

**Kırıkkale Üniversitesi**

**Bilgisayar Mühendisliği Bölümü**

**Proje 1 Rapor**

**Ders Adi**: Proje 1

**Ders Hocası**: Bülent Gürsel Emiroğlu

**Proje Adı**: Duygu Analizi

**Hazırlayanlar**

Nurullah Cirit 170205041

Soner Kılınç 170205046

**İçindekiler**

**1. Projenin Amacı…………………………………………………………………………...2**

**2.Veri Seti İçin Verinin Toplanması………………………………………………….3**

**2.1. N11 sitesinden ürün yorumları alma**

**2.2. Toplanan Verinin Etiketlenmesi(Label)**

**2.3. Verinin Temizlenmesi**

**3.Modelin Seçimi…………………………………………………………………………….11**

**4.Instagram’dan Yorumların Alınması……………………………………………..14**

**5.Web Sayfası ile Etkileşim………………………………………………………………15**

**Kaynakça…………………………………………………………………………………………29**

**1.Projenin Amacı**

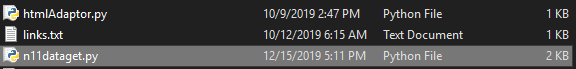
Türkçe bir cümlenin olumlu veya olumsuz olduğunu tespit ederek, bir ürüne, gönderiye veya bir filme gelen yorumların yüzde kaçı olumlu yüzde kaçı olumsuz olduğunu tespit etmektir.

**2.Veri Seti İçin Verinin Toplanması**

Projemizi gerçekleştirmek için öncelikle modelimizin eğitileceği, test edileceği bir veri kümesine(Data Set) ihtiyacımız vardır. Biz bu projeyi gerçekleştirir iken N11.com adlı sitedeki ürünlere yapılan yorumları topladık.

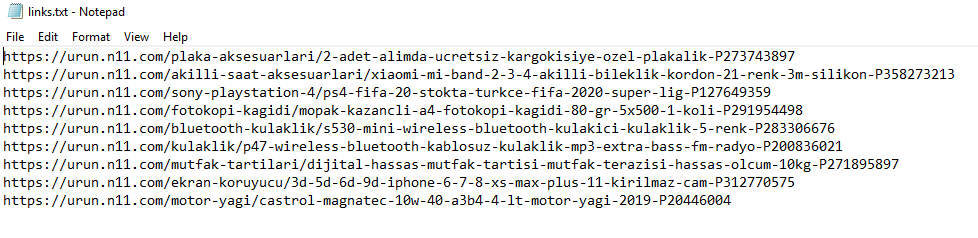
**2.1. N11 sitesinden ürün yorumları alma**

Şekil 2.1.1’de görüldüğü gibi yorum alırken 3 tane ayrı dosyamız var:



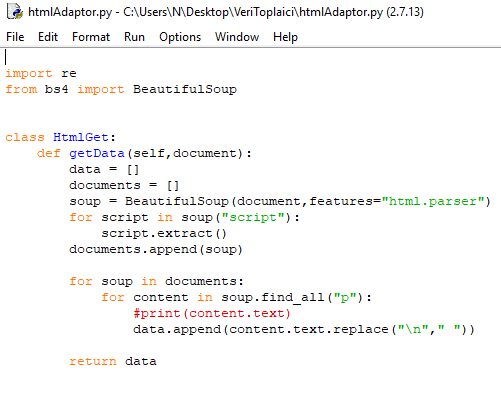
Şekil 2.1.1

**Links.txt** : Bu dosyada ürünlerin linki var. Bu linkleri biz yorumu fazla olan ürünleri göz önüne alarak seçtik. Şekil 2.1.2’de dosyanın içeriği verilmiştir.



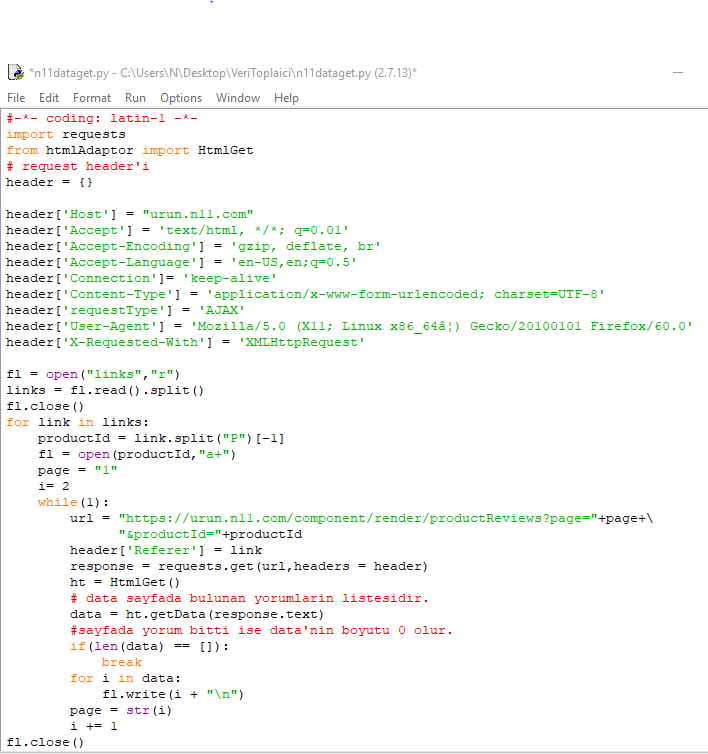
Şekil 2.1.2

**HtmlAdaptor.py**: Şekil 2.1.3’de gösterilen dosyada HTML sayfasını işleyen sınıfımız mevcut bu sınıfta olan metod ile html sayfasındaki yorumları topluyoruz.



Şekil 2.1.3

**n11dataget.py**: Şekil 2.1.4’de içeriği gösterilen dosyamızda ise links dosyasını okuyup linkleri alıp request işlemi gerçekleştiriliyor. Bu request sonucu bir HTML sayfası döndürüyor ve biz bu sayfayı Şekil 2.1.3’de bulunan getData metoduna parameter geçerek o sayfada bulunan yorumlar alıyoruz.N11 sitesinde yorumlar sayfa sayfa olup her sayfada 10 yorum bulunuyor dolayısı ile yorumlar sayfa sayfa çekiliyor.

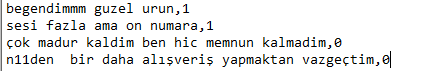


Şekil 2.1. 4

**2.2. Toplanan Verinin Etiketlenmesi(Label)**

Veri toplama işlemini bitirdikten sonra bu verilerin olumlu veya olumsuz olarak etiketlenmesi gereklidir bu işlem uzun sürebilir.

Şekil 2.2.1’e bakarsak olumlu yorumları 1 , olumsuz yorumları 0 olarak etiketledik.



Şekil 2.2. 1-Toplanılan yorumların bir kesiti.

**2.3. Verinin Temizlenmesi**

Modeli eğitirken performans artışı için gereksiz olan verilerin temizlenmesi gereklidir.

***Anlamsız Kelimeler (Stop Words)*:**Bu kelimeler cümlenin anlamına bir etkisi olmayan kelimelerdir dolayısı ile bu kelimelerin veriden çıkarılması gereklidir. Örneğin Şekil 2.3.1’de bulunan kelimeler.

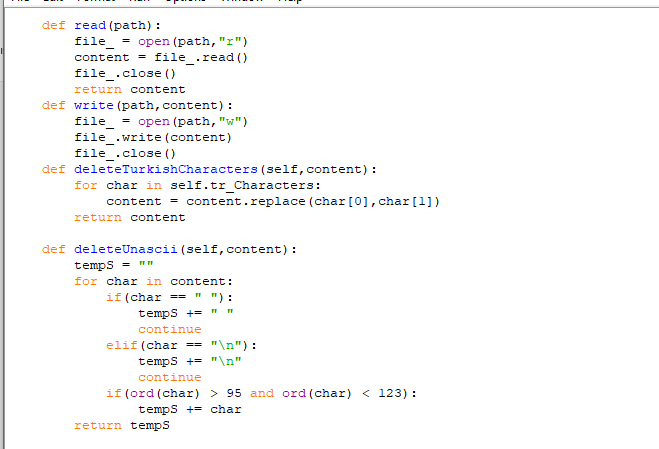


Şekil 2.3. 1-Anlamsız kelimeler dosyasının bir kesiti.

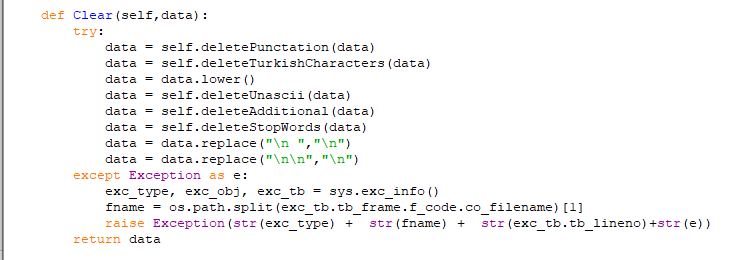
Veri temizleme kısmı için, içeriği Şekil 2.3.2’de verilen bir DataCleaner adlı sınıf oluşturduk ve bu sınıfta veri temizleme ile ilgili metodları belirledik.

**DataCleaner** **Sınıfı:** 

Şekil 2.3. 2 – DataCleaner Sınıfı



Şekil 2.3. 3-DataCleaner Sınıfı Devamı



Şekil 2.3. 4 - DataCleaner Sınıfı Devamı

DataCleaner sınıfımızda Clear adlı metod ile tüm veriler temizlenir ve yapılan işlemler sırası ile;

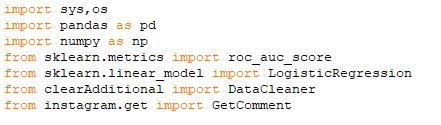
1. Veriden noktalama işaretlerini siliyoruz
2. Türkçe karakterleri asci eşdeğerleri ile değiştiriyoruz
3. Büyük harfleri küçültüyoruz
4. ASCII olmayan karakterleri siliyoruz.(Emoji vb.)
5. deleteAdditional metodu ile gereksiz boşlukları siliyoruz.
6. Gereksiz satırları siliyoruz.

Bu İşlemlerden sonra verimiz işlenebilir duruma geliyor.

**3.Modelin Seçimi**

Modelimizi seçereken; bir yorumun olumlu yada olumsuz mu olduğuna karar vermemiz gerekir bu sebeple problemimiz bir sınıflandırma problemidir.Dolayısı ile Logistic Regression modelini kullanarak problemimizi çözeceğiz. Ve modelin input’u olarak veri setimizde geçen tekli , çiftli ve üçlü kelime gruplarını kullanacağız.

**Main.py:**



Şekil 3. 1 – Main.py’de import edilen kütüphane/sınıflar

**sys,os**: Bu kütüphaneler bir hata oluştuğunda oluşan hatanın hangi

satırda olduğunu yakalamak ve program çalıştırılırken girilen parametreleri yakalamak için eklenmiştir.

**pandas:** Bu kütüphane veri setimizi okumak içindir.

**Roc\_auc\_score**: Bu sınıf ile doğruluk oranı ölçülmüştür.

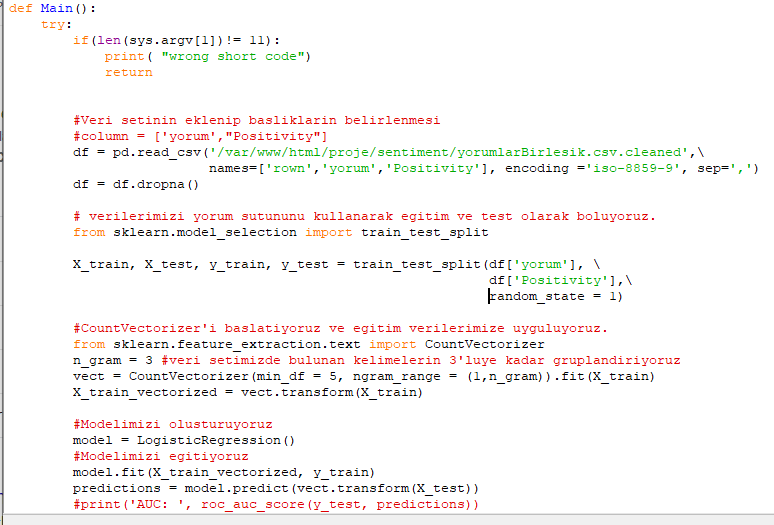
**LogisticRegression**: Bu sınıf modelimizin sınıfıdır.

**DataCleaner**: Bu sınıf instagramdan aldığımız yorumların

temizlenmesi için eklenmiştir.

**GetComment**: Bu sınıf instagramdan yorum almak için kullanılmıştır.

İleriki bölümde anlatılacak.



Şekil 3. 2 - main fonksiyonun içeriği



Şekil 3. 3 - main fonksiyonunun içeriğinin devamı

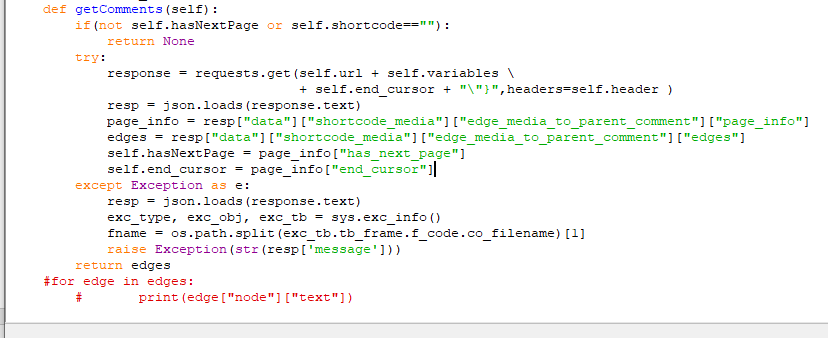
**4.Instagram’dan Yorumların Alınması**

Instagram’dan bir gönderiye ait yorumları almak için o gönderinin sahip olduğu shortcode olarak adlandırılan id’si kullanılmıştır. Şekil 4.1’de gönderiye ait shortcode ile instagram’a request atıyoruz ve yorumları çekiyoruz gelen cevap json formatında olduğu için json kütüphanesini kullanarak gelen cevabı bir dictionary’e çeviriyoruz.

**GetComment Sınıfı**:



Şekil 4. 1 – GetComment sınıfı



Şekil 4. 2 – GetComment sınıfı devamı

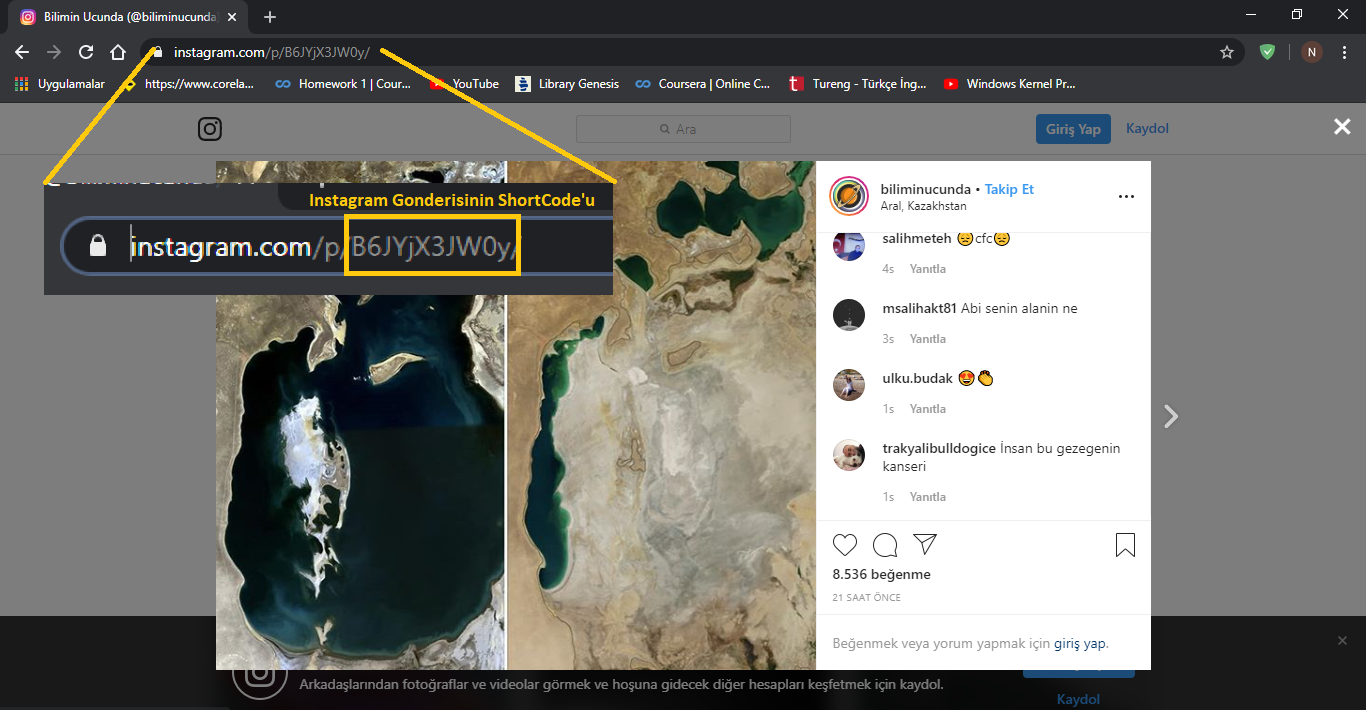
**5.Web Sayfası ile Etkileşim**

Şekil 5.1’de gösterilen uygulamamızın web arayüzüdür ve metin alanı yerine Instagram gonderisinin shortcode’unu giriyoruz.



Şekil 5. 1 – Web Arayüzü

Şimdi örnek olarak bir postun shortcode’unu alalım.



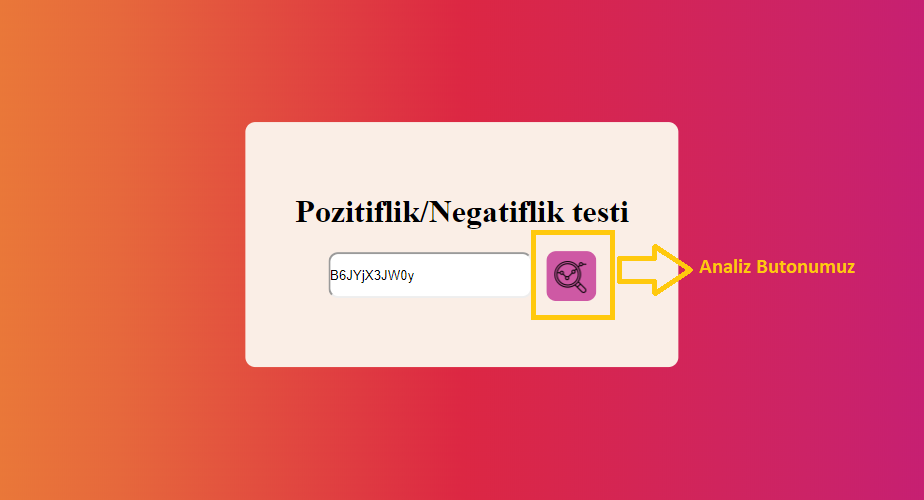
Şekil 5. 2 - Bir instagram gönderisinin shortcode'u

Şekil 5.2’de gösterilen shortcode’u kopyaladıktan sonra sitemize giriyoruz ve girdi yerine yapıştırıyoruz.



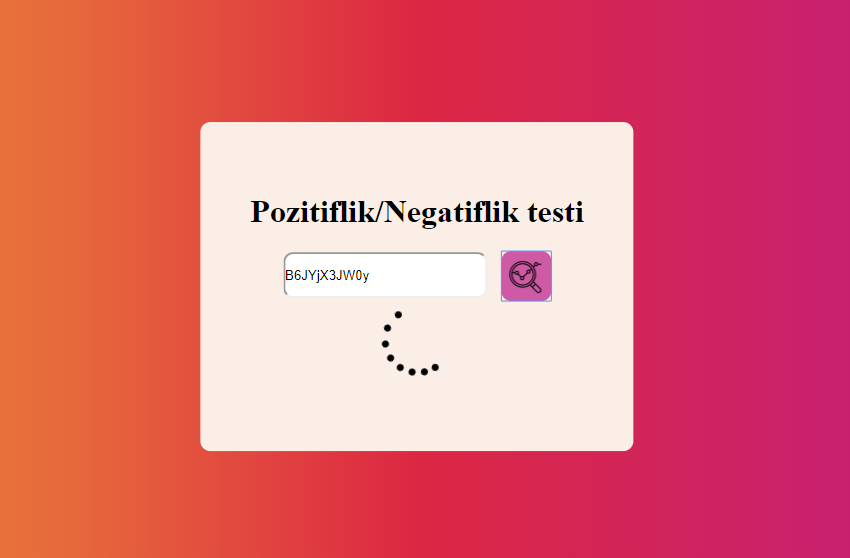
Şekil 5. 3 - Shortcode'un yapıştırıldığı yer

Kodumuzu girdikten sonra Şekil 5.4’de gösterilen resimde analiz butonuna basarak analizimizi başlatıyoruz.



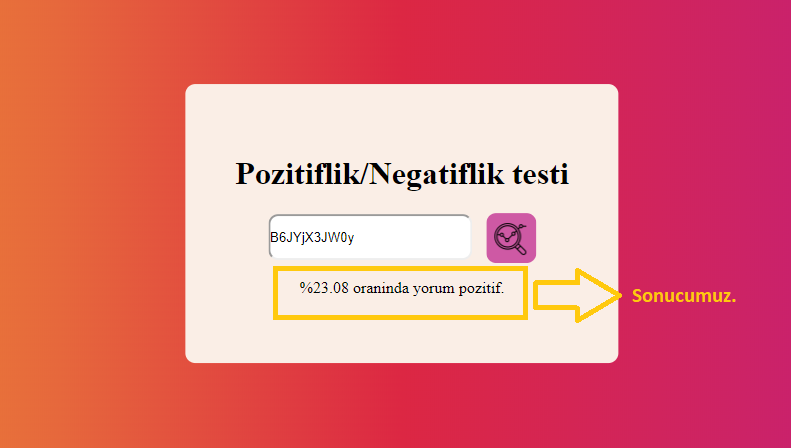
Şekil 5. 4 - Analiz işleminin başlatılması

Analiz işlemimizi gerçekleştirdikten sonra işlemimizin başladığını Şekil 5.5’de görüyoruz.



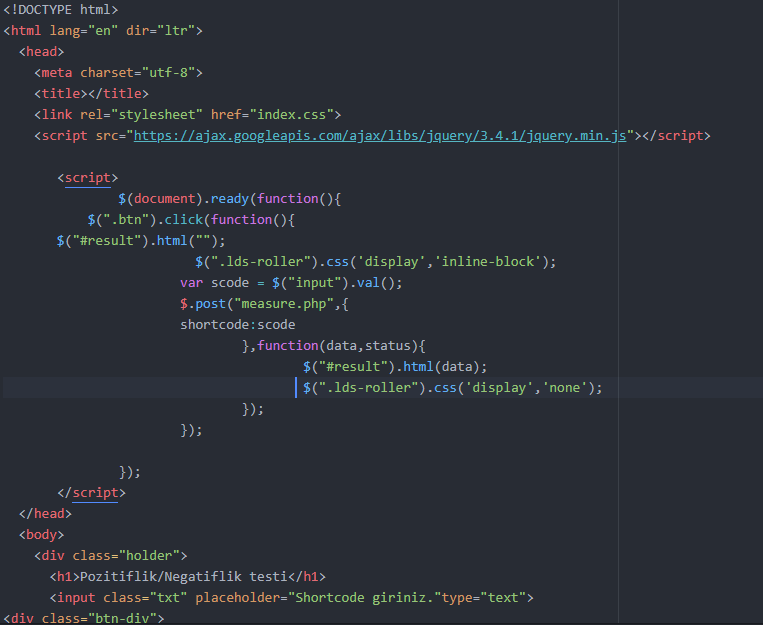
Şekil 5. 5

Son olarak sonucumuzu çıktı olarak Şekil 5.6’da görüyoruz.

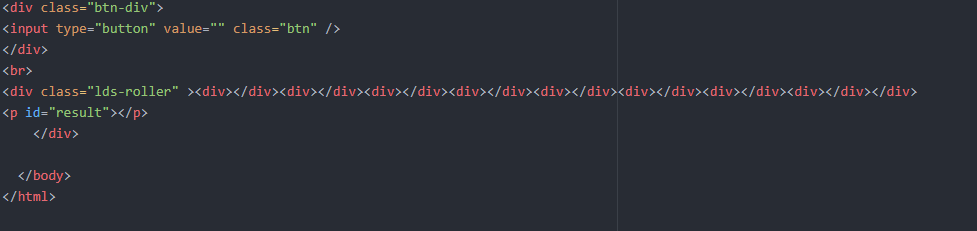


Şekil 5. 6

**Index.php:** Web arayüzümüzün HTML iskeleti buradadır ve yukarıda gördüğümüz analiz butonun fonksiyonu tanımlanmış olup burada jquery’nin asenkron post metodu ile shortcode server’a aktarılmıştır.

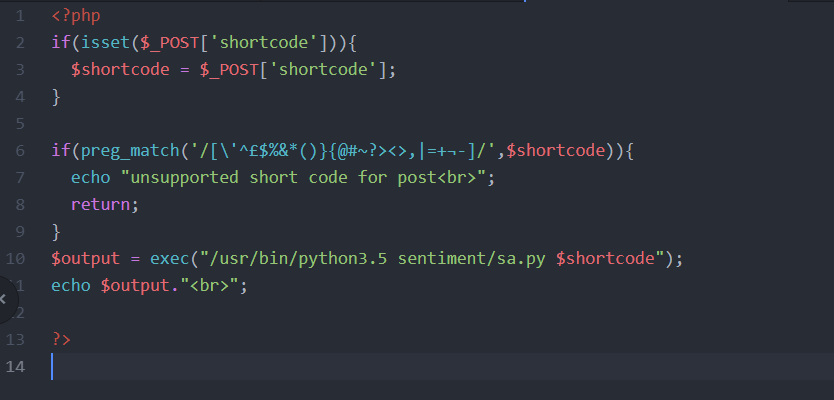


Şekil 5. 7 - index.php



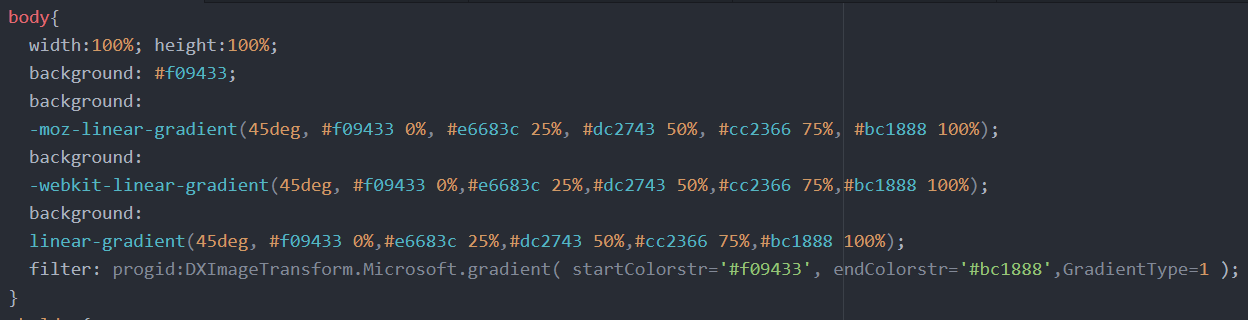
Şekil 5. 8 - index.php devamı

**Measure.php:** İçeriği Şekil 5.9’da gösterilen dosyadaki php kodu ile python’da çalışan uygulamamızı web sayfamıza bağlıyoruz ve Instagram gönderisinin shortcode’unu sistem parametresi olarak geçiyoruz .



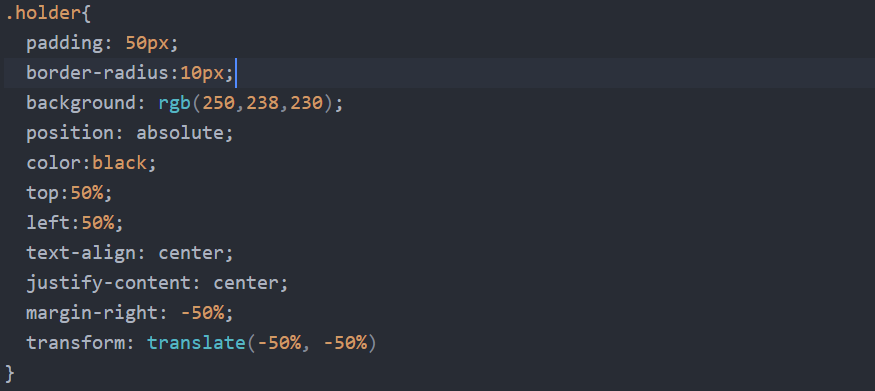
Şekil 5. 9

**Index.css**:



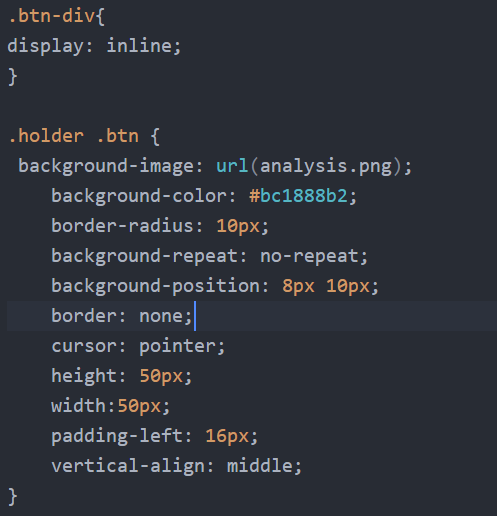
Şekil 5. 10

Şekil 5.10’da gösterilen css kodu ile web sayfasının arka planını Instagram logosunun rengi ile uyumlu hale getirdik.

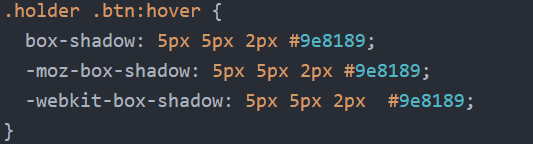


Şekil 5. 11

Şekil 5.11’de başlık,girdi ve buton’u tutan, holder sınıfına sahip bir div elemanını Şekil 5.3’de görüldüğü gibi sayfada ortaladık.

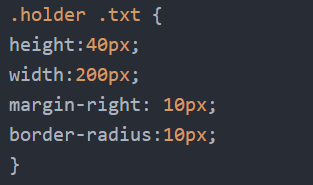


Şekil 5. 12



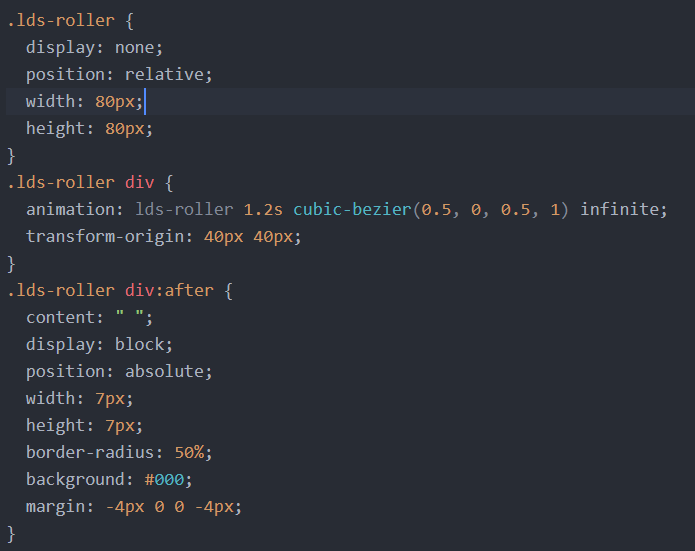
Şekil 5. 13

Şekil 5.12 ve Şekil 5.13’de css kodu verilen analiz butonunun girdi ile aynı satırda olunması sağlanıp butonun stili özelleştirilmiştir.

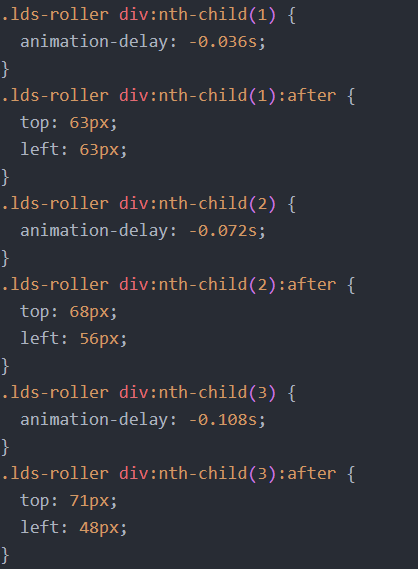


Şekil 5. 14

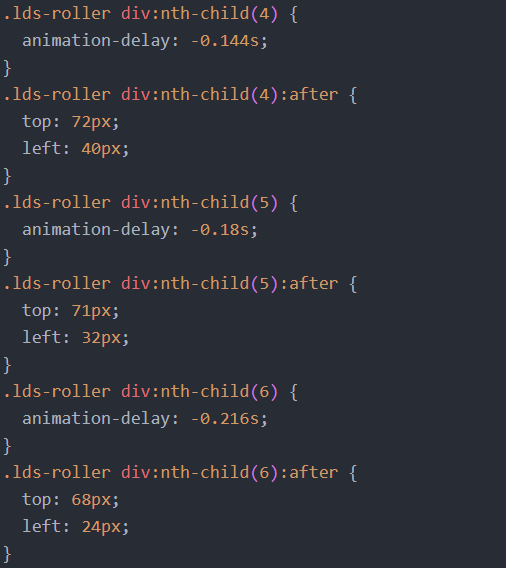
Şekil 5.14’te shortcode’u girdiğimiz alanın özellikleri belirtilmiştir.



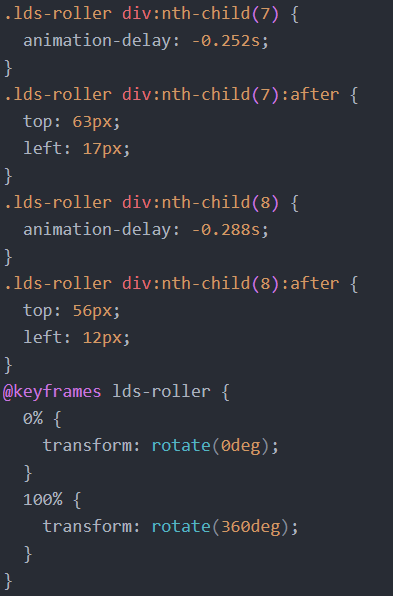
Şekil 5. 15



Şekil 5. 16



Şekil 5. 17



Şekil 5. 18

Şekil 5.15 , 5.16 , 5.17 ve 5.18’de ise analiz butonuna basıldıktan sonra oluşan yüklenme animasyonunun css kodudur.

**KAYNAKÇA**

https://www.twilio.com/blog/2017/12/sentiment-analysis-scikit-learn.html

https://realpython.com/python-requests/