**Sunan:**

**Ömer Faruk Çelik**

İçindekiler

[Proje Sunumu : 4](#_Toc42947728)

[Kullanıcı Giriş Bilgileri: 4](#_Toc42947729)

[Projenin tanımı 4](#_Toc42947730)

[Analiz : 4](#_Toc42947731)

[Eczane Yönetim Sistemi Nasıl Çalışır : 4](#_Toc42947732)

[Eczane Yönetim Sistemi Nedir : 5](#_Toc42947733)

[Eczane Yönetim Paneli Sisteminin Hedefleri Nelerdir : 5](#_Toc42947734)

[Ne Tür İşler Yapılacak : 5](#_Toc42947735)

[Eczane yönetim sistemi kimler içindir: 5](#_Toc42947736)

[Haftaların Özeti 6](#_Toc42947737)

[1 HAFTA 6](#_Toc42947738)

[Programlama Araçlarının Belirlenmesi : 6](#_Toc42947739)

[Veri Tabanı ve Tasarımı Belirlenmesi: 6](#_Toc42947740)

[Kullanılacak Editörlerin Belirlenmesi : 6](#_Toc42947741)

[2 HAFTA 6](#_Toc42947742)

[Web Site Dosyalarının Oluşturulması ve Html Siteyi PHP Uyarlamak : 6](#_Toc42947743)

[3. HAFTA 8](#_Toc42947744)

[Veritabanı ve Veritabanı Bağlantısı Oluşturmak : 8](#_Toc42947745)

[4.HAFTA 10](#_Toc42947746)

[Kullanıcı Girişi İşlemleri Yapmak : 10](#_Toc42947747)

[5. HAFTA 12](#_Toc42947748)

[Veri Tabanı Ilac Tablo ve Tablo Yapısı Oluşturmak : 12](#_Toc42947749)

[6.HAFTA 14](#_Toc42947750)

[Veri Tabanı Siparis Tablo ve Tablo Yapısı Oluşturmak : 14](#_Toc42947751)

[7. HAFTA 17](#_Toc42947752)

[CKEDİTOR NEDİR? NASIL KULLANILIR : 17](#_Toc42947753)

[CKEditor Ne İşe Yarar : 17](#_Toc42947754)

[CKEDİTÖR Örnek Kodlama : 17](#_Toc42947755)

[8.HAFTA 18](#_Toc42947756)

[İçeriklerin Düzenlenmesi Ve İçeriklerin Silinmesi : 18](#_Toc42947757)

[Sistem Gerekli Testlerden Geçirildi : 20](#_Toc42947758)

[Yazılımdan Ekran Görüntüleri 21](#_Toc42947759)

**Şekiller**

[Şekil 1 : Oluşturduğumuz PHP uzantılı dosyalarımız... 7](#_Toc42947711)

[Şekil 2 : Veritabanı tabloları gösterimi. 9](#_Toc42947712)

[Şekil 3 : Ayarlar Tablosu Gösterimi... 9](#_Toc42947713)

[Şekil 4 : Kullanıcı Tablosu Gösterimi... 10](#_Toc42947714)

[Şekil 5 : Veri tabanı ilac tablo yapısı görünümü... 12](#_Toc42947715)

[Şekil 6 : İlaçlar sayfası için veri kaydetme işlemi tamamlandı... 14](#_Toc42947716)

[Şekil 7 : Sipariş Kayıt Test İşlemi... 16](#_Toc42947717)

[Şekil 8 : Siparişler Sayfası Veri Kayıt Test İşlemi Tamamlandı... 17](#_Toc42947718)

[Şekil 9 : Nöbetçi Eczane Ekle Sayfası Görünümü... 21](#_Toc42947719)

[Şekil 10 : Siparişler Sayfası Görünümü... 22](#_Toc42947720)

[Şekil 11 : İlaçlar Sayfası Görünümü... 23](#_Toc42947721)

[Şekil 12 : İlaç Düzenleme Sayfası Görünümü... 24](#_Toc42947722)

[Şekil 13 : İlaç Düzenleme İçeriklerin Değiştirilmesi... 25](#_Toc42947723)

[Şekil 14 : İlaç Düzenleme İçeriklerin Değişti... 26](#_Toc42947724)

[Şekil 15 : Nöbetçi Eczane Sayfası Görünümü... 27](#_Toc42947725)

[Şekil 16 : Nöbetçi Eczaneler Sayfası Yeni Görünümü... 28](#_Toc42947726)

[Şekil 17 : Nöbetçi Eczane Düzenleme Sayfası... 29](#_Toc42947727)

# Proje Sunumu :

Projenin Gösterimi <http://alotimes.com/giris.php> adresinden bakabilirsiniz.

## Kullanıcı Giriş Bilgileri:

e-posta: vjraylex@gmail.com

Şifre: 15341534

# Projenin tanımı

Eczanelerimizin daha etkin satış ve stok takibi yapabilmelerini sağlar.

Dolayısı ile Eczacılarımızın karar alma sürelerini kısaltarak daha doğru karar vermelerine olanak tanır.

Örneğin;

* hangi kategoriden, alt kategoriden ya da markadanne kadardık stok taşındığını raporlar ve ne kadar stok bulundurmak gerektiği konusunda etkili sonuçlar üretmelerini kolaylaştırır.
* Bizce eczacılarımız Eczane Yönetim Paneli Sistemi **“endişeden uzak, güvenli bir yönetim”** için kullanmalılar. Çünkü biz biliyoruz ki eczacılarımızın başında fazlasıyla iş var ve bu işler insani ölçüler göz önüne alındığında yönetimi kolay olan işler değil.  
  Dolayısı ile, eğer işimizde (ağırlıklı olarak ilaç dışı kategorilerde) büyüme trendine girmek istiyorsak işimizi ölçmemiz ve bu doğrultuda strateji belirlememiz çok önemli, mottomuz olan “Ölçemediğiniz İşi Yönetemezsiniz” ilkesinden hareketle Eczane Yönetim Paneli Sistemi’nin bir olmazsa olmaz olduğunu düşünüyoruz.  
  Eczacılarımız kendi işlerinin yöneticileri olduğu gerçeğinden yola çıkarsak, bir yöneticinin en temel sorumluluğu işini en verimli şekilde yönetmesidir. Dolayısı ile bu yönetimi sergilerken yardımcıya ihtiyaç duymak ise kaçınılmazdır.
* İşte Eczane Yönetim Paneli Sistemin de tam bu yüzden geliştirildi ve hizmete sunuldu.

# Analiz :

Eczane Yönetim Paneli Sistemin de;

* Satış ve Stok Takibi
* Fiyat ve Bilgilendirme
* Stok Kontrol ve Sayım İşlemleri
* İlaçların Son kullanma tarihleri
* İlaç Bitim Takibi ve Analizleri

## Eczane Yönetim Sistemi Nasıl Çalışır :

Eczanelerin satış ve stok bilgilerini, **eczacılarımızın izni dahilinde eczacılarımızdan**, sadece ürün bazında alarak kendi bünyesinde analizlere tabi tutması yöntemi ile çalışır.

Bununla birlikte, eczacılarımızın için çok büyük bir iş yükü olan ürün kategori eşleşmelerine gerek kalmaz, çünkü Eczane Yönetim Paneli Sistemi içerisinde güçlü bir ürün ağacı bulunmaktadır ve  Eczane Yönetim Paneli Sistemi eşleşmeleri kendi bünyesinde otomatik yapar.

İşte tam bu aşamada, Eczane Yönetim Paneli Sistemi bu sıkıntıyı ortadan kaldırıp doğru ve güvenilir bir ürün veri tabanını bünyesinde barındırmakla kalmaz ayrıca zengin analizlere zemin oluşturarak eczacılarımızın hizmetine sunar.

Ayrıca web tabanlı bir sitemdir, yani evinizin rahatlığında ya da internet erişimi olan her yerde size özel dizayn edilmiş analizlerimize erişebilirsiniz.

## Eczane Yönetim Sistemi Nedir :

Hedeflediğimiz projemiz Web ve mobil tabanlı bir sistem olup, platform bağımsızdır.

Dolayısı ile web teknolojisinin imkanlarından yararlanmak ve eczacılarımızı yararlandırmak ve işlerini kolaylaştırmak maksadı ile geliştirilmiştir.

Ayrıca Eczane Yönetim Paneli Sistemin de;

* Rapor çekmezsiniz hazır grafikler girişte açılır.
* Takip etmeniz gereken her bir kriteri aynı anda görürsünüz.
* Web tabanlıdır, her yerden girebilirsiniz.
* Kullanımı oldukça kolaydır.
* Siz talep etmeden sürekli gelişim felsefesi ile sürekli yeni özellikler eklenir
* Güçlü ve güvenilir altyapısı sayesinde 7/24 hizmet verebilmektedir.

Özetle Eczane Yönetim Paneli Sistemi, yöneticinin sağ koludur ve bir eczane yönetim paneli sistemidir.

# Eczane Yönetim Paneli Sisteminin Hedefleri Nelerdir :

Türkiye'de hizmet veren tüm eczacılarımızın işlerini daha kolay ve seri biçim de kontrol etme ve hata yapma olanaklarını ortadan kaldırmak için amaçlanan bir sistemdir.

## Ne Tür İşler Yapılacak :

* Açık eczaneler listesi ve düzenleme
* İlaç listesi ve düzenleme
* İlaç sipariş etme ve düzenleme
* Personel çalışma saatleri ve programları
* Eczane ekleme ve düzenleme
* Stok takibi ve düzenleme
* Son kullanım tarihi geçmiş ilaçların bildirilmesi ve gerekli işlemlerin yapılması ve bunların sistemde dinamik bir şekilde hazırlanmasını sağlayıp sunmaktır.

## Eczane yönetim sistemi kimler içindir:

Başarı elde etmek isteyen tüm eczane'lerin kesinlikle kullanması gereken bir sistemdir.

# Haftaların Özeti

# 1 HAFTA

Yazılım Kaynakları Belirlemek:

Profesyonel anlamda **web tasarım** işlemi **html** ve **css** programları sayesinde kurulmuş, üzerine canlı dinamik hale getirebilmek için **php**, **asp**, gibi programları ile veri tabanlarını anahtarlarla bağlantılı hale getirmektir. **Web sitesi** **html** ve **css** ile yapılmış olsa da sadece görsellik yeterli olmayıp online işlemler gerçekleştirebilmek için Dinamik tasarımlar yani **html** ve **css**'ler ile birlikte **asp**, **php** ile **sql**, **mysql** vb. veri tabanları kullanılır. Kodlamakta kullanılan dil ile veri tabanına ulaşılarak oradan veri çağırılması ve sunuma getirilmesi için amaçlanır. Kendi projemiz de hangi programlama dillerini kullanacağımızı seçelim şimdi de.

## Programlama Araçlarının Belirlenmesi :

Proje geliştirilmesi aşamasında kullanılması gereken sisteme uygun programlama araçlarını tespit ettik.

Proje İçin Kullanılacak Programlar ve Programlama Dilleri listemiz:

* Notepad ++
* Adobe Photoshop Cs6
* Xampp
* Javascript
* HTML
* CSS
* PHP 7: PDO
* Bootstrap

## Veri Tabanı ve Tasarımı Belirlenmesi:

Projemiz de PHP Teknolojisi kullanacağımız için açık kaynak bir veri tabanı seçimi yapmam gerekecek bunun için de MYSQL kullanacağım. Tasarımını PHPMYADMIN üzerinden gerçekleştireceğim.

**MYSQL** İnternet üzerinde kullanılan en popüler ve en açık kaynak kodlu veri tabanı programı olması nedeniyle en çok kullanılanıdır. Yüksek performansa sahip bir veri tabanıdır. Aşırı yüklenmelerde dahi veri hatası yaşanmaz.

## Kullanılacak Editörlerin Belirlenmesi :

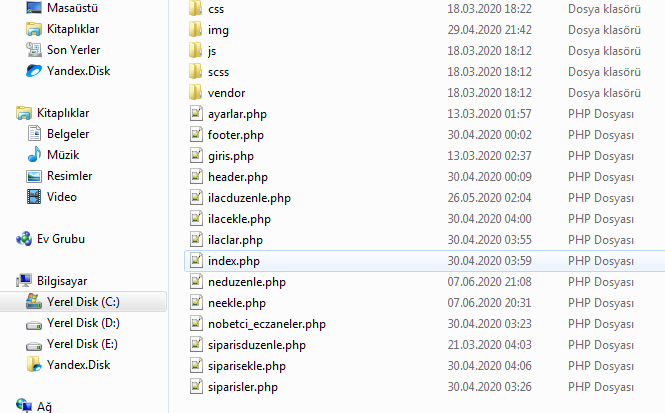
CKEDİTÖR kullanılacaktır.

# 2 HAFTA

## Web Site Dosyalarının Oluşturulması ve Html Siteyi PHP Uyarlamak :

1.hafta'da belirlediğim programlar ile yaptığım çalışmalar neticesinde gözle görünür şekilde bir tema oluşturabildim bu hafta hazırlamış olduğum temayı php uyumlu yapabilmek için parçalama işlemi yapacağım tabi öncesinde php uzantılı tema dosyalarımızı oluşturalım; dosya yolumuz bu şekilde olacaktır "C:\xampp\htdocs\eczane" html, css, js ve img klasörlerimizi htdocs\ klasörleri oluşturmamız veya varsa kaydetmemiz gerekiyor.

* index.php
* header.php
* footer.php
* giris.php
* ayarlar.php
* ilaclar.php
* nobetci\_eczaneler.php
* siparisler.php



Şekil 1 : Oluşturduğumuz PHP uzantılı dosyalarımız...

<?php include 'header.php' ?>

<?php include footer.php' ?>

ve benzeri php dosyaları oluşturmamız gerekiyor yukarıda görüldüğü gibi tema dosyalarımızı php uyumlu olabilmesi için html kodlarımız içerisinde bulunan <head> tagları arasında bulunan kodlarımızı header.php sayfasına include etmemiz gerekiyor, içerikler alanımız yani <body> içerisinde bulunan kodları index.php sayfasına kopyalamamız gerekiyor son olarak anasayfamız da bulunan footer yani alt kısım da bulunan kodları footer.php içerisine include etmemiz gerekiyordu bu işlemleri yaptıktan sonra xampp bağlantılarını aktif ederek localhost adresine giriş yapıp web sitemizin kontrolünü başarılı bir şekilde yapmamız mümkün olacaktır. include komutu aşağıdaki gibi olacaktır.

# 3. HAFTA

## Veritabanı ve Veritabanı Bağlantısı Oluşturmak :

Bu haftaki çalışmamız da geçen hafta yapmış olduğumuz işlerden yani html temayı php uyulumlu olması için parçalayıp ardından veri tabanı bağlantısını aktif edebilmemiz için baglan.php adında bir dosya oluşturdum sonrasında kurs adında veritabanı oluşturdum bu kodları yazdım.

<?php

$host="localhost";

$veritabani\_ismi="kurs";

$kullanici\_adi="root";

$sifre="";

try{

$db=new PDO("mysql:host=$host;dbname=$veritabani\_ismi;charset=utf8",$kullanici\_adi,$sifre);

}

catch (PDOException $e){

echo $e->getMessage();

echo "bağlantı başarısız";

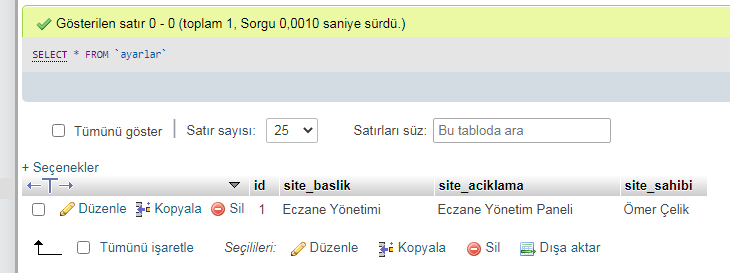
} ?>

ve ardından tarayıcıdan localhost/phpmyadmin adresine gidip kurs adında veritabanı oluşturdum sonrasında site içerinde yapılacak işlerle alakalı tablolar oluşturdum yapacağımız işleri proje detayını ilk sayfada açıklamıştım zaten neler yaptığımıza görsel den devam edelim.

# 2020-06-12_17-30-25.png

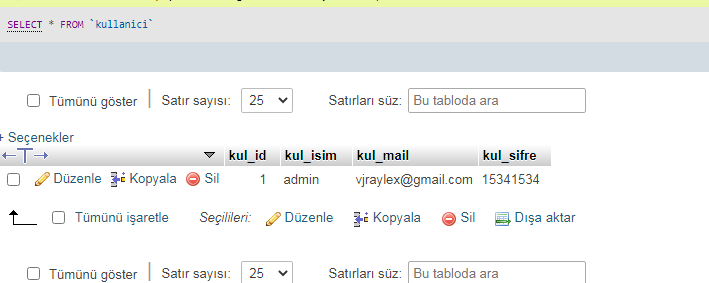
Şekil 2 : Veritabanı tabloları gösterimi.

(Şekil 3) te görüldüğü gibi ayarlar , ilac, kullanici, nobetci\_eczane, proje ve siparis adında tablolarımız mevcut ayarlar tablosun da yapılacak işler (Şekil 4) te görüldüğü gibi site bilgileri yani site başlık , site açıklaması ve site sahip bilgilerinin girileceği yerlerdir.



Şekil 3 : Ayarlar Tablosu Gösterimi...

Yönetici girişi yapılabilmesi için ise kullanıcı adında tablo oluşturduk yapılacak işlem belli olacağı gibi bir kullanıcının site ye girebilmesi için e-mail ve şifre ile giriş yapabilmesi için veriler kaydettik (Şekil 5) bakınız.



Şekil 4 : Kullanıcı Tablosu Gösterimi...

Kullanıcı mail (Şekil 5) gösterildiği gibi bilgilerden oluşuyor bir sonraki hafta oluşturduğumuz tabloları aktif bir şekilde web site içerisin de kullanımını gerçekleştireceğiz.

# 4.HAFTA

## Kullanıcı Girişi İşlemleri Yapmak :

Geçen haftaki çalışmamızdan kullanıcı girişi ile ilgili bilgiler vermiştim ve mysql içerisinde kullanici adında tablo oluşturmuştum bu hafta kullanıcı girişi yapabilmemiz için gereken kodları yazmaya başlayalım ki sitemiz daha güvenli olabilsin ve bizden başka kimse giremesin siteye giris.php dosyamızın içerisine girelim ve kodlarımızı yazmaya başlayalım.

<form class="user" action="islemler/islem.php" method="POST">

<div class="form-group">

<input name="kul\_mail" type="email" class="form-control form-control-user" id="exampleInputEmail" aria-describedby="emailHelp" placeholder="Enter Email Address..."> </div>

<div class="form-group">

<input name="kul\_sifre" type="password" class="form-control form-control-user" id="exampleInputPassword" placeholder="Password"> </div>

<div class="form-group">

<div class="custom-control custom-checkbox small">

<input type="checkbox" class="custom-control-input" id="customCheck">

<label class="custom-control-label" for="customCheck">Remember Me</label></div></div>

<button name="oturumac" type="submit" class="btn text-white btn-primary btn-user btn-block">Oturum Aç</button>

<hr><a href="index.html" class="btn btn-google btn-user btn-block">

<i class="fab fa-google fa-fw"></i> Login with Google</a>

<a href="index.html" class="btn btn-facebook btn-user btn-block"> <i class="fab fa-facebook-f fa-fw"></i> Login with Facebook</a> </form></hr>

Kullanıcı girişi sayfası için form alanımızı oluşturduğumuza göre şimdi php kodlarımızı yazıp sonrasın da <form> içerisine girilen bilgilerin sorgulanması için islemler/islem.php dosyamıza gidip gereken kodları aşağıda anlatacağım.

<?php

include 'islemler/baglan.php';

$ayarsor=$db->prepare("SELECT \* FROM ayarlar");

$ayarsor->execute();

$ayarcek=$ayarsor->fetch(PDO::FETCH\_ASSOC);

?>

yukarıda yazdığımız kodlar ile php ayarların sorgusunu oluşturduk bu kodlar giriş bilgilerinin doğruluğunu test etmek için veritabanı içerisinde bulunan ayarlar tablosuna gidip sorgulama işlemi yapacaktır.

if (isset($\_POST['oturumac'])){

$kullanicisor=$db->prepare("SELECT \* FROM kullanici WHERE

kul\_mail=:mail

AND

kul\_sifre=:sifre");

$kullanicisor->execute(array(

'mail' => $\_POST['kul\_mail'],

'sifre' => $\_POST['kul\_sifre'] ));

$sonuc=$kullanicisor->rowcount();

if($sonuc==0) {

echo "Mail Veya Şifreniz Yanlıştır";

}else

header("location:../index.php");

$\_SESSION['kul\_mail']=$\_POST['kul\_mail'];

}

Oturum açma işlemini islemler.php dosyası üzerinde kaydettik ve deneme işlemi yaptık sorunsuz çalıştı ve artık kullanıcı e-mail ve parolamız ile giriş yapabileceğiz tabi veritabanında girmiş olduğumuz kullanıcı bilgilerinin de kayıtlı olması gerekiyor yanlış bilgiler girersek kullanıcı girişi başarısız hatası alıyoruz.

# 5. HAFTA

## Veri Tabanı Ilac Tablo ve Tablo Yapısı Oluşturmak :

Bu haftaki çalışmamda ilac adında tablo oluşturdum ilaclar.php, ilacekle.php ve ilacduzenle.php adında dosyalar oluşturdum veritabanı entegrasyonunu tamamladım ve admin panelinden veri girdiğim zaman veritabanına kayıt olması için bu kodları yazdım.



Şekil 5 : Veri tabanı ilac tablo yapısı görünümü...

Veri tabanı Tablo Yapısı (Şekil 5) görüldüğü gibi oluşturuldu.Form dan veri cekmek için kullandığımız kodlar aşağıdaki gibidir.

<?php

$sayi=0;

$ilacsor=$db->prepare("SELECT \* FROM ilac");

$ilacsor->execute();

while($ilaccek=$ilacsor->fetch(PDO::FETCH\_ASSOC)) {

$sayi++; ?>

<tr>

<td><?php echo $sayi ?></td>

<td><?php echo $ilaccek['i\_adi']?></td>

<td><?php echo $ilaccek['i\_tarih'] ?></td>

<td><?php echo $ilaccek['i\_durum'] ?></td>

<td><?php echo $ilaccek['i\_detay'] ?></td>

<td><?php echo $ilaccek['i\_kategori'] ?></td>

<td><?php echo $ilaccek['barkod'] ?></td>

</tr>

<?php } ?>

Bu kodlar ile veritabanındaki kodları web siteme çağırmayı başardım.İşlemler sayfasın da veritabanına veri kaydetmek için neler yaptığımıza bakalım şimdi de.

<?php

if (isset($\_POST['i\_ekle'])) {

$i\_ekle=$db->prepare("INSERT INTO ilac SET

i\_adi=:i\_adi,

i\_tarih=:i\_tarih,

i\_durum=:i\_durum,

i\_kategori=:i\_kategori,

barkod=:barkod,

i\_detay=:i\_detay");

$i\_ekle->execute(array(

'i\_adi' => $\_POST['i\_adi'],

'i\_tarih' => $\_POST['i\_tarih'],

'i\_durum' => $\_POST['i\_durum'],

'i\_kategori' => $\_POST['i\_kategori'],

'barkod' => $\_POST['barkod'],

'i\_detay' => $\_POST['i\_detay']));

if($i\_ekle) {

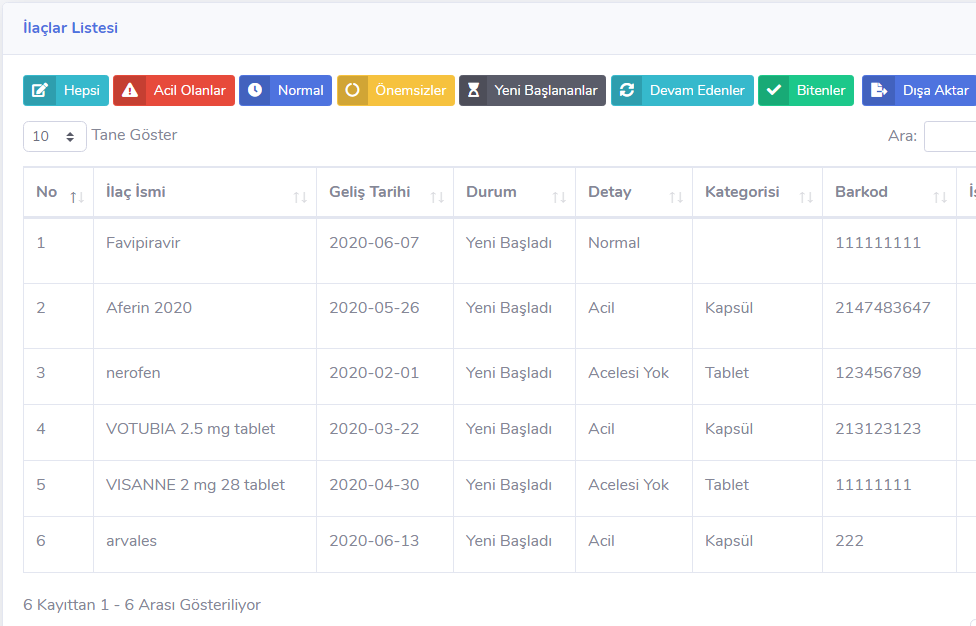
header("location:../ilaclar.php");

}else{

echo "BAŞARÍSÍZ";

exit; } } ?>

İlaçlar girdisi tamamlandı (Şekil 6) da görüldüğü gibi sorunsuz bir şekilde ilaç ekleyebiliyoruz sisteme.



Şekil 6 : İlaçlar sayfası için veri kaydetme işlemi tamamlandı...

Bu haftaki çalışmamız da bu kadar gelecek hafta siparişler sayfasını oluşturacağız.

# 6.HAFTA

## Veri Tabanı Siparis Tablo ve Tablo Yapısı Oluşturmak :

Bu haftaki çalışmamda proje adında tablo oluşturdum siparisler.php, siparisekle.php ve siparisduzenle.php adında dosyalar oluşturdum veritabanı entegrasyonu işlemini tamamladım ve admin panelinden veri girdiğim zaman veritabanına girilmesi aşağıda gösteridiğim işlemleri sırayla gerçekleştirdim. 5.hafta yaptığım ilk çalışma gibi ilk başta sql tablomun yapısını oluşturdum



Sql veri kaydetme işlemi için form alanımızı düzenledik php kodlarımızı aşağıda görüldüğü şekilde hazırladık .

<?php

$sayi=0;

$siparissor=$db->prepare("SELECT \* FROM siparis");

$siparissor->execute();

while($sipariscek=$siparissor->fetch(PDO::FETCH\_ASSOC)) {

$sayi++; ?>

<tr>

<td><?php echo $sayi ?></td>

<td><?php echo $sipariscek['sip\_baslik']?></td>

<td><?php echo $sipariscek['sip\_teslim\_tarihi'] ?></td>

<td><?php echo $sipariscek['sip\_aciliyet'] ?></td>

<td><?php echo $sipariscek['sip\_durum'] ?></td>

</tr>

<?php } ?>

bu kodlar ile islemler klasörümün içerisindeki islem.php dosyamı bir birine bağladım ve veritabanına bu yolla ulaşabildim. siparisler.php sayfam için veritabanında kayıtlı olan bilgileri bu kodlar ile siteme çektim. Sonraki işlem sql tablomuzun sorgularını işlemek.

<?php

if (isset($\_POST['siparisekle'])) {

$siparisekle=$db->prepare("INSERT INTO siparis SET

musteri\_isim=:isim,

musteri\_mail=:mail,

musteri\_telefon=:telefon,

sip\_baslik=:baslik,

sip\_teslim\_tarihi=:teslim\_tarihi,

sip\_aciliyet=:aciliyet,

sip\_aciliyet=:acelesiyok,

sip\_durum=:durum,

sip\_ucret=:ucret,

sip\_detay=:detay

");

$ekleme=$siparisekle->execute(array(

'isim' => $\_POST['musteri\_isim'],

'mail' => $\_POST['musteri\_mail'],

'telefon' => $\_POST['musteri\_telefon'],

'baslik' => $\_POST['sip\_baslik'],

'teslim\_tarihi' => $\_POST['sip\_teslim\_tarihi'],

'aciliyet' => $\_POST['sip\_aciliyet'],

'durum' => $\_POST['sip\_durum'],

'ucret' => $\_POST['sip\_ucret'],

'detay' => $\_POST['sip\_detay']

));

if($siparisekle) {

header("location:../siparisler.php");

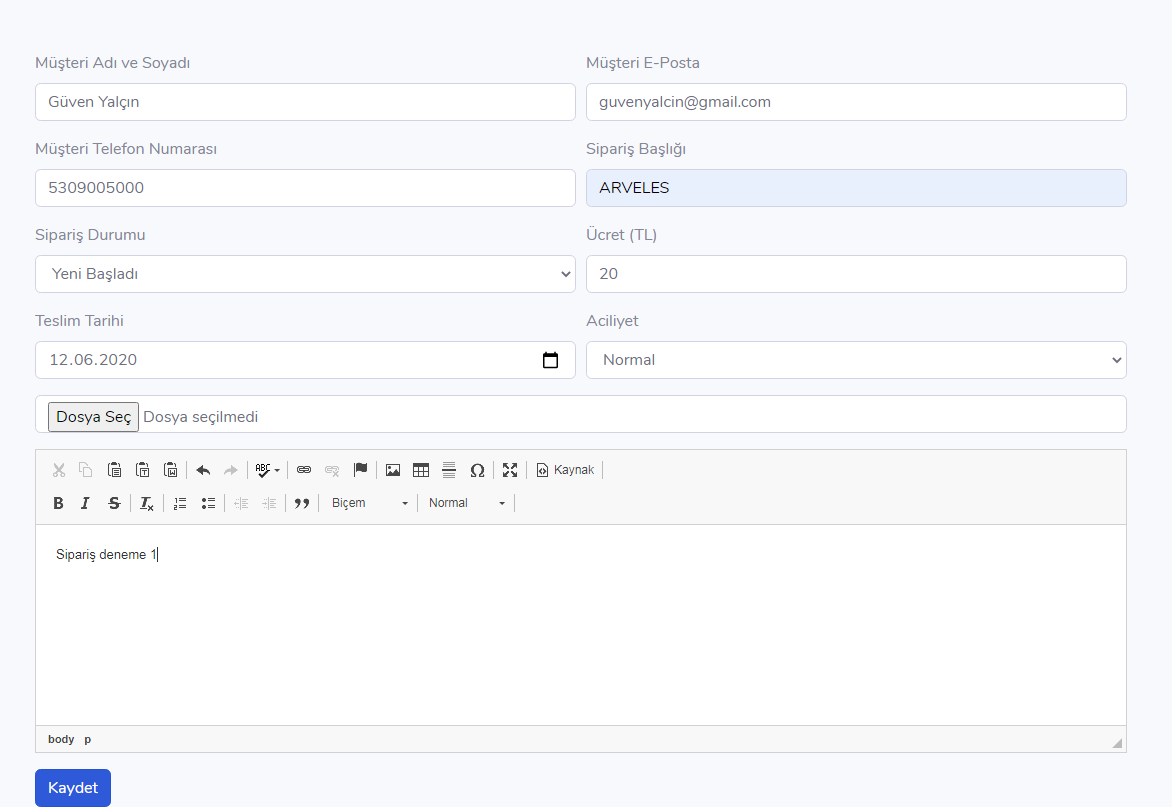
}else{

echo "BAŞARÍSÍZ";

exit;

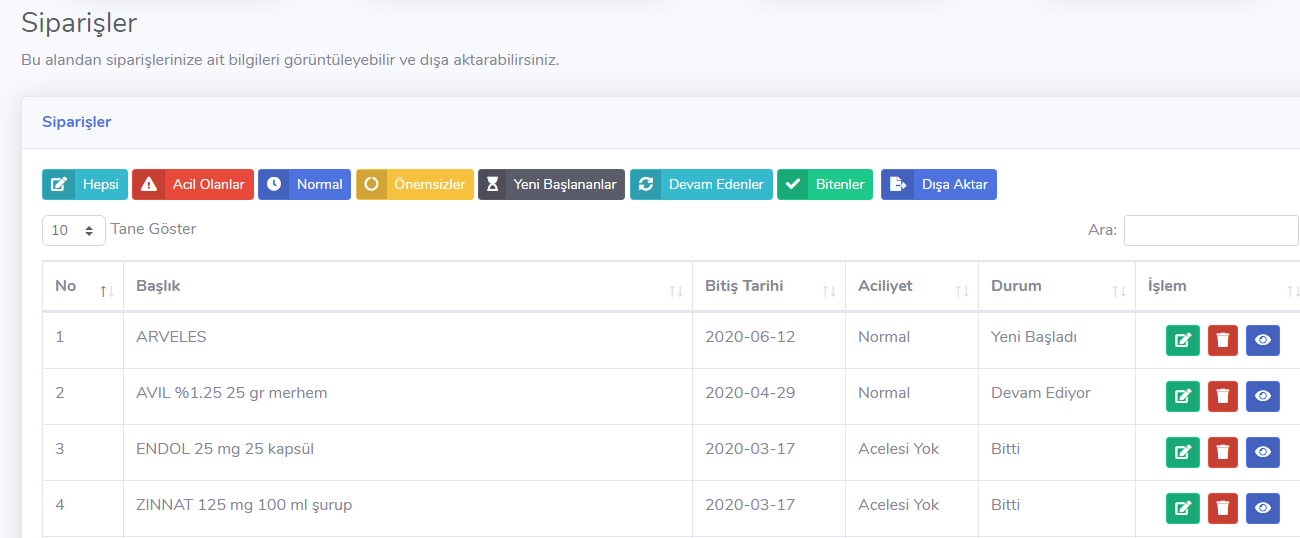
}}?>

Yukarıda yazılan kodlar ile test işlemini gerçekleştirelim şimdi de.



Şekil 7 : Sipariş Kayıt Test İşlemi...

(Şekil 7) de test işlemi için form alanlarımızı doldurduk ve kayıt işlemini gerçekleştirdik.



Şekil 8 : Siparişler Sayfası Veri Kayıt Test İşlemi Tamamlandı...

Veri kayıt işlemi başarı ile tamamlanmıştır. Bir sonraki hafta için yapılacak işimiz başında CKEDİTÖR yani web site içerisine zenginleştirilmiş bir metin editörü ekleyip bunu tanıtmak olacaktır.

# 7. HAFTA

## CKEDİTOR NEDİR? NASIL KULLANILIR :

CKEditor; Kolay kullanımlı, ücretsiz, açık kaynak, hızlı ve güvenli bir zenginleştirilmiş metin editörüdür.

# CKEditor Ne İşe Yarar :

Dinamik içerikli web sitelerine yazı eklemek istediğimiz an, yönetim panelinde kullanabileceğimiz, gördüğünüzün aynısını çıktı olarak alabileceğiniz bir editör. Yönetim panelinde kullanımı ve entegrasyonu çok kolaydır.

# CKEDİTÖR Örnek Kodlama :

**index.html sayfası içerisin de footer alanına kaydetmemiz gereken ckeditör aktifleştirme kodu aşağıdaki gibidir.**

<script src="//cdn.ckeditor.com/4.5.10/full/ckeditor.js"></script>

Head tagları arasına bu kodu ekledikten sonra, body tagları arasına aşağıda ki kodu ekliyoruz.

<form action="goster.php" method="post">

 <label for="icerik">CKEditor Yazı Alanı</label>

 <textarea id="icerik" name="icerik"></textarea>

 <button class="btn btn-success" type="submit">Gönder</button>

</form>

Formumuzu oluşturduktan sonra, aşağıda ki kodu da ekliyoruz.

<script>

            CKEDITOR.replace( 'icerik ile beraber bu alanı silmeyi unutmayınız' );

</script>

**goster.php** adında php dosyamız olduğunu düşünelim ve aşağıdaki kodu goster.php dosyamıza ekleyelim.

$yazi=$\_POST["icerik"];

echo $yazi;

# 8.HAFTA

Bu haftaki çalışmamda islemler.php dosyası içerisinde değişiklikler yaptım. Bu değişikliklerin amacı veritabanında kayıt olan bilgileri istediğim sayfaya tablo bilgileri ile entegre etmekti ve bu içerisinde gerekli düzenlemelerin mysql üzerinden değil de yönetici panelinden yapılmasıydı bunlardan bazıları oturum açma, ayar kaydetme, ilaçlar tablosundaki bilgileri çekme bilgileri site içerisinde düzenleme ve silme gibi işlemleri bu dosya üzerinden gerçekleştireceğim.

## İçeriklerin Düzenlenmesi Ve İçeriklerin Silinmesi :

İçerik düzenleme ve silme işlemlerini iki yol ile yapabiliriz.

1. Veritabanı üzerinden
2. Web site yönetici panelinden

Bizim işimiz de tam olarak ikinci yani web site yönetici panelinden yapmak bunun için ilk önce düzenleme sayfalarını hazırlamak sonrasında ise işlemler dosyamız dan sql kodlarımızı hazırlamak. Birinci yolu html form elemanı kodları yazacağız yani bir nevi içerik ekleme sayfalarındaki kodların aynısını düzenleme sayfasına aktaracağız php kodlar üzerine yazıp işlemi tamamlayacağız mantık bu şekilde. Birinci yolu gösterelim hemen.

<?php

if (isset($\_POST['ne\_id'])) {

$nesor=$db->prepare("SELECT \* FROM nobetci\_eczane WHERE ne\_id=:id");

$nesor->execute(array(

'id' => $\_POST['ne\_id'] ));

$necek=$nesor->fetch(PDO::FETCH\_ASSOC); }?>

Yukarıdaki kodlar ile nöbetçi eczane düzenleme sayfası hazırlayalım

<input type="text" required name="ne\_adi" class="form-control" value="<?php echo $necek['ne\_adi'] ?>"> </div>

<input type="number" class="form-control" required name="ne\_telefon" value="<?php echo $necek['ne\_telefon'] ?>"></div>

Bu bilgileri veritabanı içerisin de düzenlemek için form içerisinde bulunan value değerine bir takım kod yazmam gerekiyor.

<?php

if (isset($\_POST['neduzenle'])) {

$neduzenle=$db->prepare("UPDATE nobetci\_eczane SET

ne\_adi=:ne\_adi,

ne\_adres=:ne\_adres,

ne\_telefon=:ne\_telefon,

ne\_gun=:ne\_gun,

ilce=:ilce,

bolge=:bolge WHERE ne\_id=:ne\_id ");

$neduzenle->execute(array(

'ne\_adi' => $\_POST['ne\_adi'],

'ne\_adres' => $\_POST['ne\_adres'],

'ne\_telefon' => $\_POST['ne\_telefon'],

'ne\_gun' => $\_POST['ne\_gun'],

'ilce' => $\_POST['ilce'],

'bolge' => $\_POST['bolge'],

'ne\_id' => $\_POST['ne\_id'] ));

if($neduzenle) {

header("location:../nobetci\_eczaneler.php");

}else{

echo "BAŞARÍSÍZ";

exit; }}

if (isset($\_POST['ne\_sil'])) {

$ne\_sil=$db->prepare("DELETE FROM nobetci\_eczane WHERE ne\_id=:ne\_id");

$kontrol=$ne\_sil->execute(array(

'ne\_id' => $\_POST['ne\_id']

));

if ($kontrol) {

header("location:../nobetci\_eczaneler.php");

}else{

echo "BAŞARÍSÍZ";

exit;

}

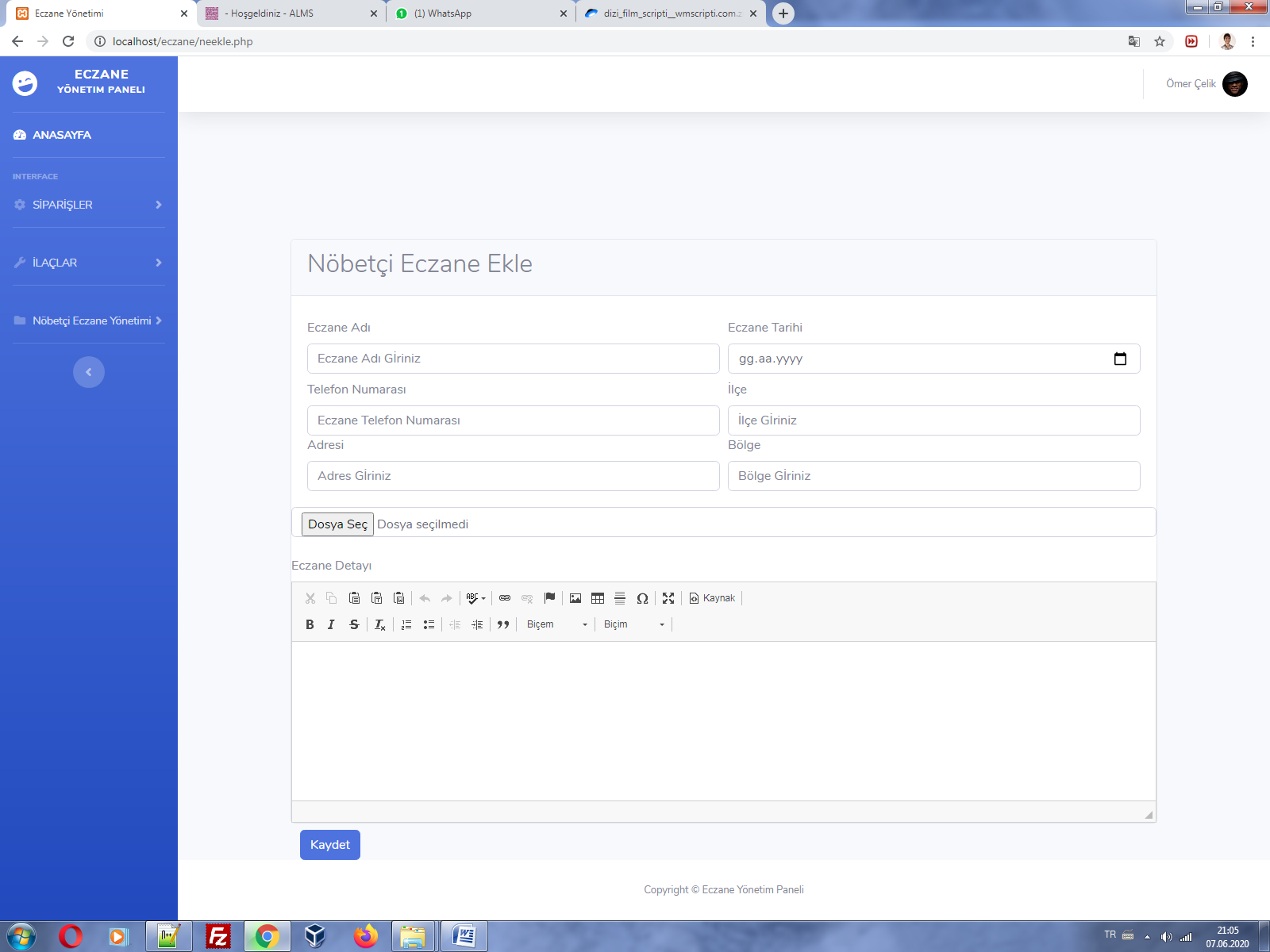
} ?>

Yazılım Aşamasında genellikle bu kodlar kullanılmıştır bundan sonra sistem kontrol testleri yapılacaktır

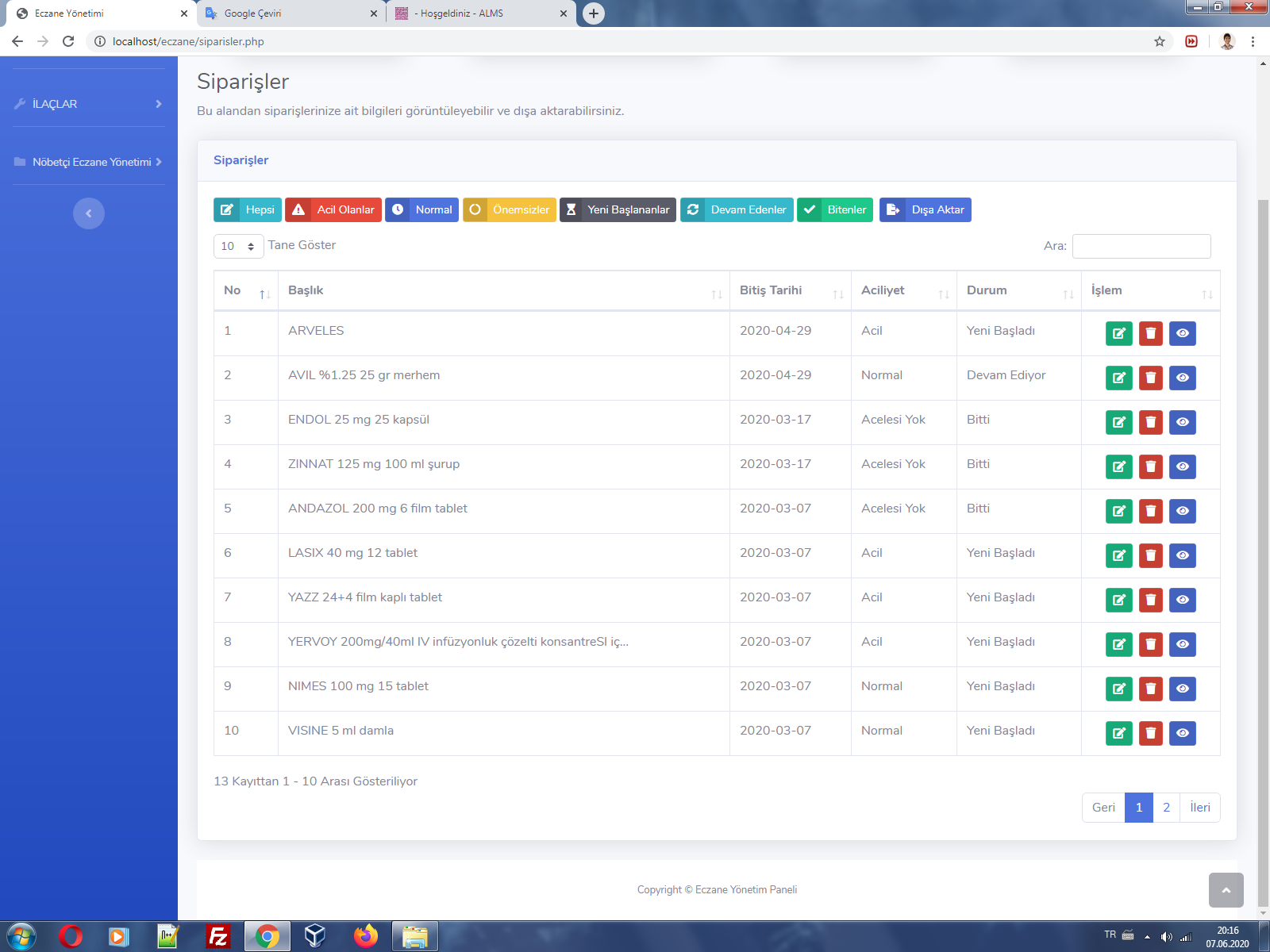
# Sistem Gerekli Testlerden Geçirildi :

fonksiyonların çalışması kontrol edilerek gerekli sınamalar yapıldı. form elemanları ve veritabanı bağlantısı ve sistemin doğru sonuçlar verip vermediği konuları test edildi.

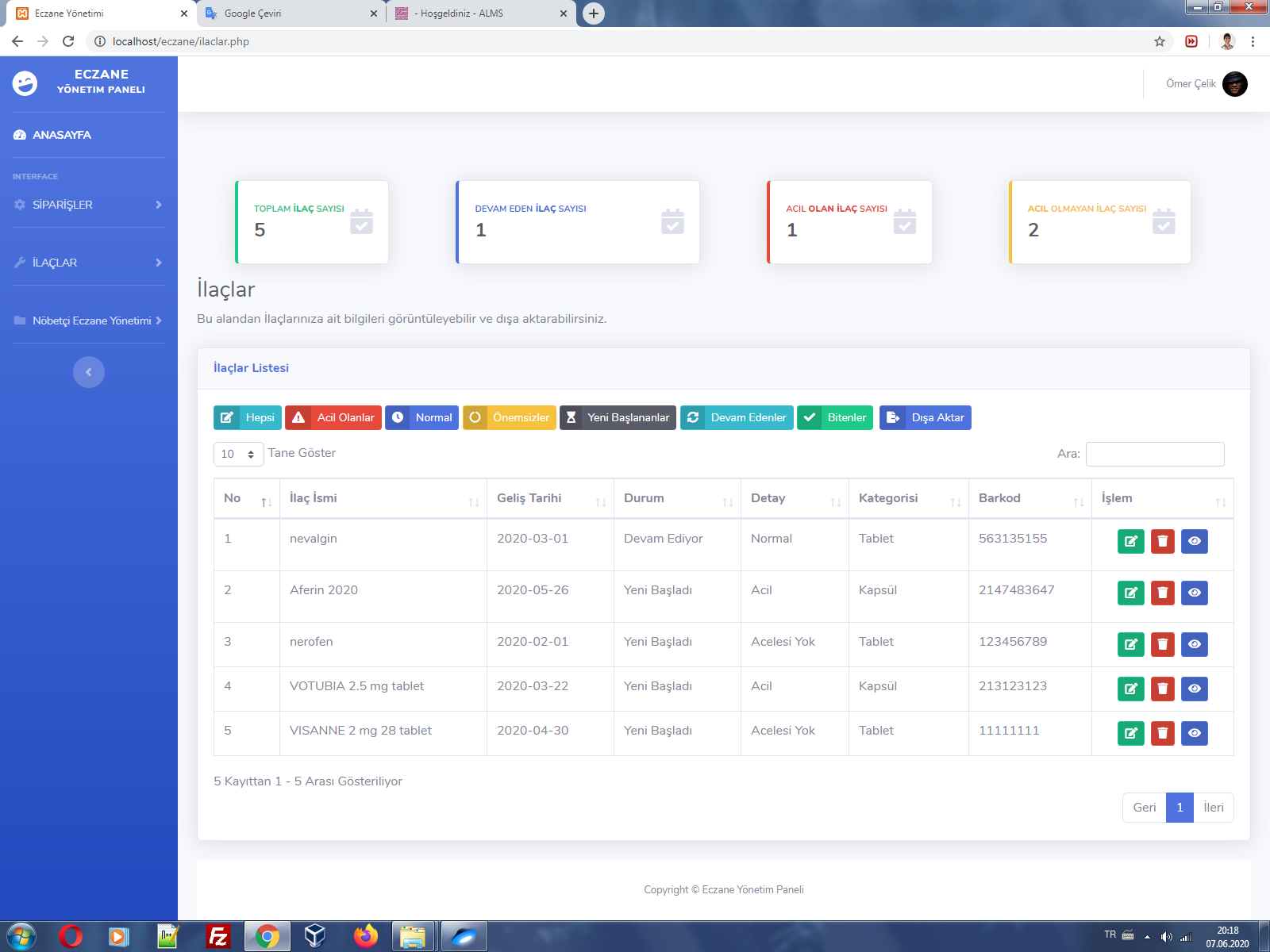
# Yazılımdan Ekran Görüntüleri



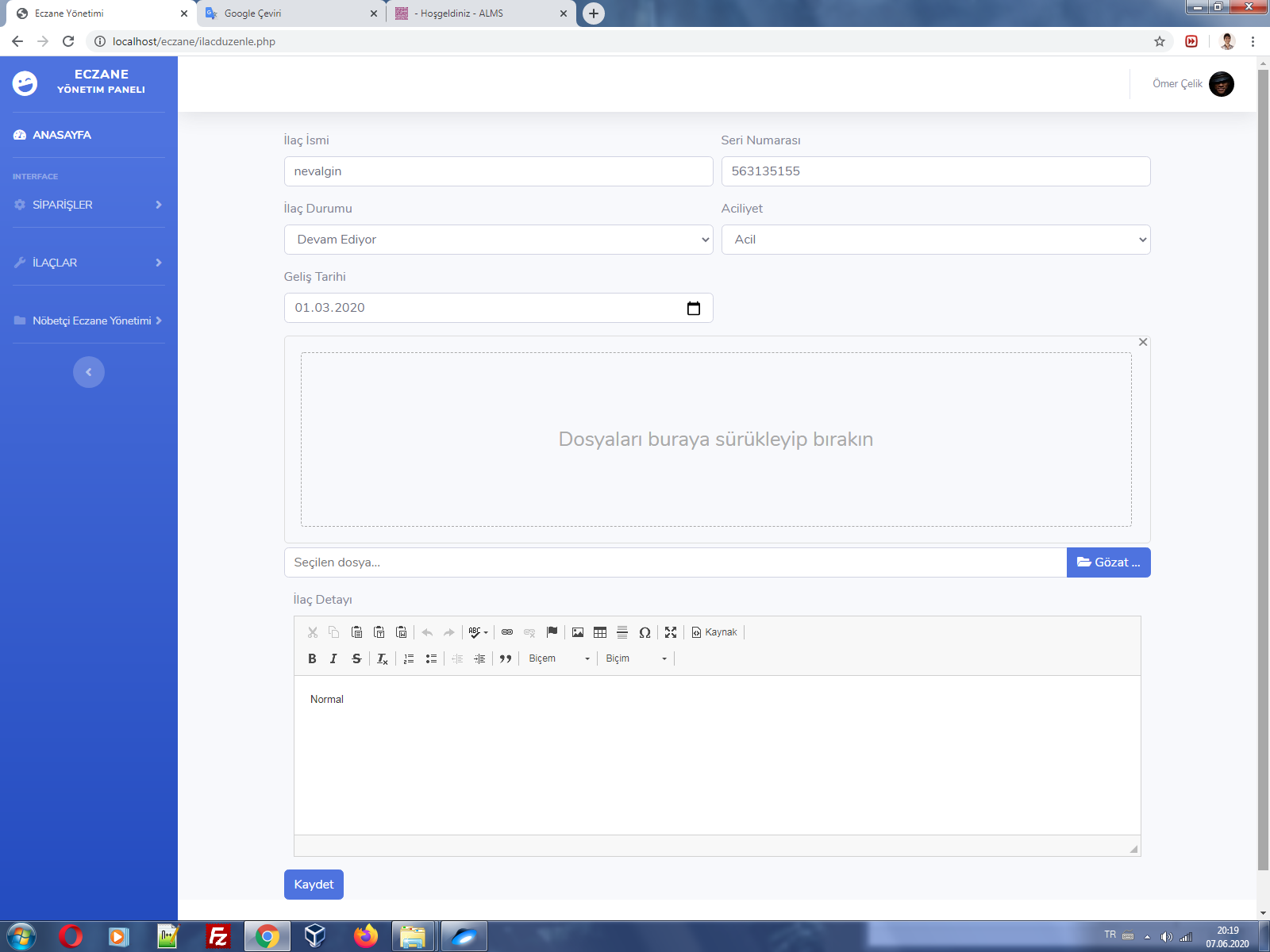
Şekil 9 : Nöbetçi Eczane Ekle Sayfası Görünümü...



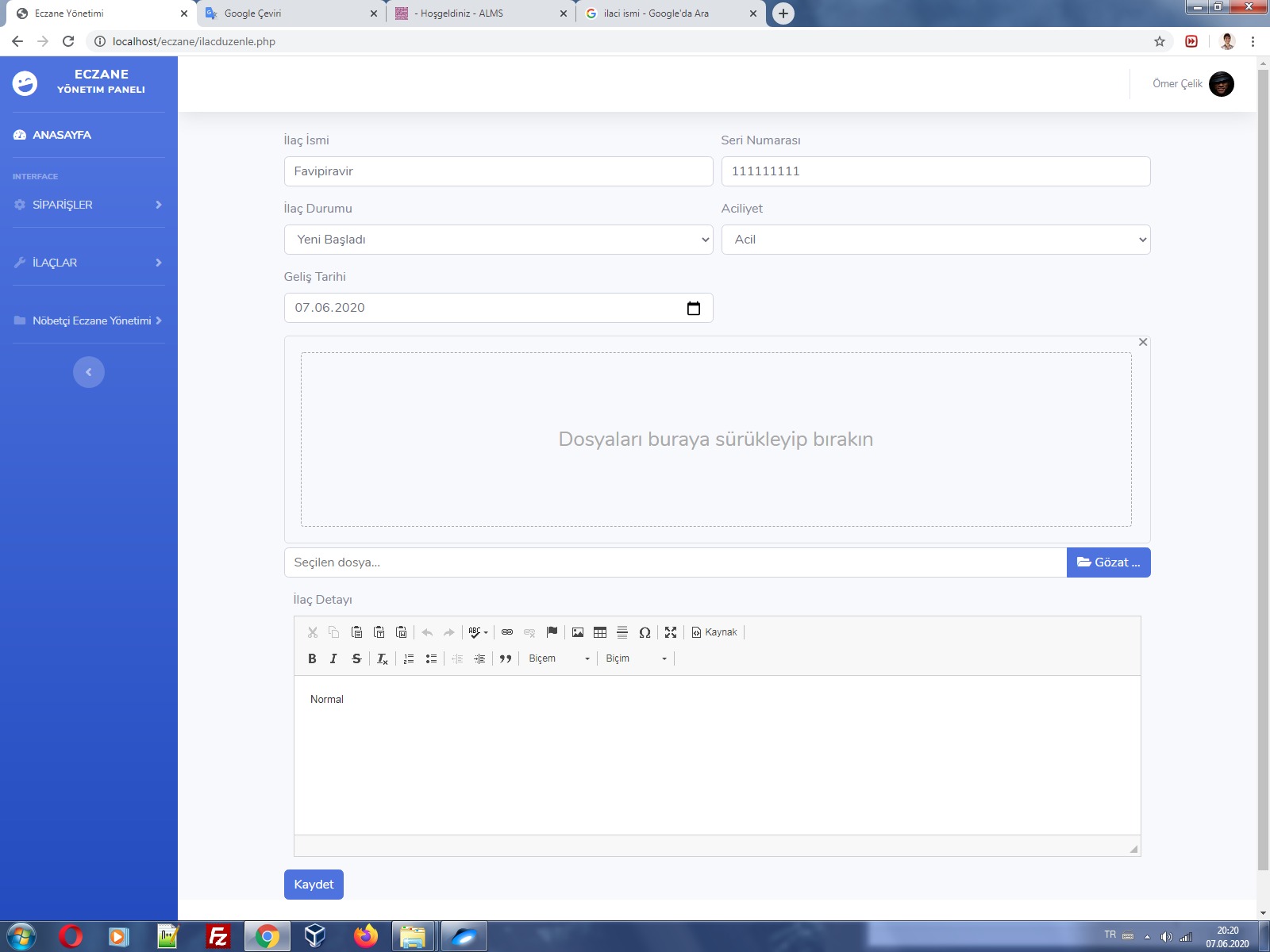
Şekil 10 : Siparişler Sayfası Görünümü...



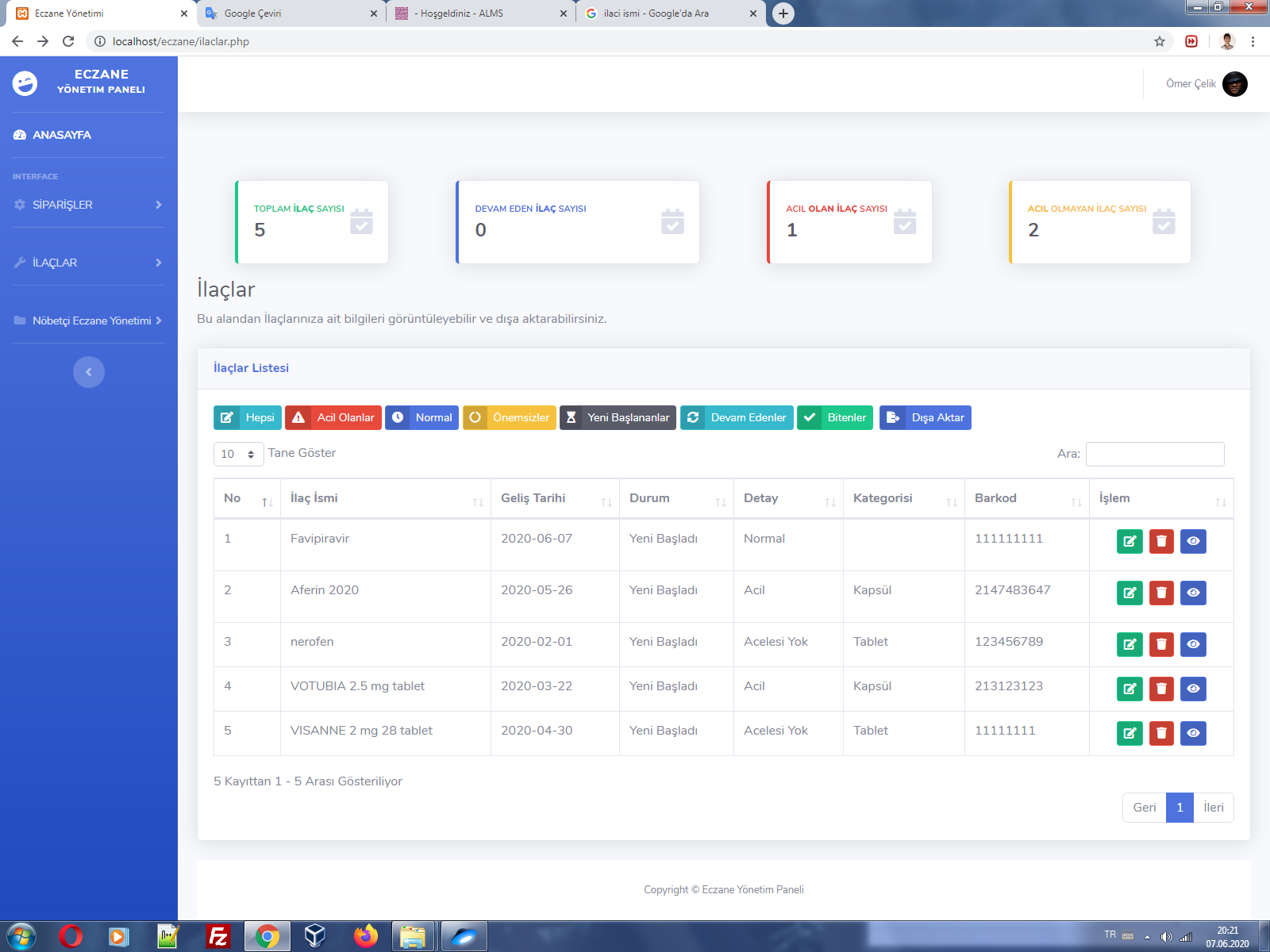
Şekil 11 : İlaçlar Sayfası Görünümü...



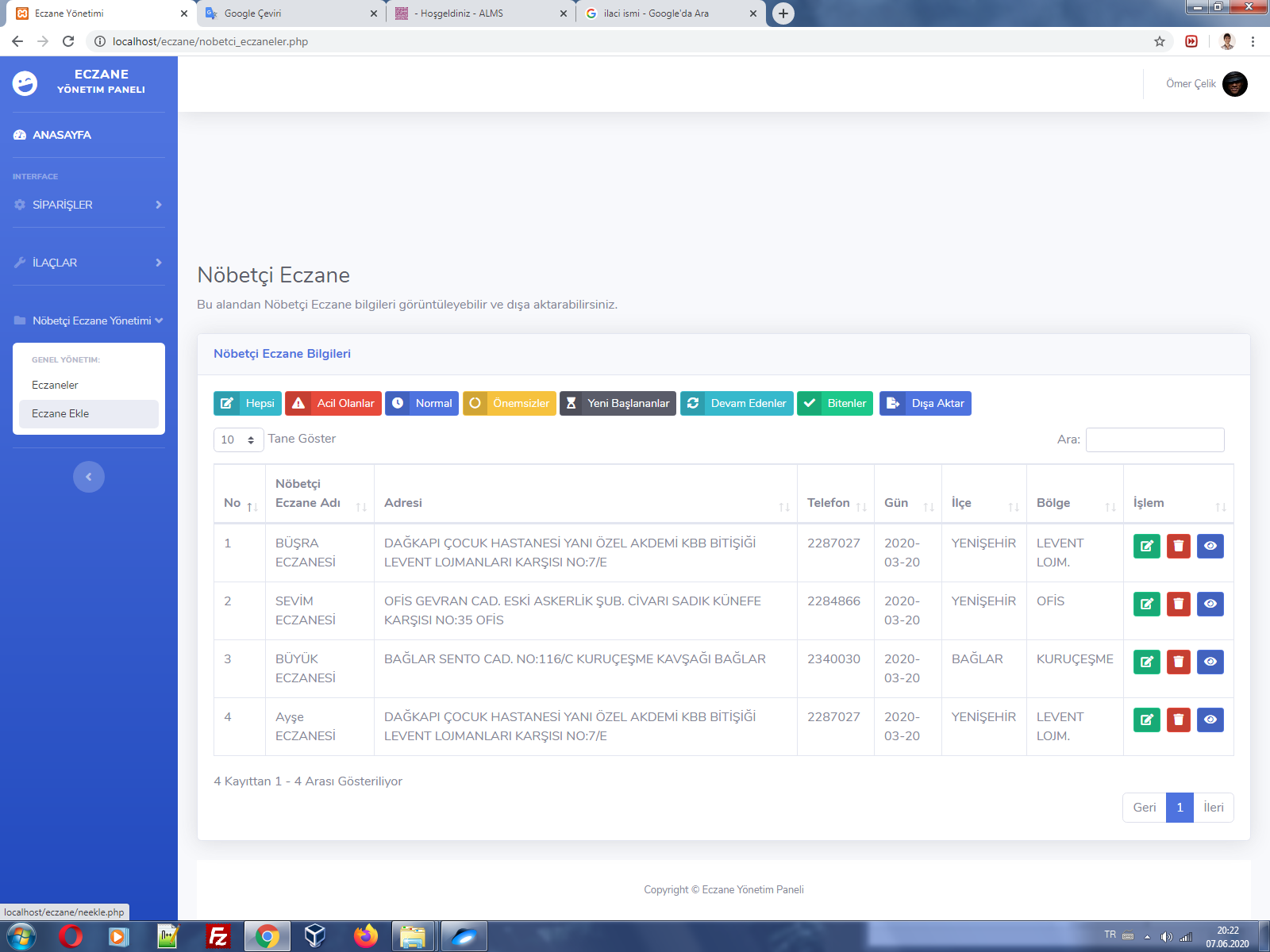
Şekil 12 : İlaç Düzenleme Sayfası Görünümü...



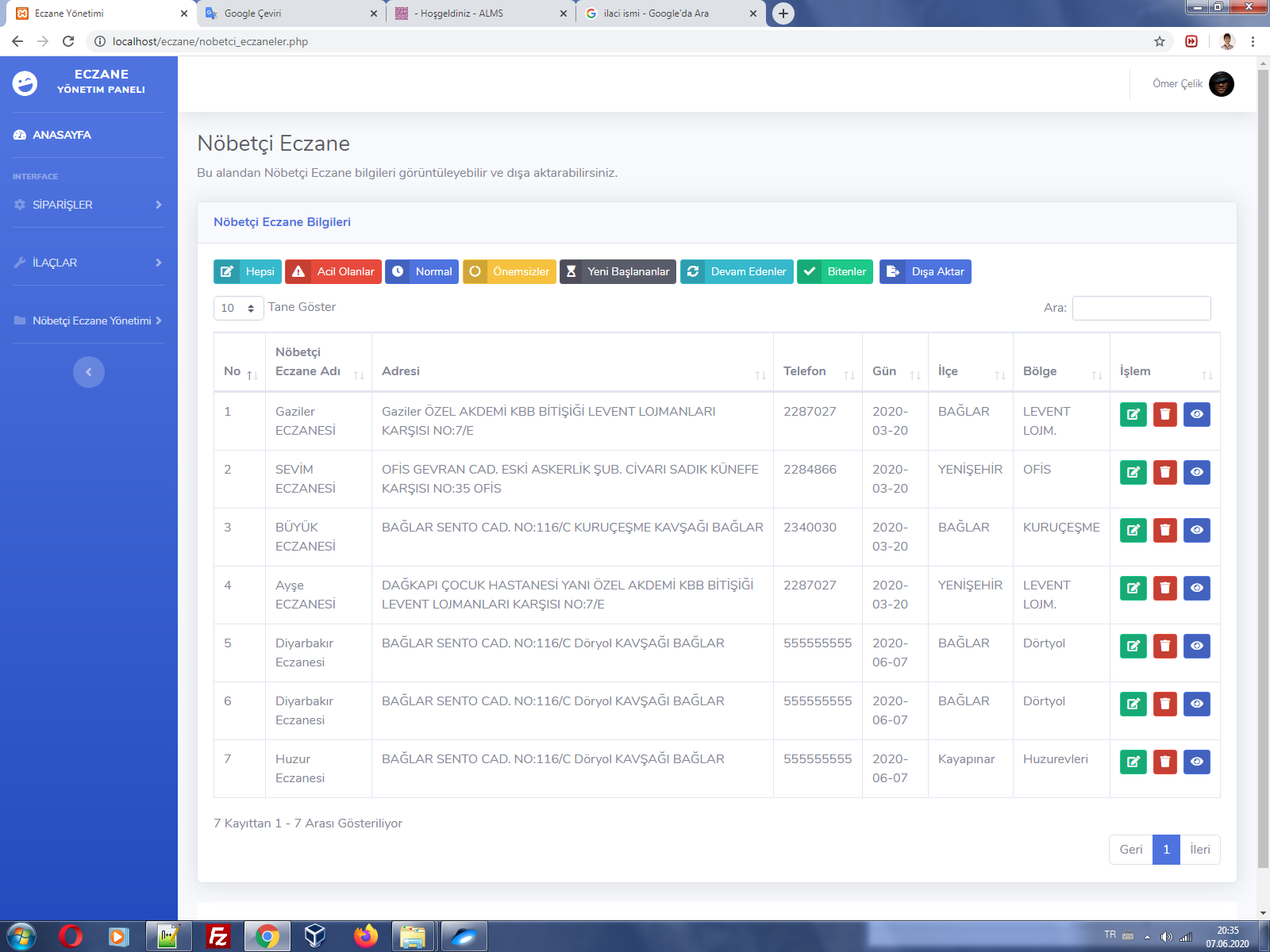
Şekil 13 : İlaç Düzenleme İçeriklerin Değiştirilmesi...



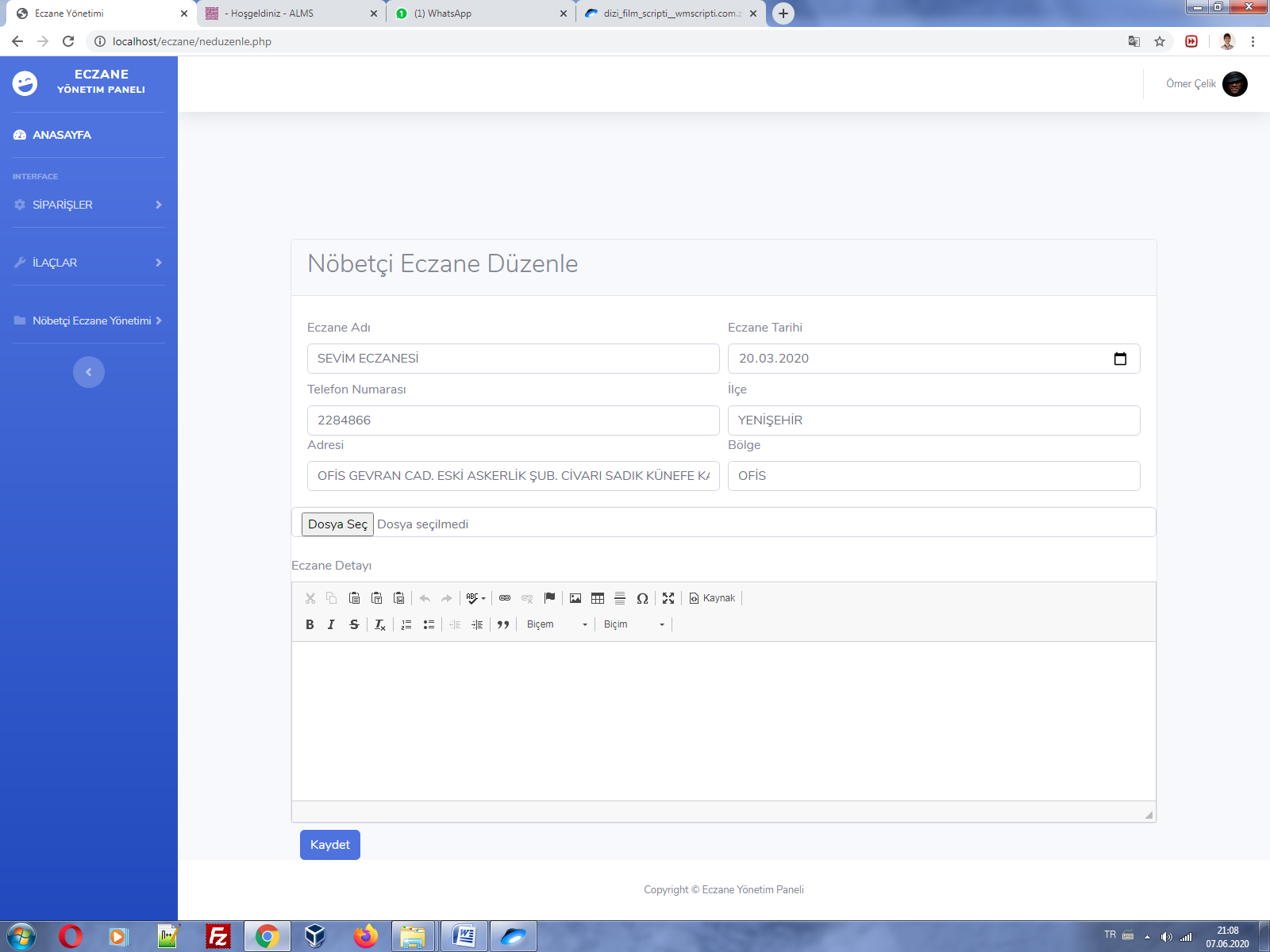
Şekil 14 : İlaç Düzenleme İçeriklerin Değişti...



Şekil 15 : Nöbetçi Eczane Sayfası Görünümü...



Şekil 16 : Nöbetçi Eczaneler Sayfası Yeni Görünümü...



Şekil 17 : Nöbetçi Eczane Düzenleme Sayfası...