



SELÇUK ÜNİVERSİTESİ  
TEKNOLOJİ FAKÜLTESİ



T.C.  
SELÇUK ÜNİVERSİTESİ  
TEKNOLOJİ FAKÜLTESİ  
**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

**Muhasebe Programları ile Opencart  
Entegrasyonu**

**MUHAMMET VAROL**

**BİTİRME PROJESİ**

**OCAK 2019  
KONYA  
Her Hakkı Saklıdır**

## BİTİRME PROJESİ KABUL VE ONAYI

..... tarafından hazırlanan “.....” adlı bitirme proje çalışması .../.../... tarihinde aşağıdaki juri üyeleri tarafından oy birliği/oy çokluğu ile Selçuk Üniversitesi Teknoloji Fakültesi Bilgisayar Mühendisliği bölümünde bitirme projesi olarak kabul edilmiştir.

**Jüri Üyeleri**

**Başkan**

**İmza**

.....

**Danışman**

.....

**Üye**

.....

**Üye**

.....

**Üye**

.....

Yukarıdaki sonucu onaylarım.

Prof. Dr. Fatih BAŞÇİFTÇİ  
Bilgisayar Mühendisliği  
Bölüm Başkanı

## **PROJE BİLDİRİMİ**

Bu projedeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edildiğini ve proje yazım kurallarına uygun olarak hazırlanan bu çalışmada bana ait olmayan her türlü ifade ve bilginin kaynağına eksiksiz atıf yaptığını bildiririm.

## **DECLARATION PAGE**

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all materials and results that are not original to this work.

İmza  
Muhammet VAROL  
Tarih:  
10.01.2019

## ÖZET

### Muhasebe Programları ile Opencart Entegrasyonu

**Muhammet VAROL**

**Selçuk Üniversitesi Teknoloji Fakültesi  
Bilgisayar Mühendisliği Bölümü**

**Danışman: Öğr. Gör. Dr. Mustafa Nevzat Örnek**

**2019, 40 Sayfa**

### Jüri

**Öğr. Gör. Dr. Nevzat ÖRNEK  
Dr. Öğr. Üyesi Humar KAHRAMANLI ÖRNEK  
Arş. Gör. Kübra UYAR  
Dr. Öğr. Üyesi Gülnihal GÜĞÜL**

Günümüzde teknolojinin gelişmesi ile alışverişlerin de birçoğu web sitelerinden yapılmaktadır. Birçok e-ticaret siteleri mevcut ve bu siteler milyonlarca ürün satışı yapmaktadır. Türkiye'nin yalnızca 1 aylık e-ticaret kullanıcı etkinliklerine baktığımızda, internet kullanıcılarının yüzde 52'sinin satın almak üzere çevrimiçi ürün veya hizmet aradığını, bir çevrimiçi perakende mağazasını ziyaret ettiğini, yüzde 43'ünün de bu satın almayı yaptığıını biliyoruz. Hal böyle iken e-ticaret siteleri bu kadar müşteri hizmet vermek için çok fazla ürün satışı yapmaktadır. Bu ürünlerin web sitesine eklenmesi ve bu ürünlerin muhasebe işlemlerinin yapılması için aynı ürünlerin muhasebe programında eklenmesi gerekmektedir. Küçük bir işletme iseniz sizin için büyük bir problem olmaya bilir ama büyük işletmeler için büyük bir çile. Bu sorunu çözmek için muhasebe uygulaması ile e-ticaret sitemizin arasına bir ara katman yazılım ile iki uygulamanın entegrasyonunu sağlanmıştır. Muhasebe uygulamasında yapılan değişikler e-ticaret sitemize de yansıyacaktır.

**Anahtar Kelimeler:** entegrasyon, e-ticaret, muhasebe, opencart

## **ABSTRACT**

### **Opencart Integration with Accounting Programs**

**Muhammet VAROL**

**THE FACULTY OF TECHNOLOGY  
SELÇUK UNIVERSITY  
COMPUTER ENGINEERING**

**Advisor: Öğr. Gör. Dr. Mustafa Nevzat Örnek**

**2019, 38 Pages**

### **Jury**

**Öğr. Gör. Dr. Nevzat ÖRNEK  
Dr. Öğr. Üyesi Humar KAHRAMANLI ÖRNEK  
Arş. Gör. Kübra UYAR  
Dr. Öğr. Üyesi Gülnihal GÜĞÜL**

Nowadays, with the development of technology, most of the purchases are made from web sites. Many e-commerce sites are available, and these sites sell millions of products. When we look at Turkey's only 1 month old e-commerce user activity, Internet users to purchase of the 52 percent of online products or services calling, an online have visited a retail store, 43 percent are aware that in doing this purchase. While this is the case, e-commerce sites are selling too many products to serve so many customers. These products must be added to the website and the same products must be included in the accounting program for the accounting of these products. If you're a small business you might not be a big problem for you but a great ordeal for big businesses. In order to solve this problem, an application between the accounting application and our e-commerce site is integrated with two middleware. Changes made in the accounting application will be reflected in our e-commerce site.

**Keywords:** accounting, e-commerce, integration, opencart

## ÖNSÖZ

Bu tez çalışmasında muhasebe ve e-ticaret ile uğraşan insanların işlerini hızlandırmak ve daha kolay ve zahmetsiz iş yapabilmeleri için ara bir yazılım geliştirilmiştir. Geliştirilen yazılımın veri tabanı yapısı, kullanılan modüller, kullanılan yazılımlar ve kod parçaları bu tez ile sunulmuştur.

Öncelikle bu tez konusunu seçmemi sağlayan ve bana yardımcı olan tez danışmanım Öğr. Gör. Dr. Nevzat ÖRNEK' a teşekkürlerimi sunarım.

MUHAMMET VAROL  
KONYA-2019

## İÇİNDEKİLER

<b>PROJE BİLDİRİMİ .....</b>	<b>iiiv</b>
<b>ÖZET .....</b>	<b>iv</b>
<b>ABSTRACT.....</b>	<b>v</b>
<b>ÖNSÖZ .....</b>	<b>vi</b>
<b>İÇİNDEKİLER .....</b>	<b>vii</b>
<b>1. GİRİŞ .....</b>	<b>1</b>
<b>2. KAYNAK ARAŞTIRMASI .....</b>	<b>2</b>
<b>2.1. Opencart E-Ticaret Uygulaması.....</b>	<b>2</b>
<b>2.2. Opencart Kurulum .....</b>	<b>6</b>
<b>2.3. Akaunting Muhasebe Uygulaması.....</b>	<b>9</b>
<b>2.4. Akaunting Kurulum .....</b>	<b>13</b>
<b>2.5. Zirve Muhasebe Uygulaması.....</b>	<b>15</b>
<b>2.6. Zirve Muhasebe Kurulum.....</b>	<b>16</b>
<b>3. MATERYAL VE YÖNTEM.....</b>	<b>18</b>
<b>3.1. Mysql Binary Log.....</b>	<b>21</b>
<b>3.1.1 Mysql-event modül.....</b>	<b>21</b>
<b>3.2 Akaunting İşlemleri .....</b>	<b>22</b>
<b>3.2. Zirve Muhasebe İşlemleri.....</b>	<b>28</b>
<b>4. ARAŞTIRMA BULGULARI VE TARTIŞMA .....</b>	<b>32</b>
<b>4.1. Web Uygulaması İşlemleri .....</b>	<b>32</b>
<b>4.1.1 Express js .....</b>	<b>32</b>
<b>4.2 API Oluşturma .....</b>	<b>33</b>
<b>4.3. Pug .....</b>	<b>35</b>
<b>4.4. Toplu Ürün Ekleme .....</b>	<b>36</b>
<b>5. SONUÇ .....</b>	<b>38</b>
<b>KAYNAKÇA .....</b>	<b>39</b>
<b>ÖZGEÇMİŞ .....</b>	<b>40</b>

## 1. GİRİŞ

Yapılan araştırmalara göre firmaların muhasebe programlarında yaptıkları değişikliklerin aynısını e-ticaret sitelerinde de yapmaktadır. Bu iş hem zaman kaybı ve yapan kişi tarafından çok zahmetli bir iş haline gelmektedir. Bu yazılım ile muhasebe ile ilgili yapılan işlemlerin tek bir kanaldan kontrol etmemizi sağlamaktadır. Böylelikle firmaların muhasebe hesaplarının hatasız ve hızlı bir şekilde e-ticaret sitesinden veya muhasebe uygulamasından yapılması sağlanmıştır.

Projede kullanılacak e-ticaret sitesi Opencart. OpenCart, açık kaynak kodlu bir çevrimiçi alışveriş platformudur. OpenCart' ta bağlanacak muhasebe uygulamaları ise akaunting ve zirve yazılımıdır. Akaunting, açık kaynak kodlu ücretsiz ön muhasebe platformudur. Zirve, Sql Server veritabanına sahip C# tabanlı bir muhasebe programıdır.

Rapor içeriğinde zirve ve akaunting yazılımlarında oluşan değişiklerin fark edilip OpenCart e-ticaret uygulamasına aktarılması için gerekli olan yapı, kod parçacıkları, kullanılan programlama dilleri anlatılmıştır.

## **2. KAYNAK ARAŞTIRMASI**

Yaptığım araştırmalara göre bu tarz uygulamayı çoğu şirket talep etmekte. Türkiye de bulunan n11, hepsiburada, gittigidiyor gibi büyük şirketler de bu yöntem ile işlemleri hızlandırip tek kanaldan işlemleri yapmaktadır. Zirve muhasebe uygulaması bir çok şirket kullanmakta ve OpenCart e-ticaret sitesine böyle bir entegrasyon yazılımı bulunmamaktadır. Akaunting muhasebe yazılımda ise böyle bir yazılımı eklenti olarak yazıp satabiliyoruz. Akaunting Mysql veri tabanını kullanmakta. OpenCart ‘da Mysql veri tabanını kullanmakta. Aynı veri tabanını kullandıkları için Mysql log yapısı ile değişikleri algılayıp senkronizasyonu kolay bir yol ile yapabiliyorum. Zirve ise SQL Server veri tabanını kullanmaktadır. SQL Server da bulunan CDC (Change Data Capture) özelliği veri tabanının da yapılan değişikleri izliyor eğer değişiklik olursa bize değişimleri veriyor fakat bu özellik SQL Server Enterprise ve Developer sürümlerinde desteklemektedir bu sürümlerde aşırı pahalı olduğu için farklı bir yol ile çözüme gidildi.

### **2.1. Opencart E-Ticaret Uygulaması**

OpenCart açık kaynak kodlu bir çevrimiçi alışveriş platformudur. Çevrimiçi satıcılar için e-ticaret iş platformu diyebiliriz. OpenCart kendi çevrimiçi mağaza yönetim sisteminizi kurmanıza ve etkin bir şekilde kazanmanıza yardımcı olur. Çevrimiçi işin her bölümünü kapsar.( Celiloğlu , 2017 a)

Başlangıçta OpenCart, 1998'de CDROM için Christopher G. Mann ve daha sonra da ücretsiz BSD Mall tarafından geliştirildi. OpenCart, Mayıs 1999'da halka açık bir şekilde harekete geçti. Daniel Cerr tarafından geliştirildi ve PHP 'de yazdığı kendi e-ticaret yazılımlarının temeli olarak kullanıldı. Uzun bir mücadele ve komplikasyondan sonra, son olarak Şubat 2009'da Google'a güvenilir ve kararlı bir yazılım olarak yayınladılar.

Piyasada pek çok benzer platform var ama OpenCart aşağıdaki özellikler nedeniyle diğer rakiplerinin önüne geçiyor.

## Temel Özellikler:

- **Kullanıcı yönetimi:**

Bir çevrimiçi mağazayı organize etmek ve birçok insanla iletişim kurmak için bir yönetim sistemi gereklidir. OpenCart, kullanıcılar ve kullanıcı gruplarına ayrı erişim olanağı sunar.

- **Seçenekler ve Özellikler:**

Hepimiz farklı seçenek, farklı boyutta ve farklı renklerde bulunan ürün ve malları biliyoruz. OpenCart uygulaması gelişmiş sürümleri olan ürünlerde size değişkenlik sunuyor.

- **İndirim, Teklifler ve Kuponlar:**

Bir müşteri pazarda bir şey satın almak istediği zaman, ürünleri ve fiyatlarını kesinlikle karşılaştırır. Bu yüzden, OpenCart uygulaması, perakendeciler için olduğu kadar müşteriler için de bir platform sağlar.

- **Ödeme seçenekleri:**

OpenCart uygulaması çek, Paypal ve banka havalesi gibi farklı ödeme seçenekleri sunar.

- **Çeşitli platformda indirme seçeneği:**

OpenCart açık bir kaynaktır ve ücretsizdir. Şeffaflık sağlar ve topluluklarının istediği bir形象 paylaşır. OpenCart indirmek ve kullanıcı dostu kolaydır

- **Sınırsız kategori:**

Müşteri ve perakendecinin ürün almak için çok fazla kategori aldığı platformdur.

- **İnceleme ve Puanlama:**

İnceleme ve derecelendirmeler, hangi ürünün ve şirketin daha iyi olduğuna ve hangisinin satın alınacağına karar vermek için en önemli faktördür.

OpenCart, müşterinin en iyi derecelendirme ürünlerini gözden geçirme ve yorumlama ile birlikte sunar.

- **Ödül Puanları:**

OpenCart, müşterinin yaşam boyu değerini ve geri dönüşü sağlar ve ayrıca ürün ve kaynak şirketleri hakkında kendinden emin olduğunu gösterir.

- **Ürünleri ve filtreleri kolayca bulun ve satın alın:**

OpenCart kolayca satın almak ve satmak olanak sağlar. Perakendeciler ve müşteriler için harika bir platform. OpenCart ürün kategorileri oluşturun. Müşteri istediği ürünü rafine eder.

- **Dil ve para birimi:**

OpenCart açık kaynaklı bir platformdur. Yani, OpenCart uygulamasını kendi dilinizde kullanabilirsiniz oldukça kolaydır. İşletmenizi birden çok dilde genişletebilirsiniz.

- **Etiket desteği ve satış raporu:**

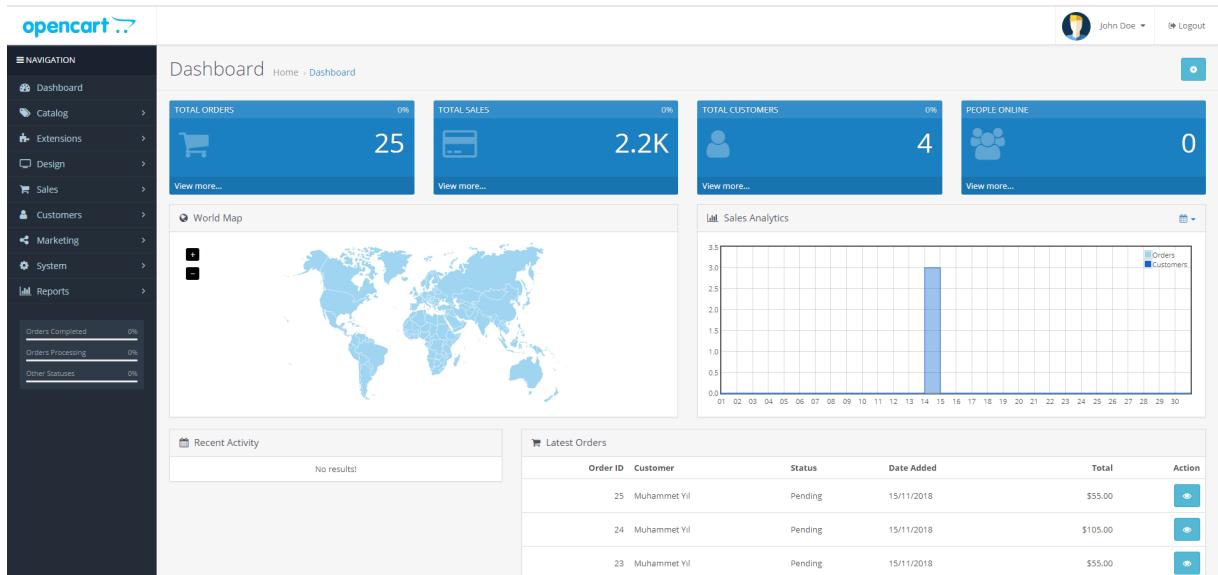
OpenCart ürün raporları sağlar, böylece ürününe ve şirket çevrimiçi mağazasına müşteriye güvenmek kolaydır. OpenCart ayrıca yıllık ürün satış raporları sağlar.

- **Destek:**

OpenCart sertifikalı ortaklardan çevrimiçi yardım sağlar. Altın, gümüş ve platin gibi çevrimiçi mağaza ortaklarına derecelendirme sağlar. Herkes, en çok satan uzmanlardan ihtiyacınıza göre öneri alabilir.

- **Arama motoru:**

OpenCart uygulamasını indirmek kolaydır ve en iyi arama motoru uzantısını sağlar. OpenCart, web sitenizi yüksek arama motoruna taşımak için yüksek maliyet gerektirmez.



**Şekil 2.1** Opencart Admin Panel

Yönetici panelini kullanarak uygulamada aşağıdaki şeyleri değiştirebilir veya ekleyebiliriz.

- Banner widget'ı
- Kategori
- Müşteriler
- Ürün Ekleme
- Uygulama Altbilgisi
- Uygulama Logosu
- Arkaplan
- Tema renk
- Buton rengi

OpenCart pazarında, işe başlamak, büyümek ve genişletmek için 13000'den fazla modül ve tema bulunmaktadır. Herhangi bir sektör, hizmet entegrasyonu, ödeme sağlayıcıları, gönderim yöntemleri, sosyal medya, pazarlama, muhasebe, raporlama, satış ve dil paketleri hakkında güzel temalar bulunmaktadır. (Roy, 2016)

## 2.2. Opencart Kurulum

OpenCart'ın en son sürümü (v.3.0.2.0) OpenCart web sitesinden (son kullanıcı için önerilir) veya doğrudan Github' dan(web geliştiricisi için önerilir ) indirilebilir. OpenCart'ın sıkıştırılmış arşivi bir zip dosyası indirilir. Zip dosyasını açınca, aşağıdaki dosyalar görünür olacaktır.

"install" klasörü

"license.txt" dosyası

"readme.txt" dosyası

"install" klasörü, OpenCart'ı bir web sunucusuna yüklemek için gereken tüm dosyaları içerir. "License.txt" dosyası, sitenizde OpenCart kullanımıyla ilgili lisans sözleşmesini içerir. "Readme.txt" dosyası, OpenCart web sitesinde mevcut yükleme ve yükseltme talimatlarına bağlantılar sağlar. .( Celiloğlu , 2017 b)

Bundan sonraki aşamada bir web sunucuna dosyaları atmak gerekiyor. Ben Wamp ile bir web sunucu çalıştırıp dostları local' de çalıştırıldım.

Bir sonraki adım, bir mağazanın verilerini depolamak için OpenCart için MySQL sunucusunda bir veri tabanı oluşturmaktır. Sitenin kontrol paneline giriş yapmalı ve MySQL veri tabanlarını bulmalısınız. MySQL veri tabanlarını kullanarak, bu veri tabanına erişmek için bir veri tabanı adı ve kullanıcı adı / şifre girerek yeni bir veri tabanı oluşturabilirsiniz. Yeni oluşturulan kullanıcının gerekli tüm izinleri etkinleştirerek veri tabanına eklenmesi gereklidir. Bu veri tabanı bilgilerini, daha sonra otomatik yükleyiciyi kullanarak OpenCart'ı yapılandırırken kullanacağız.

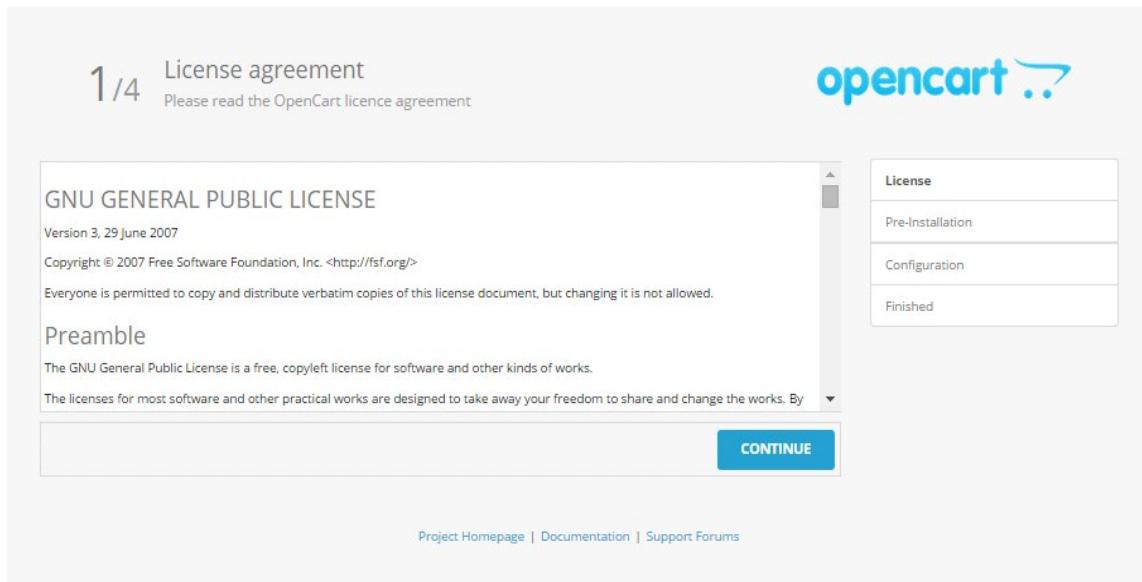
## Veritabanları



**Şekil 2.2** Mysql Yeni Veri Tabanı Oluşturma

<http://localhost/phpmyadmin> adresine girerek yeni bir veri tabanı oluşturabiliriz. Yeni oluşturulmuş veritabanıyla, OpenCart'ı doğrudan bir web sitesine kurmaya hazırız. Bir web tarayıcısı açmalı ve OpenCart'ı yüklediğimiz web adresini girmeliyiz. "install"

içindeki "install" klasörü doğru şekilde yükleniyse, Şekil 2.3' deki sayfa tarafından otomatik olarak karşılanmalıyız.( Vargonon, 2017)



**Şekil 2.3** OpenCart Yükleme Sayfası

Lisans sözleşmesini okuduktan sonra devam edebilir ve ikinci adıma geçebiliriz. Bu adım, web sunucusunun teknik gereksinimleri ve OpenCart'ı doğru şekilde kurmak için doğru izinlere sahip olup olmadığını kontrol eder. Kırmızı işaretler 1 veya 2'de görünürse, bu bileşenlerin web sunucusuna düzgün şekilde yüklediğinden emin olmanız gereklidir. Bununla ilgili herhangi bir sorun varsa, barındırma hizmetinize başvurmalısınız. 3 veya 4 üzerinde kırmızı işaretler varsa, FTP istemcisini kullanarak dosyaların yüklenmesiyle ilgili bir sorun olabilir. Her şey doğru şekilde yapılandırılmışsa ve yeşil işaretler görünürse Şekil 2.4 'de göründüğü gibi "Devam Et" düğmesine basabilirsiniz.

1. Please configure your PHP settings to match requirements listed below.

PHP Settings	Current Settings	Required Settings	Status
PHP Version	5.6.31	5.4+	<span style="color: green;">✓</span>
Register Globals	Off	Off	<span style="color: green;">✓</span>
Magic Quotes GPC	Off	Off	<span style="color: green;">✓</span>
File Uploads	On	On	<span style="color: green;">✓</span>
Session Auto Start	Off	Off	<span style="color: green;">✓</span>

2. Please make sure the PHP extensions listed below are installed.

Extension Settings	Current Settings	Required Settings	Status
Database	On	On	<span style="color: green;">✓</span>
GD	On	On	<span style="color: green;">✓</span>
cURL	On	On	<span style="color: green;">✓</span>
OpenSSL	On	On	<span style="color: green;">✓</span>
ZLIB	On	On	<span style="color: green;">✓</span>
ZIP	On	On	<span style="color: green;">✓</span>

4. Please make sure you have set the correct permissions on the files list below.

Files	Status
C:/wamp64/www/OpenCart2/upload/config.php	Missing
C:/wamp64/www/OpenCart2/upload/admin/config.php	Unwritable

5. Please make sure you have set the correct permissions on the directories list below.

Directories	Status
C:/wamp64/www/OpenCart2/upload/image/	Writable
C:/wamp64/www/OpenCart2/upload/image/cache/	Writable
C:/wamp64/www/OpenCart2/upload/image/catalog/	Writable
C:/wamp64/www/OpenCart2/upload/system/storage/cache/	Writable
C:/wamp64/www/OpenCart2/upload/system/storage/logs/	Writable
C:/wamp64/www/OpenCart2/upload/system/storage/download/	Writable
C:/wamp64/www/OpenCart2/upload/system/storage/upload/	Writable
C:/wamp64/www/OpenCart2/upload/system/storage/modification/	Writable

BACK
CONTINUE

**Şekil 2.4** OpenCart Kurulum Öncesi Bilgilendirme sayfası

Mağazanın veri tabanını MySQL’ de oluşturduğunuzda veri tabanı bilgilerini (kullanıcı adı / şifre, veri tabanı adı) ekleyerek 1. Bölümde alanları doldurmanız gereklidir. 2. Bölümde, OpenCart mağazasına yönetim erişimi için yeni bir kullanıcı adı ve şifre oluşturmalıyız.

Bu adımdan sonra yükleme işlemini bitirmiş oluyoruz. Artık bir OpenCart uygulamamızı kullanıyor olabileceğiz. Son olarak web sunucusunda bulunan “install” klasörünü silmek. Yükleme işimiz bittiği için güvenlik nedeniyle bu dosyayı silmemizi istiyor.

### **2.3. Akaunting Muhasebe Uygulaması**

Açık kaynak kodlu ücretsiz ön muhasebe platformudur. Muhasebe kayıtlarını tutmak ve bu kayıtları takip etmek oldukça meşakkatli. Faturalamadan gider takibine ve ön muhasebeye kadar pek özellikle sahip olan Akaunting, ücretsiz olması sayesinde ön plana çıkan bir program. Cüneyt Şentürk ve Denis Duliçi tarafından geliştirilmiştir.(Papuççıyan, 2018)

Kullanıma sunulduğu günden bu yana 10 binden fazla indirilip farklı domainlerde/sunucularda kurulmuş olan Akaunting’ in bulut hizmetlerinde ise 2 bin 500’den fazla şirket hesap açmış durumda. Nakit akışı, kolay fatura, raporlama, gider takibi, işlem kategorileri, müşteri paneli, tedarikçi yönetimi, stok takibi ve çoklu dil desteği gibi özellikler sunan Akaunting bir uygulama mağazasına da sahip.

Uygulama mağzası sayesinde eklenti / uygulama desteği de sunan Akaunting, hem ücretli hem de ücretsiz olmak üzere pek çok uygulama sunuyor. Ücretli uygulamalar üzerinden bir gelir modeli oluşturan Akaunting, uygulamalar için aylık ücretler talep ediyor.

İnternet erişimi olan herhangi bir bilgisayar, tablet veya telefondan, finansal verilerinize anında ulaşmanızı, işletmenizin mali durumunu takip etmenizi sağlıyor.

Akaunting, Laravel, Bootstrap, jQuery, RESTful API gibi modern teknoloji kullanarak geliştirilmiştir.

- Kullanımı Kolay Arayüz
- Mobil ve Tablet Uyumlu
- Çoklu Dil Yönetim ve Müşteri Paneli

- KOBİ'ler için Tasarlanmış
- %100 Veri Gizliliği
- Ömür Boyu ÜCRETSİZ

Gibi bazı özellikler sunuyor.(Akaunting, 2018 a)



**Şekil 2.5** Akaunting Yönetim Paneli

### Temel Özellikler:

- **Ücretsiz muhasebe**

Tamamen ücretsiz. Kısıtlı, süreli veya deneme sürümü yok, sadece tam sürüm. Bildiğiniz ücretsiz.

- **Online muhasebe**

Finansal verilerinize anında, dünyanın her yerinden ve herhangi bir cihazdan ulaşın.

- **Açık kaynak muhasebe**

Açık Kaynak teknolojisi size 360 derece gizlilik sağlıyor. Kendi sunucunuzda kolaylıkla kurulumu yapın.

- **Müşteri yönetim**

Müşterilerinizi yönetin ve onlara fatura gönderin. Onlara şifre göndererek panele girmelerini sağlayın.

- **Kolay fatura**

Hızlı ve kolayca fatura kesin, müşterilerinizin online olarak ödemelerini sağlayın, komisyon yok.

- **Deposito ve transfer**

Kasa ve Banka arası para transferlerini ve depozitolarını rahatlıkla tek sayfa üzerinden yönetin.

- **Tedarikçi yönetimi**

Tedarikçileriniz için ayrıntılı bilgiler içeren cari kartları yönetin, işlemleri kolayca filtreleyin.

- **Gider faturaları**

Sadece gelir değil, gider faturalarınızı da sisteme ekleyin, beklenmedik sürprizlerle karşılaşmayın.

- **Tüm giderler**

Kime, ne zaman ve ne kadar ödeme yapılacak, tüm gider kalemleriniz önünüzde olsun.

- **Stok takibi**

Fatura kestikçe otomatik güncellenen stok durumunu kolayca takip edin. Ürünler faturalamayı hızlandırın.

- **Banka hesapları**

Hangi kasa veya banka hesabından ne zaman, ne kadar para girmiş veya çekmiş görüntüleyin.

- **Çoklu kur**

Farklı kurda fatura kesip müşterilerinize gönderin. Akaunting kur dönüşümlerini sizin için otomatik yapsın.

- **Çoklu firma**

Tek ara yüz üzerinden birden fazla firma yönetin, firmalararası kolayca geçin yapın, farklı kullanıcılar atayın.

- **Anlaşılır raporlar**

Muhasebe bilgisine gerek yok, işletmenizin detaylı raporları rahatlıkla yorumlayın ve analiz edin.

- **Müşteri paneli**

İşlemleri ve faturaları müşterinizle online paylaşın, tahsilatlarınızı garanti altına alın.

- **Tekrarlama emri**

Tekrarlama emri ile günü gelen gelir/gider kalemlerinizi otomatik olusun, siz takip etmekle uğraşmayın!

- **İndirim**

Gelir veya gider faturalarınıza ara toplam tutarına indirim ekleyin, müşterilerinizi memnun edin.

- **Cari kart**

Carinizin gelir/gider geçmişini hızlıca görüntüleyin, iletişim bilgilerine anında ulaşın.

- **Sınırsız dosya**

Faturalara ve gelir-gider kalemlere istediğiniz kadar ekleyin, gerçek kanıtınız olsun.

- **İşlem kategorileri**

En çok hangi kategoriden kazandığınızı ve nereye para harcadığınızı görün, işletmenize yön verin.

- **Vergi oranları**

Dilediğiniz kadar vergi oranı oluşturun, gelir-gider kalemlerinizi istediğiniz şekilde vergilendirin.

- **Uygulama mağazası**

Akaunting panelini terk etmeden özelleştirin, herhangi bir uygulamayı satın alıp kolayca kurun.

- **Çoklu dil**

İşletmenizi istediğiniz dilde yönetin, diller arası arasında geçiş yapın.

- **Gelişmiş izinler**

Kullanıcıları farklı görevlere atayın, bu görevlere farklı izinler belirleyin.(Akaunting, 2018 b)

## **2.4. Akaunting Kurulum**

Akaunting resmi sitesini ziyaret ederek.

- Akaunting paketini indirin.
- İndirilen paketi çıkarın.
- Tüm dosyaları sunucunuza yükleyin.

Daha sonra web sunucunuza yüklediğiniz dosyalarınıza erişim sağlayın. Erişim sağladığımızda bizi Şekil 2.6 ‘deki gibi bir sayfa karşılaşacak.(Akaunting, 2018 c)



**Şekil 2.6** Akaunting Dil Seçimi

İlk adım, kullanmak istediğiniz dili seçmektir. Bu, farklı bir dili kullanmak isteyen herkes için iyi bir yoldur. Adım 2' de ise veri tabanı bağlantı detaylarını (Ana Bilgisayar Adı, Kullanıcı Adı, Şifre ve Veri tabanı Adı) ekleyerek alanları doldurmanız gereklidir.

### Adım 2/3 : Veritabanı Ayarları

<b>Sunucu *</b>	<input type="text" value="localhost"/>
<b>Kullanıcı *</b>	<input type="text" value="deneme@deneme.com"/>
<b>Şifre</b>	<input type="password" value="....."/>
<b>Veritabanı *</b>	<input type="text" value="Veritabanı Girin"/>

**İleri →**

**Şekil 2.7** Akaunting Veri Tabanı Bilgileri Giriş

Adım 3' de ise, Şirket Adı ve E-postasını eklemenize ve ayrıca Akaunting' inize erişim için yeni bir yönetici hesabı oluşturmanıza izin verir.

**Adım 3/3 : Şirket ve Yönetici Bilgileri**

---

<b>Şirket Adı *</b>	<input type="text"/>
	Şirket Adı Girin
<b>Şirket e-Postası *</b>	<input type="text"/>
	Şirket e-Postası Girin
<b>Yönetici e-Postası *</b>	<input type="text"/>
	Yönetici e-Postası Girin
<b>Yönetici Şifresi *</b>	<input type="text"/>
	Yönetici Şifresi Girin

---

**Şekil 2.8** Akaunting Şirket ve Giriş Bilgileri

Bu işlemleri yaptıktan sonra artık akaunting (ön muhasebe) uygulamamız çalışır hale gelmiştir.

## 2.5. Zirve Muhasebe Uygulaması

Windows işletim sistemi için tasarlanmış olup SQL Server veritabanı kullanır. Standart donanımlarla ekstra maliyet gerektirmeden kurulum yapılabilir. Kullanıcı dostu ara yüzü ile kolay adaptasyon sağlar. Veriler kolayca yedeklenebilir ve yine yedekten kolayca geri dönülebilir. Zengin modül içeriğine sahiptir.

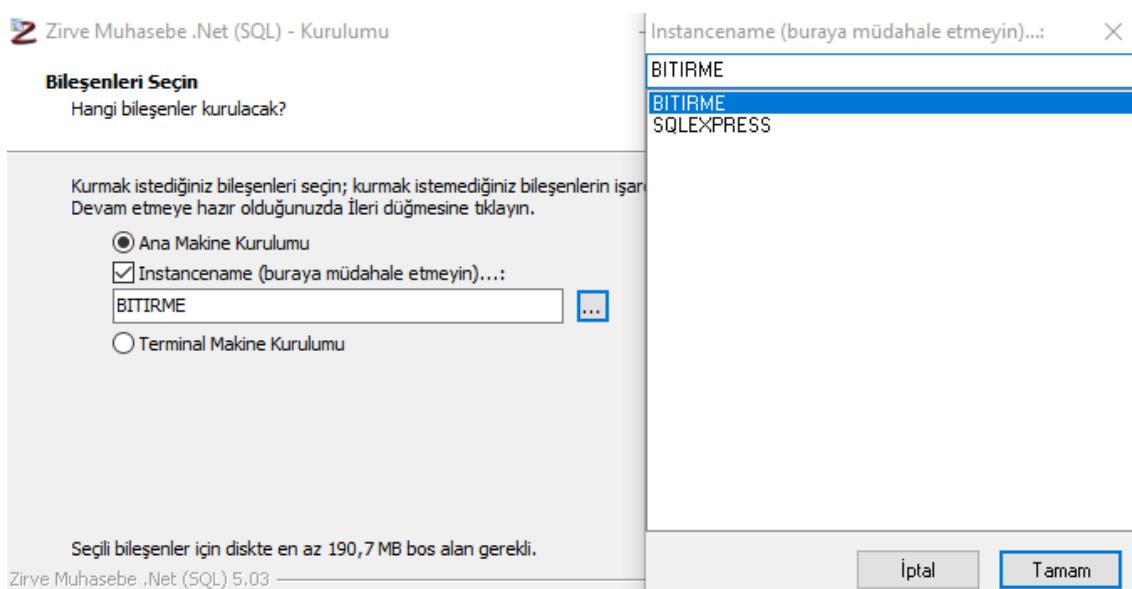
Genel Muhasebe, müşavir Stok, KDV-1 Beyannamesi, KDV-2 Beyannamesi, Muhtasar Beyannamesi, Geçici Vergi Beyannamesi, Gelir Vergisi Beyannamesi, Kurumlar Vergisi Beyannamesi, Basit Usul Gelir Vergisi, Kesin Mizan Beyannamesi, Ba - Bs Formları, Adres Etiket, Ajanda, Zirve Mevzuat gibi özellikleri bulunmakta.(Zirve, 2018 a)

### Temel özellikler:

- Tek Düzen Hesap Planı yâda hesap kırılımları tarafımızca eklenmiş Zirve Hesap Planı üzerinde çalışabilirsiniz.
- Firmadan firmaya hesap planı kopyalayabilirsiniz.
- Bordro-Personel, Demirbaşlar, Stok, KDV, Muhtasar, Geçici Vergi Beyannameleri, Serbest Meslek Makbuzu modülleri ile tam entegreli çalışabilirsiniz.
- Yıl Sonu Kapanış İşlemlerinin (yansıtma işlemleri, gelir tablosu hesaplarının kapatılması, vergi karşılıklarının ayrılması, dönem kar-zarar kaydı, kapanış fişi) fişlerini tek tuşla tam otomatik olarak hazırlayabilirsiniz.
- Dönem içerisinde istediğiniz tarihte sadece stok mevcudunu belirterek tek tuşla Gelir Tablosu alabilirsiniz.
- Kapanış işlemlerinden sonra geçmişe dönük Bilanço ve Gelir Tablosu alabilirsiniz.
- Miktarlı, dövizli, şubeye göre ve masraf merkezine göre muhasebe kayıtları yapabilirsiniz.
- Fişlerinizi fiş türüne göre sıralayabilir, yevmiye numarası verebilirisiniz.
- Fişleri kopyalayabilir ve birleştirebilirisiniz. (Zirve, 2018 b)

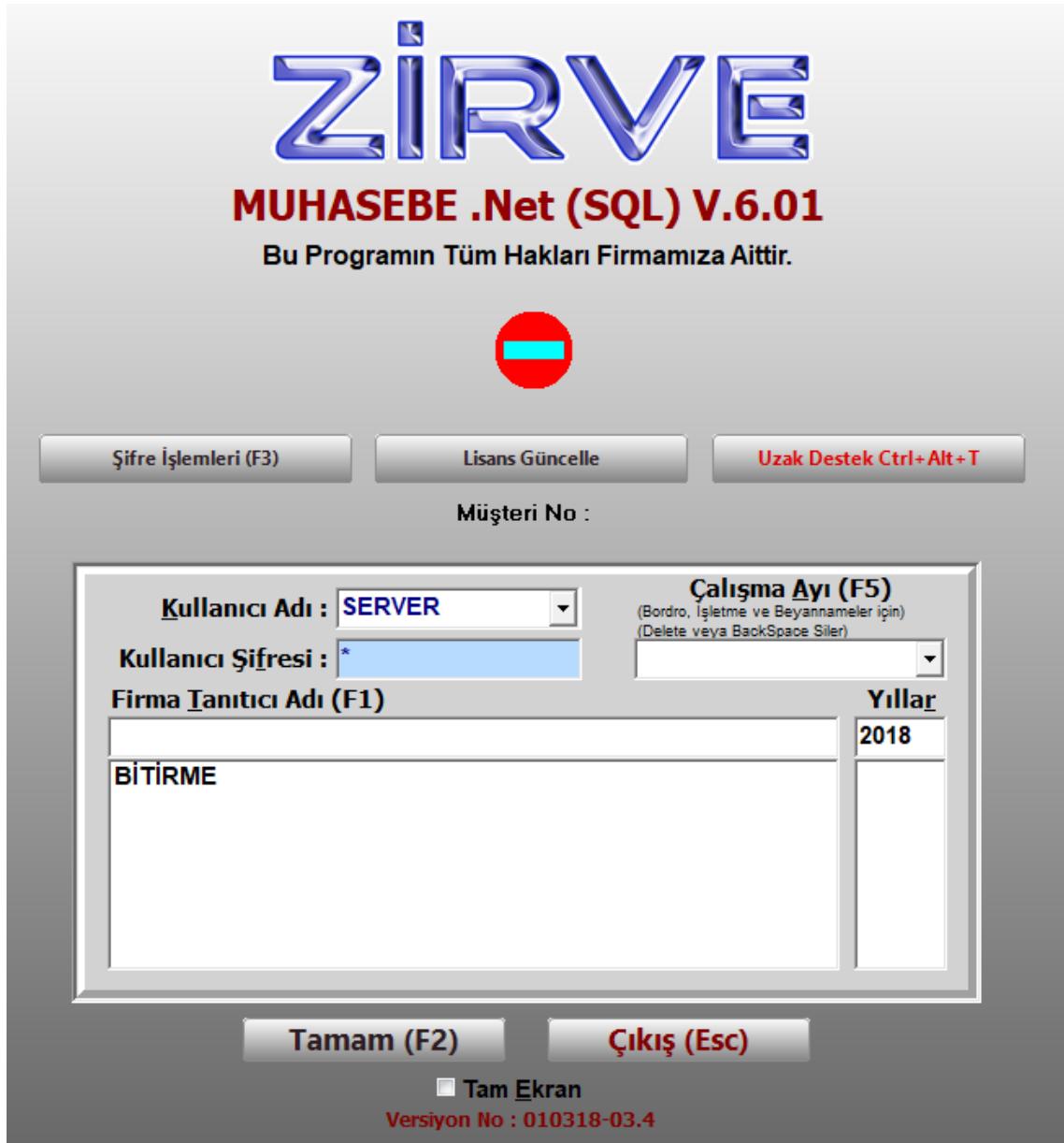
### 2.6. Zirve Muhasebe Kurulum

Zirve programı SQL Server veri tabanını kullandığı için Microsoft SQL Server 2008 Express sürümü yüklü olması gerekiyor. SQL Server' i yükledikten sonra sitesinde verilen linkten muhasebe programını indirip kurulumu başlayabiliriz.



**Şekil 2.9** Zirve Veri Tabanı Seçim Ekranı

Şekil 2.9' de programı yüklemek için seçmemiz gereken veri tabanını kullanıcısını soruyor. Benim veri tabanım BITIRME isminde olan kullanıcı. Veri tabanı bağlantısını seçtikten sonra program kurulmuş oldu.

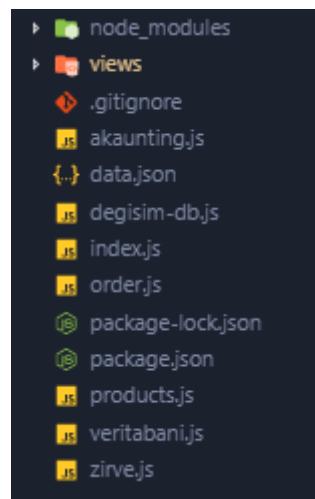


Şekil 2.10 Zirve Giriş Ekranı

Şekil 2.10' da kullanıcı adı ‘server’ ve kullanıcı şifresi ‘1’ olarak tanımlıdır. Bilgileri girdikten sonra tamam butonuna bastığımızda sorunsuz bir şekilde ana işlemler sayfası açılacaktır.

### 3. MATERİYAL VE YÖNTEM

Uygulama geliştirme ortamı olarak Visual Studio Code kullanılmıştır. Uygulama geliştirilirken kullanılan programlama dili Nodejs' dir. Nodejs kullanma amacım kendi içinde Mysql veritabanı olayları takip edici modüller bulundurması. Proje dizin yapısı Şekil 3.1' de verildiği gibi dir.



**Şekil 3.1** Proje Dizin Yapısı

Node\_modules, Nodejs modüllerinin kaydedildiği dosyadır. Views, klasörü projenin ara yüz kısımlarının bulunduğu dosyaları içerir. ‘Akaunting.js’, Akaunting ön muhasebe programında yapılan müşteri ve ürün değişimlerini Opencart' a ekleme işlemlerini içerir.

‘Data.json’, zirve muhasebe uygulamasında müşavir stok bölümde eklenen verilerin tüm bilgilerini bu dosyaya kaydediyorum. Bunu yapma sebebi Zirve muhasebe programına veri eklendiğinde ‘data.json’ dosyasındaki ile karşılaştırıyorum. Eğer değişiklik varsa o değişikliği bana veriyor.

‘Index.js’, projenin ana dizinidir. Proje ana işlemlerinin yapıldığı dosyadır.

‘Order.js’, Opencart' ta yapılan siparişleri fark edip Akaunting' e fatura olarak eklendiği dosyadır.

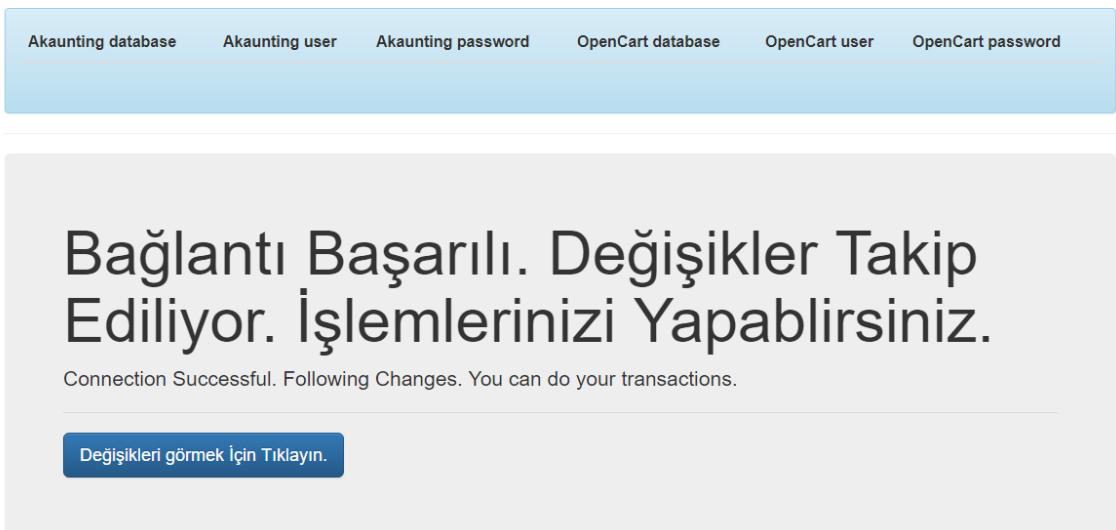
‘Zirve.js’, Zirve muhasebe programında eklenen ürünleri Opencart' a eklendiği dosyadır.

<p><b>Akaunting</b></p> <p><b>Database</b> Enter Database</p> <p><b>User</b> Enter User</p> <p><b>Host</b> Enter User</p> <p><b>Password</b> Password</p> <p><b>OpenCart</b></p> <p><b>Database</b> Enter Database</p> <p><b>User</b> Enter User</p> <p><b>Host</b> Enter User</p> <p><b>Password</b> Password</p> <p><b>Akaunting Müşteri ve Ürün Sync Başlat</b></p> <p>Sipariş Değişikleri için Senkronizasyon Başlat.</p>	<p><b>Zirve</b></p> <p><b>Database</b> Enter Database</p> <p><b>User</b> Enter User</p> <p><b>Host</b> Enter User</p> <p><b>Password</b> Password</p> <p><b>OpenCart</b></p> <p><b>Database</b> Enter Database</p> <p><b>User</b> Enter User</p> <p><b>Host</b> Enter User</p> <p><b>Password</b> Password</p> <p><b>Zirve Eşitle</b></p>
---	---

**Şekil 3.2** Proje Ana Ekran

Proje bir web uygulaması olarak tasarlanmıştır. Ana ekran Şekil 3.2 ‘deki gibidir. Kullanıcı Akaunting’ e bağlanmak isterse Akaunting yazan yerleri ve Opencart Bilgilerini doldurmalı. Zirve programına bağlanmak isterse Zirve alanını doldurmalı ve senkronizasyon işlemini başlatmalı. Akaunting müşteri ve ürün sync butonu Akaunting’ de yapılan değişikleri Opencart’ ta aktarmak için. Sipariş değişikleri için senkronizasyon başlat butonu ise Opencart’ tan alınan ürünler Akaunting’ e fatura olarak eklemek için çalıştırılması gereken kodları çalıştırır. Zirve eşitle butonu ise veri tabanında bulunana tüm stoklar ile ‘data.json’ dosyasında yazan stokları karşılaştırıp eklenen yada silinen stokların işlemlerinin yapıldığı kodları çalıştırır.

Bilgileri doldurduktan sonra eğer hata yoksa giriş başarılı bir şekilde bağlantı sağlamış oluyoruz ve Şekil 3.3’ deki sayfa sizi karşılıyor.



**Şekil 3.3** Bağlantı Başarılı Sayfası

Bağlantı başarılı olduğunda gelen sayfada değişiklikleri görmek için butonuna tıklayarak yapılan değişiklikleri takip edebilirsiniz.

Müşteriler					
İşlem	id	firstname	lastname	email	telephone
Müşteri Güncellendi	159	Çağrı	Dönmez	deneme123@deneme.com1	+90 (544) 514 78 91
Müşteri Eklendi	161	Aydın	Akulut	vrl.muhammet55600@gmail.com15	+90 (544) 514 78 91
Müşteri Güncellendi	162	Aydın	Yılmaz0	vrl.muhammet55600@gmail.com15	+90 (544) 514 78 91
Müşteri Güncellendi	164	Çağrı	Dönmez	deneme123@deneme.com1	+90 (544) 514 78 91
Müşteri Eklendi	165	Recep	Ali	deneme@deneme.com1231234	+90 (544) 514 78 91
Müşteri Güncellendi	166	Recep	Kalender	deneme@deneme.com1231234	+90 (544) 514 78 91
Müşteri Silindi	168	Ömer	Faruk	vrl.muhammet55600@gmail.com123	+90 (544) 514 78 91
Müşteri Eklendi	169	Durkan	Kaya	deneme@deneme.com1234	+90 (544) 514 78 91
Müşteri Güncellendi	170	Durkan	Ali	deneme@deneme.com1234	+90 (544) 514 78 91

**Şekil 3.4** Değişikleri Listeleme Sayfası

İlk etapta ön muhasebe (akaunting) uygulamamızda olan bütün değişikleri algılayıp OpenCart (e-ticaret) uygulamamıza eş zamanlı olarak ekleme, silme, güncelleme işlemleri yapacağız. Bu işlemleri yapmak için kendi yerel bilgisayarımızda kurulan olan iki uygulamaya tam erişim sağlamaktayız. Daha önce OpenCart ve Akaunting Mysql veri tabanını kullanmakta olduğunu söylemiştim. Akaunting veri tabanında yapılan değişiklikleri dinleyip değişikler algılandı. Bu işlem için Mysql binary log (ikili loglama) kullanıldı.

### 3.1. Mysql Binary Log

Mysql Server hangi aktivitelerin meydana geldiğini anlamamız için bazı log türleri sunmaktadır.

**Binary Log:** Silme, Güncelleme, Oluşturma, Drop gibi veriyi değiştiren sorgular içerir. Tablo oluşturma işlemleri veya tablo verisi değişiklikleri gibi veri tabanı değişikliklerini tanımlayan “ olaylar ” içerir. Ayrıca her ifadenin güncellenmiş verileri ne kadar süre aldığı hakkında bilgiler içerir.

İkili loglamanın iki önemli amacı vardır:

**Çoğaltma için**, bir ana sunucudaki işlemleri, bağımlı sunuculara göndererek veri değişikliklerinin bir kaydını sağlar. Ana sunucu, olayları, ana bilgisayar üzerinde yapılan aynı veri değişikliklerini yapmak için bu olayları yürüten işlemcilere gönderir.

**Bazı veri kurtarma işlemleri için**, ikili loglama kullanımını gerektirir. Yedeklemeden sonra kaydedilen ikili loglamada olaylar yeniden çalıştırılır. Bu olaylar, veri tabanlarını yedekleme noktasından bugüne kadar getirir. (Mysql, 2018)

İkili loglama, verileri değiştirmeyen SELECT veya SHOW ifadeleri için kullanılmaz.

İkili günlüğe kaydetme etkinleştirilmiş bir sunucuyu çalıştmak performansı biraz daha yavaş hale getirir. Ancak, çoğaltma ve geri yükleme işlemlerini ayarlamانıza olanak veren yararları genellikle bu küçük performans düşüşünden çok daha önemli.

Default olarak Windowsta Error loglarının aktif olması haricinde hiçbir loglama açık değildir. Loglamayı aktif hale getirebilmek için konfigürasyon dosyası açılır. C:\wamp64\bin\mysql\mysql5.7.19 dizini altında bulunana my.ini dosyası metin düzenleyicide açılır. “log-bin=mysql-bin” satırı bulunur ve başında bulunan yorum satırı (;) ifadesi kaldırılır ve kaydedilir. Mysql restart edilir ve artık binary loglama işlemi başlamış olur.

#### 3.1.1 Mysql-event modül

MySQL veri tabanını izleyen ve eşleşen olaylarda geri çağrıları çalıştan Nodejs paketidir. (NPM, 2018 a)

Bölüm 3.1 de bahsettiğim Mysql ikili loglama özelliğini kullanarak akaunting web uygulamamızın veri tabanını dinleyeceğim ve silme güncelleme ekleme olayları ile yapılan işlemleri yakalayacağım.

```
var MySQLEvents = require('mysql-events');
var mysql = require('mysql');
```

**Şekil 3.5** Mysql Modülleri Dahil Edilme

Mysql-events ve mysql modülünü projemize dahil ediyoruz.

### 3.2 Akaunting İşlemleri

Akaunting ve Opencart veri tabanı olarak Mysql kullanıyor. Bağlantı için Mysql modülünü kuruldu. Şimdi akaunting veri tabanı bilgileri dolduruldu ve bağlantı sağlandı.

```
var akauntingdb = {
  host: 'localhost',
  user: 'root',
  password: ''
};
var mysqlEventWatcher = MySQLEvents(akauntingdb);
```

**Şekil 3.6** Akaunting Veri Tabanı Bilgileri

Akauntingdb ile akaunting veri tabanı bilgileri giriyoruz. Veri tabanını dinlemek için bağlantı sağlıyor. mysqlEventWatcher ile bağlantı yaptığımız veri tabanını dinlemeye başlıyoruz.

```
var opencart = mysql.createConnection({
  host: 'localhost',
  user: 'root',
  password: '',
  database: 'opencart'
});

opencart.connect();
```

**Şekil 3.7** OpenCart Veri Tabanı Bilgileri

Akaunting de yapılan değişikleri OpenCart' a aktarmak için Opencart veri tabanına bağlantı yapıyoruz.

```
var mysqlEventWatcher = MySQLEvents(akauntingdb);
var watcher = mysqlEventWatcher.add(
    'akaunting',
```

**Şekil 3.8** Akaunting Olay Dinleme

mysqlEventWatcher akaunting veri tabanı bilgilerini alarak bir izleyici oluşturuyor. Bu izleyici veri tabanında hangi tabloda ne değişimleri olduğunu bildiriyor.

```
var mysqlEventWatcher = MySQLEvents(akauntingdb);
var watcher = mysqlEventWatcher.add(
    'akaunting',
    function (oldRow, newRow, event) {
        ...
        if(event.tableId==118){ ... }
        ...
        if(event.tableId==126){ ... }
    },
    'Active'
);
```

**Şekil 3.9** Akaunting Tablo Değişimleri

Event.tableId 118 ise müşteriler tablosunda değişim olduğunu eğer 126 ise ürünler tablosunda değişim olduğunu bilinmekte.

```

var product;

var watcher = mysqlEventWatcher.add(
  'akaunting.dgi_items',
  function (oldRow, newRow, event) {

    if (oldRow === null) {

      product=newRow.fields;
      let sql = `INSERT INTO oc_product(product_id, model, sku,
        upc, ean, jan, isbn, mpn, location,quantity,
        stock_status_id, image, manufacturer_id, shipping,
        price, points, tax_class_id, date_available, weight,
        weight_class_id, length, width, height,
        length_class_id, subtract, minimum, sort_order,
        status, date_added, date_modified)

      VALUES ('${product.id}',
      'Product ${product.id}',
      '${product.sku}', '', '', '', '', '',
      '${product.quantity}', '1','NULL',
      '${product.company_id}', '1',
      '${product.sale_price}', '0.0000', '1',
      '${date}', '0.00000000', '1',
      '0.00000000', '0.00000000',
      '0', '1', '1', '1', '0',
      '${product.enabled}',
      '${date}', '${date}')`;

      let sql1=`INSERT INTO oc_product_description(product_id,
        language_id, name, description, tag,
        meta_title, meta_description, meta_keyword)
      VALUES('${product.id}', '1',
      '${product.name}', '${product.description}'
      , '', '', '')`;

      var query = opencart.query(sql, function (error, results, fields) {
        if (error) throw error;
        console.log('eKLENEN Ürün Id:' + results.insertId);
      });

      var query = opencart.query(sql1, function (error, results, fields) {
        if (error) throw error;
        console.log('eKLENEN Ürün Id:' + results.insertId);
      });
    }
  }
)

```

**Şekil 3.10** Akaunting Ürün Senkronizasyon Tanımlamaları

Mysql dinleyicimize aksaunting veri tabanında olan değişikleri dinlemesini söylüyoruz. Eğer eski satır boş ise yani yeni bir şey eklenmiş demektir. Bu da demek oluyor ki “newRow” da değişikler var. Bu da demek oluyor ki veri tabanına bir veri ekleme işlemi yapılmış. Eklenen verinin bilgileri “newRows.fields” yeni satırın fields alt objesinde yer almaktadır. Verileri aldığımıza göre şimdi OpenCart’ a ekleme yapabiliriz.

Şimdi gelen verileri parçalayarak OpenCart’ ta “oc\_product” tablosuna ekleyeceğiz.

```
let sql = `INSERT INTO oc_product(product_id, model, sku,
    upc, ean, jan, isbn, mpn, location, quantity,
    stock_status_id, image, manufacturer_id, shipping,
    price, points, tax_class_id, date_available, weight,
    weight_class_id, length, width, height,
    length_class_id, subtract, minimum, sort_order,
    status, date_added, date_modified)

VALUES ('${product.id}',
'Product ${product.id}',
'${product.sku}', '', '', '', '', '',
'${product.quantity}', '1', 'NULL',
'${product.company_id}', '1',
'${product.sale_price}', '0.0000', '1',
'${date}', '0.00000000', '1',
'0.00000000', '0.00000000',
'0', '1', '1', '1', '0',
'${product.enabled}',
'${date}', '${date}')`;
```

**Şekil 3.11** OpenCart Ürün Ekleme Tanımlamaları

Yukarıdaki sql kodunda “oc\_product” tablosuna eklemek istediğimiz sql ifadesini yazıyoruz.

```
var query = opencart.query(sql, function (error, results, fields) {
    if (error) throw error;
    console.log('EKLENEN Ürün Id:' + results.insertId);
});
```

**Şekil 3.12** Sql Sorgusu Çalıştırma

Yazdığımız sql kodunu OpenCart veri tabanında çalıştırıyor ve akaunting' e eklenen veriyi OpenCart 'a eklemiş olduk.

Şimdi ise silinen verileri yakalayıp aynı şekilde Opencart' tan sileceğiz.

```
if (oldRow !== null && newRow !== null) {
    var product=oldRow.fields;
    if (event.rows[0].after.deleted_at != null) {

        let sql=`DELETE FROM oc_product WHERE product_id=${product.id}`;
```

**Şekil 3.13 Ürün Silme İşlemleri**

Eğer ki eski ve yeni satırda boş ise güncelleme var demektir. Ama akaunting de silinen veriler tamamen veri tabanından silmek yerine “deleted\_at” adın bir değişken ekleyip silinme tarihini ekleniyor. Bizim projemizde de bunu güncelliyor olarak algılıyor ama küçük bir if kontrolü ile eğer “deleted\_at” boş değil ise veri silinmiş demektir.

Silinen verinin “id” numarasını alıp gerekli sql ifadesini yazıyoruz.

```
var query = opencart.query(sql, function (error, results, fields) {
    if (error) throw error;
    console.log('SİLİİNEN Müşteri Id:' + product.id);
});
```

**Şekil 3.14 Ürün silme Sql Sorgusu Çalıştırma**

Yazdığımız sql kodunu çalıştırıyoruz ve veriyi eş zamanlı olarak silmiş oluyoruz.

Eğer ki “deleted\_at” null ise güncelleme olmuştur. Güncelleme için de Şekil 3.15'da ki kodunu yazmalıyız.

```
let sql =`UPDATE oc_product SET
    sku='${event.rows[0].after.sku}',
    price='${event.rows[0].after.sale_price}',
    quantity='${event.rows[0].after.quantity}'
WHERE product_id = ${event.rows[0].after.id}`;
```

**Şekil 3.15 Ürün Güncelleme Sql İfadesi**

```

var query = opencart.query(sql, function (error, results, fields) {
    if (error) throw error;
    console.log('GÜNCELLENEN Müşteri Id:' + product.id);
});

```

**Şekil 3.16** Ürün Güncelleme Sql Sorgusu Çalıştırma

Akaunting ürünler için yapılan bu işlemlerin aynısını müşteriler tablosu içinde yapıldı. Müşteri tablosu id'si 118 dir. Eklenen müşteriler aynı şekilde yakalanıp 'Opencart' a eklenmiştir.

```

if(event.tableId==118){
    var kullanici=newRow.fields;
    if (oldRow === null) {

        var name=String(kullanici.name);
        name=name.split(" ");

        let sql = `INSERT INTO oc_customer(customer_id, customer_group_id,
            language_id, firstname, lastname,email, telephone, fax, password,
            salt, cart, wishlist, newsletter, address_id, custom_field, ip,
            status, safe, token, code, date_added) VALUES ('${kullanici.id}',
            '1', '0', '${name[0]}', '${name[1]}', '${kullanici.email}',
            '${kullanici.phone}', 'null', 'null', 'null', NULL, NULL,
            '0', '0', '', '1', '1', '', '', '${date}')`;

        let sql2 = `INSERT INTO degisim_musteri(islem, customer_group_id,
            language_id, firstname, lastname,email, telephone, fax, password,
            salt, cart, wishlist, newsletter, address_id, custom_field, ip,
            status, safe, token, code, date_added) VALUES ("Müşteri Eklendi",
            '1', '0', '${name[0]}', '${name[1]}', '${kullanici.email}',
            '${kullanici.phone}', 'null', 'null', 'null', NULL, NULL,
            '0', '0', '', '1', '1', '', '', '${date}')`;

        var query = opencart.query(sql, function (error, results, fields) {
            if (error) throw error;
            console.log('eKLENENE Müşteri Id:' + results.insertId);
        });
    }
}

```

**Şekil 3.17** Müşteri Değişikliklerini Yakalama ve Kaydetme

Şekil 3.17 gelen olayın tablo numarası 118 ise ve eklenen satır varsa Opencart veri tabanında 'oc\_customer' tablosuna değişiklik yapılan veriyi ekliyoruz. 'degisim\_musteri' tablosu ise değişiklik yapılan veriler eklemek için. Bu tabloya kaydedilen verileri araya yüzde görsel olarak göstereceğim.

### 3.2. Zirve Muhasebe İşlemleri

Zirve muhasebe programı SQL Server kullandığı için Nodejs' de böyle bir modül yok. SQL Server' da Change Data Capture(CDC) özelliği var ama Express sürümü bunu desteklemiyor. Değişikleri algılamak için 'data.json' adında bir dosya oluşturup ürünler tablosundaki tüm ürünleri çekip bu dosyaya yazdım. Daha sonra ürün eklediğimde tüm veri çekildi 'data.json' dosyası ile karşılaştırıldı. Değişiklikler OpenCart'a kaydedildi.

```
var fs = require("fs");
const sql = require("msnodesqlv8");
const jsonfile = require('jsonfile');
var mysql      = require('mysql');
var format = require('date-format');
var jsonDiff = require('json-diff')
```

**Şekil 3.18** Modül Tanımlamaları

Fs modülü Nodejs dosya okumak için kullanılan modül. Msnodesqlv8, Nodejs ile SQL Server'a bağlanmak için kullanmadım gereken modül. JsonDiff, zirve veri tabanından çekilen tüm verileri json dosya formatında 'data.json' dosyasına aktarıyorum. jsonDiff modülü ile yapılan değişikleri alglıyorum ekleme işlemi mi silme işlemi mi yapıldığını öğrenmek için kullanıyorum.(NPM, 2018 b)

```
const connectionString = "server=localhost\\
BITİRME;Database=BİTİRME_2018;Trusted_Connection=Yes;Driver={SQL Server}"
;
```

**Şekil 3.19** Sql Server Bağlantı

Şekil 3.19' da Sql Server bağlantısı sağlandı. Program localhost' ta çalışıyor. Veri tabanı ismi BİTİRME\_2018.

```
let data = fs.readFileSync('data.json');
let datajson = JSON.parse (data);
```

**Şekil 3.20** Data.json Dosyası Okuması

Önceden yazmış olduğumuz 'data.json' dosyasını okuyoruz ve 'datajson' adlı değişkene atıyoruz. Sayfayı her yenilediğimizde bu kodlar tekrardan çalışacak ve eğer değişiklik bulunuyorsa jsonDiff ile bu değişiklikleri elde edeceğim.(NPM, 2018 c)

```

sql.query(connectionString, query, (err, rows) => {

    let diffdata = jsonDiff.diff(datajson,rows);

    if(diffdata){

        jsonfile.writeFile("./data.json", rows, function (err) {
            if (err) console.error(err)
            console.log("Dosya Yazıldı!");
        });
    }
});

```

**Şekil 3.21** Zirve Programında Değişiklikleri Kontrol Etme

Veri tabanından çekilen veriler ‘rows’ adında json formatında geliyor. Daha önceden çektiğimiz verileri ‘datajson’ adlı değişkendeydi. JsonDiff fonksiyonu ile silinen ve eklenen değerleri alıyoruz. Eklenmiş ise Şekil 3.22’ de olduğu gibi eklenmiş ürün ‘+’ ile belirtilmiş.

```

[ ' ' ]
[ '+',
  { Stk: '11',
    Ref: 80,
    Sta: 'asvz',
    Stb: '51',
    Stgm: null,
    Stgt: null,
    Stcm: null,
    Stct: null,
    Kar: null,
    Ticari: null,
    Eyontemi: null } ]
[ ' ' ]
[ ' ' ]
[ ' ' ]
[ ' ' ]

```

**Şekil 3.22** Eklenen Veri

Silinen veri de Şekil 3.23 de olduğu gibi görünecektir.

```
[ ' ' ]
[ '-',
  { Stk: '12',
    Ref: 81,
    Sta: 'avasvasdasdas',
    Stb: '141',
    Stgm: null,
    Stgt: null,
    Stcm: null,
    Stct: null,
    Kar: null,
    Ticari: null,
    Eyontemi: null } ]
[ ' ' ]
[ ' ' ]
[ ' ' ]
[ ' ' ]
```

Şekil 3.23 Silinen Veri

```
if(diffdata){

  var datach=[];
  var product;
  diffdata.forEach(element => {
    console.log(element)
    if( element[0]=="+"){
      product = element[1];
      datach.push(product);
      let sql = `INSERT INTO oc_product(product_id, model, sku,
        upc, ean, jan, isbn, mpn, location,quantity,
        stock_status_id, image,manufacturer_id, shipping,
        price, points, tax_class_id, date_available, weight,
        weight_class_id, length, width, height,
        length_class_id, subtract, minimum, sort_order,
        status, date_added, date_modified)

      VALUES ('${product.Stk}',
      'Product ${product.Ref}',
      '${product.Ref}', '', '', '', '', '',
      '${product.Stb}', '1', 'NULL', '1',
      '1',
      '500', '0.0000', '1',
      '${date}', '0.00000000', '1',
      '0.00000000', '0.00000000',
      '0', '1', '1', '1', '0',
      '1',
      '${date}', '${date}')`;
```

Şekil 3.24 Değişiklikleri Opencarta Ekleme

Değişiklikleri algıladık ve “+” ise veriyi Opencart’ ekledim eğer “-” ise Opencart’ tan sildim. Tüm işlemler bittikten sonra eğer dosyada farklı veri varsa eski “data.json” dosyasındaki verileri silip yeni gelen değerleri ekliyoruz.

Şekil 3.25’ de yazmış olduğum fonksiyon ile eğer değişim var ise veri tabanındaki “dbo.stokgenm” tablosundaki verileri “data.json” dosyasına yazıyorum.

```
var dbwrite= function(){
var fs = require("fs");
const jsonfile = require('jsonfile');
const sql = require("msnodesqlv8");

const connectionString = "server=localhost\\\
BITIRME;Database=BITİRME_2018;Trusted_Connection=Yes;Driver={SQL Server}\
";

const query = "SELECT * FROM dbo.stokgenm";

sql.query(connectionString, query, (err, rows) => {

    jsonfile.writeFile("./data.json", rows, function (err) {
        if (err) console.error(err)
        console.log("Dosya Yazıldı!");
    });
});

module.exports=dbwrite;
```

**Şekil 3.25** Veriyi Dosyaya Yazma

Nodejs de module.exports ile yazmış olduğum fonksiyonu dışarıya kullanıma açıyoruz ve kullanırken Şekil 3.26 ‘da olduğu gibi kullanıyoruz.

```
var dbwrite=require('./veritabani');
```

**Şekil 3.26** Export Edilen Fonksiyonu Çağırma

## 4. ARAŞTIRMA BULGULARI VE TARTIŞMA

E-ticaret sitelerine sahip şirketler için tasarlanan bu proje ile artık zirve, akaunting muhasebe uygulaması kullanan firmalar işlerini daha hızlı ve zahmetsz halledecekler. Uygulama müşteri, ürün ve yapılan siparişleri ekleme, silme, güncelleme işlemleri otomatik bir şekilde yapmaktadır. Projenin arka tarafta çalışacak işlemleri yapıldı. Şimdi ön yüz de görünecek işlemleri yapalım.

### 4.1. Web Uygulaması İşlemleri

Web uygulaması ile kullanıcı istediği cihazdan istediği mekândan bağlantı yapabilecek ve değişiklikleri görebilecek. Web uygulaması olmasının avantajlarından biri bu ve diğer avantajı da herhangi bir program kurma ve yükleme işlemleri yapmadan kullanabilecek. Proje arka tarafta işlemleri yapıyor ama ön yüz de görünen bir şey yok. Projenin giriş sayfasını daha önce açıklamışım giriş yaptıktan sonra gösterilecek sayfaları da göstermişim. Şimdi bu parçaları birleştirme işlemleri yapacağım.

#### 4.1.1 Express js

Express.js bir Nodejs çatısıdır. Şu an itibariyle en popüler çatıdır. Aynı kodu tekrar tekrar tekrarlamanız gerekmekz. HTTP modülüne sahip düşük seviye bir giriş / çıkış mekanizmasıdır. HTTP sunucuları için küçük ve sağlam araçlar sağlayarak tek sayfa uygulamaları, web siteleri veya genel HTTP API'leri için mükemmel bir çözümüdür.(NPM, 2018 d)

```
var express = require('express');
var app = express();
```

**Şekil 4.1** Express'i Proje Dâhil Etme

Projede kullanmak Şekil 4.1' de ki gibi tanımlama yapılmalıdır.

```
app.post('/excellread', function (req, res) {
})
app.get('/akaunting', function (req, res) {
})
```

**Şekil 4.2** Express'i Yönlendirme İşlemleri

Şekil 4.2 'de göründüğü gibi yönlendirme yapısı mevuttur. Her işlem için bir istek ve sonuç fonksiyonu çalışmakta. URL' e /akaunting olarak bir istekte bulunduğuanda yapılacak istekler burada çalışacaktır.

## 4.2 API Oluşturma

API (Application Programming Interface), bir uygulamaya ait yeteneklerin, başka bir uygulamada da kullanılabilmesi için, yeteneklerini paylaşan uygulamanın sağladığı ara yüzdür. API yapısı ile gelen isteklere göre hangi işlemler yapılacağını belirleyeceğiz. Mesela URL' e /zirve yazıldığında yada post işlemi yapıldığında zirve muhasebe senkronizasyon işlemi başlatılacak.

```
var app = express();
app.use(bodyParser.urlencoded({ extended: false }));
app.use(bodyParser.json());
app.set('view engine', 'pug');
```

**Şekil 4.3** Express ve İşlem Yöntemlerini Tanımlama

Body-Parser modülü, gönderilen post datasını obje olarak yakalamamızı sağlayan bir modüldür. Bu modül ile giriş yaparken kullanıcı bilgilerini obje olarak getirir.

- `app.use(bodyParser.json())`  
JSON veri tipinde gelecek olan verilerin kullanılabilmesi için bu tanımlamanın yapılması gerekmektedir.
- `app.use(bodyParser.urlencoded({ extended: true }))`  
Encode(kodlanmış/şifrelenmiş) edilmiş URL'ler üzerinde Body-Parser'1 kullanmak istiyorsanız eğer “extended” özelliğine true değerini vermeliyiz.

```

app.get('/excellupload', function (req, res) {
    res.render('excellupload');
})

app.get('/orderakaunting', function (req, res) {
    order();
    res.render('orderakaunting');
})

app.get('/orderakauntingch', function (req, res) {
    res.render('orderakauntingch');
})

app.get('/', function (req, res) {
    res.render('index')
})

app.get('/akaunting', function (req, res) {
    res.redirect("akauntingch")
})

app.post('/akaunting', function (req, res) {
    if (!req.body) return res.sendStatus(400)
    res.render("akaunting", req.body)
    akauntingSync();
})

app.listen(3000, () => {
    console.log("Server running on port 3000");
});

```

**Şekil 4.4** Basit API Yapısı

Şekil 4.4’ de verilen kodlarda projemizin basit api mimarisi görünmektedir. ‘orderakaunting’ sayfasına get istediginde bulunursa eğer order () fonksiyonu çalışacak ve orderakaunting HTML sayfası yüklenecektir. /akaunting sayfasına post işlemi yapılrsa eğer body’ de gelen veri var mı kontrol edilir yoksa durum kodu olarak 400 gönderilir eğer body’ de veri varsa akaunting.pug dosyası yüklenir. Sayfa yüklenirken body’ de bulunan verilerde gönderilir. Listen fonksiyonu ile 3000 portunda çalışır ve dinlenir.

### 4.3. Pug

Pug, bir HTML Preprocessor(ön işlemci)'dur. HTML kodlarını daha hızlı yazmanıza olanak tanır. Kendi yazım standarı vardır. Yazılan kodları HTML yazımına çevirir.(Pug, 2018)

```
const pug = require('pug');
app.set('view engine', 'pug');
```

**Şekil 4.5** Pug'ı Projeye Dahil Etme

Pug 'i require fonksiyonu ile projeye dâhil ediyoruz ve Nodejs' e gösterim motoru olarak pug kullanılacağını bildirmemiz gerekiyor. App.set fonksiyonu ile pug kullanılacağını söylüyoruz.

```
<html>
<head>
    <title>Örnek HTML</title>
</head>

<body>
    <p>Merhaba</p>
</body>
</html>
```

**Şekil 4.6** Html Yazım Standardı

```
html
  head
    title Örnek HTML
  body
    p Merhaba
```

**Şekil 4.7** Pug Yazım Standardı

Şekil 4.6 ve Şekil 4.7 'yi karşılaştırırsak büyük bir farklılık olduğunu görebiliriz. İkiside aynı ifadeyi temsil ediyor.

### Özellikleri:

- Layout'u extend ederek yani paylaşmamızı sağlayarak farklı dosyalarda kullanabilmemizi sağlar.

- Klasik html'de olmayan programatik yapıları kullanmamıza imkân sağlar;
  - Değişkenler
  - Döngüler
  - Kontrol Mekanizmaları
  - Diziler vs..

#### 4.4. Toplu Ürün Ekleme

Araştırmalarına göre çoğu şirket ürün eklerken Excel toplu ürün ekleme işlemi yapıyor ve bu özellik çok fazla kullanılmakta.

```
var XLSX = require('xlsx')

var workbook = XLSX.readFile(`../${data}`);
var sheet_name_list = workbook.SheetNames;
var xlData = XLSX.utils.sheet_to_json(workbook.Sheets[sheet_name_list[0]]);
console.log(xlData);

xlData.forEach(item => {

  let sql = `INSERT INTO oc_product(model, sku,
    upc, ean, jan, isbn, mpn, location,quantity,
    stock_status_id, image, manufacturer_id, shipping,
    price, points, tax_class_id, date_available, weight,
    weight_class_id, length, width, height,
    length_class_id, subtract, minimum, sort_order,
    status, date_added, date_modified)

  VALUES (
    'Product ${item.model}',
    '${item.sku}', '', '', '', '', '',
    '${item.quantity}', '1', 'NULL',
    '1', '1',
    '${item.price}', '0.0000', '1',
    '${date}', '0.00000000', '1',
    '0.00000000', '0.00000000',
    '0', '1', '1', '1', '0',
    '1',
    '${date}', '${date}')`;
```

**Şekil 4.8** Excell' den Veri Okuma

Şekil 4.8' deki kodlarda masaüstündeki Excel dosya okunur. İlk satırdaki ürün bilgileri alınır ve alt satırda ekli olan ürünler ile bağlantı yapılır en son olarak bütün

ürünleri json formatında geri döndürür. Ürünler biz dizi içindedir ve tek tek Opencart' ta 'oc\_product' tablosuna eklenir. Eklenen veriler /excelread html dosyasında gösterilir.

Toplu Ürün Ekleme

Ürünleri Excel' de (model, sku, quantity, price, name, description) şeklinde doldurunuz..

Excel Dosyasını Yükleyin.

Dosya Seç

**Yükle**

**Şekil 4.9** Excel Dosya Yükleme

Dosya Okuma Başarılı.

### Eklenen Ürünler

Eklenen ÜRünler					
<b>name</b>	<b>sku</b>	<b>model</b>	<b>quantity</b>	<b>price</b>	<b>description</b>
Xiamio Redmi Note 1	11111111	Product 45	12	100	Akıllı Telefon
Xiamio Redmi Note 2	22222222	Product 46	432	200	Akıllı Telefon
Xiamio Redmi Note 3	3333333333	Product 47	412	300	Akıllı Telefon
Xiamio Redmi Note 4	4444444444	Product 48	51	400	Akıllı Telefon
Xiamio Redmi Note 5	5555555	Product 49	41	500	Akıllı Telefon
Xiamio Redmi Note 6	666666666	Product 50	51	600	Akıllı Telefon

**Şekil 4.10** Toplu Eklenen Ürünler Listesi

## 5. SONUÇ

Proje, muhasebe uygulamaları ile Opencart(e-ticaret) uygulaması arasında bağlantı sağlayıp eş zamanlı senkronizasyon işlemlerini yapmaktadır. Amaç firmaların, şirketlerin hızlı işlemlerini gerçekleştirmek, aynı işlemleri tekrar tekrar yapmasına engel olmak ve zahmetsiz iş takibi yapmak için yapılmıştır.

Proje şuanda iki tane muhasebe uygulamasına destek vermektedir. İleriki zamanlarda bu projeyi daha popüler ve çeşitli muhasebe uygulaması eklenebilir. Aynı şekilde şuanda sadece Opencart e-ticaret sitesi ile çalışmaktadır. Daha farklı e-ticaret siteleri de eklenebilir. Bazı firmaların tek bir muhasebe uygulaması olup bir den fazla e-ticaret sitelerinde hesapları bulunmakta. Bu gibi firmalar için toplu e-ticaret sitelerine ekleme yapılabilir.

Excel' den toplu ürün ekleme özelliği bulunmaktadır. Sadece Opencart' a ekleme var. Muhasebe uygulamalarında toplu ürün ekleme var ama kullanıcı tek bir kanaldan işlemlerini yapmak isteyebilir. Bu yüzden muhasebe uygulamaları içinde toplu ürün ekleme eklenebilir.

## KAYNAKÇA

- Celiloglu, V., 2017a, <https://birbu.com/opencart-nedir/>, [Ziyaret Tarihi: 15 Ekim 2018].
- Celiloglu, V., 2017b, <https://birbu.com/opencart-nedir/>, [Ziyaret Tarihi: 15 Ekim 2018].
- Roy, 2016, <https://www.quora.com/What-is-OpenCart-1>, [Ziyaret Tarihi: 19 Ekim 2018].
- Vargonen, 2017, <https://medium.com/@vargonen/opencart-nedir-opencart-kurulum-i%C8%9F%C4%9Flemi-nas%C4%B1l-yap%C4%B1r-ef105e168b45>, [Ziyaret Tarihi: 24 Ekim 2018].
- Akaunting, 2018 a, <https://akaunting.com/tr/bilsoft-on-muhasebe-programi-karsilastirmasi>, [Ziyaret Tarihi: 12 Kasım 2018].
- Akaunting, 2018 b, <https://akaunting.com/tr/ozellikler>, [Ziyaret Tarihi: 12 Kasım 2018].
- Akaunting, 2018 c, <https://akaunting.com/docs/installation>, [Ziyaret Tarihi: 13 Kasım 2018].
- Papuççıyan, A., 2018, <https://webrazzi.com/2018/04/03/ucretsiz-on-muhasebe-programi-akaunting/>, [Ziyaret Tarihi: 21 Kasım 2018].
- Zirve, 2018, a, <http://www.zirveyazilim.net/>, [Ziyaret Tarihi: 20 Kasım 2018].
- Zirve, 2018, b, <http://www.zirveyazilim.net/genelmuh.htm>, [Ziyaret Tarihi: 20 Kasım 2018].
- Mysql, 2018, <https://dev.mysql.com/doc/refman/8.0/en/binary-log.html>, [Ziyaret Tarihi: 8 Aralık 2018].
- NPM, 2018 a, <https://www.npmjs.com/package/mysql-events>, [Ziyaret Tarihi: 13 Aralık 2018].
- NPM, 2018 b, <https://www.npmjs.com/package/msnodesqlv8>, [Ziyaret Tarihi: 15 Aralık 2018].
- NPM, 2018 c, <https://www.npmjs.com/package/json-diff>, [Ziyaret Tarihi: 19 Aralık 2018].
- NPM, 2018 d, <https://www.npmjs.com/package/mysql-events>, [Ziyaret Tarihi: 22 Aralık 2018].
- Gençay, 2018, <http://www.gencayyildiz.com/blog/node-js-express-pug-nedir-genel-bakis/>, [Ziyaret Tarihi: 29 Aralık 2018].

## ÖZGEÇMİŞ

### KİŞİSEL BİLGİLER

**Adı Soyadı** : Muhammet Varol  
**Uyruğu** : T.C  
**Doğum Yeri ve Tarihi** : Terme-01.03.1996  
**Telefon** : 5445147891  
**Faks** :  
**e-mail** : vrl.muhammet@gmail.com

### EĞİTİM

<b>Derece</b>	<b>Adı, İlçe, İl</b>	<b>Bitirme Yılı</b>
Lise	: Terme Bülent Çavuşoğlu Anadolu Lisesi	2014
Üniversite	: Selçuk Üniversitesi	2019
Yüksek Lisans :		
Doktora	:	

### İŞ DENEYİMLERİ

<b>Yıl</b>	<b>Kurum</b>	<b>Görevi</b>
2018	Taze Bilgisayar Teknolojileri	Stajyer

### UZMANLIK ALANI

Ön yüz geliştirme  
Javascript programlama

### YABANCI DİLLER

Yok

### BELİRTMEK İSTEĞİNİZ DİĞER ÖZELLİKLER

Yok

<b>Kontrol Edilecek Hususlar</b>	<b>Evet</b>	<b>Hayır</b>
Sayfa yapısı uygun mu?		
Şekil ve çizelge başlık ve içerikleri uygun mu?		
Denklem yazımları uygun mu?		
İç kapak, onay sayfası, Proje bildirimi, özet, abstract, önsöz ve/veya teşekkür uygun yazıldı mı?		
Proje yazımı; Giriş, Kaynak Araştırması, Materyal ve Yöntem (veya Teorik Esaslar), Araştırma Bulguları ve Tartışma, Sonuçlar ve Öneriler sıralamasında mıdır?		
Kaynaklar soyadı sırasına göre verildi mi?		
Kaynaklarda verilen her bir yayına proje içerisinde atıfta bulunuldu mu?		
Kaynaklar açıklanan yazım kuralına uygun olarak yazıldı mı?		
Proje içerisinde kullanılan şekil ve çizelgelerde kullanılan ifadeler Türkçe'ye çevrilmiş mi? (Latince ve Özel kelimeler hariçtir)		
Projenin içindekiler kısmı, proje içerisinde verilen başlıklara uygun hazırlanmış mı?		

Yukarıdaki verilen cevapların doğruluğunu kabul ediyorum.

Unvanı Adı SOYADI

İmza

**Öğrenci :** .....

**Danışman :** .....