

# Yazılım Geliştirme Laboratuvarı

## Umuttepe Turizm

1. Selin Avcı  
Bilişim Sistemleri Mühendisliği  
Kocaeli Üniversitesi  
Kocaeli, Türkiye

2. Helen Hacer Uzunçayır  
Bilişim Sistemleri Mühendisliği  
Kocaeli Üniversitesi  
Kocaeli, Türkiye

**Özetçe—**Bu Proje, PHP ve CodeIgniter framework'ü kullanılarak geliştirilmiş bir otobüs bilet sistemi üzerine odaklanmaktadır. Sistem, kullanıcıların bilet alımı yapmalarına olanak tanıırken, yöneticilerin biletlerle ilgili bilgi girişlerini gerçekleştirebileceği bir yönetici paneli sunar.

**Keywords —** codeigniter; web; php; framework.

### I. SİSTEM TANIMI

Sistem, kullanıcıların otobüs biletlerini çevrimiçi olarak satın almalarını sağlar. Kullanıcılar, seyahat tarihlerini seçebilir, koltuk tercihlerini belirleyebilir ve kartla ödeme işlemlerini gerçekleştirebilirler. Ayrıca, yöneticiler, otobüs seferleri ve biletlerle ilgili bilgileri yönetmek için özel bir yönetici paneline erişebilirler.

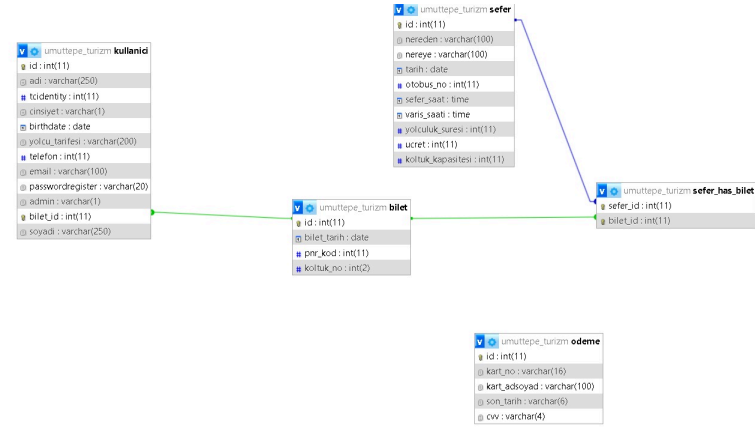
### II. VERİTABANI

Tabanında PHP ve CodeIgniter framework'ü kullanılarak geliştirilen otobüs bilet sisteminde PHPMyAdmin kullanılarak bir veritabanı yönetim aracı sağlanmıştır. PHPMyAdmin, MySQL veritabanılarını web tarayıcısı üzerinden kolayca yönetmeyi sağlayan ücretsiz ve açık kaynaklı bir araçtır. Veritabanı ile ilgili işlemleri grafiksel bir arayüz kullanarak gerçekleştirmek için tercih ettik.

PHPMyAdmin'in kolay kullanımı sayesinde, projenin veritabanıyla ilgili işlemler hızlı ve etkili bir şekilde gerçekleştirilebilir hale getirdik. Veritabanı yapılarının yönetimi ve veri manipülasyonu gibi karmaşık işlemler, kullanıcı dostu bir arayüz üzerinden rahatlıkla gerçekleştirilebilir hale getirdik.

Projenin hayata geçirdiği otobüs bilet sistemi için MySQL Workbench kullanarak veritabanı tablolarını oluşturmak kritik bir adımdır. Bu adım, sistemdeki bilet alımı, yönetim paneli ve bilet sorgulama gibi işlevlerin temelini oluşturan veritabanı yapısını sağlar. Otobüs bilet sistemi için gerekli olan veritabanı yapısını planladık. Bu, seferlerin, biletlerin, müşteri bilgilerinin ve diğer ilgili verilerin nasıl saklanacağını belirledik.

MySQL Workbench üzerinde tasarladığımız veritabanı yapısına uygun olarak tabloları oluşturduk(Şekil2.1). Bu tablolar, seferlerin bilgilerini içeren bir tablo, müşteri bilgilerini saklayan bir tablo, biletlerin bilgilerini tutan bir tablo gibi bileşenleri içerir.



Şekil2.1

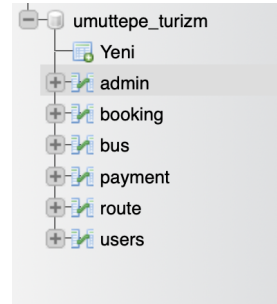
Tablolar arasındaki ilişkileri belirledik. Oluşturulan tablolar üzerinde gerekli verileri ekledik. Bu, sefer bilgileri, müşteri bilgileri ve bilet detayları gibi verileri içerir.

Bu kullandığımız tabloları en sonunda PHPMyAdmin veritabanına içe aktarma yöntemiyle veritabanımıza ekledik.

Şekil2.2

### III. SİSTEM YAPISI

MVC (Model-View-Controller), yazılım geliştirme süreçlerinde sıkça kullanılan bir tasarım desendir. Bu desen, bir uygulamanın bileşenlerini mantıksal olarak üç farklı katmana ayırarak organize eder. MVC modeli şu üç ana bileşenden oluşur:



### 1. Model (Model):

Model, uygulamanın veri katmanını temsil eder. Veritabanı işlemleri, veri manipülasyonu ve iş mantığı bu katmanda yer alır. Model, genellikle uygulamanın içinde bulunan veritabanı tablolarını temsil eden sınıflardan oluşur. Bu sınıflar, verilerin saklanması, alınması, güncellenmesi ve silinmesi gibi işlemleri gerçekleştirir.

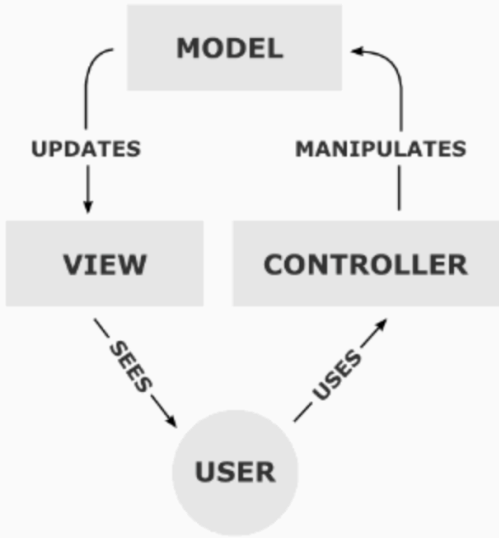
### 2. View (Görünüm):

View, uygulamanın kullanıcı arayüzünü temsil eder. Kullanıcıya sunulan HTML, CSS, JavaScript gibi dosyalar bu katmanda yer alır. View, kullanıcıya verileri gösterir ve kullanıcı tarafından yapılan işlemleri Controller'a ileterek işlemleri gerçekleştirir.

### 3. Controller (Denetleyici):

Controller, Model ve View arasında iletişimi sağlar ve uygulamanın iş mantığını yönetir. Kullanıcının isteklerini karşılar, gerekli veri işlemlerini yapar ve sonuçları kullanıcı arayüzüne yansıtır. Controller, kullanıcı girişlerini alır, bu girişleri işler, Model katmanında gerekli veritabanı işlemlerini gerçekleştirir ve sonuçları uygun View'a iletir.

MVC modelinin temel amacı, uygulamanın kod tabanını modüler hale getirerek bakımını ve genişletilmesini kolaylaştırmaktır. Ayrıca, farklı katmanların bağımsız olarak değiştirilmesine ve test edilmesine olanak tanır. Bu nedenle, birçok modern web uygulaması ve çerçeve MVC modelini kullanır. Bu model, uygulamaların daha organize ve yönetilebilir olmasını sağlar. numaralandırılacağından ayrıca sizin numaralandırmanıza gerek yoktur.



Şekil3.1

Kısaltmaları yazı içinde ilk defa kullanıldıklarında tanımlayınız. Başlıklarda kısaltma kullanmayınız. IEEE, SI, CGS vb. gibi çok bilinmiş kısaltmaları tanımlamanıza gerek yoktur.

### IV.

### CodeIgniter

CodeIgniter, PHP tabanlı bir web uygulama geliştirme çerçevesidir. Hafif yapısı, hızlı öğrenme eğrisi ve esnekliğiyle bilinir. İşte CodeIgniter yapısı hakkında kısa bir bilgi:

#### -Rutin Görevler İçin Kütüphaneler ve Yardımcılar:

CodeIgniter, sık kullanılan işlevleri gerçekleştirmek için bir dizi kütüphane ve yardımcı fonksiyon içerir. Örneğin, form doğrulama, veritabanı işlemleri, oturum yönetimi gibi işlevler için hazır kütüphaneler bulunur.

#### -Veritabanı Desteği:

CodeIgniter, birçok farklı veritabanı sistemini destekler. MySQL, PostgreSQL, SQLite gibi popüler veritabanı sistemleriyle entegrasyon sağlar.

CodeIgniter, hızlı bir şekilde web uygulamaları geliştirmek için ideal bir seçenektir. Basit ve etkili yapısıyla geliştirme sürecini kolaylaştırır ve zaman kazandırır.

#### -Rutin Görevler İçin Kütüphaneler ve Yardımcılar:

CodeIgniter, sık kullanılan işlevleri gerçekleştirmek için bir dizi kütüphane ve yardımcı fonksiyon içerir. Örneğin, form doğrulama, veritabanı işlemleri, oturum yönetimi gibi işlevler için hazır kütüphaneler bulunur.

### V.

### Özellikler

- Bilet Alımı: Kullanıcılar, sistem üzerinden otobüs biletleri satın alabilirler. Ödeme işlemleri kartla yapılır.

Rezerv İşlemleri							
ID	Kayıt Şehri	İni Şehri	Otobüs İsmi	Tarih	Zaman	Ücret	Rezerv
1	izmit	Ankara	K	2024-04-12	04:35:00.000000	1000	<a href="#">Bilet Al</a>
2	Manisa	izmit	M	2024-03-13	12:32:00.000000	1000	<a href="#">Bilet Al</a>
4	Manisa	izmir	Z	2024-03-12	12:43:00.000000	250	<a href="#">Bilet Al</a>
5	Ankara	istanbul	K	2024-04-12	12:00:00.000000	600	<a href="#">Bilet Al</a>
6	Ankara	izmir	L	2024-02-13	13:09:00.000000	1000	<a href="#">Bilet Al</a>
7	Manisa	istanbul	Q	2024-09-13	21:34:00.000000	1000	<a href="#">Bilet Al</a>

Şekil 5.1

- Yönetici Paneli: Yöneticiler, otobüs seferleri, bilet fiyatları ve diğer ilgili bilgileri yönetmek için bir yönetici paneline erişilebilir.

- PNR Kodu : Kullanıcılar, PNR kodunu tanımlar

İsim	E-mail	Adres	Şehir	Güncelle	Silme
aise	aise@gmail.com	sakarya	mumbai	Güncelle	Sil
aise	aise@gmail.com	sakarya	mumbai	Güncelle	Sil
aise	aise@gmail.com	sakarya	mumbai	Güncelle	Sil
aise	aise@gmail.com	hnbjm	ghjk	Güncelle	Sil
aise	aise@gmail.com	hnbjm	ghjk	Güncelle	Sil
aise	aise@gmail.com	hnbjm	ghjk	Güncelle	Sil

Şekil 5.2

-Bilet Sorgulama:Kullanıcı için bilet sorgusu yapma.

-Kayıt ve Login:Kullanıcının kayıt olup sistemde işlem yapabilmesi.

-Bilet İşlemleri:Bilet iptali gibi özellikler.

ID	Kalın Sahel	İniş Sahel	Person	Tarih	Zaman	Ücret	Güncelleme	Silme
1	izmit	Ankara	K	2024-04-12	04:35:00.000000	1000	Güncelle	Sil
2	Manisa	izmit	M	2024-03-13	12:32:00.000000	1000	Güncelle	Sil
4	Manisa	izmir	Z	2024-03-12	12:43:00.000000	250	Güncelle	Sil
5	Ankara	istanbul	K	2024-04-12	12:00:00.000000	600	Güncelle	Sil
6	Ankara	izmir	L	2024-02-13	13:09:00.000000	1000	Güncelle	Sil
7	Manisa	istanbul	Q	2024-09-13	21:34:00.000000	1000	Güncelle	Sil

Şekil 5.3

-Otobüs Kuş Bakışı görünümü:Otobüse kuş bakışı olarak bilet seçebilme.

-Güzergah:Kullanıcının gitmek istediği yere istediği konumdan rota oluşturabilme.

-Otobüs Koltuğu Seçimi:Otobüs koltuk seçiminde kadın yanına erkek oturması engelleniyor ve izin verilmiyor.

-Ödeme Fiyatlandırması:Ödeme fiyatlandırması kategoriye göre değişiklik gösterir her yolcuda aynı oranda ödeme alınmaz.

Şekil 5.4

-Kullanıcı Kayıt Formu Oluşturma

Kullanıcıların kayıt olabileceği bir form oluşturun. Bu form genellikle ad, soyad, e-posta, kullanıcı adı, şifre gibi temel bilgileri içerir. Form, HTML ve CSS kullanılarak tasarlanır.

-Form Verilerini Kontrol Etme:

Sunucu tarafında, gönderilen form verilerini doğrulamak için bir script yazın. Bu adımda, gelen verilerin geçerli olduğunu kontrol edebilirsiniz. Örneğin, e-posta adresinin geçerli olup olmadığını veya şifrenin belirli kriterleri karşılayıp karşılamadığını kontrol edebilirsiniz.

Şekil 5.2

-Kullanıcıyı Veritabanına Kaydetme

Doğrulama adımlarından sonra, geçerli kullanıcı bilgilerini bir veritabanına kaydedin. Bu adımda, kullanıcıya ait bilgileri bir veritabanı tablosuna ekleyebilirsiniz. Genellikle, şifrelerin güvenli bir şekilde saklanması için hashleme yöntemleri kullanılır.

-Kullanıcı Giriş Formu Oluşturma

Kullanıcıların giriş yapabileceği bir form oluşturun. Bu form genellikle kullanıcı adı/e-posta ve şifre alanlarını içerir. Kullanıcı adı/şifre doğrulaması yapılacaktır.

-Kullanıcıyı Doğrulama:

Gelen kullanıcı adı/e-posta ve şifreyi doğrulayın. Bu adımda, veritabanından kullanıcı bilgilerini alın ve gelen şifreyi hashleme algoritmasıyla karşılaştırarak doğrulama yapın.

-Oturum Yönetimi:

Başarılı bir giriş işleminden sonra, kullanıcıyı oturum yönetimi mekanizması ile oturum açık tutabilirsiniz. Bu, kullanıcının tarayıcıyı kapattığı sürece oturumunun aktif kalmasını sağlar.

-Giriş Kontrolü ve Yönlendirme:

Kullanıcı oturum açıkken, kullanıcıyı giriş sayfasına yönlendirebilir veya diğer sayfalara erişim izni verebilirsiniz. Oturum kapalıysa, giriş yapma sayfasına yönlendirme yapılabilir.

## KAYNAKLAR

- <https://www.codeigniter.com/>
- <https://tr.wikipedia.org/wiki/Model-View-Controller>

**ÖDEMLENDİRME**  
Ücret :  
200,00  
İsim Soyisim :  
aise  
Email :  
aise@gmail.com  
Adres :  
Ülke :  
Türkiye  
Şehir :  
Zip Kodu :  
ÖDEME YAP

**ÖDEME İŞLEMLERİ**  
Kart Kabul Edildi :  
PayPal MasterCard Visa  
Kart İsmi :  
Kredi Kartı Numarası :  
Biriş Ayc :  
Biriş Yılı :  
CVV :  
ÖDEME YAP