

Veritabanı Yönetim Sistemleri Proje Ödevi

Turizm Acentası-Grup 23

Muhammet İkbāl Çakır ,Ayşenur Elibüyük, Mustafa Toprak

Bilişim Sistemleri Mühendisliği

Kocaeli Üniversitesi

Umuttepe Kampüsü,41380, Kocaeli-Türkiye

(mamicakir30@gmail.com, ayselibuyuk171@gmail.com, musttoprakk@gmail.com)

Abstract - Bu proje, turizm sektöründe faaliyet gösteren bir acenta için geliştirilen Mobil ve Web tabanlı bir otomasyon programının tasarım ve uygulamasını tanıtmaktadır. Uygulamanın içeriğinde rezervasyon yapmak, rezervasyonları yönetmek, turlar sunmak, popüler otelleri görmek gibi işlemler mevcuttur. Hem mobil cihazlar hem de web tarayıcıları üzerinden kullanılabilir olması, kullanıcılara her açıdan kolaylık sağlar.

Keywords- turizm, otomasyon, mobil, web, veritabanı, turlar

1-Giriş

Turizm acentaları, müşterilere çeşitli hizmetler ve turlar (tekne turları ve yurtdışı turlar) sunmak, kolaylıkla tatil planı yapmak, istedikleri tatile rezervasyon yapmak ve müşteri memnuniyetini sağlamak için oluşturulmuş otomasyon programlarıdır. Mobil ve web projelerinde ortak veri tabanı kullanılmıştır. Veriler birbiriyle uyum sağlamaktadır. Yaptığımız programların mobil ve web tabanlı olması, kullanıcıların erişimini kolaylaştırır. Müşteriler kendi tercih ettikleri cihazlar üzerinden otelleri ve turları inceleyip diledikleri gibi rezervasyon yapabilirler. Bu çalışmanın devamında, otomasyon programlarının tasarımı, uygulaması ve sonuçları detaylı bir şekilde ele alınacaktır.

2-Problem Tanımı

Turizm sektörü, son dönemlerde yoğun rekabet göre sektörlerden biridir. Dönemimizin teknoloji dönemi olmasının yanında insanların çoğu artık işlerini internet üzerinde kolayca halletmek istemektedir. Bununla birlikte artık çoğu sorun mobil veya web üzerinden oldukça hızlı şekilde halledilebilmektedir. Turizm sektörü de buna uyum sağlamıştır. Tatil planı yapmak istiyorsanız mobil

veya web üzerinden bunu kolayca gerçekleştirebilirsiniz. Ayrıca acentanın veritabanında müşteri bilgilerinin, rezervasyon detaylarının, tur paketi bilgilerinin güvenli ve tutarlı bir şekilde yönetilmesi büyük önem taşımaktadır.

3-Projenin Yapısı

Projenin Web Yapısı

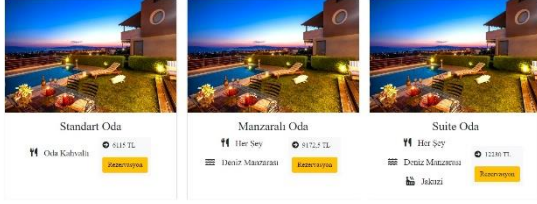
Web sitesini temel olarak; Ana Sayfa, Oteller, Tekne Turları, Yurtdışı Turlar, İletişim, Giriş Yapma Ekranı ve Ayarlar olmak üzere toplam 7 ayrı bölümden oluşur. Ana sayfada; mevcut tatil temaları, popüler bölgeler, tavsiye edilen birkaç otel ve bu sitenin belirlenmiş hizmetleri bulunmaktadır. Popüler bölgeler, swiper kullanılarak tasarlanmıştır. Her bir popüler bölge butonun içerisinde yer almaktadır. Yani istediğimiz popüler bölgeye tıkladığımızda ilgili sayfaya gitmektedir. Tavsiye edilen oteller bölümünde ilimizi çeken otele hemen rezervasyon yapılabilir.

Otellerde; veri tabanında kayıtlı olan tüm oteller, filtreleme ve sıralama işlemleri mevcuttur. Tüm oteller arasında filtreleme veya sıralama işlemi yaparak kriterlerimize uygun otellere ulaşabiliriz. Herhangi bir otelin detay sayfasına gitmek istersek yani oteli ayrıntılı şekilde incelemek istersek otel card'ının üstüne tıklamalıyız [**Şekil_1**].



[Şekil_1]

Tıkladığımızda o otele ait konum bilgisi, puan bilgisi, otel olanakları, oda çeşitleri ve genel şartlar bölümlerini görebilmekteyiz. Oda çeşitleri; standart oda, deniz manzaralı oda ve suite oda şeklinde 3 çeşide ayrılmakta [Şekil_2].



[Şekil_2]

İstenilen oda çeşidinde “rezervasyon yap” butonuna tıkladığımızda bizi rezervasyon sayfasına gönderiyor. Bu sayfaya geldiğimizde rezervasyon yapmak için kişi sayısı, giriş ve çıkış tarihi seçiyoruz. Kişi sayısı ise yetişkin ve çocuk olmak üzere 2’ye ayrılıyor. Kişi sayısı kadar önümüze doldurmamız gereken form çıkıyor. Yetişkin formlarının her birinde müşteriden isim, soyisim, doğum tarihi, telefon numarası ve e-posta adresi bilgilerini doldurulması istenmektedir. Çocuk formlarında ise telefon numarası ve e-posta adresi bilgilerine gerek olunmamaktadır. Bu işlemlerin hepsini gerekli şekilde tamamladıktan sonra rezervasyonu tamamlama butonunu kullanarak rezervasyon işlemimizi onaylayabilmekteyiz.

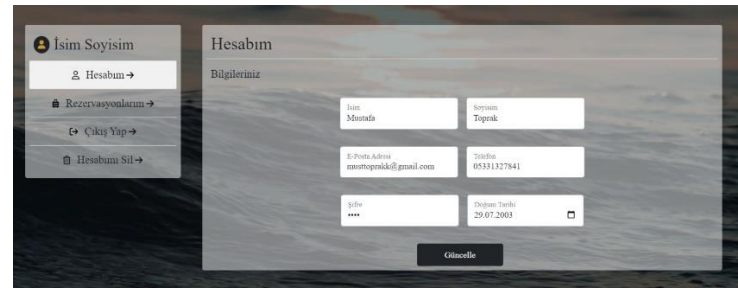
Tekne turlarında, aynı otellerde olduğu gibi mevcut turlar, filtreleme ve sıralama işlemleri vardır. Burada da kriterlerimize uygun tekne turlarını filtreleme ve sıralama işlemleri yaparak bulabiliriz. Seçtiğimiz tekne turuna tıkladığımızda tekne turu detay sayfasına ulaşırız. Bu sayfada o tekne turuna ait detaylı tur programı ve genel şartlar bölümleri vardır. Turla alakalı yeterli bilgi edindikten sonra rezervasyon yap butonuna tıklayarak rezervasyon sayfasına gidebiliriz. Burada da kişi sayısı, giriş ve çıkış tarihi bulunmakta. Otellerden farklı olarak tekne turlarının daha önceden belirlenmiş tarihleri olduğundan burada herhangi bir tarih seçmeyiz. İlgili formları doldurduktan sonra rezervasyon işlemimiz tamamlanır.

Yurtdışı turlar sayfasında, oteller ve tekne turlarında olduğu gibi filtreleme ve sıralama işlemleri vardır. Onun dışında mevcut yurtdışı turları bulunmakta. İstedığımız yurtdışı turuna tıkladığımızda bizi detay sayfasına göndermektedir. Detay sayfasında o tura ait tur programı ve genel şartlar bölümleri bulunmaktadır. İnceledikten sonra rezervasyon yap butonuna tıklayarak rezervasyon sayfasına gidebilmekteyiz. Burada da tekne turlarında olduğu gibi sadece kişi sayısını belirleyip formları doldurduktan sonra rezervasyon yap butonuyla işlemimizi tamamlayabiliriz. Tekne turları ve yurtdışı turlarının diğer ortak noktası da vize durumunun bulunması.

İletişim sayfası, form ve harita bölümlerinden oluşmaktadır. Müşteri bilgilenebilir ve soru sormak için mesaj gönderebilmektedir. Harita ise rastgele ayarlanmış bir konumu göstermektedir. Bu konumu aslında şirketin bulunduğu konum olarak düşünebiliriz.

Giriş yap ya da kayıt ol sayfasında müşteri eğer daha önce üye olmamışsa kolayca üye olabilir. Sonrasında herhangi bir rezervasyon işleminde sayfaya giriş yapması yeterli olacaktır. Son olarak ayarlar sayfası, hesabım, rezervasyonlarım, çıkış yap ve hesabımı sil olmak üzere 4 bölümden oluşmaktadır.

Hesabım kısmında [Şekil_3] müşterinin bilgileri tutulmaktadır. Müşteri istediği takdirde bilgilerini güncelleyebilmektedir.



[Şekil_3]

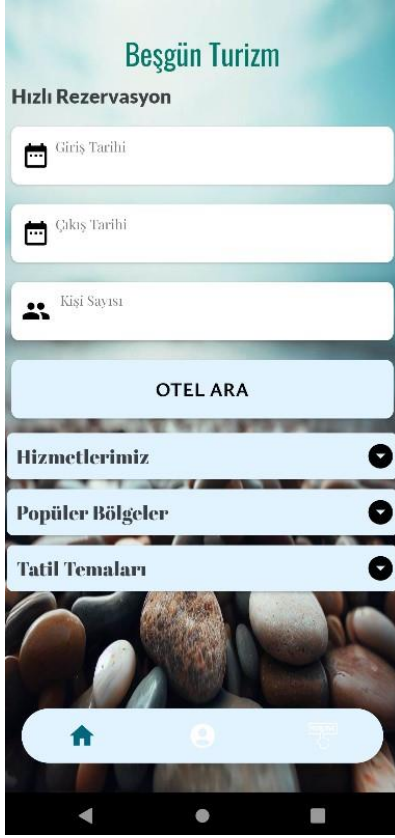
Rezervasyonlarım kısmında daha önce yapılan rezervasyonlar gözükmektedir [Şekil_4]. İstenilen rezervasyon iptal edilebilir veya güncellenebilir. Müşteri dilediği zaman hesaptan çıkış yapabilir ve hesabını silebilir.



[Şekil_4]

Projenin Mobil Yapısı

Projemizin mobil uygulama kısmında bizi çok şey bekliyor. Uygulamamızda birçok sayfa bulunuyor bunlardan bahsetmek istersek bunlar MainActivity, HizmetlerActivity, GirişYapActivity, KayıtOlActivity, AyarlarActivity, HesabımActivity, TekneTurDetayActivity, YurtdışıTurDetayActivity, OteldetayActivity, RezervasyonYapActivity vb. sayfalara sahibiz. Uygulamayı açınca bizi MainActivity sayfası karşılıyor. Bu sayfada uygulama ismi, hızlı rezervasyon, hizmetlerimiz, popüler bölgeler ve tatil temaları Cardlarını görüyoruz [Şekil_5]. Burada hızlı rezervasyon ile mobil kullanıcı hiç vakit kaybetmeden giriş tarihi, çıkış tarihi, yetişkin sayısı ve çocuk sayısı seçerek otel araması yapabiliyor.



Şekil_5

Uygulamamızı kullanan kullanıcıyı hızlı rezervasyon kısmından sonra hizmetlerimiz Cardview'i karşılıyor. Burada [Şekil_6] Şirketin yaptığı faaliyetlerden bahsediyoruz. Otel rezervasyonları, Yurt Dışı Turları , Tekne Turları olmak üzere 3 bölümden oluşuyor.Kullanıcı burada bulunan açıklamaları okuyarak şirketin yaptığı hizmetler hakkında bilgi ediniyor.İlgisini çeken ya da rezervasyon yapmak istediği bir hizmet varsa ona tıklayıp ilgili hizmetin verilerinin bulunduğu hizmetler sayfasına gidiyor. Kullanıcı MainActivity'den başka bir sayfaya henüz geçmediyse hala göreceği birkaç şey var [Şekil_7]. Bunlardan birisi Hizmetlerimiz CardView'inin altında bulunan Popüler Bölgeler ScrollView bulunuyor. Bu ScrollView'de uygulama kullanıcıya popüler bölgeleri soldan sağa doğru gösteriyor.Bu popüler bölgelerden bazıları Bodrum,Çanakkale,Fethiye,Alaçatı gibi şehirler. Kullanıcı bu şehirlerden birisine tıklarsa hizmetler sayfasında bu şehire ait oteller listeleniyor ve kullanıcı istediği şehirdeki otelleri görüyor.Popüler bölgelerin altında ise aynı popüler bölgeler gibi bir ScrollView bulunuyor. Bulunan ScrollView'de Tatil Temaları verileri yer alıyor.Bu tatil temalarından bazıları Termal Oteller, Şehir Otelleri, Butik Oteller gibidir. Kullanıcı istediği tatil temasını seçtiğinde kullanıcı ilgili tatil tema bilgisi ile hizmetler sayfasına gidiyor. Burada kullanıcının seçtiği tema ile eşlesen oteller hizmetler sayfasında listeleniyor.



Şekil_6

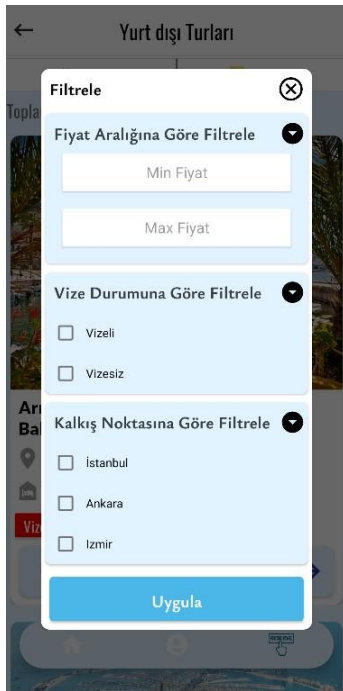


Şekil_7

Kullanıcı ilgili hizmeti (Otel Rezervasyonu, Yurt dışı Tur Seyahati ve Gemi Tur Seyahati) seçerek Hizmetler sayfasına geliyor. Hizmetler sayfamızda [Şekil_8] kullanıcıyı ilgili hizmetin verilerinin bulunduğu bir sayfa karşılıyor. Bu verileri ilgili hizmetin bilgilerinin bulunduğu bir CardView'de barındırıyoruz. Bu verilerden bahsetmek istersek bunlar hizmetin fotoğrafı, fiyatı , başlangıç tarihi , bitiş tarihi, başlangıç konumu , bitiş konumu vb. bilgilerden oluşuyor. Sayfanın en üstünde sırala ve filtrele butonları bulunuyor. Bu butonlar ilgili hizmetin verilerini istenilen filtrelemeye göre filtreliyor. Bu filtremeler Fiyatına göre, Başlangıç konumuna göre, vizesine göre, temasına göre ve puanına göre olabilir [Şekil_9]. Kullanıcı ilgili hizmetin verilerinden herhangi bir tanesine tıkladığında ilgili verinin detay sayfasına gidiyor.



Şekil_8



Şekil_9

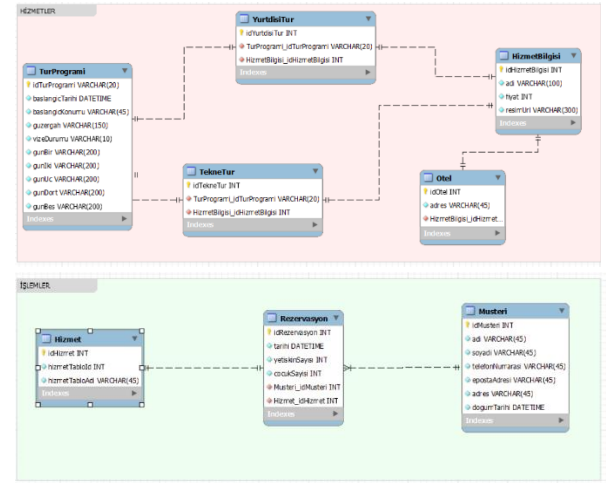
Projenin Veritabanı Yapısı

Veritabanımız, müşteri, hizmet, rezervasyon, otel, tekne turu ve yurtdışı turu gibi tabloları içermektedir. Müşteri ve rezervasyon tabloları arasında bir ilişki vardır. Her rezervasyon, bir hizmete sahiptir ve hizmet tablosunda, hizmetin adı ve hizmetin verildiği yerin (otel, yurtdışı turu, tekne turu gibi) ID'si tutulur. Otel, yurtdışı turu ve

tekne turu hizmetleri için ayrı tablolar vardır. Bu tabloların hizmet bilgisi tablosu, hizmetin adı, fiyatı, resim URL'si gibi bilgileri içerir. Tekne turu ve yurtdışı turu için özel bir tablo olan tur programı tablosu da vardır. Bu tablo, tekne turu veya yurtdışı turu hizmeti için sunulan tur programı hakkında bilgi içerir.

Bu projede oluşturulan veritabanı, işletme sahiplerinin müşteri bilgilerini, hizmetleri ve rezervasyonları yönetmelerine olanak sağlar. Bu nedenle, işletmenin günlük işleyişini kolaylaştırmak için kullanılır. Müşteriler için de bu veritabanı faydalıdır. Müşteriler, rezervasyon yaparken hizmetlerin detaylı bilgilerini, fiyatlarını ve hizmetlerin sunulduğu yerleri görebilirler. Bu sayede, müşteriler işletmenin sunduğu hizmetler hakkında daha iyi bilgilendirilirler ve karar verme süreci kolaylaşır. Sonuç olarak, bu projede oluşturulan veritabanı, işletme sahiplerinin işlerini daha iyi yönetmelerine ve müşterilerin hizmetler hakkında daha iyi bilgilendirilmelerine olanak sağlar. Bu nedenle, bu veritabanı işletme sahipleri ve müşteriler için büyük bir fayda sağlar.

Şimdi ise veritabanını görelim



Şekil_10

Veritabanımıza bakınca artık veritabanının ne işe yaradığını daha belli oldu bir Turizm Acentası uygulamalarının verilerini tuan ve müşteri işlemlerini depolayan veritabanıdır. Bu veritabanımızın tablolar dışında bir çok işlevi daha vardır örnek olarak burada gözükmeyen bir tablomuz daha vardır bu da müşteri bilgileri değişirse yeni bilgisini tutan bir trigger tablosudur. Adı "newmusteri" adlı bir tablodur ve şu şekildedir [Şekil_11]

| Column Name | Datatype | PK | NN | UQ | B | UN | ZF | AI | G |
|-----------------|-------------|--------------------------|-------------------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|
| idNewMusteri | INT | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| adi | VARCHAR(45) | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| soyadi | VARCHAR(45) | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| sifre | VARCHAR(45) | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| telefonNumarasi | VARCHAR(45) | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| epostaAdresi | VARCHAR(45) | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| dogumTarihi | DATETIME | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| idMusteri | INT | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |

Şekil_11

Göründüğü gibi burada müşteri bilgileri ile aynıdır sadece ekstra olarak kendi Id'si vardır ama PrimaryKey değildir sadece referans edeceğimiz tablonun PrimaryKey'ini ilgili trigger tablosuna ForeignKey olarak tanımlamamız gerekiyor burda ise idMusteri değişkeni ForeignKey olarak tanımlanmıştır. Bu tablomuzun işlemlerine elle müdahale etmeyiz çünkü veritabanı bunu bir trigger olarak kullanır ve istenilen durum tetiklendiğinde kendisi otomatik olarak çalışır. Kodunu göstererek nerede tetiklendiğini açıklayalım.

```
DELIMITER $$
create trigger after_reservasyon_update after update on
musteri
FOR EACH ROW
BEGIN
INSERT INTO newmusteri (adi,soyadi,sifre,telefonnumarasi,epostaAdresi,dogumTarihi,idMusteri)
VALUES (old.adi,old.soyadi,old.sifre,old.telefonnumarasi,old.epostaAdresi,old.dogumTarihi,old.idMusteri);
END$$
DELIMITER ;
```

Şekil_12

Burada musteritablosunda eğer bir müşteri güncellenirse o müşterinin yeni bilgilerini tutan bir trigger olarak kullanılmıştır [Şekil_12] .

Trigger dışında veritabanımızda bulunan bir başka şey ise arama işlemleri yapılırken daha kolay işlem yapılsın diye kullandığımız Indexlerdir [Şekil_13]. Bunlarda şöyledir.

```
CREATE INDEX idx_eposta ON musteritablosu (epostaAdresi);
CREATE INDEX idx_sifre ON musteritablosu (sifre);
select epostaAdresi from musteritablosu where epostaAdresi="musticane@gmail.com";
select sifre from musteritablosu where sifre ="123";
```

[Şekil_13]

Burada görünüşü gibi musteritablosunda epostaAdresi ve sifresine bir index atanmıştır bu atamanın sebebi kullanıcı girişi yaptığında daha hızlı bir şekilde kullanıcının girişi yapmasını sağlar. Bu nasıl olur diye sorarsak indexler bir tablonun istenilen sütünuna index atar ve bu index o sütün ile işlem yapılırsa diğer yerlere bakmadan ilgili sütün üzerinden direkt işlem yapar ve bize hız kazandırır.

Veritabanımızda kullandığımız bir başka şey ise Viewlerdir [Şekil_14] . Bu viewler uygulamanın kodunu yazarken ve uygulamasında çok şey yarayan bir işlem olmuştur. Çünkü viewler ilişkili tabloları birleştirerek bir tabloymuş gibi davranır ve kullanışı çok kolaylaşır biz kendi veritabanımızda hizmetlerimizi birleştirerek bir hizmetin herhangi bir bilgisine direkt o viewde görebilir

hale geliyoruz ve bu çok kullanışlıdır.

5 • SELECT * FROM turizmacentasi.otelview;

| idOtel | adres | puan | tema | adi | fiyat | resimUrl |
|--------|---------------------------|------|-------------|---------------------------------|-------|-----------------------------|
| 1 | Torba, Bodrum, Muğla | 7.5 | Balayı Otel | Voyage Torba | 5000 | https://images.etstur.com/ |
| 2 | Çamyuva, Kemer, Antalya | 8.6 | Termal Otel | Asteria Kemer Resort | 3100 | https://images.etstur.com/ |
| 3 | Ürgüp, Kapadokya, Muğla | 9.2 | Butik Otel | Sacred House Hotel | 6115 | https://cdng.jollytur.com/f |
| 4 | Bafra, Samsun | 7.2 | Balayı Otel | Nuhun Gemisi Deluxe Hotel & Spa | 2850 | https://images.etstur.com/ |
| 5 | Selimiye, Marmaris, Muğla | 7.9 | Termal Otel | Les Terrasses De Selimiye | 2162 | https://cdng.jollytur.com/f |
| 6 | Gayrettepe, İstanbul | 8.5 | Şehir Otel | Melas Hotel İstanbul | 4232 | https://cdng.jollytur.com/f |
| 7 | Abant, Bolu | 9.7 | Tatil Köyü | Abant Lotus Otel | 2562 | https://cdng.jollytur.com/f |
| 8 | Osmangazi, Bursa | 8.7 | Tatil Köyü | Grand Bursa Hotel | 1200 | https://cdng.jollytur.com/f |
| 9 | Bornova, İzmir | 8.2 | Şehir Otel | Villa Levante Hotel | 3203 | https://cdng.jollytur.com/f |
| 10 | Sapanca, Sakarya | 6.5 | Termal Otel | Elite World Grand Sapanca Hotel | 3795 | https://cdng.jollytur.com/f |

5 • SELECT * FROM turizmacentasi.tekneturview;

| idTekneTur | adi | fiyat | resimUrl |
|------------|---|-------|---|
| 1 | Bavyera'nın Güzellikleri: Münih, Regensburg, Nü... | 44000 | https://cdn2.setur.com.tr/image/tour/large/cru... |
| 2 | Sırbistan'ı Keşfedin: Sava Nehri, Derdap Milli Par... | 21000 | https://wp.oggusto.com/wp-content/uploads/2... |
| 3 | Hollanda'nın Tarihi Kanallarında Geleneksel Köyle... | 27000 | https://www.karavancruises.com/shipphotos/fh... |
| 4 | Panama Macerası: Kanal Keşif, Adaların Keyfi ve ... | 34000 | https://images.gemiturkari.info/gemi/turlar/fvisio... |
| 5 | Batı Amerika'nın En İkonik Şehirleri: Los Angeles, ... | 57000 | https://www.goldenbaytour.com/UserFiles/Ima... |
| 6 | Ege Denizi'nin İncileri: Santorini, Mykonos, Rodos... | 43000 | https://ucdn.tatilbudur.net/tur/Tour-Category/... |
| 7 | Ege Adaları'nın Tarihi ve Doğal Güzellikleri: Kuşa... | 49000 | https://library.coraltatil.com/resources/pageIm... |
| 8 | Baltık Denizi'nin İncileri: Helsinki, St. Petersburg, ... | 40000 | https://www.karavancruises.com/uploads/baltik... |
| 9 | Tropical Paradise: Bahamalar, Jamaika, Kaiman ... | 70000 | https://images.gemiturkari.info/gemi/turlar/coco... |
| 10 | Beş Günde Macaristan'ın En Güzel Şehirleri: Bud... | 56000 | https://www.mngturizm.com/images/tour/1214... |

5 • SELECT * FROM turizmacentasi.yurtdisiturview;

| idYurtdisiTur | adi | fiyat | resimUrl |
|---------------|--|-------|--|
| 1 | Arnavutluk:Keşfedilmeyi Bekleyen Balkan Turu | 14000 | https://media.istockphoto.com/id/529358575/tr/... |
| 2 | Dubai Turu:Lüks,Macara ve Modernitenin Buluş... | 18000 | https://cdnp.flypgs.com/files/Sehirler-long-tail/... |
| 3 | İtalya:Tarihi, Lezzetli ve Sanat Dolu Bir Akdeniz ... | 24000 | https://media.istockphoto.com/id/539115110/tr/... |
| 4 | Bulgaristan: Tarihi ve Kültürel Zenginlikleriyle Dol... | 8000 | https://media.istockphoto.com/id/621112110/tr/... |
| 5 | İran: Doğunun Tarihi, Doğal Güzellikleri ve Kültü... | 25000 | https://media.istockphoto.com/id/1134992307/... |
| 6 | İspanya: Lezzetli Mutfak, Renkli Festivali ve ... | 20000 | https://cdn.ytur.net/fit-in/697x465/filters:quali... |
| 7 | Sırbistan: Bir Balkan Kaçamağı | 10000 | https://media.istockphoto.com/id/1326144217/... |
| 8 | Yunanistan: Antik Harikaları, Güneşli Adaları ve ... | 30000 | https://media.istockphoto.com/id/1145450965/... |
| 9 | Maldivler: Beyaz Kumlu Plajları, Mercan Resifleri ... | 50000 | https://media.istockphoto.com/id/155139968/tr/... |
| 10 | Mısır: Antik Harikaları, Nil Nehri'ni Keşif ve Kültür... | 29000 | https://blog.obilet.com/wp-content/uploads/20... |

[Şekil_14]

Veritabanımız genel olarak bu işlemler uygulanarak tasarlanmıştır. Ancak bu veritabanımızı iki farklı platformda kullandığımız için farklı şekillerde bağlantısını sağladık. Bağlantılar iki şekildedir biri web üzerinden kullanılır biri ise mobil uygulamadan kullanılır. İlk olarak web nasıl bir bağlantı sağladığımızı açıklayalım.

Web uygulamamızı bir MVC core kullanılarak oluşturulmuştur ve MVC ile veritabanı bağlantısı bir ConnectionString üzerinden alınan bilgilerle veritabanına erişerek ilgili veritabanımıza bir migration oluşturur ve veritabanımız ile bağlantısını kurmuş oluruz bağlantı şekili şu şekildedir

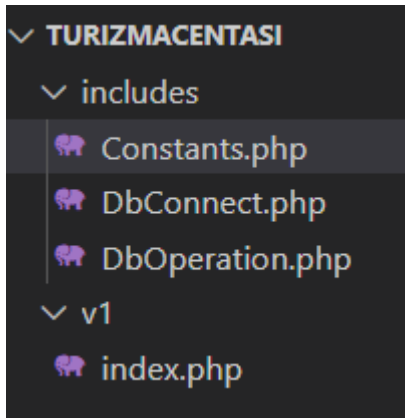
```
"ConnectionStrings": {
  "Default": "server=localhost;user=root;database=turizmacentasi;password=;port=3306"
},

// Add services to the container.
builder.Services.AddDbContext<MyDbContext>(options =>
{
    options.UseMySQL(builder.Configuration.GetConnectionString("Default"));
});
builder.Services.AddControllersWithViews();
```

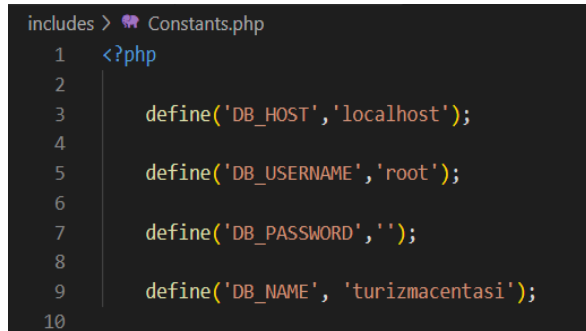
Burada ConnectionStrings ile bağlantı bilgileri alınır ve uygulamanın Program.cs bölümünde Services üzerinden bağlantısı sağlanır bağlantı şekli bir DbContext classını miras alan MyDbContext üzerinde tutulur ve migrationlar bu class ile yönetilir. Ve Mysql bağlantısı sağlanır.

Şimdi ise mobil bağlantımızın nasıl olduğunu açıklayalım . Mobil uygulamamız bir kotlin dili üzerinden yazılmıştır. Kotlin dili üzerinden bağlantı sorunları yaşadığımız için mobil uygulamamıza veritabanı bağlantısı bir API üzerinden sağladık API kontrolünü ise PHP kodları ile GET,POST işlemleri ile kontrol ederek veritabanı işlemlerini sağladık size şimdi nasıl bir bağlantı yaptığımızı göstereyim.

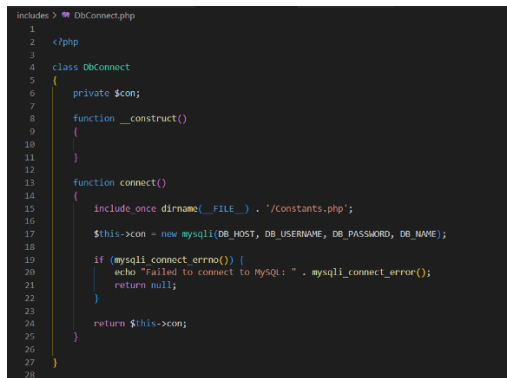
1.Adım



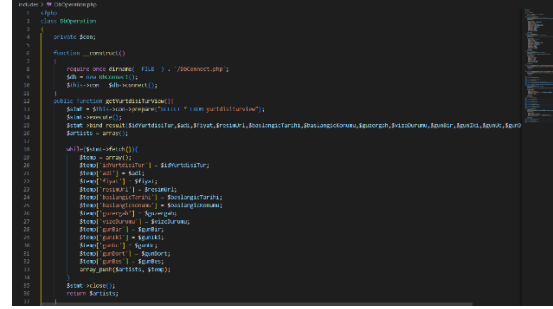
2.Adım



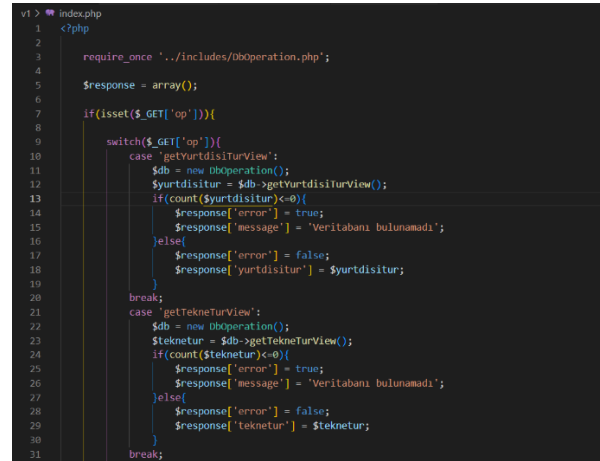
3.Adım



4.Adım



5.Adım



Burada bağlantı şu şekildedir veritabanımız bir localhost oluşturarak yapılmıştır localhost işlemlerini xampp adı verilen bir uygulama ile yönettik xampp uygulamasını hem web için hem de mobil için kullanılan API için kullandık.

Xampp içinde bulunan dosyalar ile bir API oluşturduk

API dosyaları görüldüğü gibidir ilk önce veritabanı bağlantısı sağlanır ve o bağlantı üzerinde istenilen sql sorguları ile kontrol edilerek veriler API'ye çekilir ve yazılır ilgili url üzerinden mobil uygulama ile POST ve GET işlemleri ile veritabanı işlemleri tamamlanır örnek olarak işlemleri nasıl yaptığımızı açıklayalım [Şekil_15].



Şekil_15

4-Yazılım Mimarisi

Yazılım mimarisi, bir yazılım sisteminin bileşenleri ve bu bileşenler arasındaki ilişkilerin yapısal bir düzenlemesi olarak tanımlanabilir. Yazılım mimarisi, sistemlerin daha kolay anlaşılmasını, geliştirilmesini, sürdürülmesini ve ölçeklendirilmesini sağlamak için tasarlanmıştır.

Projemizde Yazılım mimarisi 3 temel başlıktan oluşuyor.

1. Kullanıcı arayüzü: Bu katman, müşterilerin ve turizm acentası çalışanlarının yazılımı kullanması için tasarlanmış bir arayüzdür. Bu arayüz, müşterilerin seyahat planlarını yönetmelerine ve turizm acentası çalışanlarının müşteri taleplerini ve seyahat planlarını takip etmelerine olanak tanır.
2. Uygulama mantığı: Bu katman, turizm acentası yazılımının iş mantığını barındırır. Bu katmanda, müşteri verileri, seyahat planları, fiyatlandırma bilgileri vb. gibi tüm veriler yönetilir. Turizm acentası çalışanları, bu katman üzerinden müşteri verilerini düzenleyebilir ve yeni seyahat planları oluşturabilir.
3. Veri tabanı: Bu katman, turizm acentası yazılımının tüm verilerini saklar. Bu veriler, müşteri bilgileri, seyahat planları, bilet rezervasyonları ve diğer ilgili bilgileri içerir. Bu katman, uygulama mantığı katmanı ile etkileşim halindedir ve verilerin güvenli bir şekilde depolanmasını sağlar.

Yazılım mimarisini uygulamak için birçok yazılım mimarisi deseni vardır. Bunlardan bazıları MVC ve MVVM'dir. Biz projemizde MVC desenini kullandık. Mvc deseninin Turizm için nasıl kullandığımızdan bahsetmek gerekirse bunu şöyle açıklayabilirim.

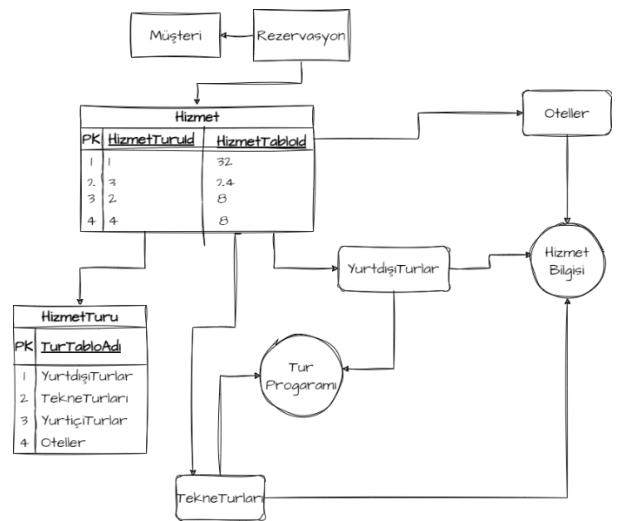
1. Model: Seyahat paketleri, rezervasyonlar, müşteri bilgileri ve diğer verilerin işlenmesi ve saklanması için gereken veri tabanı veya veri kaynağıdır. Bu bileşen, verilerin mantıksal yapısını temsil eder ve uygulama kodu tarafından erişilir.
2. Görünüm: Müşterilerin tur seçimleri, seyahat tarihleri, otel tercihleri ve diğer seyahat detaylarını seçebilecekleri ve ayarlayabilecekleri web sayfası veya mobil uygulama arayüzüdür. Bu bileşen, müşterilerin tur seçimlerini yapmalarına ve seyahatleri hakkında bilgi edinmelerine olanak tanır.
3. Kontrolör: Müşterilerin seyahat paketi seçimlerini, ödeme işlemlerini ve diğer etkileşimleri yönetir. Bu bileşen, müşterilerin yaptığı işlemleri yakalar, veri modelinde gerekli güncellemeleri yapar ve sonuçlarını müşterilere gösterir.

Örneğin, bir müşteri bir hizmet için rezervasyon yapmak isterse, Kontrolör hizmet seçimini alır, Model veri tabanında hizmet bilgilerini günceller ve Görünüm, müşteriye seyahat planı hakkında bilgi verir.

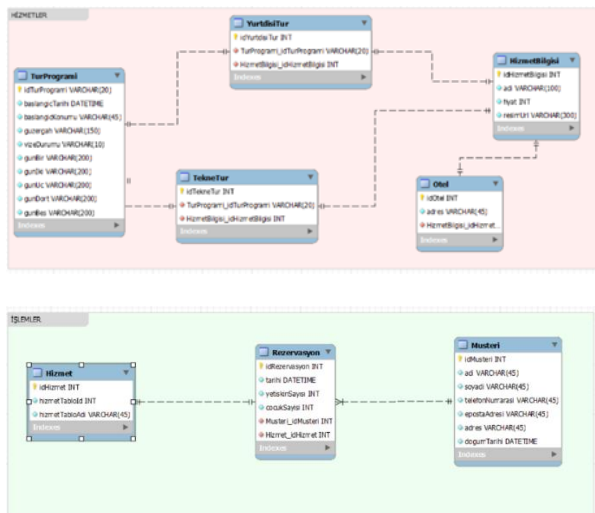
5-Yapılan Araştırmalar

Projemizi kodlarken birçok hata aldık. Bunlardan bizi en çok uğraştıran, en çok vaktimizi alan şey veri tabanını mobil uygulamaya bağlamak olayıydı. Birçok kaynağa göz gezdirdik, birçok video izledik, okulumuzdan birçok kişiye sorduk fakat günlerce kimse bize bu konuda yardımcı olamadı. Fakat biz umudumuzu kaybetmeyip tekrar araştırmaya devam ettik. Mysql veritabanının ara bağlantısını Php ile yazacağımızı sene başında hocamız bize söylemişti. Buradan yola çıkarak araştırmalarımıza devam ettik. İşin ilerleyen kısmında işimize yarayan birçok kaynak bulduk. Bu kaynaklardan yola çıkarak mobil-veri tabanı ve web – veritabanı bağlantımızı kurduk. Bu bağlantıda 3 farklı sayfalık bir php kodu yazdık. Bunların adı DbConnect, DbOperation ve Indexdir. DbConnect adındanda belli olacağı gibi veritabanına bağlanma sayfasıdır. Veritabanına bağlanırken yazılması gereken fonksiyonları bu sayfaya yazıyoruz ve bir istek gönderiyoruz. İstek başarılı ise veri tabanına bağlantımız başarılı bir şekilde tamamlanmış demektir eğer başarısız ise fonksiyon geriye null döndürür. DbOperation sayfası, veri tabanında yapmak istediğimiz işlemlerin fonksiyonlarını yazdığımız kısımdır. Bu kısımda veri çekebilir, sorgu yazabilir, veri silebilir veya güncelleyebiliriz. Index sayfasında ise DbOperation tarafından istediğimiz fonksiyonu kullandığımız kısımdır. İlgili fonksiyonu index sayfasında çağırarak isteği gerçekleştiriyoruz.

6-Akış Şeması



7-Veri tabanı Diyagramı



8-Referanslar

1. <https://www.php.net>
2. <https://www.w3schools.com/js/>
3. <https://stackoverflow.com>
4. <https://kotlinlang.org/docs/home.html>
5. <https://developer.android.com/docs?hl=tr>
6. <https://getbootstrap.com/docs/5.1/utilities/flex/>
7. <https://swiperjs.com/>
8. <https://learn.microsoft.com/tr-tr/aspnet/mvc/>
9. <https://www.apachefriends.org/docs/>
10. <https://www.udemy.com/course/asp-net-core-api-web-cok-katmanli-mimari-api-best-practices/>

