



**Teknoloji Fakültesi**

**MARMARA ÜNİVERSİTESİ**

**TEKNOLOJİ FAKÜLTESİ**

**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

**BİTİRME PROJESİ**

**E-ticaret Sektöründe**

**Dinamik Fiyatlandırma ve Ürün Takibi**

**Öğrencinin Adı Soyadı**

Oğuz Kaan Arda

**DANIŞMAN**

Doç. Dr. Buket DOĞAN

**İSTANBUL, 2024**



**Teknoloji Fakültesi**

**MARMARA ÜNİVERSİTESİ**

**TEKNOLOJİ FAKÜLTESİ**

**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

**BİTİRME PROJESİ**

**E-ticaret Sektöründe**

**Dinamik Fiyatlandırma ve Ürün Takibi**

**Öğrencinin Adı Soyadı**

Oğuz Kaan Arda

**DANIŞMAN**

Doç. Dr. Buket DOĞAN

**İSTANBUL, 2024**

**İÇİNDEKİLER**

<b>İÇİNDEKİLER</b>	<b>1</b>
<b>ÖZET</b>	<b>3</b>
<b>ABSTRACT</b>	<b>4</b>
<b>KISALTMALAR</b>	<b>5</b>
<b>ŞEKİL LİSTESİ</b>	<b>6</b>
<b>TABLO LİSTESİ</b>	<b>7</b>
<b>1. GİRİŞ</b>	<b>8</b>
1.1. İlgili Çalışmalar	10
1.2. Amaç	15
<b>2. MATERYAL VE YÖNTEM</b>	<b>15</b>
2.1. Uygulama Kapsamının Belirlenmesi	15
2.2. Literatür Taraması Yapılması	16
2.2.1. Dil ve Kütüphane Seçimi	17
2.2.2. Kazıma Algoritmasının Hazırlanması	17
2.3. Veriseti Hazırlanması	18
2.3.1. Türkiye Pazaryeri Durum İstatistikleri	18
2.3.2. Veri toplama	20
2.4. Kullanılacak Teknolojilerin Belirlenmesi	23
2.4.1. Veritabanı Tasarımı ve Gerçekleştirimi	23
2.4.2. Arkayüz Proje Gerçekleştirimi	24
2.4.3. Django Framework	26
2.4.4. Makine Öğrenmesi	29
2.4.5. Verilerin Hazırlanması	31
2.4.6. Önyüz Proje Tasarımı, Ekran Tasarımı ve Gerçekleştirimi	33
<b>3. SONUÇLAR</b>	<b>36</b>
<b>4. KAYNAKLAR</b>	<b>38</b>

## ÖZET

### E-TİCARET SEKTÖRÜNDE DİNAMİK FİYATLANDIRMA VE ÜRÜN TAKİBİ

E-ticaret sektörü son 10 yılda hızlı bir büyüme trendi sergilemiştir. Dünya genelinde perakende e-ticaret hacmi 2010'dan bu yana yaklaşık 7 kat artış göstermiş olup bu büyümede, internetin yaygınlaşması ve hükümetlerin e-ticareti desteklemesi önemli rol oynamıştır.

Türkiye'de de e-ticaret sektörü son yıllarda hızlı bir gelişim göstermiştir. 2019 yılında 13,9 milyar dolar olan sektörün hacmi, 2020'de 226 milyar liraya, 2021'de ise 381,5 milyar liraya ulaşmış, 2022 yılında ise bir önceki yıla göre %109 artarak 800 milyar TL'ye yükselmiştir. 2019-2022 yılları arasında Türkiye'nin e-ticaret hacmi yıllık ortalama %80,6 oranında büyümüştür.

E-ticaret sektöründe yaşanan hızlı büyüme ve yüksek enflasyonlu ortam, ürün fiyatlarının ve stok durumlarının sıklıkla değiştiği dinamik bir pazar oluşturmuştur. Bu çalışma derin öğrenme ve veri kazıma destekli bir ürün yönetim platformu geliştirerek e-ticaret sektöründeki dinamik fiyatlandırma ve ürün takibinin zorluklarına çözüm bulmaktadır. Oluşturulacak platformun veri kazıma bileşeni sayesinde toplanan verileri kullanarak fiyat değişikliklerini ve pazar büyüklüğünü takip edebilmekte ve bu sayede daha bilinçli satın alma kararları alınmasına olanak tanımaktadır.

Platformun dinamik fiyatlandırma bileşeni, veri kazıma ile toplanan verileri kullanarak derin öğrenme algoritmaları ile ürün fiyatları hakkında tahminde bulunmaktadır. Bu yaklaşım, platformun değişen pazar koşullarına uyum sağlamasını ve gerçek zamanlı olarak doğru fiyat tahminleri sunmasını sağlar. Platformun ürün takibi bileşeni ise toplanan verileri kullanarak ürün kategori ve markalarını izlemekte ve fiyat değişikliklerini takip ederek daha bilgilendirilmiş satın alma kararları alınmasına olanak tanımaktadır.

**Anahtar Kelimeler:** E-ticaret, Dinamik fiyatlandırma, Makine öğrenmesi, Fiyat tahmini

## **ABSTRACT**

### **DYNAMIC PRICING AND PRODUCT TRACKING IN THE E-COMMERCE SECTOR**

The e-commerce sector has exhibited a rapid growth trend over the past 10 years. Worldwide, the retail e-commerce volume has increased by approximately 7 times since 2010, with the widespread use of the internet and government support for e-commerce playing a significant role in this growth.

In Turkey, the e-commerce sector has also shown rapid development in recent years. With a volume of \$13.9 billion in 2019, the sector's volume increased to \$226 billion in 2020 and \$381.5 billion in 2021, and to \$800 billion in 2022, representing a 109% increase compared to the previous year. The e-commerce volume in Turkey has grown by an average of 80.6% annually between 2019 and 2022.

The rapid growth and high inflation in the e-commerce sector have created a dynamic market where product prices and stock levels change frequently. This study aims to address the challenges of dynamic pricing and product tracking in the e-commerce sector by developing a product management platform supported by deep learning and data scraping. The platform's data scraping component enables the tracking of price changes and market size using the collected data, allowing for more informed purchasing decisions.

The platform's dynamic pricing component uses deep learning algorithms to make predictions about product prices based on the collected data. This approach enables the platform to adapt to changing market conditions and provide real-time accurate price estimates. The platform's product tracking component monitors product categories and brands using the collected data and tracks price changes to enable more informed purchasing decisions.

**Keywords:** E-commerce, Dynamic pricing, Machine learning, Price prediction

## **KISALTMALAR**

<b>B2C</b>	:Business to Consumer
<b>SEO</b>	:Search Engine Optimization
<b>İVTYS</b>	:İlişkisel veri tabanı yönetim sistemi
<b>SQL</b>	: Structured Query Language
<b>IDE</b>	: Integrated Development Environment
<b>API</b>	:Application Programming Interface

## ŞEKİL LİSTESİ

Şekil 2.1 Uygulama Adımları	15
Şekil 2.2 Ürün Kategorilerinin Aldığı Pay	19
Şekil 2.3 Kategori kodları	20
Şekil 2.4 Örnek soup yapısı	21
Şekil 2.5 Google Chrome kaynağı inceleme	21
Şekil 2.6 Örnek bir “beautifulsoup” sorgusu	22
Şekil 2.7 Django Yapısı	25
Şekil 2.8 Örnek Url yönlendirmesi	26
Şekil 2.9 Örnek model yapısı	27
Şekil 2.10 Örnek view yapısı	27
Şekil 2.11 Django DTL Örneği	28
Şekil 2.12 Kategori Temizliği	32
Şekil 2.13 Sütun Temizliği	32
Şekil 2.12 Ürün kazıma ekranı	34
Şekil 2.13 Envanter Ekleme Ekranı	34
Şekil 2.14 Kategori filtreleme	35
Şekil 2.15 Marka filtreleme	35
Şekil 2.16 Ürün Detay	36

## TABLO LİSTESİ

Tablo 2.1 Veri Kazıma Algoritma Tablosu	17
Tablo 2.2 Yaşa Göre Tüketicilerin En Sık Tercih Ettiği Pazaryeri	18
Tablo 2.3 Vazgeçilmez olarak gören kullanıcılar	19
Tablo 2.4 Örnek Veriler	22
Tablo 2.5 Örnek Veriler Temizliği	33
Tablo 3.1 Örnek Veriler Temizliği	37



## 1. GİRİŞ

E-ticaret, internet ticaretinden kısaltılan bir terim olup, çevrimiçi olarak mal ve hizmetlerin alım satımını ve para ile veri işlemlerini internet aracılığıyla gerçekleştirir. Mobil internet çağının avantajıyla, giderek daha fazla insan artık akıllı telefonlara sahip ve internete erişim her zamankinden daha kolay hale gelmiştir. Bu nedenle, E-ticaret müşterileri uluslararası pazarda her yerden mal ve hizmet satın alabilmektedirler. Ayrıca, geleneksel çevrimdışı işletmelere kıyasla daha düşük yatırım veya iş sermayesi gerektirdiği için E-ticaret maliyet açısından etkili bir yöntemdir.

2010 ve 2020 yılları arasında E-ticaret insan hayatını yavaş yavaş değiştirdi ve çevrimiçi alışveriş giderek daha popüler hale geldi[15]. 2015'ten 2018'e kadar E-ticaret, geleneksel çevrimdışı işletmelerin 13 katı daha fazla pazar payına sahip oldu[16]. 2018 itibari ile E-ticaret satışları tüm dünya perakende satışlarının %11.9'unu oluşturmakta ve 2021'de bu oranın %17.5'e ulaşması bekleniyor[17]. Araştırmacılar, B2C web sitesinin bilgi içeriği, tasarımı ve güvenliği dahil olmak üzere dört temel faktörün, tüketicilerin kararlarını etkileyebileceğini bulmuşlardır. Buna karşılık, güvenlik ve gizliliğin tüketicilerin satın alma niyetleri üzerinde daha büyük bir etkisi olduğunu da eklemişlerdir.[18]

Makine öğrenmesi, E-ticaret alanında giderek daha fazla kullanılmaya başlanmıştır.(**kaynak ve revize**)Evrışimli sinir ağları, ürünlerin geçmiş satış verilerindeki içsel bağlantıları otomatik olarak çıkartıp bu özellikleri kullanarak gelecek haftanın toplam ürün satışlarını tahmin etmek için kullanılmıştır.[19] Ayrıca, [20] L1 düzenlemesini dikkate alan bir derin öğrenme modeli (ANN), %86'lık bir satış tahmini doğruluk oranına ulaşmıştır. Makine öğrenmesi algoritmaları, E-ticaret alanında ürün talebini tahmin etmek, müşteri davranışlarını analiz etmek ve pazarlama stratejilerini optimize etmek için kullanılmaktadır.

Bu çalışmada e-ticaret ile uğraşan kişilerin stok ve fiyatlandırma konusunda işlerini kolaylaştırabilecek bir yazılım geliştirilmiştir. Yazılım geliştirme aşamaları aşağıda belirtildiği gibidir:

- İlk olarak pazaryerlerini(Trendyol, Hepsiburada vb.) inceleyerek ürün gamı ve pazar payı yüksek olan bir pazaryeri araştırması yapılmıştır.

- İkinci adım olarak Trendyol pazaryerini seçtim ve websitesi üzerinde veri kazıma işlemi yapabilmek için bazı araştırmalarda bulunulmuştur.
- Ardından veri kazıma ile ilgili yapılmış çalışmaları inceleyerek bir literatür taraması gerçekleştirdim. V. Srividhya ve P.Megala, Shilpa Chaudhari ve Amalia'ın yaptıkları çalışmaları inceledikten sonra beautifulsoup kütüphanesi kullanarak ilerlemeye karar verilmiştir. Trendyol'un önyüzünü detaylı bir şekilde ineleterek ürün adı, fiyat, tarih gibi kullanılabilecek verilerin nereden alınabileceğine dair bir araştırma yapıldı. Bu araştırma sonucunda Trendyol'da satılan ürünlerin ürün detay sayfasından çekilmesine karar verildi.
- Genellikle seo performansı artırmak için kullanılan ürün şemasının içinde bulunan verileri listeledik. Ve veri kazıma algoritmasını Python üzerinden yazmaya karar verilmiştir.
- Veriseti oluşturma amaçlı 3 ana kategori(Vitamin, Mineral ve Kolajen) belirlendi ve bu kategorilerde sayfaları gezerek ürünlerin detay sayfasına giriş yapan bir algoritma hazırlandı.
- Diğer teknolojiler olarak veritabanı için MySQL, web arayüzü için Django veri kazıma için beautifulsoup kütüphanesi, makine öğrenmesi için Cat Boost kütüphanesi kullanılmıştır. IDE olarak Visual Studio Code tercih edilmiştir.
- Periyodik olarak veri kazıma işlemi gerçekleştirilip bir veri seti oluşturmuş ve makine öğrenmesi algoritmasında kullanılmak üzere düzenlenmiştir.
- Ardından ürün yönetim, veri kazıma ve fiyat tahmini fonksiyonlarının gerçekleştirebildiği ön yüz çalışmaları tamamlanmıştır.
- Veri seti için yeterli veri toplandıktan sonra makine öğrenme algoritması eğitilerek ön yüz'e entegre edilmiştir.

## 1.1. İlgili Çalışmalar

E-ticarete fiyat hesaplama yöntemlerinin son on yılda analizi, çeşitli teknolojik ve metodolojik gelişmelerden etkilenen dinamik bir alan oluşturmaktadır. Önemli araştırmalar, e-ticaret operasyonlarını optimize etmek için tahmine dayalı modeller ve fiyatlandırma stratejileri, makine öğreniminden, veri madenciliği ve dinamik fiyatlandırma algoritmalarından yararlanan bir çok çalışma bulunmaktadır.

Chunli Yin ve Jinglong Han tarafından yapılan araştırmada, çeşitli pazar koşullarına etkili bir şekilde uyum sağlamak için derin öğrenmeyi kullanan bir dinamik fiyatlandırma sistemi geliştirmektedirler. Çalışma, kârlılığı ve müşteri memnuniyetini artırmayı amaçlayan, değişen pazar ve tüketici katılım senaryoları altında en uygun fiyatlandırma stratejilerini değerlendiren iki dönemli bir oyun modeli oluşturmaktadır.

Chunli Yin ve Jinglong Han hem olgun hem de gelişmekte olan pazarlardaki fiyatlandırma stratejilerini araştırıyor ve farklı stratejik kombinasyonlar altında optimal fiyatlandırmayı ve kârı belirlemeye ve piyasa dengelerini karşılaştırarak Nash dengelerini çözmeye odaklanarak tüketici deneyim düzeyinin fiyatlandırma stratejilerini nasıl etkilediğine dair içgörü sağlamaktadır.

Vurgulanan başlıca sonuçlardan biri, modelin fiyatları gerçek zamanlı ve dinamik olarak ayarlama yeteneğidir; bu, ürün fiyatlarının ve stok mevcudiyetinin hızlı bir şekilde dalgalanabildiği e-ticarete kritik öneme sahiptir. Model, çeşitli kaynaklardan toplanan verileri kullanarak bunları pazar taleplerine ve stok değişikliklerine anında yanıt vermek için fiyatlandırma stratejisine dahil ediyor.

Ayrıca çalışmada veri madenciliği tekniklerinin dinamik fiyatlandırma modeline entegrasyonu tartışılmaktadır. Bilgiye dayalı fiyatlandırma kararları vermek için verilerin etkili bir şekilde işlenmesine ve kullanılmasına yardımcı olan, veri toplama, analiz ve karar verme katmanlarından oluşan üç katmanlı bir yapının ana hatlarını çizer. Veri madenciliğinin kullanılması, dinamik fiyatlandırma stratejilerini etkileyen faydalı kalıpların ve eğilimlerin ortaya çıkarılmasına yardımcı olur.

Modelin uygulanmasından elde edilen sonuçlar, derin takviyeli öğrenme yaklaşımının fiyatlandırma stratejilerini önemli ölçüde optimize ederek onları pazar değişikliklerine ve tüketici davranışlarına daha uyumlu hale getirebileceğini göstermektedir. Bu yaklaşım, e-ticaret platformlarının kârlarını en üst düzeye çıkarmasına yardımcı olmakla kalmıyor, aynı

zamanda gerçek zamanlı piyasa koşullarına göre rekabetçi ve adil fiyatlar sunarak müşteri memnuniyetini de artırıyor.

Genel olarak, Yin ve Han tarafından yapılan çalışma, e-ticaret dinamik fiyatlandırmasında derin takviyeli öğrenmenin uygulanması için bir temel sunmakta ve pazaryeri seçimi ve derin öğrenme teknolojisini kullanma yolunda çalışmamıza yardımcı olmuştur.[22]

Ürün fiyat optimizasyonu, şirketlerin karlarını maksimize etmek için müşteri davranışlarını ve pazar dinamiklerini analiz ederek en uygun fiyatı belirleme sürecidir. Fiyatlandırma, şirketlerin karlılığını, pazar payını ve müşteri memnuniyetini doğrudan etkileyen kritik bir parametredir. Optimum fiyat belirlemek, ürünün ne çok pahalı ne de çok ucuz olmasını sağlayarak hem tüketicilerin hem de satıcıların memnuniyetini dengelemeyi amaçlar. Mandava Jaswanth ve arkadaşlarının yaptığı çalışmada, fiyat optimizasyonu sürecinde en küçük kareler regresyon yönteminin uygulanması ele alınmıştır. Dinamik, statik, rekabetçi ve rekabetçi olmayan stratejiler gibi çeşitli fiyatlandırma modellerini karşılaştırarak bunların uygulamalarını ve bir şirketin geliri üzerindeki potansiyel etkileri üzerine çalışılmış. Dinamik fiyatlandırma modelleri, talebin değişken olabileceği e-ticaret ve perakende gibi sektörler için ideal olarak gösterilmiş ve fiyatları talep ve arz dalgalanmalarına dayalı olarak gerçek zamanlı olarak ayarlama yetenekleri nedeniyle çalışma içerisinde sıkça vurgulanmaktadır. Çalışmanın önemli bir kısmı En Küçük Kareler Yöntemine ayrılmış olup, gözlemlenen değerler ile model tahminleri arasındaki kareleri alınmış farkları en aza indiren bir regresyon modeli uygulanmıştır. Bu teknik, geçmiş satış verilerini ve pazar eğilimlerini analiz ederek, kar marjlarını en üst düzeye çıkarmak için gelecekteki fiyatlandırma stratejilerini tahmin ederek en uygun fiyat noktalarını tahmin etme konusunda başarılı bir performans sergilemiştir. Çalışma aynı zamanda farklı fiyat noktalarının tüketici satın alma kararlarını ve talep seviyelerini nasıl etkileyebileceğini tahmin etmek için piyasa simülatörlerinin kullanımını da araştırmakta.

Fiyat optimizasyonundaki zorluklara da değinilmiş; Çalışmada veri kalitesi, piyasa oynaklığı ve tüketici psikolojisinin entegrasyonu gibi fiyatlandırma modellerinin doğruluğunu ve etkinliğini önemli ölçüde etkileyebilecek bazı maddelerden bahsedilmiş. Araştırma, daha karmaşık makine öğrenimi modellerini ve gerçek zamanlı veri analizi tekniklerini birleştirmeye yönelik gelecekteki çalışmalar için önerilerle sonuçlanıyor. Bu, fiyatlandırma modellerinin öngörü doğruluğunu ve yanıt verebilirliğini artıracak ve piyasa koşullarına ve

tüketici tercihlerine dinamik olarak uyum sağlayan daha uyarlanabilir ve akıllı fiyatlandırma sistemlerinin kullanmasını önermektedir.[23]

Günümüzde farklı pazaryerleri aynı ürünü farklı fiyatlara satabilmektedir. Bu direkt olarak firmalar için rekabeti arttıran bir faktör olup ürün araştırma aşamasının birden fazla pazaryeri içermesi önemlidir. Bu nedenle araştırma süreçlerinin elle yapılması insan ve vakit kaybı yaratmaktadır. Bu süreçleri otomatikleştirmek için veri kazıma tekniklerini bir çözüm olarak düşündük. Veri kazıma, web sitelerinden toplanan HTML gibi işaret dillerinin analiz edilip ilgili verinin işlenerek kullanılmasına verilen addır.

David Mathew Thomas ve arkadaşlarının yaptığı çalışmada veri kazıma ve veri analiz etme süreçlerinden bahsedilmiştir. Veri kazıma, internetten veri çekme ve bu verileri analiz etme sürecine verilen addır. Python programlama dili, geniş kütüphane desteği ve topluluk kaynakları sayesinde bu süreç çalışmamız için oldukça uygun görünmektedir. Bu çalışmada, Python 3.6 kullanılarak Scrapy web tarayıcısı ile veri çekme ve analiz süreci incelenmiştir. Scrapy, kullanıcının ihtiyaç duyduğu verileri toplamak için tasarlanmış açık kaynaklı bir web tarayıcısıdır. Bu yazılım, e-ticaret siteleri gibi kaynaklardan veri çekmek, bu verileri temizlemek, organize etmek ve analiz etmek için kullanılmaktadır. Bu çalışmada, e-ticaret siteleri ve sosyal medya platformları gibi çeşitli kaynaklardan veri çekilmiş ve analiz edilmiştir. Örneğin, bir e-ticaret sitesinden ürün detayları, kullanıcı yorumları ve ürün değerlendirmeleri gibi veriler çekilmiş. Sosyal medya platformlarından ise kullanıcı etkileşimleri ve popüler içerikler gibi veriler toplanmıştır. Çekilen bu veriler, CSV dosyası formatında saklanmış ve analiz edilmiştir.

Analiz sonucunda, e-ticaret sitesindeki en çok aranan ürün kategorileri, kullanıcıların en çok beğendiği ve yorum yaptığı ürünler belirlenmiştir. Ayrıca, kullanıcı yorumları ve değerlendirmeleri analiz edilerek, ürünlerin genel memnuniyet düzeyi ve sıkça dile getirilen sorunlar tespit edilmiştir. Sosyal medya verileri ise kullanıcıların en çok etkileşimde bulunduğu içerik türlerini ve popüler paylaşımları ortaya koymuştur.[24]

Ayat Abodayeh ve arkadaşlarının yaptığı veri kazıma ile ilgili diğer bir çalışmada ise amaçları, Python'un BeautifulSoup kütüphanesi kullanılarak geliştirilen bir web scraper ile çeşitli web sitelerinden veri çekmek ve bu verileri analiz etmektir. Web scraper, Amazon web sitesinden belirli ürün bilgilerini toplamakta ve bu verileri analiz ederek kullanıcıya anlamlı bilgiler sunmaktadır. Scraper, çekilen verileri grafiklerle görselleştirerek sonuçların anlaşılmasını kolaylaştırmaktadır. veri kazıma alanında yapılan çalışmalar, çeşitli yöntemler

kullanarak verilerin nasıl çekildiğini ve analiz edildiğini incelemiştir. Örneğin, Färholt, JavaScript'in Puppeteer kütüphanesini kullanarak veri kazıma tekniği geliştirmiştir. Bu çalışma, web sitelerindeki güvenlik mekanizmalarını atlatmayı başarmış olup Chaulagain ve Pandey ise Selenium ve WebDriver API kullanarak dinamik veri taleplerini yönetebilen bir veri kazıma aracı geliştirmiştir. [25]

Önerilen metodoloji, BeautifulSoup4 ve Requests kütüphanelerini kullanarak veri çekme ve analiz etme süreçlerini kapsamaktadır. Amazon web sitesine yapılan istekler sonucunda elde edilen HTML verileri, BeautifulSoup ile işlenerek istenen bilgiler toplanmıştır. Toplanan veriler, Matplotlib kütüphanesi kullanılarak grafiklerle görselleştirilmiş ve bir kullanıcı arayüzü aracılığıyla sunulmuştur. Geliştirilen web scraper, Amazon'dan Apple iPhone 14 Pro Max ve Raspberry Pi 4 gibi ürünler hakkında veri çekmiş ve bu verileri analiz etmiştir. Geliştirilen veri kazıyıcı, beş sayfa veriyi on saniyede toplayıp analiz edebilmiştir. Geliştirilen web scraper, sadece belirli ürün adları için çalışmakta ve sınırlı sayıda bilgi toplamaktadır. Amazon'un HTML yapısında yapılacak değişiklikler, scraper'ın çalışmasını etkileyebilmektedir.[26]

V. Srividhya ve arkadaşlarının yaptığı bir çalışmaya göre büyük miktarda veriye erişim ve bu verileri analiz etme ihtiyacı, günümüz araştırmalarının ve iş dünyasının merkezinde yer almaktadır. Veri kazıma, web sayfalarından veri çekme sürecini otomatikleştirerek hızlı ve etkili bir çözüm sunmayı amaçlamaktadırlar. Geleneksel yöntemlerle manuel olarak veri toplamakta ve özellikle büyük web siteleri veya büyük veri projeleri için oldukça zaman alıcı ve iş gücü gerektiren bir süreç olmaktadır. Bu nedenle, veri kazıma gibi otomatik yöntemler, hem araştırmacılar hem de işletmeler için değerli bir araç haline gelmiştir. Çalışmada önerilen metodoloji, BeautifulSoup4 ve Requests kütüphanelerini kullanarak veri çekme ve analiz etme süreçlerini anlatmaktadır. İlk adım olarak, Amazon web sitesine yapılan HTTP istekleri sonucunda elde edilen HTML verileri, BeautifulSoup ile işlenerek istenen bilgiler toplanmış toplanan bu veriler, Python'un Pandas kütüphanesi kullanılarak bir veri çerçevesine dönüştürülmüş ve Matplotlib kütüphanesi ile grafiksel olarak görselleştirilmiştir. Kullanıcı arayüzü olarak PySimpleGUI kütüphanesi kullanılmıştır, bu sayede kullanıcılar verileri kolayca inceleyebileceği bir arayüz tasarlanmıştır.

Çalışmada, Amazon web sitesinden Apple iPhone 14 Pro Max ve Raspberry Pi 4 gibi popüler ürünler hakkında veri çekilmiştir. Örneğin, Apple iPhone 14 Pro Max için yapılan analizde, ürün fiyatlarının ve kullanıcı değerlendirmelerinin dağılımı grafiklerle görselleştirilmiştir.

Elde edilen veriler, CSV dosyası formatında saklanmış ve istatistiksel analizler yapılmıştır. Analiz sonucunda, ürünlerin fiyat aralıkları, en yüksek ve en düşük fiyatlar, ortalama fiyatlar ve kullanıcı değerlendirme oranları gibi önemli bilgiler elde edilmiştir. Ayrıca, ürünlerin yorum sayıları ve bu yorumların puan dağılımları da incelenmiştir. Fakat geliştirilen web scraper, bir önceki çalışmada olduğu gibi sadece belirli ürün adları için çalışmakta ve sınırlı sayıda bilgi toplamaktadır. Amazon'un HTML yapısında yapılacak değişiklikler, scraper'ın çalışmasını etkileyebilmektedir. Bu nedenle, küçük projelerde kullanılmak üzere tasarlanmıştır ve farklı web sitelerinde test edilmesi önerilmektedir. Ayrıca, scraper'ın veri toplama sürecinde karşılaşılan zorluklar arasında, dinamik içeriklerin işlenmesi ve bazı web sitelerinin scraping işlemlerine karşı aldığı önlemler yer almaktadır. Bu sınırlamaların aşılması için, scraper'ın daha esnek ve uyarlanabilir hale getirilmesi gerekmektedir.

Çalışmamızın makine öğrenmesi kısmında ise Liudmila Prokhorenkova ve arkadaşlarının yaptığı çalışmaya göre cat boost algoritmasını inceledik. Çalışmanın odak CatBoost algoritmasının geliştirilmesi ve performansının diğer gradient boosting yöntemleriyle karşılaştırılmasıdır. Bu çalışmada, CatBoost'un temel yenilikleri olan sıralı boosting ve kategorik özelliklerin işlenmesi için yeni bir algoritmanın tanıtılması ve bu yeniliklerin teorik ve pratik avantajlarının detaylandırılması amaçlanmıştır.[27]

Thammisetty Swetha ve arkadaşlarının yaptığı çalışmada CatBoost kullanılarak çevrimiçi alışveriş yapanların satın alma niyetlerinin tahmin edilmesini ele almaktadır. Çevrimiçi alışveriş, tüketicilerin fiziksel mağazalara gitmeden ürün veya hizmet satın almasına olanak tanır. Ancak, birçok kullanıcı alışveriş amacı olmadan siteleri ziyaret eder ve bu, e-ticaret firmaları için bir zorluk oluşturmaktadır. Bu zorluğu çözmek için, çevrimiçi alışveriş yapanların satın alma niyetlerini tahmin etmek gereklidir. Bu çalışma, çeşitli sınıflandırma algoritmalarını (Random Forest, Decision Tree, Support Vector Machine, Gradient Boosting, AdaBoosting, LightGBM ve CatBoost) kullanarak bu niyeti tahmin etmeyi amaçlamaktadır. Deneysel sonuçlar CatBoost sınıflandırıcısının en yüksek doğruluğu (yüzde 98.73) sağladığını göstermektedir. [28]





Dolayısıyla bu çalışmada kapsam literatürde yapılan benzer çalışmaların araştırılarak, veri kazıma algoritması için hangi algoritmaların ve kütüphanelerin kullanılabileceği, fiyat tahmini için kullanılacak olan makine öğrenme algoritmasının sisteme ve verilere uyumluluğunun araştırılması ve bunların tasarlanacak önyüz üzerinden kullanıcılar tarafından yönetilebilmesi olarak belirlenmiştir.

## **2.2. Literatür Taraması Yapılması**

Çalışmamızın iki ana konusundan biri olan veri kazıma probleminin çözümü birkaç maddeden oluşmaktadır.

Veri kazıma kısaca özetlemek gerekirse Veri kazımanın ilk örnekleri, internetin yaygınlaşmaya başladığı 1990'larda ortaya çıktı. Bu dönemde, web sitelerinden manuel veri çekme işlemleri yapılyordu. 2000'li yılların gelmesi ile veri kazıma araçlarından 2 önemli gelişme olmuştur.

2004 yılında Leonard Richardson, HTML ve XML dosyalarını kolayca işleyebilmek için BeautifulSoup adlı kütüphaneyi geliştirdi. Bu kütüphane, Python programlama dili ile yazılmış olup, web scraping işlemlerini oldukça kolaylaştırdı.

2008 yılında geliştirilmeye başlanan Scrapy, açık kaynaklı bir web scraping çerçevesidir. Scrapy, daha karmaşık scraping işlemlerini yönetmek ve otomatikleştirmek için güçlü araçlar sunmaktadır.

### 2.2.1. Dil ve Kütüphane Seçimi

Dil ve kütüphane seçimi için incelediğimiz ilk çalışma Daniel Glez-Penca, Anaalia Lourenço ve arkadaşlarının yaptığı çalışmada açık kaynaklı veri kazıma kütüphane ve yöntemleri listelenmiştir[29]

**Tablo 2.1** Veri Kazıma Algoritma Tablosu

	C:HTTP istemcisi P:Parsing F:Framework	Etki alanına özgü dil	API/ Bağımsız	Dil	R: Regex H: HTML parsed tree X: XPath C: CSS selectors
UNIX shell	CP	Hayır	Bağımsız	bash	R
Curl/libcurl	C	Hayır	İkisinde	C+	
Web-Harvest	F	Evet	İkisinde	Java	RX
Jsoup	CP	Hayır	API	Java	HC
HttpClient	C	Hayır	API	Java	
jARVEST	F	Evet	İkisinde	JRuby/ Java	RXC
WWW::Mechanize	CP	Hayır	API	Perl	RX
Scrapy	F	Hayır	İkisinde	Python	RX
BeautifulSoup	P	Hayır	-	Python	H

Yapılan bu çalışmalar ışığında Python dili ve BeautifulSoup kütüphanesi kullanmaya karar verilmiştir.

### 2.2.2. Kazıma Algoritmasının Hazırlanması

Çalışmada kullanılacak algoritmanın hazırlanabilmesi için Narendra Kumar Rao ve arkadaşları tarafından hazırlanan çalışmayı inceledik[30]. Çalışma içerisinde;

- Verilerin spesifikasyonu
- Toplanması

- Düzenlenmesi
- Temizlenmesi
- Algoritmaların uygulanması
- Nihai sonuçların elde edilmesi

maddelerini inceleyip kendi çalışmamıza entegre etme kararı veridik

### 2.3. Veriseti Hazırlanması

Veriseti olarak için kendi verisetimizi hazırlamaya karar verdik. İlk olarak verilerin kazıma yapılacağı pazaryerini ardından kazıma işlemi yapacağımız kategorileri belirledik.

Pazaryerlerinin aldığı trafiğin önemli olmasının yanı sıra içerdiği ürün sayısı, ürünlere yapılan yorum ve değerlendirme sayıları da veri seti oluşturmak için önemli olduğundan dolayı bu maddeler hakkında da bazı araştırmalar gerçekleştirdik.

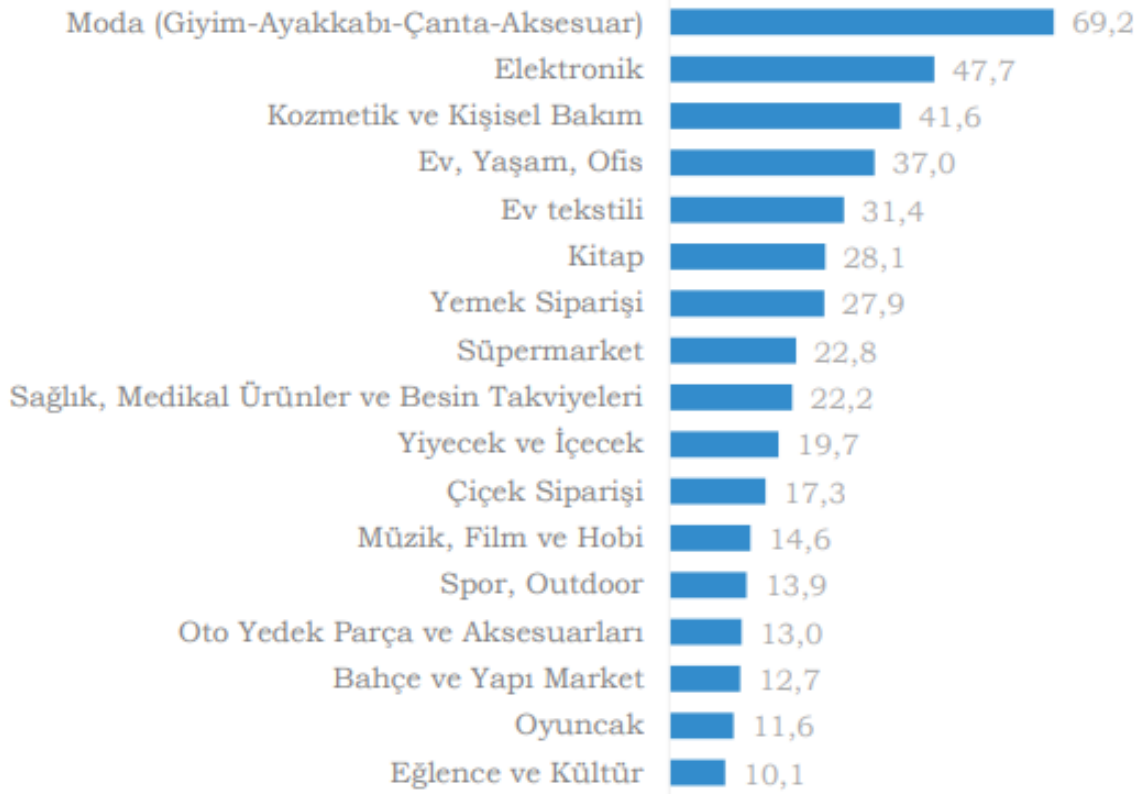
#### 2.3.1. Türkiye Pazaryeri Durum İstatistikleri

Pazaryeri olarak pazar payı ve ürün sayısının en çok olduğu platform olan Trendyol'u seçtik.

**Tablo 2.2** Yaşa Göre Tüketicilerin En Sık Tercih Ettiği Pazaryeri

Yaş	Trendyol	Hepsibur ada	GittiGidi yor	N11	Amazon	ÇiçekSe peti	AliExpre ss
18-24	62,9	22,1	5,6	5,0	2,4	0,3	0,9
25-34	56,4	23,9	5,8	6,5	2,9	1,4	1,6
34-44	42,2	27,5	11,8	8,7	2,1	2,9	1,9
45+	32,5	33,1	11,5	12,4	2,2	1,9	2,2

Kategori olarak ürün ve kozmetik ve kişisel bakım kategorisinin altında bulunan vitamin mineral ve kolajen kategorisini seçtik.



Şekil 2.2 Ürün Kategorilerinin Aldığı Pay

Tablo 2.3 Vazgeçilmez olarak gören kullanıcılar

Pazaryeri	Vazgeçilmeme Oranı	Baz
Trendyol	48,9	3978
Hepsiburada	27,3	3405
N11	22,8	4245
Morhipo	15,8	95
GittiGidiyor	14,6	2534
Aliexpress	10,9	406
Amazon	9,5	1242
PttAvm	6,7	934

Tablo 2.3.1.2, Şekil 2.3.1.1 ve Tablo 2.3.1.1’de görüldüğü gibi Pazaryeri üstünlüğü olarak trendyolun net bir şekilde önde olduğunu görebiliriz.

### 2.3.2. Veri toplama

Harsh Khatter et.[11] ve ekip arkadaşlarının çalışmalarını baz alarak veri kazıma işlemini 5 adımda ele almaya karar verdik.

Kullanıcıdan veri girişi olarak url oluşturulması

Hedef site üzerinde veri kazıma işlemi gerçekleştirebilmek için ilk adım url belirlemektir. Dinamik bir yapıya sahip olmak için kullanıcıdan veri girişi almaya karar verdik. Kullanıcı pazaryeri ve kategori seçerek farklı url üzerinden veri kazıma işlemi yapabilir. Bu işlem hedef url’lerde bulunan kategori kodları çıkarılarak başlar. Örneğin Trendyol’da bulunan vitamin kategorisinin kodu “vitamin-x-c105087” şeklinde belirlenmiştir. bu mantık ile url’de bulunan anahtar kelimeler değiştirilerek farklı kategoriler için url’ler oluşturulur.

```
<option value="vitamin-x-c105087">Vitamin</option>
<option value="mineral-x-c106093">Mineral</option>
<option value="kolajen-x-c109664">Kolajen</option>
```

Şekil 2.3 Kategori kodları

Kullanıcı aracı oluşturulması:

Veri kazıma algoritmalarında kullanıcı aracı oluşturmak kazıma işleminin hedef site tarafından tespit edilmesini epey zorlaştırmakta[12]. Örnek kullanıcı aracı “User-Agent": "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/XXX.0.0.0 Safari/XXX.36”

“Beautiful Soup” objesinin oluşturulması:

Url belirlendikten ve kullanıcı aracı oluşturulduktan sonra sıradaki adımımız soup objesi oluşturmak oldu. Python’da bs4 kütüphanesini import ederek “beautiful soup” kütüphanesini kullanmaya başlayabiliriz. “Beautiful Soup” sitenin kaynak HTML kodunu argüman olarak alır ve bir soup objesi oluşturur. Oluşturulan soup objesi HTML etiketlerini içeren bir ağaç yapısına sahip. “Beautiful Soup” ikinci bir argüman olarak soup objesinin nasıl yaratılacağına dair bir argüman alır biz “HTML.parser kullanmaya karar verdik. Oluşan soup objesi HTML etiketlerinden oluşan bir ağaç

yapısına sahip olduğu için çeşitli soup fonksiyonları ile farklı işlemler gerçekleştirebilmekteyiz.

```
url=("https://www.trendyol.com/"+kategori+"?pi="+str(i))

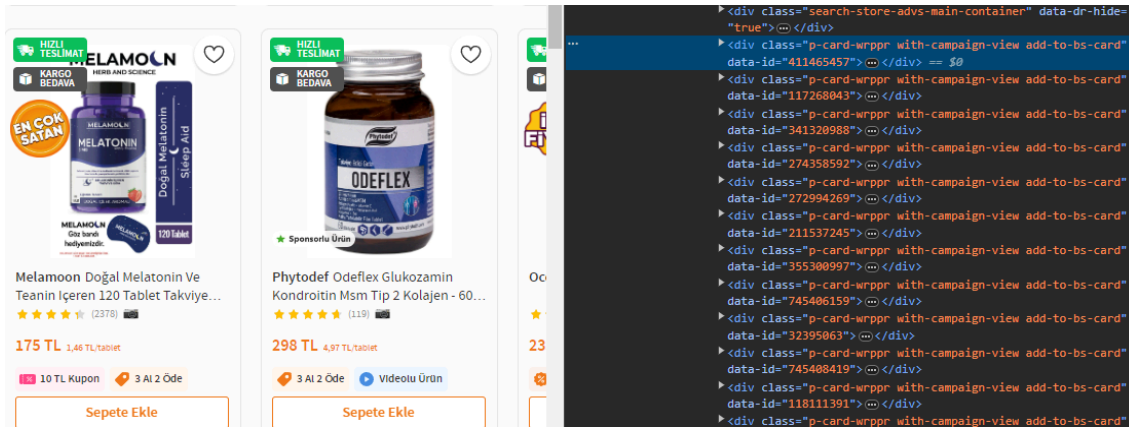
page = requests.get(url,headers=headers)

htmlPagePListing=BeautifulSoup(page.content,"html.parser")
```

Şekil 2.4 Örnek soup yapısı

Soup objesinden ilgili dataların çekilmesi:

Bu adım bir önceki adımın çıktısını girdi olarak alır. Temel olarak ürün hakkında anlamlı veriler üretebileceğimiz fiyat, ürün puanı gibi çıktılar aramaktayız. Bu sayede taradığımız ürünlerin veritabanında bir kaydını oluşturarak ürünü kayıt altına alabiliriz. Bu adımda da “Beautiful Soup” kütüphanesinden yardım aldık find(), findAll() gibi fonksiyonlar ile HTML kaynak kodunda bulunan class ve tagler üzerinde arama yapabilmekteyiz. Trendyol örneğinde ana sayfadaki ürünler div etiketine ve “p-card-wrppr with-campaign-view add-to-bs-card” adlı bir class yapısının içindeler. Bu sorgu ile sayfada bulunan ürünleri tespit ettik. Bulunan ürünlerin ismini kullanarak ikinci bir url oluşturduk. Bu url’i kullanarak tekrar bir “beautifulsoup” sorgusu çalıştırdık ve ürün detay sayfasından ürün ile ilgili detaylı verilere ulaştık. Bahsettiğimiz taglere ve sınıflara google chrome’un developer araçlarından kaynağı inceleyerek ulaştık.



Şekil 2.5 Google Chrome kaynağı inceleme

```

for p in htmlPagePlisting.findAll("div", class_="p-card-wrapper with-campaign-view add-to-bs-card"):
    anchor=p.find('a')
    href=anchor.get('href')
    #print(href)
    url1="https://www.trendyol.com/"+href

    page=requests.get(url1,headers=headers)
    htmlPagePDetail=BeautifulSoup(page.content,"html.parser")
    try:
        j=htmlPagePDetail.find("script", type="application/ld+json").get_text()
    except:
        print("err")
    productDetailJson=json.loads(j)
    print("tarama")
    try:
        data.extend([productDetailJson["@context"],productDetailJson["@type"],productDetailJson["@id"],productDetailJson["name"],productDetailJson["image"],productDetailJ
        data.extend([kategori])
        InsertProductJson(data)
    except:
        print("")
    data.clear()

```

**Şekil 2.6** Örnek bir “beautifulsoup” sorgusu

Trendyol’da ürün detay sayfası içerisinde “*application/ld+json*” etiketine sahip alanlarda ürünlere ait bilgilerin tutulduğunu gördük. HTML kaynak kodundan bu alanı ayıkladıktan sonra python json kütüphanesini kullanarak bu verileri json formatına dönüştürdük. Tarama sonucunda ürün hakkında edinebildiğimiz bilgiler için Tablo 2.3.2.1.

**Tablo 2.4** Örnek Veriler

Veri Adı	Veri İçeriği
context	<a href="https://schema.org/">https://schema.org/</a>
type	Product
id	https://www.trendyol.com/nutra
name	Nutraxin Magnezyum Complex
image	https://cdn.dsmedn.com/ty1010/
desc.	"Nutraxin Magnezyum Complex
sku	450797981
gtin13	8680512632108
brand.type	Brand
url	https://www.trendyol.com/nutra
offers.name	offer

offers.url	https://www.trendyol.com/nutra
offers.priceCurrency	TRY
offers.price	174.8
offers.itemCondition	https://schema.org/NewConditio
offers.availability	https://schema.org/InStock
aggregateRating.@type	AggregateRating
aggregateRating.ratingValue	4.7
aggregateRating.ratingCount	1476
aggregateRating.ratingCount	888

## 2.4. Kullanılacak Teknolojilerin Belirlenmesi

Çalışma kapsamında yapılan uygulamada, önyüz teknolojileri olarak HTML/CSS ve bootstrap, arkayüz teknolojisi olarak django kütüphanesi, veri kazıma için beautifulsoup kütüphanesi, fiyat tahmini için cat boost makine öğrenme algoritması, veritabanı için MySQL kullanılmıştır.

### 2.4.1. Veritabanı Tasarımı ve Gerçekleştirimi

Kazınan verilerin bilgisayar ortamına aktarılması için MySQL veritabanı tercih edilmiştir. MySQL, İVTYS tabanlı açık kaynaklı yazılımdır. Web siteleri, e-ticaret platformları ve mobil uygulamalar dahil olmak üzere çok çeşitli uygulamalarda kullanılır. SQL ise, MySQL'in temelini, yani çekirdek yapısını oluşturmaktadır. MySQL ismi ise "SQL (Structured Query Language)" ve Michael Widenius'un kızının adının (My) birleşiminden almaktadır.

Çalışmada tercih edilme sebepleri:

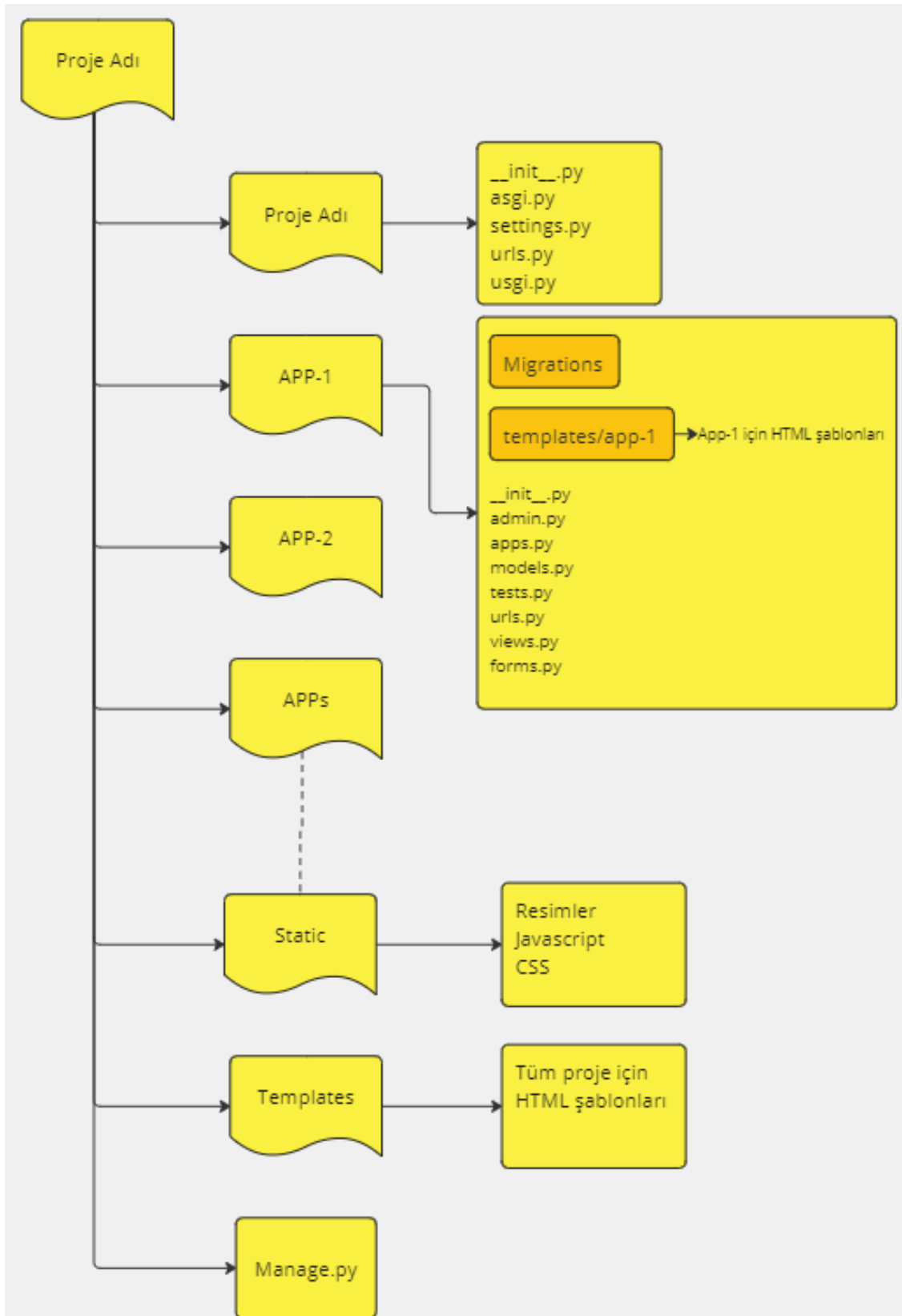
- Ücretsiz olan kullanılabilen açık kaynak bir veritabanıdır
- Microsoft SQL Server ve Oracle gibi diğer veritabanlarına kıyasla kullanımı basittir. Tüm programlama dilleriyle kullanılabilir.
- Büyük verilerde bile yüksek performans elde edilebilir.
- Kullanıcı sayısı yüksek olduğu için online destek ve alacağınız hataların çözümleri konusunda çözüme ulaşma ihtimali daha yüksektir.



- Tablo ve veritabanı boyutunda bir limit bulunmamaktadır.

#### **2.4.2. Arkayüz Proje Gerçekleştirimi**

Bu adımın temel amacı verileri daha anlaşılabilir ve mantıklı şekilde işleyerek kullanmak ve bir arkayüz oluşturmaktır. Bu yüzden Django framework kullanarak bir web sitesi oluşturmaya karar verdik.



Şekil 2.7 Django Yapısı

### 2.4.3. Django Framework

Django, Python dilinde yazılmış açık kaynaklı web uygulama geliştirmek için kullanılan bir framework'dür. MVT(Model View Template) dizayn yapısını kullanır. Django modeli veri tabanı yönetimi olarak görev yapmaktadır. *python manage.py makemigrations* ve *python manage.py migrate* komutlarını kullanarak önce veritabanı üzerinde yapılmış değişiklikler taranır ve ardından tüm değişiklikler kaydedilerek bir model oluşturulur. Ardından *views.py* dosyası üzerinden istekler yönetilerek kullanıcılara oluşturulan HTML şablonları gösterilir.

Django'yu kullandığımız "*local environment*"a kurmamız gerekmektedir. Ardından "*django-admin startproject projectName*" komutu ile yeni bir proje oluşturmamız gerekir. Projeyi oluşturduktan sonra "*python manage.py runserver*" komutu ile server lokal olarak çalıştırılır. [Http://localhost:8000/](http://localhost:8000/) adresini ziyaret ederek varsayılan django sayfasını görüntüleyebiliriz.

Django ile web sitesi oluşturmak için ilk olarak uygulama oluşturmamız gerekmektedir. Bunun için *python manage.py startapp <UYGULAMAADI>* komutu kullanılır. Uygulamayı çalıştırmak için *setting.py* dosyasının içerisinde bulunan *INSTALLED\_APPS* alanında uygulamamızı eklememiz gerekmektedir. Bu adımlar tamamladığında django uygulaması oluşmuş demektir. Oluşturduğumuz şablonu kullanmak için *urls.py* içerisinde bir yönlendirme işlemi yapmamız gerekmektedir.

```
urlpatterns=[
    path('',views.index,name='dashboard'),
    path('detay/',views.detayPage,name='detay'),
    path('detay/brands/',views.brands,name='brands'),
    path('detay/kategoriler/',views.kategoriler,name='kategoriler'),
    path('detay/fiyatHesaplama/',views.fiyatHesaplama,name='fiyatHesaplama'),
    path('detay/urunEkleme/',views.urunEkleme,name='urunEkleme'),
```

Şekil 2.8 Örnek Url yönlendirmesi

Django model veri tabanından alınan verileri oluşturmaya, almaya, güncellemeye ve silmeye olanak tanıyan bir veritabanı soyutlama API'si sağlar. Geneler olarak her model bir veri tabanı ile ilişkilendirilir. Biz projemizde MySql veri tabanını kullandık.

```

class Kategoritablosu(models.Model):
    checkbox_id = models.AutoField(db_column='checkbox_id', primary_key=True) # Field name made lowercase.
    name = models.CharField(max_length=45, blank=True, null=True)
    is_checked = models.IntegerField(blank=True, null=True)
    pazaryeri = models.CharField(max_length=45, blank=True, null=True)

    class Meta:
        managed = False
        db_table = 'kategoritablosu'

```

**Şekil 2.9** Örnek model yapısı

CRUD(Create, Read, Updating, Deleting) yani oluşturma, okuma, güncelleme ve silme işlemlerini ORM(Object Relational Mapper) adı verilen bir yapıya sahiptir. veri tabanlarında verilerin çoğu bir tür SQL tarafından düzenlenir, ancak her veritabanı SQL'i kendi yöntemiyle kullanır. SQL'in öğrenilmesi ve kullanılması zor olabilir. ORM aracı, bir nesne (ORM'deki 'O') ile temel bir veritabanı arasında basit bir harita sağlayarak veritabanı sistemini basitleştirir.

Django views bir dizi işlev sınıfıdır. Her şey uygulama dizinindeki view.py dosyasının içinde bulunur. Views, aslında bir web sitesini görüntülerken tarayıcıda gördüğünüz bir kullanıcı arayüzüdür.

```

def index(request):
    productjson=Productjson.objects.values("name","url","brandname","kategori","offers_price").distinct()
    context={'productjson':productjson}
    if 'scrape' in request.POST:
        if request.method == 'POST':
            data=request.POST["sayfasayisi"],request.POST["kategori"],request.POST["pazaryeri"]
            if data is not None:
                scrape_(data[0],data[1],data[2])
                return redirect('dashboard')
            else:
                return render (request,'accounts/login.html', {'error':'err'})
    elif 'detayPage' in request.POST:
        if request.method == 'POST':
            data=request.POST["detayPage"]
            print(data)
            pj_OneRow=Productjson.objects.filter(name=data).values()
            for item in pj_OneRow:
                item['image_urls'] = item['image'].split(',')
            context1={'productjson':pj_OneRow}
            return render(request,"detay.html",context1)
    elif 'brandDetay'in request.POST:
        if request.method == 'POST':
            data=request.POST["brandDetay"]
            context=Productjson.objects.filter(brandname=data).values("name","url","kategori","offers_price").distinct()
            context={'productjson':context}
            return render(request,"brandDetay.html",context)
    elif 'kategoriDetay'in request.POST:
        if request.method == 'POST':
            data=request.POST["kategoriDetay"]
            context=Productjson.objects.filter(kategori=data).values("name","url","brandname","offers_price").distinct()
            context={'productjson':context}
            return render(request,"kategoriDetay.html",context)
    else:
        return render(request,"index.html",context)

```

**Şekil 2.10** Örnek view yapısı

Django iki farklı view tipine sahiptir. Fonksiyon bazlı ve Class bazlı view.

Fonksiyon Bazlı

Bir web isteğini alan ve bir web yanıtı döndüren bir python işlevidir. Yanıt biçimi bir HTML içeriği, XML belgesi, 404 hatası vb. olabilir. Tüm görünüm işlevleri, ilk parametre olarak bir HttpRequest nesnesini alır.

#### Class Bazlı

Görünümlerin işlevler yerine python nesneleri olarak yürütülmesini sağlayan bir yapıdır. Fonksiyon bazlı view'ların yerini almazlar ancak bazı farklılıkları ve avantajları vardır.

Django, şablonlar ile güçlü HTML oluşturmanın kolay bir yolunu sunar. Django şablonları genellikle HTML, CSS ve Javascript kullanılarak oluşturulur. Ve son kullanıcının görebileceği HTML sayfaları üretir. Şablonlar temel olarak üç parametre alır.

- Request
- Şablonların bulunduğu dosya dizini
- Sözlük parametreleri

#### Django Şablon Dili (DTL)

Kullanıcının karşılaştığı programın frontend kısmını tanımlamak için küçük bir dil sağlar. Django şablonu, Django şablonunun diliyle kullanılan bir Python thread parçacığıdır. Bu sayede HTML dokümanı içerisinde değişkenler ve döngüler atayabilmekteyiz.

```
{% for item in productjson %}
<tr>
  <td>
    <!--<a href="{%url 'detay'%}">{{item.name}}</a-->
    <button type="submit" name="detayPage" value="{{item.name}}" class="btn-link">{{item.name}}</button>
  </td>
  <td>{{item.url}}</td>
  <td>
    <button type="submit" name="brandDetay" value="{{item.brandname}}" class="btn-link">{{item.brandname}}</button>
  </td>
  <td>
    <button type="submit" name="kategoriDetay" value="{{item.kategori}}" class="btn-link">{{item.kategori}}</button>
  </td>
  <td>{{item.offers_price}}</td>
</tr>
{% endfor %}
```

**Şekil 2.11** Django DTL Örneği

HTML formları, modern web sitelerinin temel bileşenlerinden biridir. Web sitesi ziyaretçilerinden ve kullanıcılarından bilgi toplamanın temel kaynağıdır. Django, HTML formları oluşturmak için kullanılan bir Form sınıfını içinde barındırır.

#### 2.4.4. Makine Öğrenmesi

Makine öğrenmesi ile fiyat tahmini aşamasında 3 farklı algoritma kullanmaya karar verilmiştir

- CatBoost
- RandomForestRegressor
- LGBMRegressor

Makine öğreniminde, özellikle denetimli öğrenme alanında, karar ağaçları tabanlı algoritmalar önemli bir yer tutar. Gradient boosting, bu ağaçları ardışık olarak eğiterek hataları azaltma amacı güden bir tekniktir. CatBoost (Categorical Boosting), Yandex tarafından geliştirilen ve kategorik veri setleriyle etkili bir şekilde başa çıkabilen güçlü bir gradient boosting algoritmasıdır.

CatBoost, diğer gradient boosting algoritmalarının (XGBoost, LightGBM) aksine, özellikle kategorik değişkenlerin işlenmesinde devrim niteliğinde yenilikler sunar. Temel teorik katkılarından bazıları şunlardır:

- Ordered Boosting

CatBoost, overfitting (aşırı öğrenme) riskini minimize etmek için "Ordered Boosting" tekniğini kullanır. Bu teknik, her bir ağacın eğitiminde rastgele bir veri alt kümesi kullanarak ağaçları ardışık olarak eğitir. Böylece, modelin her iterasyonunda aynı veriyi kullanarak overfitting riskini azaltır.

- Kategorik Özelliklerin Kodlanması

Kategorik verilerin işlenmesi genellikle zordur ve yanlış kodlama, model performansını olumsuz etkileyebilir. CatBoost, kategorik verileri işlemeyi kolaylaştıran ve bilgi kaybını minimize eden benzersiz bir kodlama tekniği kullanır. Bu teknik, kategorik değişkenleri sayısal değerlerle temsil ederken, hedef değişkenin (target variable) ortalamalarını kullanarak kodlamayı gerçekleştirir.

- Veri Sırasının Önemi

CatBoost, veri sırasının önemini vurgular ve verilerin sıralı bir şekilde işlenmesini sağlar. Bu, modelin genelleme kabiliyetini artırarak daha sağlam tahminler yapmasına olanak tanır.

RandomForestRegressor, makine öğreniminde yaygın olarak kullanılan ve regresyon problemlerini çözmek için tasarlanmış güçlü bir modeldir. Bu algoritma, temel olarak karar ağaçlarının bir araya gelerek oluşturduğu bir orman (forest) modelidir.

RandomForestRegressor, bir ensemble (topluluk) öğrenme yöntemidir ve Breiman'ın 2001 yılında tanıttığı Random Forest algoritmasının bir parçasıdır. Temel çalışma prensipleri şu şekildedir:

- Karar Ağaçları (Decision Trees)

RandomForestRegressor, temel yapı taşı olarak karar ağaçlarını kullanır. Karar ağaçları, veri集中的 örnekleri belirli kurallar dizisi ile dallara ayırarak sonuç tahmininde bulunur. Her bir yaprak düğümünde (leaf node), hedef değişkenin tahmini yapılır.

- Bootstrap Aggregating (Bagging)

RandomForestRegressor, bootstrap örnekleme yöntemini kullanarak farklı karar ağaçları oluşturur. Bootstrap, orijinal veri setinden rastgele ve tekrarlı olarak örnekler seçmeyi içerir. Her bir karar ağacı, bu rastgele örneklenmiş veri seti üzerinde eğitilir.

- Rastgele Özellik Seçimi (Random Feature Selection)

Her bir düğümde bölünme noktası (split point) seçilirken, tüm özellikler yerine rastgele seçilen bir alt küme dikkate alınır. Bu yaklaşım, ağaçlar arasındaki korelasyonu azaltır ve modelin genelleme kabiliyetini artırır.

- Tahminlerin Birleştirilmesi

Eğitilmiş tüm karar ağaçlarının tahminleri bir araya getirilir. Regresyon problemlerinde, bu tahminler ortalaması alınarak nihai tahmin elde edilir. Bu yöntem, hataları ve varyansı azaltarak daha doğru sonuçlar sağlar.

LGBMRegressor, LightGBM (Light Gradient Boosting Machine) kütüphanesinin bir parçası olarak, özellikle büyük veri setleri ve yüksek performans gerektiren regresyon problemleri için tasarlanmış güçlü bir makine öğrenimi algoritmasıdır.

LGBMRegressor, Microsoft tarafından geliştirilen LightGBM kütüphanesine dayanmaktadır ve gradient boosting framework'ünün bir varyantıdır. Temel çalışma prensipleri şu şekildedir:

- Gradient Boosting

Gradient boosting, zayıf tahmincilerden (weak learners), genellikle karar ağaçlarından, güçlü bir tahminci oluşturmak için kullanılan bir yöntemdir. LGBMRegressor, her iterasyonda hataları minimize etmek için mevcut modelin hatalarını düzelten yeni bir ağaç ekler.

- Leaf-wise Tree Growth

LGBMRegressor, ağaçları büyütürken leaf-wise (yaprak bazlı) strateji kullanır. Bu strateji, her iterasyonda en yüksek kazanç sağlayan yaprağı genişleterek daha derin ve dolayısıyla daha doğru tahminler yapabilen ağaçlar oluşturur. Bu, level-wise (seviye bazlı) stratejiden farklıdır ve genellikle daha iyi performans sağlar.

- Gradient-based One-Side Sampling (GOSS)

LGBMRegressor, veri setinin tamamını kullanmak yerine, yalnızca büyük gradient'lere sahip örnekleri seçerek hesaplamaları optimize eder. Bu, modelin eğitim süresini kısaltırken performansı düşürmez.

- Exclusive Feature Bundling (EFB)

EFB, birbirini dışlayan (mutually exclusive) özellikleri bir araya getirerek veri boyutunu azaltır. Bu özellik, özellikle yüksek boyutlu veri setlerinde modelin hızını ve verimliliğini artırır.

- Özellik Önem Sıralaması

LGBMRegressor, her bir özelliğin modeldeki önemini hesaplayarak, modelin hangi özelliklere daha fazla odaklandığını belirler. Bu bilgi, modelin yorumlanabilirliğini artırır ve özellik mühendisliği sürecine katkı sağlar.

## 2.4.5. Verilerin Hazırlanması

Veri kazıma sonrası verilerin makine öğrenme algoritmalarını eğitmek için verileri düzenlemeniz gerekmektedir.

İlk olarak algoritmalara göndereceğimiz veri sütunlarını seçerek başladık. Kullanacağımız sütunlar

- SKU

Her ürün benzersiz bir sku'ya sahip olduğu için ürünler arası ayrıştırma bu data üzerinden ilerlemektedir.



- Fiyat

Fiyat verisi en önemli datalardan biridir. Tarih, marka, kategori vb. bir çok veriye göre değişkenlik göstermesinden ötürü kullanmaya karar verdik.

- Ürün Puanı

Ürün puanı kullanıcının ürün satın alma sonrası ürüne verdiği puanı göstermekte olup ürün fiyatına etkisi bulunmaktadır.

- Kategori
- Marka

Kategori ve marka verisinin de fiyata etkisi bulunmaktadır.

- Verinin Okunduğu Tarih

Ürünün fiyat değişimini gözlemleyebilmek için tarih verisine ihtiyacımız olduğundan tarih verisini de kullanmaya karar verdik.

Sütunlar belirlendikten sonra verilerin algoritmalar tarafından işlenebilmesi için veri temizliği adımıyla başladık.

```
dropped_data['kategori']=dropped_data['kategori'].str.split('-', n=1, expand=True)[1]
dropped_data['kategori']=dropped_data['kategori'].str[3:]
dropped_data['kategori'] = dropped_data['kategori'].astype(int)
dropped_data['kategori'] = np.log(dropped_data['kategori'])
```

Şekil 2.12 Kategori Temizliği

```
dropped_data['offers_price'] = np.log(dropped_data['offers_price'].astype(float))
###
dropped_data['sku'] = dropped_data['sku'].astype(int)
dropped_data['sku'] = np.log(dropped_data['sku'])
###
enc_nom = (dropped_data.groupby('brandname').size()) / len(dropped_data)
enc_nom
dropped_data['brandname_encode'] = dropped_data['brandname'].apply(lambda x : enc_nom[x])
###
from datetime import datetime
dropped_data['dt_created'] = pd.to_datetime(dropped_data['dt_created'])
dropped_data['Month'] = [date.month for date in dropped_data.dt_created]
###
dropped_data.drop(['brandname', 'dt_created'],axis=1,inplace = True)
```

Şekil 2.13 Sütun Temizliği

- Kategori Dönüşümü: kategori sütunu, belirli bir formatta temizlenir ve logaritmik dönüşüm uygulanır.
- Fiyat ve SKU Dönüşümü: offers\_price ve sku sütunları logaritmik dönüşüme tabi tutulur.
- Kategorik Kodlama: brandname sütunu sayısal olarak kodlanır.
- Zaman Dönüşümü: dt\_created sütunu zaman damgasından ay bilgisine dönüştürülür ve ardından silinir.

- Eksik Verilerin İşlenmesi: aggregateRating.ratingValue sütunu sayısal formata dönüştürülür.

Veri temizleme işlemi tamamlandıktan sonra kazıma işlemi ile toplanan veriler ile eğitilen modellere ilgili ürünün verileri gönderilir ve fiyat tahmini alınır

**Tablo 2.5** Örnek Veriler Temizliği

SKU	Fiyat	Ürün Puanı	Kategori	Marka	Verinin Okunduğu Tarih
19.926530	5.163642	4.7	11.562544	0.030374	12
19.835236	5.105945	4.4	11.562544	0.004673	12
19.429947	5.556828	4.5	11.745521	0.010514	12
19.424961	5.798790	4.2	11.745521	0.049065	12

#### 2.4.6. Önyüz Proje Tasarımı, Ekran Tasarımı ve Gerçekleştirimi

Önyüz’de teknoloji olarak bootstrap kütüphanesi kullanılmıştır.

Bootstrap, web siteleri geliştirmek için yaygın olarak kullanılan bir önyüz kütüphanesidir. Açık kaynaklı ve ücretsizdir ve HTML, CSS ve JavaScript dillerini kullanır. Bootstrap, duyarlı tasarım, önceden hazırlanmış bileşenler ve özelleştirme seçenekleri gibi birçok avantaj sunar. Çalışmada bu kütüphanenin kullanılmasının sebebi farklı ekran boyutlarına otomatik olarak uyum sağlama, çok kolay öğrenilebilir ve uygulanabilir olması, butonlar, formlar, navigasyon menüleri, modaller ve daha fazlası gibi birçok önceden hazırlanmış bileşen içermesi gibi özellikler ön plana çıkmaktadır.

Hazırladığımız panelde ürün kazıma işlemi bu ekran Şekil 2.4.3.1’de bulunan ekran üzerinden ilerlemektedir. Pazaryeri, kategori ve kazınacak sayfa sayısı seçildikten sonra scrape butonu ile kazıma işlemi başlar.

Pazaryeri  
Trendyol

Kategori  
Vitamin

Sayfa Sayısı

Scrape

Scrape Sonuçları

10 entries per page

Search...

Ürün Adı	Ürün Linki	Marka Adı	Kategori Adı	Fiyat
Nutraxin Magnezyum Complex 60 Tablet 125 Mg - Bisiglinat - Taurat - Malat - Sitrat - B6	<a href="https://www.trendyol.com/nutraxin/magnezyum-complex-60-tablet-125-mg-bisiglinat-taurat-malat-sitrat-b6-p-450797981">https://www.trendyol.com/nutraxin/magnezyum-complex-60-tablet-125-mg-bisiglinat-taurat-malat-sitrat-b6-p-450797981</a>	Nutraxin	vitamin-xx-c105087	174.8
anocin MAGNEZYUM X4 COMPLEX CITRATE TAURATE BISGLYCINATE MALATE BROMELAIN + KOLAJEN + HASIT-D + B12 + B6	<a href="https://www.trendyol.com/anocin/magnezyum-x4-complex-citrate-taurate-bisglycinate-malate-bromelain-kolajen-h-asit-d-b12-b6-p-745406817">https://www.trendyol.com/anocin/magnezyum-x4-complex-citrate-taurate-bisglycinate-malate-bromelain-kolajen-h-asit-d-b12-b6-p-745406817</a>	anocin	vitamin-xx-c105087	262
Melamoon Doğal Melatonin Ve Teanin İçeren 120 Tablet Takviye Edici Gıda 3 Mg- Daha Kaliteli ve Stresiz uyku	<a href="https://www.trendyol.com/melamoon/dogal-melatonin-ve-teanin-iceren-120-tablet-takviye-edici-gida-3-mg-daha-kaliteli-ve-stresiz-uyku-p-411465457">https://www.trendyol.com/melamoon/dogal-melatonin-ve-teanin-iceren-120-tablet-takviye-edici-gida-3-mg-daha-kaliteli-ve-stresiz-uyku-p-411465457</a>	Melamoon	vitamin-xx-c105087	165
Phytodef Demir + Folik Asit + Vitamin B12 + Vitamin C - 30 Tablet	<a href="https://www.trendyol.com/phytodef/demir-folik-asit-vitamin-b12-vitamin-c-30-tablet-p-274358592">https://www.trendyol.com/phytodef/demir-folik-asit-vitamin-b12-vitamin-c-30-tablet-p-274358592</a>	Phytodef	vitamin-xx-c105087	259
Ocean Extramaq 200 Mg Magnezyum 60 Tablet	<a href="https://www.trendyol.com/ocean/extramaq-200-mg-magnezyum-60-tablet-p-272994269">https://www.trendyol.com/ocean/extramaq-200-mg-magnezyum-60-tablet-p-272994269</a>	Ocean	vitamin-xx-c105087	329.9

Şekil 2.12 Ürün kazıma ekranı

Ürünü envantere ekleme, geçmiş fiyatları ve puanları görüntüleme, referans ürün seçerek makine öğrenmesi ile fiyat tahmini görüntüleme işlemleri Şekil 2.4.3.2’de görünen ekrandan gerçekleşir.

## Araçlar

Ürün Ekleme

### Eklenecek Ürün

SKU

Ürün Adı

Marka

Aliş Fiyatı

KDV

Kar Marjı

Stok Miktarı

Ürün Ekle

### Referans Ürün

Ürün Adı

Ürünü Bul

Kaynak: <https://www.trendyol.com/sorvagen/magnezyum-malat-taurin-ve-vitamin-b6-p5p-60-tablet-p-471017633>

Ürün Adı: Sorvagen Magnezyum Malat, Taurin Ve Vitamin B6 (p5p) 60 Tablet

#### Fiyat Geçmişi

Fiyat	Tarih
338	Dec. 16, 2023, 3:50 p.m.
338	Dec. 18, 2023, 9:58 a.m.
369	March 8, 2024, 1:32 a.m.
376	May 13, 2024, 2:27 a.m.
376	May 13, 2024, 2:28 a.m.

#### Rating Geçmişi

Type	Value	RatingCount	ReviewCount	Tarih
AggregateRating	4.6	895	537	Dec. 16, 2023, 3:50 p.m.
AggregateRating	4.6	902	540	Dec. 18, 2023, 9:58 a.m.
AggregateRating	4.63	1011	578	March 8, 2024, 1:32 a.m.
AggregateRating	4.6	1092	612	May 13, 2024, 2:27 a.m.
AggregateRating	4.6	1092	612	May 13, 2024, 2:28 a.m.

#### Fiyat Tahmini

CatBoost	RF
327.11394915541564	299.69479027656416
327.11394915541564	299.69479027656416
327.11394915541564	299.69479027656416
327.11394915541564	299.69479027656416
327.11394915541564	299.69479027656416

LGBM
380.2837902521053
380.2837902521053
380.2837902521053
380.2837902521053
380.2837902521053

Şekil 2.13 Envanter Ekleme Ekranı

Kategori filtreleme ekranı Şekil 2.4.3.3’de görüldüğü gibidir.

## Kategori

Kategoriler

kolajen-x-c109664

Filtrele

inler

10 entries per page

Search...

Ürün Adı	Ürün Linki	Marka Adı	Fiyat
Nutrain Magnezyum Complex 60 Tablet 125 Mg - Bisglisinat - Taurat - Malat - Stral - B6	https://www.trendyol.com/nutrain/magnezyum-complex-60-tablet-125-mg-bisglisinat-taurat-malat-stral-b6-p-450797981	Nutrain	174.8
anocin MAGNEZYUM X4 COMPLEX CITRATE TAURATE BISGLYCINATE MALATE + BROMELAIN + KOLAJEN + HASIT + D3 + B12 + B6	https://www.trendyol.com/anocin/magnezyum-x4-complex-citrate-taurate-bisglycinate-malate-bromelain-kolajen-h-asit-d-b12-b6-p-745406817	anocin	262
Melamoon Doğal Melatonin Ve Teanin İçeren 120 Tablet Takviye Edici Gıda 3 Mg- Daha Kaliteli Ve Stresiz Uyku	https://www.trendyol.com/melamoon/dogal-melatonin-ve-teanin-iceren-120-tablet-takviye-edici-gida-3-mg-daha-kaliteli-ve-stresiz-uyku-p-411465457	Melamoon	165
Phytodef Demir + Folik Asit + Vitamin B12 + Vitamin C - 30 Tablet	https://www.trendyol.com/phytodef/demir-folik-asit-vitamin-b12-vitamin-c-30-tablet-p-274358592	Phytodef	239
Ocean Extramao 200 Mg Magnezyum 60 Tablet	https://www.trendyol.com/ocean/extramao-200-mg-magnezyum-60-tablet-p-272994269	Ocean	329.9
Perfectil Original	https://www.trendyol.com/perfectil/original-p-25804959	Perfectil	265
Youpilus Vitamin C Çinko & Propolis 20 Efervesan Tablet	https://www.trendyol.com/youpilus/vitamin-c-cinko-propolis-20-efervesan-tablet-p-222432324	Youpilus	64.55
Phytodef Kalsiyum Magnezyum Çinko - 60 Tablet & Magnezyum B6 D3 - 30 Tablet /CALCIUM MAGNESIUM ZINC/	https://www.trendyol.com/phytodef/kalsiyum-magnezyum-cinko-60-tablet-magnezyum-b6-d3-30-tablet-calcium-magnesium-zinc-p-211537245	Phytodef	376
Ocean D3k2 Damla 20 ml	https://www.trendyol.com/ocean/d3k2-damla-20-ml-p-6476282	Ocean	202.5
SOVITAL Sieroop Melatonin 3 Mg / 6 Mg & Magnezyum Uyku Destek Vegan Gummy 60 Adet	https://www.trendyol.com/sovital/sieroop-melatonin-3-mg-6-mg-magnezyum-uyku-destek-vegan-gummy-60-adet-p-470297563	SOVITAL	425

wjwing 1 to 10 of 6173 entries

1

2

3

4

5

6

7

...

618

Şekil 2.14 Kategori filtreleme

Marka filtreleme ekranı Şekil 2.4.3.3’de görüldüğü gibidir.

## Marka

Markalar

"Vitanil's Nutrition"

Filtrele

Örünler

10 entries per page

Search...

Ürün Adı	Ürün Linki	Kategori Adı	Fiyat
Nutrain Magnezyum Complex 60 Tablet 125 Mg - Bisglisinat - Taurat - Malat - Stral - B6	<a href="https://www.trendyol.com/nutrain/magnezyum-complex-60-tablet-125-mg-bisglisinat-taurat-malat-stral-b6-p-450797981">https://www.trendyol.com/nutrain/magnezyum-complex-60-tablet-125-mg-bisglisinat-taurat-malat-stral-b6-p-450797981</a>	vitamin-x-c105087	174.8
anocin MAGNEZYUM X4 COMPLEX CITRATE TAURATE BISGLYCINATE MALATE + BROMELAIN + KOLAJEN + HASIT + D3 + B12 + B6	<a href="https://www.trendyol.com/anocin/magnezyum-x4-complex-citrate-taurate-bisglycinate-malate-bromelain-kolajen-h-asit-d-b12-b6-p-745406817">https://www.trendyol.com/anocin/magnezyum-x4-complex-citrate-taurate-bisglycinate-malate-bromelain-kolajen-h-asit-d-b12-b6-p-745406817</a>	vitamin-x-c105087	262
Melamoon Doğal Melatonin Ve Teanin İçeren 120 Tablet Takviye Edici Gıda 3 Mg- Daha Kaliteli Ve Stresiz Uyku	<a href="https://www.trendyol.com/melamoon/dogal-melatonin-ve-teanin-iceren-120-tablet-takviye-edici-gida-3-mg-daha-kaliteli-ve-stresiz-uyku-p-411465457">https://www.trendyol.com/melamoon/dogal-melatonin-ve-teanin-iceren-120-tablet-takviye-edici-gida-3-mg-daha-kaliteli-ve-stresiz-uyku-p-411465457</a>	vitamin-x-c105087	165
Phytodef Demir + Folik Asit + Vitamin B12 + Vitamin C - 30 Tablet	<a href="https://www.trendyol.com/phytodef/demir-folik-asit-vitamin-b12-vitamin-c-30-tablet-p-274358592">https://www.trendyol.com/phytodef/demir-folik-asit-vitamin-b12-vitamin-c-30-tablet-p-274358592</a>	vitamin-x-c105087	239
Ocean Extramao 200 Mg Magnezyum 60 Tablet	<a href="https://www.trendyol.com/ocean/extramao-200-mg-magnezyum-60-tablet-p-272994269">https://www.trendyol.com/ocean/extramao-200-mg-magnezyum-60-tablet-p-272994269</a>	vitamin-x-c105087	329.9
Perfectil Original	<a href="https://www.trendyol.com/perfectil/original-p-25804959">https://www.trendyol.com/perfectil/original-p-25804959</a>	vitamin-x-c105087	265
Youpilus Vitamin C Çinko & Propolis 20 Efervesan Tablet	<a href="https://www.trendyol.com/youpilus/vitamin-c-cinko-propolis-20-efervesan-tablet-p-222432324">https://www.trendyol.com/youpilus/vitamin-c-cinko-propolis-20-efervesan-tablet-p-222432324</a>	vitamin-x-c105087	64.55
Phytodef Kalsiyum Magnezyum Çinko - 60 Tablet & Magnezyum B6 D3 - 30 Tablet /CALCIUM MAGNESIUM ZINC/	<a href="https://www.trendyol.com/phytodef/kalsiyum-magnezyum-cinko-60-tablet-magnezyum-b6-d3-30-tablet-calcium-magnesium-zinc-p-211537245">https://www.trendyol.com/phytodef/kalsiyum-magnezyum-cinko-60-tablet-magnezyum-b6-d3-30-tablet-calcium-magnesium-zinc-p-211537245</a>	vitamin-x-c105087	376
Ocean D3k2 Damla 20 ml	<a href="https://www.trendyol.com/ocean/d3k2-damla-20-ml-p-6476282">https://www.trendyol.com/ocean/d3k2-damla-20-ml-p-6476282</a>	vitamin-x-c105087	202.5
SOVITAL Sieroop Melatonin 3 Mg / 6 Mg & Magnezyum Uyku Destek Vegan Gummy 60 Adet	<a href="https://www.trendyol.com/sovital/sieroop-melatonin-3-mg-6-mg-magnezyum-uyku-destek-vegan-gummy-60-adet-p-470297563">https://www.trendyol.com/sovital/sieroop-melatonin-3-mg-6-mg-magnezyum-uyku-destek-vegan-gummy-60-adet-p-470297563</a>	vitamin-x-c105087	425

Showing 1 to 10 of 6173 entries

1 2 3 4 5 6 7 ... 618

Şekil 2.15 Marka filtreleme

Ürünün geçmiş fiyatları, puanı, yorum sayısı, kazıma tarihi, fotoğrafları ve pazaryeri linki Şekil 2.4.3.5’de görüldüğü gibidir.

## Detay

Ocean D3k2 Damla 20 ml

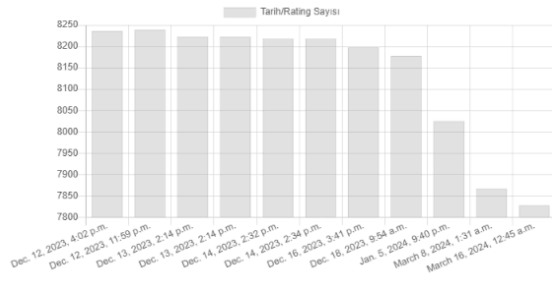
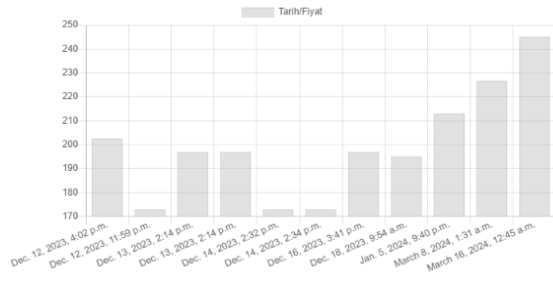
Yınak: <https://www.trendyol.com/ocean/d3k2-damla-20-ml-p-6476282>

Fiyat Geçmişi

Fiyat	Tarih
202.5	Dec. 12, 2023, 4:02 p.m.
172.84	Dec. 12, 2023, 11:59 p.m.
197	Dec. 13, 2023, 2:14 p.m.
197	Dec. 13, 2023, 2:14 p.m.
172.84	Dec. 14, 2023, 2:32 p.m.
172.84	Dec. 14, 2023, 2:34 p.m.
197	Dec. 16, 2023, 3:41 p.m.
195	Dec. 18, 2023, 9:54 a.m.
213	Jan. 5, 2024, 9:40 p.m.
226.7	March 8, 2024, 1:31 a.m.
245.16	March 16, 2024, 12:45 a.m.

Rating Geçmişi

Value	RatingCount	ReviewCount	Tarih
4.8	8236	4340	Dec. 12, 2023, 4:02 p.m.
4.8	8239	4341	Dec. 12, 2023, 11:59 p.m.
4.8	8223	4332	Dec. 13, 2023, 2:14 p.m.
4.8	8223	4332	Dec. 13, 2023, 2:14 p.m.
4.8	8218	4329	Dec. 14, 2023, 2:32 p.m.
4.8	8218	4329	Dec. 14, 2023, 2:34 p.m.
4.8	8198	4322	Dec. 16, 2023, 3:41 p.m.
4.8	8178	4306	Dec. 18, 2023, 9:54 a.m.
4.8	8025	4226	Jan. 5, 2024, 9:40 p.m.
4.76	7867	4115	March 8, 2024, 1:31 a.m.
4.75	7828	4094	March 16, 2024, 12:45 a.m.



Şekil 2.16 Ürün Detay

### 3. SONUÇLAR

Bu çalışmada, e-ticaret sektöründe dinamik fiyatlandırma ve ürün takibi konusundaki zorluklara çözüm bulmak amacıyla derin öğrenme ve veri kazıma destekli bir ürün yönetim platformu geliştirilmiştir. Araştırma süreci boyunca elde edilen bulgular ve sonuçlar şu şekilde özetlenebilir:

- Veri Toplama ve İşleme:

Veri kazıma teknikleri kullanılarak çeşitli e-ticaret sitelerinden ürün fiyatları, kullanıcı yorumları, puanlamalar ve stok bilgileri toplanmıştır. Bu veriler, Python'un BeautifulSoup ve Scrapy kütüphaneleri kullanılarak etkin bir şekilde işlenmiş ve analiz edilmiştir.

Verilerin temizlenmesi ve ön işleme süreçleri kapsamında, eksik veriler doldurulmuş, kategorik veriler sayısal formata dönüştürülmüş ve zaman damgaları analiz için uygun hale getirilmiştir.

- Makine Öğrenmesi Modelleri:

Dinamik fiyat tahmini için CatBoost, LightGBM ve RandomForestRegressor gibi güçlü makine öğrenmesi algoritmaları kullanılmıştır. Modeller, fiyat değişikliklerini doğru bir şekilde tahmin etmek için eğitilmiş ve test edilmiştir.

Modellerin performansları karşılaştırılmış ve CatBoost algoritmasının, özellikle e-ticaret fiyat tahmini için en yüksek doğruluğu sağladığı belirlenmiştir .

- Dinamik Fiyatlandırma:

Geliştirilen platform, derin öğrenme algoritmaları kullanarak dinamik fiyatlandırma yapabilmekte ve gerçek zamanlı fiyat tahminleri sunabilmektedir. Bu sayede, platform değişen pazar koşullarına hızlı bir şekilde uyum sağlayabilmektedir.

**Tablo 3.1** Örnek Veriler Temizliği

	CatBoost	RFG	LGBM
RMSE	0.598	0.546	0.587

- Ürün Takibi:

Ürünlerin kategori ve markalarına göre takibi yapılmış, fiyat değişiklikleri gibi bilgiler periyodik olarak izlenmiştir.

Platform, kullanıcıların daha bilinçli satın alma kararları alabilmeleri için detaylı ürün ve fiyat analizleri sunmaktadır.

Bu çalışma, e-ticaret sektöründe dinamik fiyatlandırma ve ürün takibi konusundaki mevcut zorluklara etkili çözümler sunmakta ve bu alandaki literatüre önemli katkılarda bulunmaktadır. Gelecekte, daha geniş veri setleri ve gelişmiş algoritmalar kullanılarak platformun daha da geliştirilmesi ve yaygınlaştırılması hedeflenmektedir

#### 4. KAYNAKLAR

##### Referanslar

- [1] Desheng Dash Wu & Anthony Ross, 2018. "Pricing optimization, channel and uncertainty," *International Journal of Production Research*, Taylor & Francis Journals, vol. 56(5), pages 1733-1737, March.
- [2] Giorgio Alfredo Spedicato, Christophe Dutang and Leonardo Petrini. (2018). "Machine Learning Methods to Perform Pricing Optimization," *A Comparison with Standard GLMs* Variance 12 (1): 69-89.
- [3] Sukanta Nayak, (2020) "Fundamentals of Optimization algorithms," 1st Edition Academic Press, ISBN: 9780128211267.
- [4] M. Jaswanth, N. K. L. Narayana, S. Rahul, R. Subramani. and K. Murali., "Product Price Optimization using Least Square Method," 2022 IEEE 2nd International Conference on Mobile Networks and Wireless Communications (ICMNWC), Tumkur, Karnataka, India, 2022, pp. 1-5, doi: 10.1109/ICMNWC56175.2022.10031834.
- [5] Emmanuelle Clément, Damien Lamberton and Philip Protter (2002). "An analysis of a least squares regression method for American option pricing," *Finance Stochast.* 6: 449–471.
- [6] Anusha, Sangeetha.H.S, Riyanka.R, Sushmitha.N and Chitra.R. (2017). "Dynamic Price Optimization for the Future of E-Commerce," *International Journal of Advanced Research in Computer and Communication Engineering* 6 (4): 879-883.
- [7] T. Matthew, *php|architect's Guide to Web Scraping with PHP*, 2010, Marco Tabini & Associates, Inc..
- [8] V. Srividhya<sup>1</sup>, P.Megala, "Scraping and Visualization of Product Data from E-commerce Websites", *IJCSE International Journal of Computer Sciences and Engineering*, May 2019
- [9] Amalia Amalia, Rizky Maulidya Afifa, Herriyance Herriyance, "Resource Description Framework Generation for Tropical Disease Using Web Scraping", 2018 IEEE International Conference on Communication, Networks and Satellite (Commnetsat)
- [10] Rizqi Putri Nourma Budiarti, Nanang Widyatmoko, Mochamad Hariadi and Mauridhi Hery Purnomo, "Web Scraping for Automated Water Quality Monitoring System: A case

study of PDAM Surabaya”, 2016 International Seminar on Intelligent Technology and Its Application

[11]H. Khatter, Dravid, A. Sharma and A. K. Kushwaha, "Web Scraping based Product Comparison Model for E-Commerce Websites," 2022 IEEE International Conference on Data Science and Information System (ICDSIS), Hassan, India, 2022, pp. 1-6, doi: 10.1109/ICDSIS55133.2022.9915892.

[12]<http://rochi.utcluj.ro/rrioc/articole/RRIOC-11-1-Gheorghe.pdf>

[13]<https://www.annalsofrscb.ro/index.php/journal/article/view/6301/4788>

[14]C. Zhan, J. Li, W. Jiang, W. Sha and Y. Guo, "E-commerce Sales Forecast Based on Ensemble Learning," 2020 IEEE International Symposium on Product Compliance Engineering-Asia (ISPCE-CN), Chongqing, China, 2020, pp. 1-5, doi: 10.1109/ISPCE-CN51288.2020.9321858.

[15] K Das and Afreen Ara. Growth of e-commerce in india. Growth, available at: [http://ijcem.in/wpcontent/uploads/2015/08/Growth of E Commerce in India. pdf](http://ijcem.in/wpcontent/uploads/2015/08/Growth%20of%20E%20Commerce%20in%20India.pdf) (accessed 9 December 2015), 2015.

[16] Azizi Othman. National ecommerce strategic roadmap overview, 12 2018.

[17] Statista. E-commerce share of total global retail sales from 2015 to 2023, 2020.

[18] C. Ranganathan and Shobha Ganapathy. Key dimensions of business-to-consumer web sites. *Information & Management*, 39(6):457 – 465, 2002.

[19] Kui Zhao and Can Wang. Sales forecast in e-commerce using convolutional neural network. *arXiv preprint arXiv:1708.07946*, 2017.

[20] Yuta Kaneko and Katsutoshi Yada. A deep learning approach for the prediction of retail store sales. In *IEEE International Conference on Data Mining Workshops*, 2016.

[21]A. Abodayeh, R. Hejazi, W. Najjar, L. Shihadeh and R. Latif, "Web Scraping for Data Analytics: A BeautifulSoup Implementation," 2023 Sixth International Conference of Women in Data Science at Prince Sultan University (WiDS PSU), Riyadh, Saudi Arabia, 2023, pp. 65-69, doi: 10.1109/WiDS-PSU57071.2023.00025. keywords: {Codes;Data visualization;Data science;Big Data applications;Libraries;Data mining;web scraping;BeautifulSoup;data gathering;data analytics;data visualization},

[22]C. Yin and J. Han, "Dynamic Pricing Model of E-Commerce Platforms Based on Deep Reinforcement Learning," *Comput. Model. Eng. Sci.*, vol. 127, no. 1, pp. 291-307. 2021. <https://doi.org/10.32604/cmes.2021.014347>

[23]M. Jaswanth, N. K. L. Narayana, S. Rahul, R. Subramani. and K. Murali., "Product Price Optimization using Least Square Method," 2022 IEEE 2nd International Conference on



Mobile Networks and Wireless Communications (ICMNBC), Tumkur, Karnataka, India, 2022, pp. 1-5, doi: 10.1109/ICMNBC56175.2022.10031834. keywords:

{Industries;Biological system modeling;Pricing;Companies;Production;Machine learning;Predictive models;Marketing Industry;Quantity Optimization;Dynamic Pricing;Revenue;Least Square Method;Decision Making Strategy},

[24]D. M. Thomas and S. Mathur, "Data Analysis by Web Scraping using Python," 2019 3rd International conference on Electronics, Communication and Aerospace Technology (ICECA), Coimbatore, India, 2019, pp. 450-454, doi: 10.1109/ICECA.2019.8822022.

keywords: {Data mining;Conferences;Python;Business;Crawlers;Databases;Aerospace electronics;Data analysis;Web Scraping;Implementing Web Scrape},

[25]Chaulagain, Ram & Pandey, Santosh & Basnet, Sadhu & Shakya, Subarna. (2017). Cloud Based Web Scraping for Big Data Applications. 138-143. 10.1109/SmartCloud.2017.28.

[26]A. Abodayeh, R. Hejazi, W. Najjar, L. Shihadeh and R. Latif, "Web Scraping for Data Analytics: A BeautifulSoup Implementation," 2023 Sixth International Conference of Women in Data Science at Prince Sultan University (WiDS PSU), Riyadh, Saudi Arabia, 2023, pp.

65-69, doi: 10.1109/WiDS-PSU57071.2023.00025. keywords: {Codes;Data visualization;Data science;Big Data applications;Libraries;Data mining;web scraping;BeautifulSoup;data gathering;data analytics;data visualization},

[27]Vasudevan, Srividhya & Megala, P.. (2019). Scraping and Visualization of Product Data from E-commerce Websites. International Journal of Computer Sciences and Engineering. 7. 1403-1407. 10.26438/ijcse/v7i5.14031407.

[28]T. Swetha, R. R, T. Sajitha, V. B, J. Sravani and B. Praveen, "Forecasting Online Shoppers Purchase Intentions with Cat Boost Classifier," 2024 International Conference on Distributed Computing and Optimization Techniques (ICDCOT), Bengaluru, India, 2024, pp. 1-6, doi: 10.1109/ICDCOT61034.2024.10515309. keywords: {Support vector

machines;Focusing;Boosting;Prediction algorithms;Classification algorithms;Electronic commerce;Forecasting;Purchase Intention Prediction;Classification Algorithms;Cat Boost;Hyperparameter tuning},

[29]Daniel Glez-Peña, Anália Lourenço, Hugo López-Fernández, Miguel Reboiro-Jato, Florentino Fdez-Riverola, Web scraping technologies in an API world, Briefings in Bioinformatics, Volume 15, Issue 5, September 2014, Pages 788–797,

<https://doi.org/10.1093/bib/bbt026>

