

Kocaeli Üniversitesi

Bilgisayar Mühendisliği Bölümü

Programlama Laboratuvarı II

Proje III

Emirhan Koç

200202068

Mustafa Kahveci

200202036

I. GİRİŞ

Proje Python dilinde Microsoft Visual Studio Code IDE'si kullanılarak yazılmıştır.

Proje tasarlanırken HTML,CSS,Django ve veritabanları ile ilgili teknik bilgiler araştırılmış ve gerekli algoritmaların kurulmasında karar kılınmıştır.

Kendinize ait bir E-ticaret sitesi oluşturmalsınız ve bu site Admin tarafından güncellenebilir olmalıdır.

Sitede yapılacak değişikliklerde ;

- Bir notebook ürünün fiyatının değiştirilmesi
- Notebook ürünü ait bir kaydın kaldırılması
- Yeni notebook ürünün eklenmesi
- Notebook ürünün puanında değişiklik yapılması
- Ürün bilgilerinin güncellenmesi

bilgilerin web scraping ile çekilmesi gerekmektedir. Alınan tüm bu değişiklikler farklı sitelerdeki ürünlerin listelendiği diğer web sitesinde anlık olarak gösterilmelidir.

Ürün bilgisinin gösterildiği web sayfasında tüm notebook ürünlerinin listelenmesi ve ürün bilgilerinin yer aldığı ana başlıklarının oluşturulması gerekmektedir. Bu başlığa tıklayınca ürün hakkında bilgilerin (fiyat bilgileri dahil) olduğu sayfa açılmalıdır.

Ayrıca ürünlerin listelendiği sayfadaki E-ticaret linki üzerine tıklayınca ürünün bu E-ticaret sitesindeki sayfasına kullanıcı yönlendirilmelidir.

Dinamik arama ile (yukarıda istenilen tüm arama kriterleri dahil olacak şekilde) Asus ..vb modele göre arama işleminin yapılması

Dinamik arama ile (yukarıda istenilen tüm arama kriterleri dahil olacak şekilde) hepsiburada ..vb E-ticaret sitelerinin aranması

Ürün bilgilerinin gösterildiği web sayfası üzerinde Filtreleme işleminin yapılması

Fiyat bilgisine göre küçükten büyüğe veya büyüktan küçüğü sıralama ayrıca puanı yüksek olan ürüne göre sıralama işleminin yapılması

Sizin oluşturacağınız E- Ticaret sitesine kayıtlı olan bir ürünün aynı bilgileri ile kayıt edilmeli ve veritabanında Duplicate Kontrolü yapılmalıdır.

Sizin oluşturacağınız E- Ticaret sitesinde iki farklı kayıt işleminin yapılması ve Near Duplicate kontrolünün yapılması

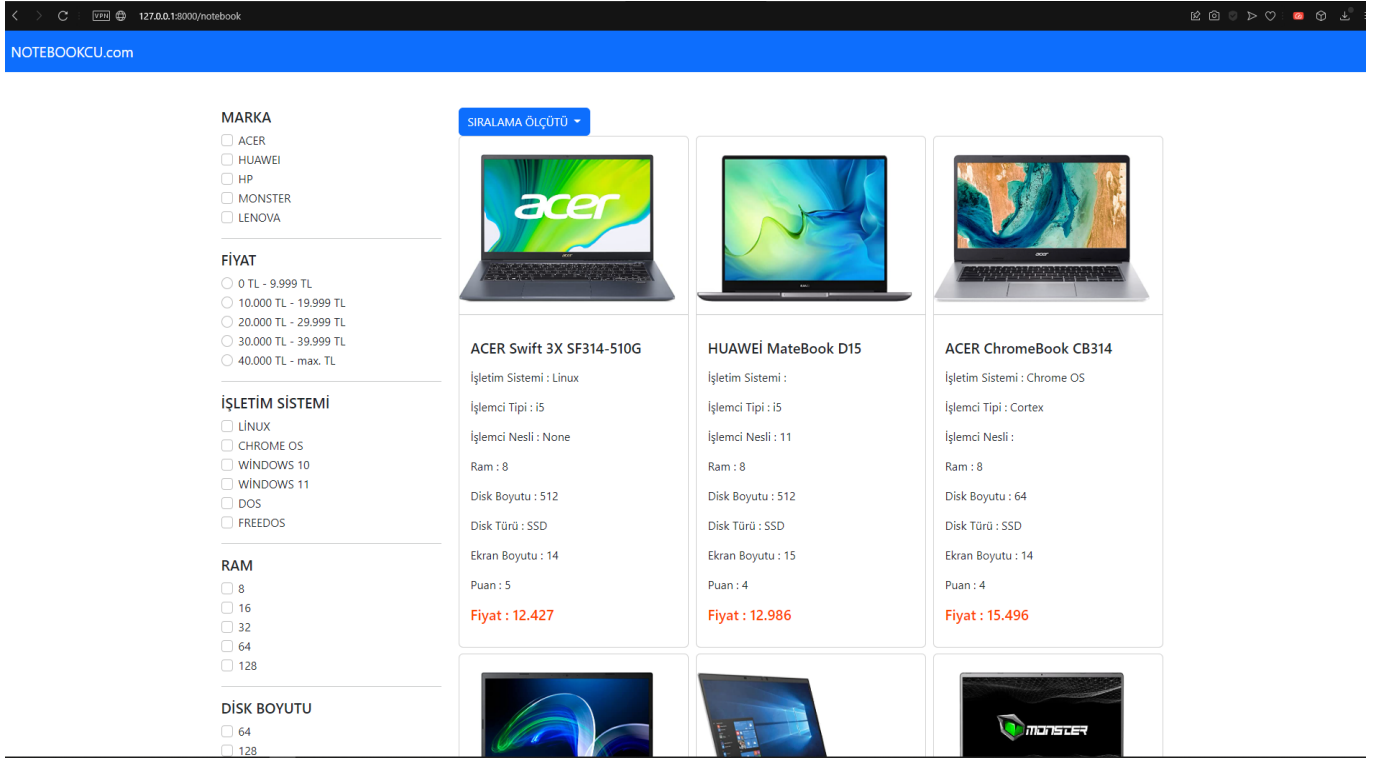
II. YÖNTEM

- Başlangıçta proje detaylı bir şekilde analiz edildi.

- Python dili,Django,HTML ve CSS hakkında gerekli araştırmalar yapıldı.

- Yapılması istenen web siteleri ve veritabanları oluşturuldu.

- Oluşturulan web siteleri ve veritabanları :





ACER Swift 3X SF314-510G

İşletim Sistemi : Linux

İşlemci Tipi : i5

İşlemci Nesli : None

Ram : 8

Disk Boyutu : 512

Disk Türü : SSD

Ekran Boyutu : 14

Puan : 5

Sepete Ekle

Yeni Veritabanı

Veritabanı Aç

Değişiklikleri Kaydet

Değişiklikleri Geri Al

Proje Aç

Projeyi Kaydet

Veritabanı Ekle

Veritabanı Kapat

Veritabanı Yapısı

Veriyi Görüntüle

Pragmaları Düzenle

SQL kodunu yürüt

Tablo:

notebook_notebook

III. YALANCI KOD

- import product.
- veritabanına bağlanmak için ilgili kütüphane ekle.
- import random
- from xml.sax.xmlreader import AttributesImpl.
- import numpy as np
- import requests.
- from bs4 import BeautifulSoup.
- import sqlite3.
- bağlan database.
- tanımla pages.
- tanımla liste
- tanımla requests.
- tanımla soup.
- tanımla products.
- tanımla headers.
- oluştur döngü.
- tanımla puan.
- tanımla flink.
- tanımla ptitle.
- tanımla a.
- tanımla price.
- tanımla detail.
- parçala detail.
- tanımla tdetails
- Constructor oluştur.
- tanımla name.
- oluştur döngü.
- bastır siteIsmi
- bastır urunLink.
- bastır marka.

- aktar liste[].
- oluştur try-except.
- bastır urunModel.
- aktar liste[].
- bastır urunPuan.
- aktar liste[].
- oluştur try-except.
- bastır urunFiyat.
- aktar liste[].
- oluştur döngü.
- tanımla cart.
- tanımla curt.
- oluştur iç içe if bloğu.
- gereksiz bilgileri sil.
- aktar liste[].
- bastır cart-curt
- database tanımla.
- tanımla sql.
- tanımla sayac.
- tanımla flag.
- oluştur for döngüsü.
- tanımla if.
- bastır bağlantı başarılı.
- oluştur try-except.
- bağlan database.
- tanımla degerler.
- tanımla cur:
- execute cur.
- tanımla con.
- commit con.
- flag = false.

IV. DENEYSEL SONUÇLAR

Yapılan işlemler ve gerçekleştirilen algoritmalar sonucunda ortaya çıkan sonuçlar şu şekildedir :

- Metodların düzenlice işlenmesi için kullanılan tabloların classları oluşturulması fark edildi.
- Veritabanına sürekli bağlanma gereği duyulduğunda veritabanı bağlama işlemleri ayrı bir classta oluşturuldu ve gerektiğinde çağrıldı.
- Yapılan işlemler sayesinde isterlerin gerçekleşmesi sağlandı.

- <https://www.youtube.com/watch?v=qW92hLp2VLcllist=PLzYuDcHZcoZfeAnad>

- <https://www.youtube.com/watch?v=tRSHp8jsZdwt=162s>

- <https://www.youtube.com/watch?v=ZHSXaFfx84I>

- <https://www.youtube.com/watch?v=T4EXSBMicBYlist=PL>

V. SONUÇ

Yapılan işlemler sonucunda bizden istenilen web scraping başarılı bir şekilde tamamlanmıştır ancak veritabanına aktarma sırasında veri düzensizliğinden kaynaklı olarak hatalar alınmaktadır. Veritabanında yaşanan düzensizlik sebebiyle web sitesinde de bilgiler isterlerde istenen şekilde gösterilememektedir.

VI. KAYNAKÇA

- <https://www.udemy.com/course/python-django-dersleri/>

- <https://www.youtube.com/watch?v=hbx39adciact=1326s>

- <https://www.sadikturan.com/python-dersleri>

- <https://www.sadikturan.com/python-django-dersleri>

- <https://www.djangoproject.com/>

- <https://www.youtube.com/watch?v=QAVoXJtQ9QM>