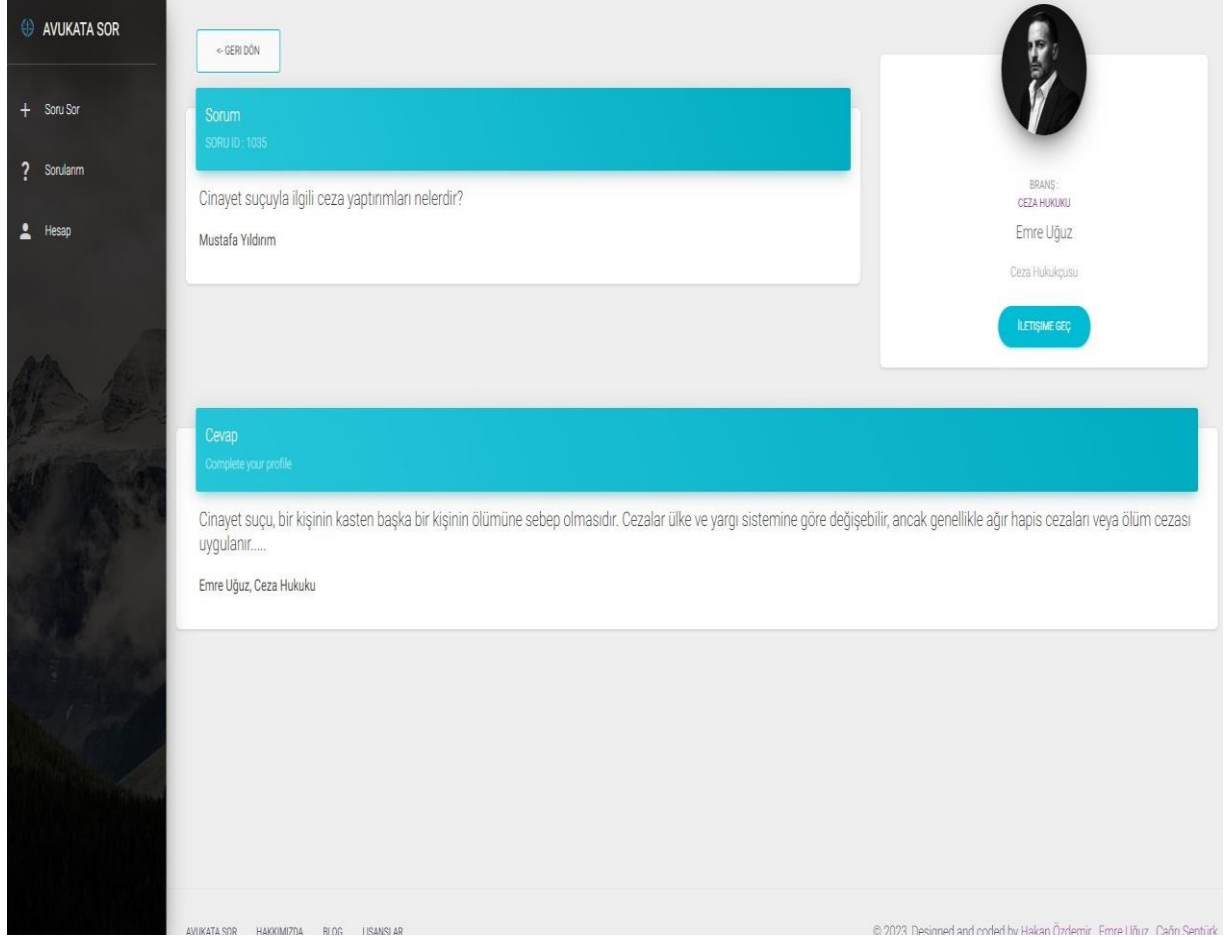
#### 4.4. Soru sorma ekranı tasarımı

Çalışmamızda soru sorma ekranı tasarımında öncelikle, kullanıcının üst tarafta sorunun hangi kategoriye ait olduğunu seçmesi gerekiyor. Bu işlemden sonra önüne o kategoriye ait avukatlar listeleniyor. Kullanıcı, seçilen avukata soruyu bu ekrandan gönderebiliyor. Aynı zamanda sol tarafta kullanıcının sorduğu soruları görebileceği bir sayfaya gidebileceği bir navbar bulunuyor.

Şekil 7:Soru sorma ekranı tasarımı

#### 4.5.Soru detay ekranı tasarımı

Çalışmamızda ki soru detay ekranı tasarımında, kullanıcı kendi sorduğu soruyu, soruyu sorduğu avukatı ve cevap kısmını görüyor.



Şekil 8:Soru detay ekranı tasarımı



#### 4.6. Avukat soru listesi sayfası tasarımı

Çalışmamızda ki avukat soru listesi sayfası tasarımında, giriş yapan avukata gelen tüm sorular listelenir. Bu listede soruların durumları da mevcuttur. Cevaplanmış sorular işaretlenir. Aynı zamanda şimdiye kadar kaç adet soruya cevap verdiği ve cevaplanmayan soruların sayıları da bu sayfada avukata gösterilir.

AVUKATA SORU

Soru Listesi

Diğer Avukatlar

Emre Uğuz

Search...

Cevaplanan Sorular 23

Cevaplanmayan Sorular 1

TÜM SORULAR CEVAPLANMAYAN SORULAR CEVAPLANAN SORULAR

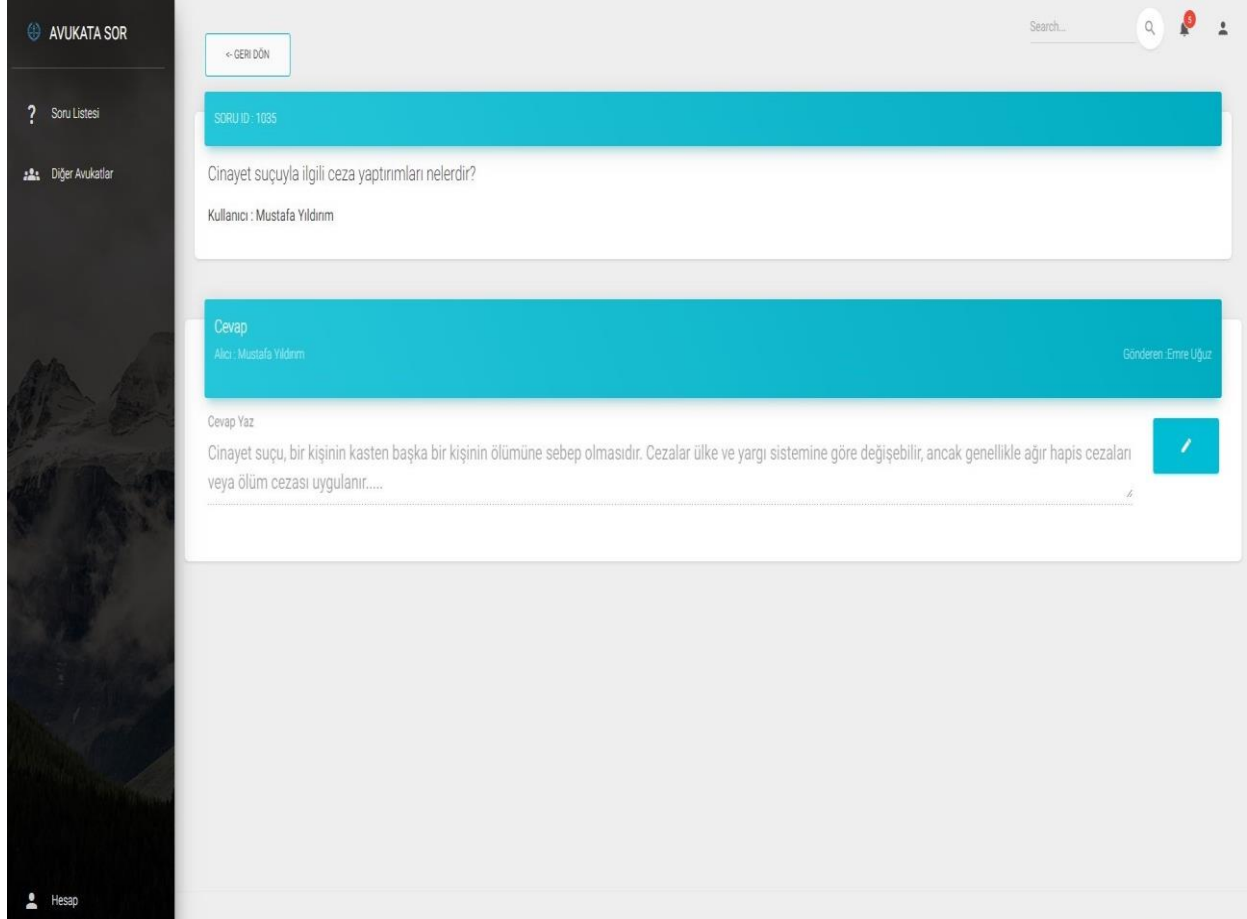
Sorulan Sorular

Soru Id	Soru	Durum
1081	Hırsızlık suçuyla ilgili cezai sorumluluk nedir?	Cevaplandı
1082	Cinayet suçuyla ilgili ceza yaptırımları nelerdir?	Cevaplandı
1083	Tecavüz suçuyla ilgili yasal süreç nasıl işler?	Cevaplandı
1084	Kasten yaralama suçuyla ilgili ceza hukuku nedir?	Cevaplandı
1085	Trafik kazası sonucu ölüm durumunda ne gibi yaptırımlar uygulanır?	Cevaplandı
1086	Ceza hukuku nedir ve amacı nedir?	Cevaplandı
1087	Cinayet ile cinayet teşebbüsü arasındaki fark nedir?	Cevaplandı

Şekil 9:Avukat soru listesi sayfası tasarımı

#### 4.7. Avukat cevap sayfası tasarımı

Çalışmamızda Avukat cevap sayfası tasarımında, seçilen soru tüm detayı ile görüntülenir. İlgili avukat bu sayfadan kullanıcıya cevabı gönderir.



Şekil 10:Avukat cevap sayfası tasarımı

### 5. BULGULAR VE TARTIŞMA

Çalışmamızda, veri seti olarak ....’dan 100 adet soru kullanıldı. Bu sorular veri tabanında Question tablosuna eklendi. Aynı zamanda cevapları da Answers tablosuna eklendi. Çalışmamızın ne kadar oranda başarılı çalıştığını görmek için bir test çalışması yaptık. Bu testte, farklı sosyolojik gruplardan tam .. kişi ile uygulamamızı denedik. Test için kullanılan her bir kişiden 3 soru sormasını istedik. Bu kişilerin bilgileri aşağıda ki tabloda gösteriliyor.

Soru soran kullanıcıların yaşları	Yaş grubuna ait toplam kişi sayısı	Yaş grubuna ait toplam soru sayısı	Soruların konuları
10-20	2	8	4 adet iş hukuku 4 adet medeni hukuk
20-30	10	40	6 adet vergi hukuku 20 adet medeni hukuk 12 adet ceza hukuku 2 adet iş hukuku
30-40	7	28	4 adet vergi hukuku 22 adet medeni hukuk 1 adet ceza hukuku 1 adet iş hukuku
40+	1	4	1 adet medeni hukuk 2 adet iş hukuku 1 adet ceza hukuku

Tablo 1: Test için kullanılan kişilerin bilgileri

Çalışmamızda kullandığımız arayüz, anlaşılır ve sade bir arayüz olduğu için soru soran 20 kişi de soru sorma konusunda bir sorun yaşamadı. Soru sorması için seçtiğimiz kişilerin, sosyolojik olarak birbirlerinden farklı gruplar içinde yer almalarına dikkat ettik. Seçtiğimiz kişiler arasında, öğrenciler, emekliler, çalışanlar gibi farklı gruplara dahil olan kişiler var. Bu sayede çok daha doğru sonuçlar elde edilebileceğine karar verdik.

Çalışmamızda, kullanıcıların soru sorma işlemleri oldukça kısa sürdü. Normal şartlarda bir avukata ulaşip soru sormak, insanlar için saatler sürebiliyor. Ancak uygulamamıza kullanan kullanıcılar, dakikalar içinde seçtikleri avukata sorularını iletmiş oldular.

## Tartışma

Çalışmamızda, kullanıcılardan soru almak çok önemli bir yer tutuyor. Kullanıcılardan aldığımız sorular uygulamanın ne kadar başarılı olduğu konusunda bize ipuçları sunuyor. Yapılan anket sonucunda, sorulan 80 sorunun yüzde 28.75'ine önceden cevap verilen sorunun cevabı otomatik olarak verilmiş.

Bu duruma baktığımızda, kullanıcıların verilen cevaplardan ne kadar tatmin olduğu da çok önemlidir. Sorulan soruya verilen cevabın düzgün ve ihtiyacı giderecek olması gerekiyor. Otomatik gelen cevaplarda, gelen cevaba göre memnuniyet oranı ise yüzde 60. Bu oranın

bize gösterdiği ise, sorulan soruların bir kısmı avukata iletilmeden yeterli şekilde cevaplanmış olduğudur. Bu oranı arttırmak için tutarlılığı arttırmak gerekiyor. Verilen cevapların tutarlılığı arttıkça uygulamanın da verimi o kadar artacaktır.

Sorulan sorular incelendiğinde ise, bu soruların basit düzeyde ve genellikle cevapları çok karışık olmayan sorular olduğu görülmüştür. Bu açıdan bakıldığında, soruların karmaşıklığı arttıkça uygulamanın verimi azalır. Spesifik sorulara otomatik cevap göndermek, bu sisteme göre oldukça zordur.

Çalışmanın en önemli avantajlarından biri, avukatların iş yükünün azaltılmasıdır. Geleneksel olarak, avukatlar müşteri sorularını bireysel olarak yanıtlamak zorunda kalırken, uygulamamız sayesinde birçok tekrarlayan soru otomatik olarak cevaplanabilir. Bu, avukatların daha karmaşık ve spesifik konulara odaklanmalarına olanak sağlar. İş yükünün azalmasıyla birlikte avukatlar, daha fazla vakit ve enerjiyi müşteri danışmanlığı, hukuki analizler ve davanın yönetimi gibi kritik görevlere ayırabilir. Bu da avukatların verimliliğini artırır ve hizmet kalitesini yükseltir.

Ayrıca, uygulamamız sayesinde yanıtlar daha hızlı bir şekilde sunulabilir. Kullanıcılar, beklemek zorunda kalmadan sorularına anında cevap alabilirler. Bu da müşteri memnuniyetini artırır ve müşteri deneyimini olumlu yönde etkiler. Hızlı yanıtların sunulması, müşteri güvenini ve sadakatini de artırabilir.

Çalışma, aynı zamanda tutarlılığı sağlama konusunda da avantaj sağlar. İnsan avukatlar arasında cevapların tutarlılığı her zaman garanti edilemezken, otomatik sistemler her soruya aynı şekilde ve tutarlı bir şekilde yanıt verir. Bu da kullanıcıların güvenini artırır ve hukuki bilgiye erişimde eşitlik sağlar.

Bununla birlikte, çalışma ile ilgili olarak belirtmek gerekir ki, bu sistem sadece destekleyici bir rol oynamalıdır. Karmaşık ve spesifik sorular genellikle insan avukatların uzmanlığını gerektirir. Çalışma, avukatları desteklerken, insan avukatların kişisel yorumlarını, analizlerini ve deneyimlerine hala ihtiyaç duymaktadır. Bu şekilde, insan ve yapay zeka arasındaki işbirliği en iyi şekilde sağlanabilir.

Çalışmada yapılan uygulama, verimlilik artışı ve iş yükünün azaltılması gibi olumlu etkileriyle önemli bir araçtır. Avukatların daha karmaşık sorunlara odaklanmalarını sağlar, hızlı ve tutarlı cevaplar sunar, müşteri memnuniyetini artırır ve hukuki bilgiye erişimi kolaylaştırır.

## **6.SONUÇ**

Bu tez çalışması, .NET Core 6 API kullanarak geliştirilen bir uygulamanın, otomatik cevaplama mekanizmasıyla birlikte nasıl çalıştığını ve kullanıcılara nasıl yardımcı olduğunu incelemiştir. Yapılan analizler ve anket sonuçları, otomatik cevaplama sisteminin belirli soruların yüzde 55'ine doğru ve tutarlı cevaplar sağlayabildiğini göstermiştir. Bu durum, avukatların iş yükünü azaltarak verimliliği artırmak adına önemli bir adımdır.

Sonuçlar, uygulamadan gelen otomatik cevapların, kullanıcılar tarafından belli oranda olumlu karşılandığını ve memnuniyet oranının yüzde 60 olduğunu göstermektedir. Ancak, sistemin karmaşık ve spesifik sorulara tutarlı cevaplar sunma konusunda sorunları olduğu tespit edilmiştir. Bu nedenle, insan avukatların uzmanlığı tecrübesi hala önemli bir değer taşımaktadır.

Bu çalışma, yapay zeka kullanarak geliştirilen bir uygulamanın, hukuki danışmanlık alanında iş yükünün azaltılması ve verimliliğin artırılması gibi olumlu etkilere sahip olduğunu ortaya koymaktadır. Ancak, insan ve yapay zeka işbirliğinin dengeli bir şekilde yönetilmesi gerekmektedir. İnsan avukatların tecrübesi, karmaşık sorunların çözümünde hala vazgeçilmezdir.

Gelecekteki çalışmalar, hukuk alanında yapay zeka kullanımının daha da geliştirilmesi, doğruluğunun ve tutarlılığının artırılması, kullanıcı memnuniyetinin yükseltilmesi ve sistemin daha karmaşık soruları yanıtlayabilme yeteneğinin geliştirilmesi üzerinde odaklanabilir. Ayrıca, güvenlik ve gizlilik konularına daha fazla önem verilerek, kullanıcıların bilgilerinin korunması sağlanmalıdır.

Bu tez çalışması, yapay zekanın hukuk alanında kullanılmasının, önemli bir potansiyele sahip olduğunu ve gelecekteki çalışmalarla daha da geliştirilebileceğini göstermektedir. İnsan ve yapay zeka işbirliği, daha etkili ve verimli hukuki danışmanlık hizmetlerinin sağlanmasına katkı sağlayabilir.

## Kaynakça:

- [1] Lewis, T. (2021). A Brief History of Artificial Intelligence. <https://www.livescience.com/49007-history-of-artificial-intelligence.html>
- [2] İŞERİ, İsmail; AYDIN, Özkan; TUTUK, Kaan. Müşteri Hizmetleri Yönetiminde Yapay Zeka Temelli Chatbot Geliştirilmesi. *Avrupa Bilim ve Teknoloji Dergisi*, 2021, 29: 358-365.
- [3] AMASYALI, M. Fatih; BANU, D. İ. R. İ. Bir soru cevaplama sistemi: Baybilmiş. *Türkiye Bilişim Vakfı Bilgisayar Bilimleri ve Mühendisliği Dergisi*, 2005, 1.1.
- [4] İşler S., Amasyalı M. F., Tatlı E., (2000), “Bir Türkçe Metindeki Kelimelerin Cümle İçindeki Durumlarına Bakılarak Eklerine Ayrılması ”, Bilişim Zirvesi'01, İstanbul
- [5] IŞIK, Ali Hakan; YAĞCI, Ayşenur. Sıradan Sıraya LSTM Modeli ile Telegram Bot Uygulaması.
- [6] KILIÇ, Berker; YÜKSEL, Ö. N. E. R. Yargıtay Kararlarının Suç Türlerine Göre Makine Öğrenmesi Yöntemleri İle Sınıflandırılması. *Veri Bilimi*, 4.3: 61-71.
- [7] Liu, X., Liu, Y., & Huang, X. (2009). A keyword extraction method based on the frequency and distribution of words. In Proceedings of the Third International Conference on Natural Computation (Vol. 5, pp. 611-615).
- [8] Turney, P. D. (1992). Automatic keyword classification for information retrieval. *Journal of the Association for Information Science and Technology*, 43(5), 339-353.
- [9] Uzun, E. (2005). Turkish text categorization using n-gram-based probabilistic language models. In Proceedings of the International Symposium on Computer and Information Sciences (pp. 491-496).
- [10] Joachims, T. (1998). Text categorization with support vector machines: Learning with many relevant features. In European conference on machine learning (pp. 137-142).
- [11] Ikonomakis, M., Anagnostopoulos, I., & Vakali, A. (2005). Neural networks and SVM in document categorization: A comparative study. In International Conference on Artificial Neural Networks (pp. 199-204).
- [12] Aseervatham, L. (2011). Sentiment classification using supervised machine learning techniques. *International Journal of Computer Science Issues (IJCSI)*, 8(2), 488-495.
- [13] CETİNER, Meltem; AKGÜL, Yusuf Sinan. Emsal Hukuk Dokümanlarının Otomatik Belirlenmesi. *Düzce Üniversitesi Bilim ve Teknoloji Dergisi*, 2021, 9.6: 83-94.

- [14] HÖFLER, Stefan. Assessing the Quality of Legislation by means of an Automatically Annotated Corpus.
- [15] Ortadoğu Teknik Üniversitesi, “Kodex Bilişim’ den Yargıtay Davalarına Yapay Zeka Destek”,<https://odtuteknokent.com.tr/tr/haber/kodexbilisimden-yargitay-davalarina-yapay-zekalidestek>
- [16] Lane, G. (2005). Using NLP techniques to identify legal ontology components: concepts and relations. In Law and the Semantic Web (pp. 169-184). Springer, Berlin, Heidelberg.
- [17] Breuker, J., Valente, A., ve Winkels, R. (2004). Legal ontologies in knowledge engineering and information management. Artificial intelligence and law, 12(4), 241-277.