

KOCAELİ ÜNİVERSİTESİ BİLGİSAYAR MÜHENDİSLİĞİ YAZILIM LAB. 2- 1. Proje

Seda Nur Ekici-200201050
Begüm Erva Şahin-200201020

• Özet

Bu rapor Yazılım Laboratuvarı 2 Dersinin 1.projesini açıklamak ve sunumunu gerçekleştirmek amacıyla oluşturulmuştur. Raporda projenin tanımı, isterleri, yapım aşaması kullanılan araç ve yöntemler, kod parçacıkları vb. Bulunmaktadır. Proje aşamasında yararlanılan kaynaklar raporun son bölümünde bulunmaktadır.

• Giriş

Bu projede bizden istenen kitap, dergi vb. gibi metinlerden elde edilen cümleler/kelimeler arasındaki benzerlik, frekans gibi özelliklerin bulunması, metinlerin birleştirip birleştirilmeyeceğine karar verilmesi ve elde edilen verilerden karakter tabanlı cümlelerin/kelimelerin birleştirilmesi beklenmektedir.

• İlerleyiş ve Yöntem

1. Başlamadan Önce

Bu projeye başlamadan önce proje dokümanında verilen bilgileri derleyip kullanabileceğimiz metodları araştırdık.

Kütüphaneden ve internet üzerinden yararlanabileceğimi kaynakları araştırdık. Ve javada spring boot üzerinden yapmaya karar verdik.

2. Başlangıç

Öncelikle <https://start.spring.io/> adresinden bir spring projesi temeli oluşturduk. Dili java seçtik ve kullanmak istediğimiz dependencies leri ekledik (lombok – mongodb – Spring Boot devtools – Spring web – Thymeleaf).

İndirdiğimiz projeyi eclipse uygulamasında açtık . Algoritma kısmını oluşturmak için model paketinin altında book adlı bir klas oluşturduk ve web arayüzünden alıcak kelimeleri ve süreyi tanımladık.

```
2
3
4* import java.util.ArrayList;
11
12 @Getter
13 @Setter
14 @ToString
15 @Document(collection="Seva")
16 public class Book {
17
18     private double sure;;
19     private String kelime1;
20     private String kelime2;
21     // private String son;
22     private String birlesmisSon;
23 /*
24 public String Total()
25 {
26     birlesmisSon=kelime1+kelime2;
27     return birlesmisSon;
28 }
29 */
30* public String Total() {
197
198
199 }
```

Arayüz kısmı için templates klasörü altında iki tane html uzantılı dosya oluşturduk. İlk ekranda kullanıcıdan alınacak kelimeler için text boxlar oluşturduk ve birleşme algoritmasına gitmesi için birleştir butonu tanımladık. Arka kısımda ise bir controller classı açıp verileri alıp ikinci ekrana dönmesi için bir method yazdık. İkinci ekranda ise birleştirme kodundan dönen birleşmiş kelime için bir text book ve algoritmanın Süresi için bir text book oluşturduk. Ayrıca bir kaydet butonu koyduk. Bu buton verileri mongodb ye kaydetmekte. Arka planda ise controoler classında save işlemi için methodunu tanımladık.

Ara yüzde iyileştirme yapmak için ise bootstrap ve css den yardım aldık.

Mongodb kısmında ise bir proje oluşturduk. İçine seva adında collection başlattık. Spring projesinde ise aplication.properties kısmında gerekli yönlendirmeleri yaptık. Repo aldı paketin Bookrepo interface inde ise mongo db yi exthens ettik. Book classında collection name i de belirttikten sonra data kısmını da tamamlamış olduk

3. İLERLEYİŞ

Algoritma kısmı için ise öncelikle girilen değerleri split() metoduyla boşluklardan kelime kelime olacak şekilde parçaladık ve her kelimeyi dizilere attık. Eğer giriş yapılmadıysa ekrana ve mongoya mesaj gönderdik.

Eğer girilen değerleri parçalayıp atadığımız dizilerin ikisin de uzunluğu bir ise tek kelimedenden oluştuğu için karakter karakter karşılaştırarak benzerlik durumunu inceledik. Girilen iki kelimeyi de dizilere aktardık ve iç içe for döngüleriyle eşit olan karakterleri dizi3 adında String tipinde ArrayList te sakladık. Dizi3 ün elemanlarını toString() metoduyla son adındaki String değişkeninde sakladık. Girilen ilk kelimedeki ortak olmayan karakterleri StringBuilder ile sbs adında bir değişkende sakladık ve String tipindeki kelime1son değişkenine atadık. Aynı şekilde girilen ikinci kelimedeki ortak olmayan karakterleri de kelime2son değişkenine atadık. Eğer ortak karakter yoksa ekrana ve mongoya “2 kelimedede ortak karakter yok ” şeklinde mesaj gönderdik. Girilen iki kelimedede ortak karakterler varsa birlesmisSon değişkenininde öncelikle ortak olan karakterleri sonrasında kelime1 de ortak olmayan karakterleri daha sonrasında ise kelime2 de ortak olmayanları birleştirirerek atadık. Eğer girilen değerler cümlelerden oluşuyorsa bu cümlelerdeki kelimeleri dizilere atamıştık. Bu iki dizedeki ortak kelimeleri bulmak için iç içe for dongüsüyle kelimeleri tek tek inceledik ve ortak kelimeleri kelime3 ArrayList e atadık. Eğer girilen iki cümlede ortak kelime yoksa ekrana ve mongoya mesaj gönderdik. Girilen ilk kelimedede ortak olmayan kelimeleri iç içe for dongüsü yardımıyla contains () ve replace ()

metodları yardımıyla kelime1parcala adındaki ArrayList e atadık. Ortak olmayan kelimeleri kelime1ortakolmayanlar adındaki String değişkeninde sakladık. Aynı şekilde girilen ikinci cümlede ortak olmayan kelimeleri de kelime2parcala ArrayList e atadık ve sonrasında kelime2ortakolmayanlar adındaki String değişkenine atadık. Girilen iki cümlede ortak olan kelimeleri cumleString adındaki String değişkenine atadık. En son girilen iki cümleyi birleştirirken öncelikle girilen ilk cümledeki ortak olmayan kelimeleri yazdırdık sonrasında iki cümledeki ortak kelimeleri yazdırdık ve en son da girilen ikinci cümledeki ortak olmayan kelimeleri yazdırdık. Bu iki cümlelerin birleşmiş halini de birlesmisSon değişkenine atadık.

• Arayüz fotoğrafları

SEVA

1.Metin

2.Metin

Birleştir

SEVA

Birleşik Metin :

Süre:

Kaydet

Birleşik Metin :

```
{ "_id": ObjectId("64idf2f8bcc08c3ab1e7b797"),  
  "sure": 0.0003209,  
  "kelime1": "asdfgh",  
  "kelime2": "kk",  
  "birlesmisSon": "ORTAK KARAKTER YOK -> asdfgh kk",  
  "_class": "com.example.mongodb.model.Book"
```

SEVA

Birleşik Metin :

Süre:

Kaydet

SEVA

1.Metin

2.Metin

Birleştir

```
{  
  {  
    {  
      {  
        {  
          {  
            {  
              {  
                {  
                  {  
                    {  
                      {  
                        {  
                          {  
                        {  
                      {  
                    {  
                  {  
                {  
              {  
            {  
          {  
        {  
      {  
    {  
  {  
}
```

• Son Söz

Bu proje bize çok fazla bilgi birikimi sağladı ve farklı bir bakış açısı kazandırdı. Projeyi yaparken çok fazla araştırma yapıp eksiklerimizi kısa zamanda tamamladık ve projede bizden istenenleri elimizden gelen en iyi şekilde yapmaya çalıştık. Spring boot projesinde web çalışmayı öğrendik. Algoritma yazmada büyük ölçüde geliştik.

• Kaynakça

- <https://spring.io/>
- <https://www.javatpoint.com/spring-boot-tutorial>
- <https://www.yusufsezer.com.tr/spring-boot/>
- <https://www.mongodb.com/compatibility/spring-boot>
- <https://www.digitalocean.com/community/tutorials/spring-boot-mongodb>

- <https://www.devglan.com/spring-boot/spring-boot-mongodb-configuration>
- <https://getbootstrap.com/>
- <https://spring.io/guides/gs/serving-web-content/>
- <https://www.geeksforgeeks.org/spring-boot-thymeleaf-with-example/>

