

Refactoring, Spagetti Koddan Nasıl Kurtulurum?

Anıl Özselgin

Spagetti Kodun Özellikleri

- Global değişkenler içerir.
- Herhangi bir fonksiyon içinde bulunmayan kodlar vardır.
- Tutarlı bir kod stili yoktur. Boşluklar, girintiler, isimlendirmeler farklılık gösterir.
- Fonksiyonlar çok uzundur ve bir çok iş yaparlar.
- Kod organizasyonu yoktur.

Yanlış Bilinenler

- Spagetti kod, prosedürel değildir. Prosedürel programlamada, prosedürler -yani fonksiyonlar- ile organizasyon sağlanır.
- Spagetti kod hızlı değildir. Değişkenler çok geniş bir alanda tanımlı oldukları için çöp toplayıcı (garbage collector) tarafından toplanamazlar. Hafıza (memory) şişer.
- Başlangıçta hızlı yazılım geliştirilir, sonra sizi çıkan hatalardan (bug) dolayı yavaşlatır.

Refactoring Nedir?

- Bir bilgisayar programının çıktılarını ve işlevlerini değiştirmeden iç yapısının yeniden düzenlenerek geliştirilmesidir.
- Ufak adımlarla kodda değişiklik yapılır. Her değişiklikten sonra sistemin düzgün çalıştığı test edilir.

Refactoring Örneği

- Refactor edeceğimiz kod: e-ticaret sitesi yorum sayfası
- İlgili iki tablo bulunuyor: urunler ve yorumlar
- Tek bir dosya içerisinde ilgili bütün işlemler yapılıyor.
- İlk önce fonksiyonlara böleceğiz.
- Sonra fonksiyonları sınıflara taşıyacağız.
- Veri katmanı, iş katmanı ve görsel katmanı ayıracağız.

Yorum Sayfası



localhost/yorumlar.php?id=1

Arçelik Nofrost 5847 NFY

No	Başlık	Yorum
1	Biraz ufak	Hacmi ufak bence
2	Çok kullanışlı	Ben çok memnunum, 2 aydır kullanıyorum. Kaynıma da aynısından aldık.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
  <link href="bootstrap.min.css" rel="stylesheet">
  <title>Ürün Yorumları</title>
</head>
<body>
<div class="container">
  <?php
    mysql_connect("localhost", "root", "") or die ("Veritabanına bağlanırken bir hata oluştu!" . mysql_error());
    mysql_select_db("site") or die("Veritabanında bir hata oluştu!" . mysql_error());
    mysql_query("SET NAMES UTF8");
    $urun_id = $_GET['id'];
    $result = mysql_query("SELECT ad FROM urunler WHERE urunler.id=\"\$urun_id\"");
    $row = mysql_fetch_row($result);
    echo "<h1>{$row[0]}</h1>";
    ?>
    <table class="table table-striped">
      <tr><th>No</th><th>Başlık</th><th>Yorum</th></tr>
      <?php
        $result = mysql_query("SELECT yorumlar.*, urunler.ad FROM yorumlar, urunler
                                WHERE yorumlar.urun = urunler.id AND urunler.id=\"\$urun_id\"");
        while($arr = mysql_fetch_array($result))
        {
          echo "<tr><td>{$arr['id']}</td><td>{$arr['baslik']}</td><td>{$arr['icerik']}</td></tr>";
        }
        ?>
      </table>
    </div>
  </body>
</html>
```

Fonksiyonlara Taşıma

2. Version

Eski kod:

```
mysql_connect("localhost", "root", "") or die ("Veritabanına bağlanırken bir hata oluştu!" . mysql_error());  
mysql_select_db("site") or die("Veritabanında bir hata oluştu!" . mysql_error());  
mysql_query("SET NAMES UTF8");
```

Yeni kod:

```
function veritabani_ayarlarini_yap(){  
    mysql_connect("localhost", "root", "") or die ("Veritabanına bağlanırken bir hata oluştu!" . mysql_error());  
    mysql_select_db("site") or die("Veritabanında bir hata oluştu!" . mysql_error());  
    mysql_query("SET NAMES UTF8");  
}  
  
veritabani_ayarlarini_yap();
```


Fonksiyonlara Taşıma

3. Version

Eski kod:

```
$result = mysql_query("SELECT ad FROM urunler WHERE urunler.id=\"$urun_id\"");  
$row = mysql_fetch_row($result);  
echo "<h1>{$row[0]}</h1>";
```

Yeni kod:

```
function urun_adi_al($urun_id){  
    $result = mysql_query("SELECT ad FROM urunler WHERE urunler.id=\"$urun_id\"");  
    $row = mysql_fetch_row($result);  
    return $row[0];  
}  
  
$urun_adi = urun_adi_al($urun_id);  
echo "<h1>$urun_adi</h1>";
```

Fonksiyonlara Taşıma

4. Version

Eski kod:

```
$result = mysql_query("SELECT yorumlar.*, urunler.ad FROM yorumlar, urunler
                        WHERE yorumlar.urun = urunler.id AND urunler.id=\"$urun_id\"");
while($arr = mysql_fetch_array($result))
{
    echo "<tr><td>{$arr['id']}</td><td>{$arr['baslik']}</td><td>{$arr['icerik']}</td></tr>";
}
```

Yeni kod:

```
function yorumlari_getir($urun_id){
    $result = mysql_query("SELECT yorumlar.*, urunler.ad FROM yorumlar, urunler
                            WHERE yorumlar.urun = urunler.id AND urunler.id=\"$urun_id\"");

    $yorumlar = array();
    while($yorum = mysql_fetch_array($result)){
        $yorumlar[] = $yorum;
    }
    return $yorumlar;
}
```

Fonksiyon Taşıma

5. Version

Eski kod:

```
$urun_id = $_GET['id'];  
$urun_adi = urun_adi_al($urun_id);  
echo "<h1>$urun_adi</h1>";
```

Yeni kod:

```
//sayfayı oluşturan ana fonksiyon  
function yorumlar_sayfasi(){  
    global $degerler;  
  
    veritabani_ayarlarini_yap();  
    $urun_id = $_GET['id'];  
    $degerler["urun_adi"] = urun_adi_al($urun_id);  
    $degerler["yorumlar"] = yorumlari_getir($urun_id);  
}  
  
yorumlar_sayfasi();
```

Html Kodlarını Ayırmak

- Html içerisindeki yoğun php kullanımını kaldırdık şu anda.
- Html içerisindeki php kodları sadece basit kontroller ve döngüleri için bulunuyor artık.

Html Kodlarını Ayırmak

6. Version

- Bir sonraki aşamada html kodlarını yorumlar_template.php dosyasının içine alıyoruz.

Sınıfa Taşıma

7. Version

- yorumlar.php dosyamızdaki kodlarımızı
yorumlar_sayfasi sınıfı oluşturup içine taşıyoruz

Front Controller Pattern'i

8. Version

- index.php dosyası oluşturuyoruz
- baslatici sınıfını index.php içine koyuyoruz. artık bütün gelen istekler buradan sayfalara dağıtılacak.
- ayarlar sınıfını oluşturup veritabani_ayarlarini_yap metodunu buraya taşıyoruz.
- index.php?sayfa=yorumlar&id=1 şeklinde istek yapılmalı artık

Veri Katmanını Ayırmak

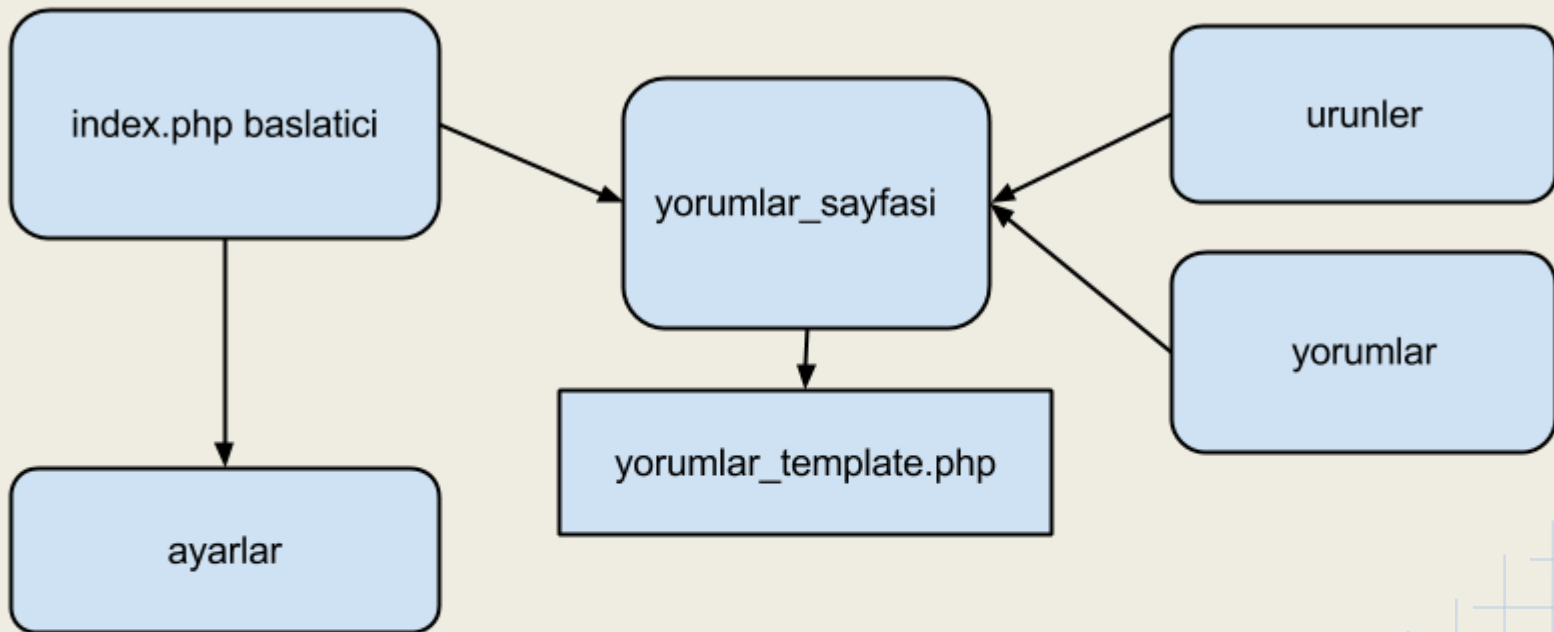
9. Version

- ürünler sınıfı oluşturup `urun_adi_al` metodunu taşıyoruz.
- yorumlar sınıfı oluşturup `yorumlari_getir` metodunu taşıyoruz.

Uygulamanın Son Hali

Uygulamanın ana sınıfı

Veri Katmanına erişen sınıflar



Sonra...

- Veri katmanından verileri array olarak aldık. Urun ve Yorum objeleri olusturup, bunlari array icinde donmemiz gerekir.
- Yorumlar sayfası direk olarak veri katmanına ulaşılıyor. Araya servis katmanı koymak gerekiyor.
- veritabanı refactoring yapılabilir. urunler tablosundaki id kolonu urun_id, yorumlar tablosundaki id kolonu yorum_id yapılmalı.



Sorular?