**Giriş:**

Bu proje, restoran yönetimi web siteleri odaklanmaktadır. Proje, restoran rezervasyonu, personel yönetimi, restoran bilgileri yönetimi, yemek yönetimi ve müşteri değerlendirmeleri gibi çeşitli işlevleri içermektedir. Bu kapsamlı proje, restoran işletmecilerinin işlerini daha verimli ve organize bir şekilde yönetmelerine yardımcı olmayı amaçlamaktadır.

OpenTable ve ResDiary gibi olacak.

**Gereksinimler:**

**Rezervasyon Yönetimi**: Müşterilerin online olarak rezervasyon yapabilmeleri,Rezervasyonların onaylanması ve yönetimi, Müşterilere rezervasyon hatırlatmaları gönderilmesi.

**Personel Yönetimi**:Personel bilgilerinin kaydedilmesi ve güncellenmesi, Vardiya ve çalışma saatlerinin planlanması, Personel performans değerlendirmelerinin yapılması.

**Restoran Bilgileri Yönetimi:** Restoranın genel bilgilerinin (adres, iletişim bilgileri, çalışma saatleri) yönetimi.

**Yemek Yönetimi** Menü oluşturma ve güncelleme, Yemeklerin malzemeleri, fiyatları ve besin değerleri gibi detaylı bilgilerinin yönetimi, Günün özel yemeklerinin tanıtılması.

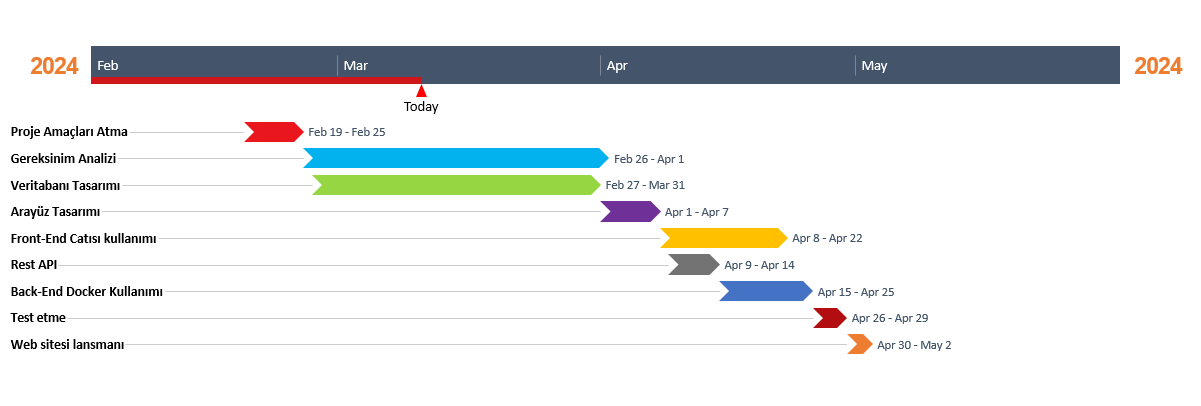
**Müşteri Değerlendirmeleri:** Müşterilerin yemekler ve hizmet hakkında değerlendirme yapabilmesi, Değerlendirmelerin analiz edilmesi ve geri bildirimlerin dikkate alınması.

**Frontend Geliştirme:**

Web arayüzü için Next.js kullanılarak kullanıcı dostu ve modern bir arayüz geliştirilecek.

**Backend Geliştirme:**

Node.js kullanılarak API'ler oluşturulacak. - Veritabanı (MongoDB) entegrasyonu sağlanacak. Authentication:JWT (JSON Web Tokens) kullanılarak güvenli kullanıcı kimlik doğrulama sistemi oluşturulacak

**Proje Planı:**

**Kulancı Senaryosu**

Senaryo Adı: DineWisel

Ana Aktör: Kullancı

ön Koşullar

.Deneyimli proje ekibi.

.Uygun donanım ve yazılım kaynakları.

.Yasal veri koruma uyumu.

.Belirlenen bütçe ve zaman çerçevesi.

Hadef Şartlar

.Rezervasyon Yönetimi

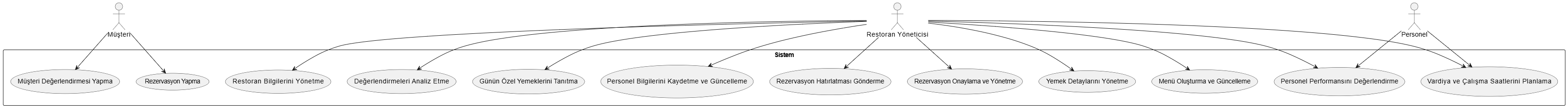
.Personel Yönetimi

.Restoran Bilgileri

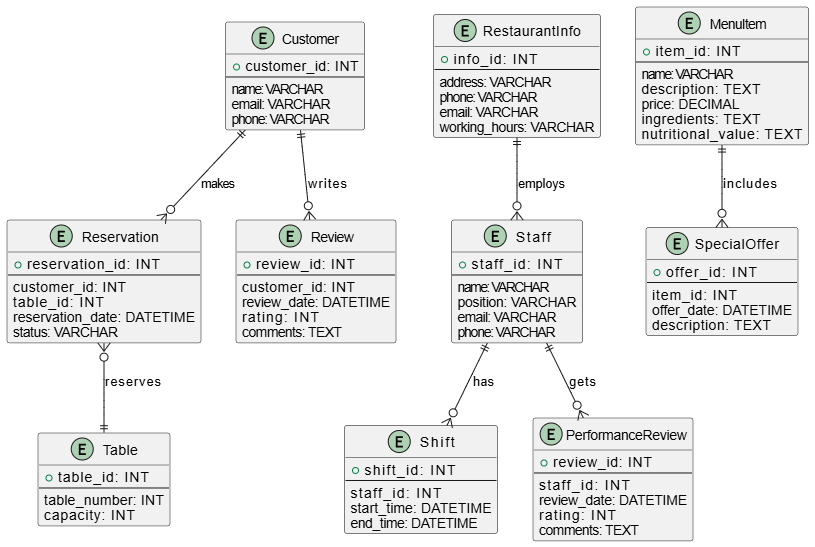
.YönetimiYemek Yönetimi

.Müşteri Değerlendirmeleri

**ardışıl diyagramı:**

****

**Veritabanı Tasarımı:**

****

**Backend Geliştirme:**

1. Next.js ve Node.js Kullanımı:

Next.js: Frontend ve backend'i aynı proje içerisinde barındıran, sunucu tarafında render edilebilen (SSR) bir React framework'üdür. Aynı zamanda API routes özelliği sayesinde backend API endpoint'leri de oluşturulabilir. Node.js: JavaScript kodlarını sunucu tarafında çalıştırmak için kullanılan bir runtime ortamıdır.

2. API Routes:

Next.js API Routes: Next.js'in sağladığı API Routes özelliği ile her bir dosya otomatik olarak bir endpoint olarak işlev görür. Bu sayede API oluşturulmasına gerek kalmadan API endpoint'leri oluşturulabilir.

3- JWT (JSON Web Tokens):

- Kimlik Doğrulama: Kullanıcı oturumları için güvenli kimlik doğrulama sağlandı. JWT ile token bazlı oturum yönetimi yapıldı.

**Veritabanı Yönetimi:**

MongoDB:

- Veri Modelleme: Mongoose kullanılarak MongoDB ile veri modelleme yapıldı. Esnek şema yapıları sayesinde, çeşitli veri türleri kolayca yönetildi.

- Veri İlişkileri: Gerektiğinde, referans ve yerleşik veri ilişkileri kurularak kompleks veri yapıları yönetildi.

**Kullanılan Yazılımsal Teknolojiler**

1. **MongoDB**

-Açıklama MongoDB, NoSQL tabanlı bir belge veritabanıdır. İlişkisel veritabanlarının aksine, esnek bir şema yapısına sahiptir ve JSON benzeri belgeleri depolar.

-Çalışma Prensibi MongoDB, koleksiyonlar ve belgeler şeklinde veri saklar. İlişkisel veritabanlarından farklı olarak, MongoDB'de tablo ve sütunlar bulunmaz. Bu sayede, veri modelleme daha esnektir ve gerektiğinde kolayca değiştirilebilir.

-Sürüm Farklılıkları Proje için belirli bir MongoDB sürümü belirtilmediği için, projenin başlangıcında en güncel sürüm ( 5.0) kullanılabilir. MongoDB, düzenli olarak güncellenir ve yeni sürümlerinde performans iyileştirmeleri ve yeni

özellikler sunar.

# 2.Multer

-Açıklama Multer, Node.js ortamında dosya yükleme işlemlerini kolaylaştıran bir middleware'dir.

-Çalışma Prensibi Multer, Express.js ile entegre edilerek kullanılır. Formdan gelen dosyaları işleyerek istemciye kaydedilmesi gereken dosyaları sunucuya aktarır.

-Dosyaları işlemek için çeşitli seçenekler sunar (örneğin, dosya türüne göre filtreleme, dosya adını özelleştirme).

-Sürüm Farklılıkları Multer sürümü, proje başlangıcında en son sürüm (şu anki sürüm: 2.1.0) olabilir. Bu sürüm, yeni özellikler veya hata düzeltmeleri içerebilir.

# 3.Next.js

-Açıklama Next.js, React uygulamalarını geliştirmek için kullanılan bir JavaScript framework'tür. Server-side rendering, static site generation ve client-side routing gibi özellikleri destekler.

-Çalışma Prensibi Next.js, projelerine bir dizi geliştirme kolaylığı ve performans artışı sağlar. Server-side rendering ile daha hızlı yükleme süreleri elde edilir ve SEO dostu uygulamalar geliştirilebilir.

-Sürüm Farklılıkları Proje kapsamında, Next.js'in en son sürümü (şu anki sürüm: 11.1.2) kullanılmış olabilir. Bu sürüm, yeni özellikler ve performans iyileştirmeleri içerebilir.

# 4.Tailwind CSS

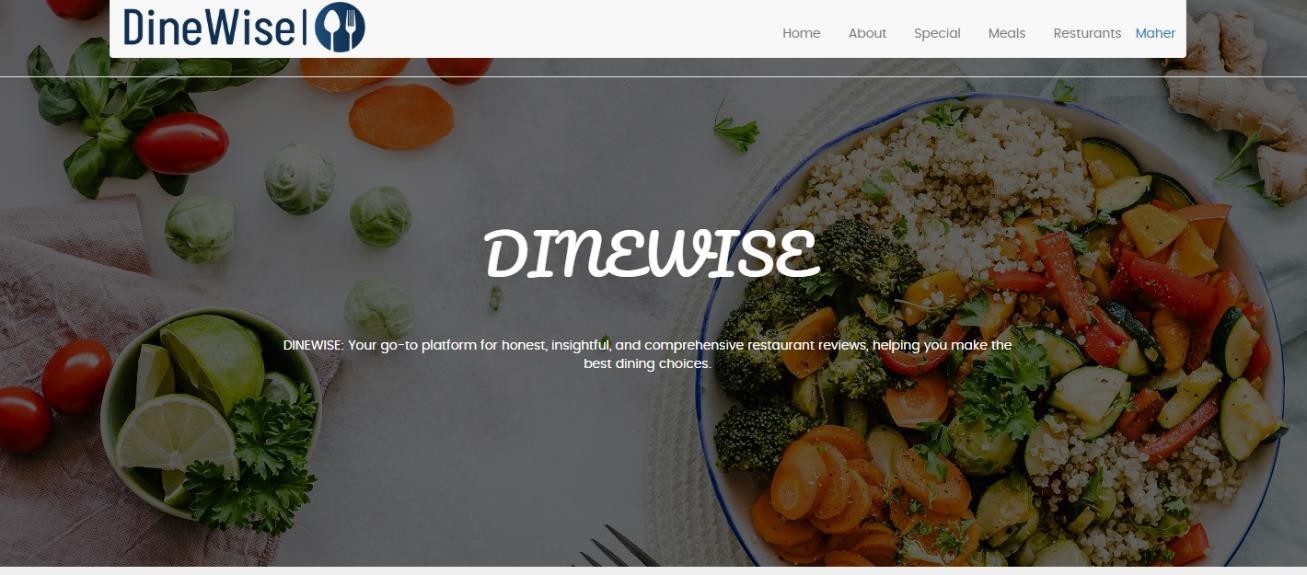
Açıklama: Tailwind CSS, CSS yazarken yeniden kullanılabilir bileşenler (component) ve utility sınıflarını kullanarak hızlı bir şekilde stil oluşturmayı sağlayan bir CSS çerçevesidir. Çalışma Prensibi: Tailwind CSS, HTML elementlerine doğrudan uygulanan utility sınıfları kullanarak stil tanımlamalarını kolaylaştırır. Bu, geleneksel CSS yöntemlerine göre daha hızlı ve verimli bir çalışma sağlar. Sürüm Farklılıkları: Proje kapsamında Tailwind CSS'in en son sürümü (şu anki sürüm: 3.0.19) kullanılmış olabilir. Bu sürüm, yeni özellikler ve geliştirmeler içerebilir..

**5.Bootstrap**

Açıklama: Bootstrap, responsif ve mobil uyumlu web tasarımları oluşturmak için kullanılan popüler bir HTML, CSS ve JavaScript kütüphanesidir. Çalışma Prensibi: Bootstrap, önceden tanımlanmış bileşenler ve stil şablonları kullanarak web sayfalarının hızlı bir şekilde oluşturulmasını sağlar. Grid sistemine dayalı olarak çalışır ve kolaylıkla özelleştirilebilir. Sürüm Farklılıkları: Proje kapsamında Bootstrap'in en son sürümü (şu anki sürüm: 5.3.0) kullanılmış olabilir. Bu sürüm, yeni bileşenler, stil güncellemeleri ve iyileştirmeler içerebilir.

**6.Next Auth**

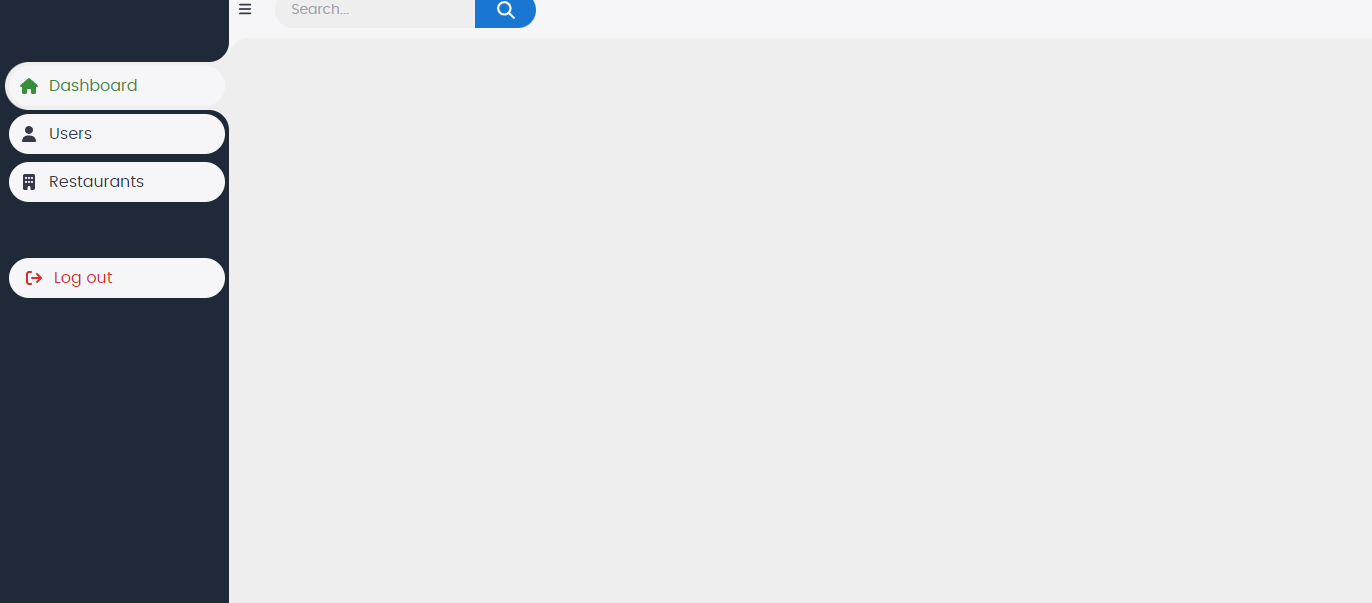
Açıklama: Next Auth, Next.js projeleri için kimlik doğrulama ve yetkilendirme işlemlerini kolaylaştıran bir kimlik doğrulama kütüphanesidir. Çalışma Prensibi: Next Auth, OAuth ve diğer kimlik sağlayıcılarını (Google, Facebook, Twitter vb.) kullanarak kullanıcı kimlik doğrulama süreçlerini yönetir. Next.js projelerinde kolayca entegre edilebilir ve çeşitli kimlik doğrulama yöntemlerini destekler. Sürüm Farklılıkları: Proje kapsamında Next Auth'in en son sürümü (şu anki sürüm: 4.0.0) kullanılmış olabilir.

****

# Şekil (Ana Sayfa )User sayfası

**Ana Sayfa İçeriği ve Tasarımı:**

- Başlık ve Logo : Sayfanın üst kısmında "DineWise" logosu ve basit bir navigasyon menüsü (Home, About, Special Meals, Restaurants, userLogin) bulunmaktadır. - Arka Plan Görseli : Yemek temalı bir arka plan görüntüsü kullanılmış. Bu görsel, ziyaretçilerin ilgisini çekmek ve platformun amacını vurgulamak için seçilmiş. - Tanıtım Metni : "DINEWISE: Your go-to platform for honest, insightful, and comprehensive restaurant reviews, helping you make the best dining choices." Bu metin platformun ne işe yaradığını ve ziyaretçilere nasıl yardımcı olacağını açıklıyor.

****

# Şekil- user dashboard

# 

**Dashboard Sayfası:**

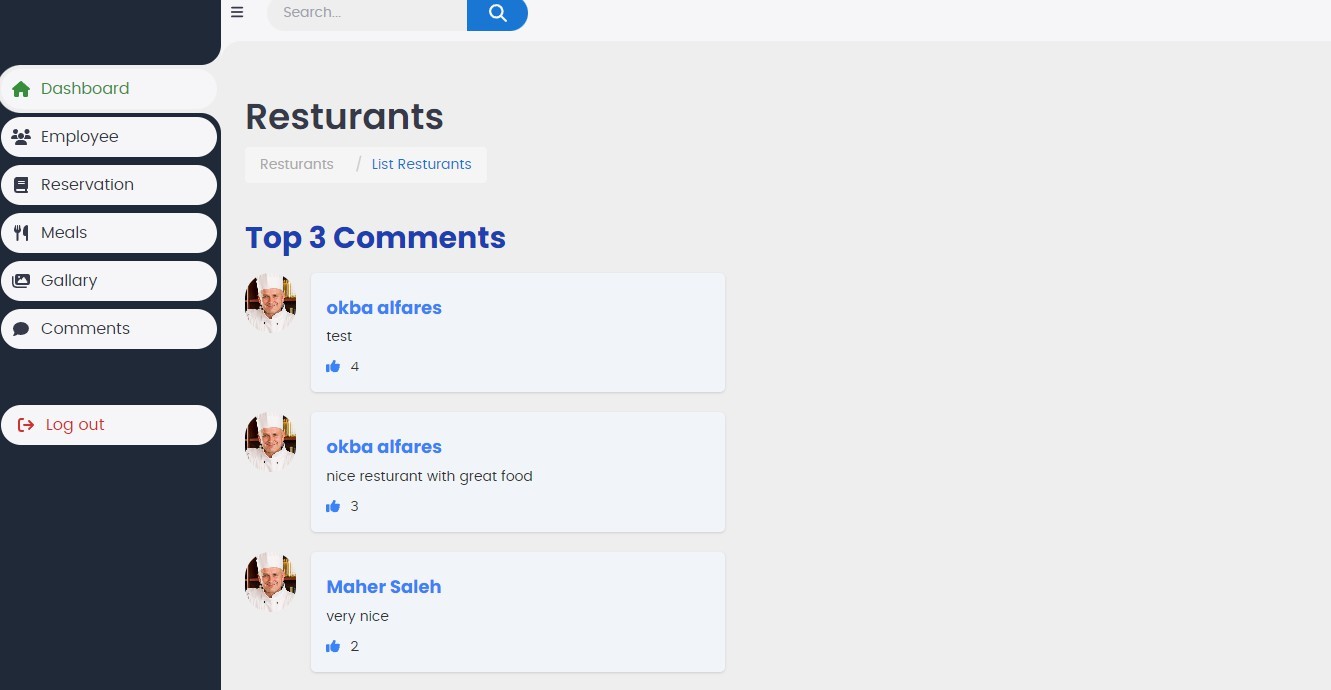
Dashboard sayfası kullanıcının ana sayfasıdır ve genel bir bakış sağlar. Bu sayfada kullanıcının işlemlerini hızlıca gerçekleştirebileceği önemli özet bilgiler ve araçlar bulunur.

**Users Sayfası:**

Bu sayfa tüm kullanıcıların listelendiği ve yönetildiği bir alandır. Bu sayfada kullanıcılarla ilgili ayrıntılı bilgiler, düzenleme seçenekleri ve bazen kullanıcıları filtreleme veya arama özellikleri de olur.

**Restaurant Sayfası:**

Bu sayfa bir restoran veya restoran zincirinin yönetildiği bir alandır. Restoranların listelendiği, detaylarının görüntülendiği, düzenlendiği ve yönetildiği bir sayfadır. Restoranların adı, adresi, menüsü gibi bilgiler bu sayfada yer olur. Yöneticiler, restoranların eklenmesi, düzenlenmesi veya silinmesi gibi işlemleri bu sayfadan yapabilirler.



# Şekil - resturant dashboard

Bir restoran yönetim paneli veya dashboard işletmenin farklı yönlerini yönetmek için bir dizi özelliği içerir. İşte bu özelliklerin bazıları:

**Employee Sayfası:**

Bu sayfa, restoran çalışanlarının yönetildiği ve izlendiği bir alandır. Burada çalışanların listesi, rolleri, iletişim bilgileri gibi bilgiler bulunabilir. Yöneticiler, buradan yeni çalışanlar ekleyebilir, mevcut çalışanların bilgilerini güncelleyebilir, rollerini değiştirebilir veya çalışanları işe alabilir veya çıkarabilir.

**Reservation Sayfası:**

Bu sayfa, restorandaki rezervasyonların yönetildiği bir alandır. Burada mevcut ve gelecekteki rezervasyonlar listelenebilir, düzenlenebilir veya iptal edilebilir. Ayrıca, yeni rezervasyonlar oluşturulabilir ve rezervasyonları tarih, saat, masa numarası gibi kriterlere göre filtreleme veya arama yapılabilir.

**Meals (Yemekler) Sayfası:**

Bu sayfa, restoran menüsündeki yemeklerin yönetildiği bir alandır. Burada restoranın sunduğu yemeklerin listesi, fiyatları, açıklamaları ve fotoğrafları bulunabilir. Yöneticiler, buradan yeni yemekler ekleyebilir, mevcut yemeklerin bilgilerini güncelleyebilir veya yemekleri kaldırabilir.

**Gallery (Galeri) Sayfası:**

Bu sayfa, restoranın fotoğraf galerisini yönetmek için kullanılır. Burada restoranın iç ve dış mekanlarıyla ilgili fotoğrafların listesi bulunabilir. Yöneticiler, buradan yeni fotoğraflar ekleyebilir, mevcut fotoğrafları düzenleyebilir veya galeriyi güncelleyebilir.

**Comment (Yorum) Sayfası:**

Bu sayfa, restoranla ilgili müşteri yorumlarının yönetildiği bir alandır. Burada müşterilerin restoran hakkında yaptıkları yorumların listesi bulunabilir. Yöneticiler, bu yorumları görüntüleyebilir, yanıtlayabilir veya gerektiğinde silebilir. Ayrıca, yorumları tarih veya puan gibi kriterlere göre filtreleyebilir veya arayabilirler.



# Şekil - (restaurant sayfası )User sayfası

**Restoranın kullanıcı yönetim paneli içinde bulunabilecek bazı özellikler şunlar olabilir:**

**Menu (Menü) Sayfası:**

Bu sayfa, restoran menüsünün yönetildiği bir alandır. Burada restoranın sunduğu yemeklerin listesi, fiyatları, açıklamaları ve fotoğrafları bulunabilir. Yöneticiler, buradan yeni yemekler ekleyebilir, mevcut yemeklerin bilgilerini güncelleyebilir veya yemekleri kaldırabilir.

**Team (Ekip) Sayfası:**

Bu sayfa, restoran çalışanlarının listelendiği ve yönetildiği bir alandır. Burada çalışanların adı, pozisyonu, iletişim bilgileri gibi bilgiler yer alabilir. Yöneticiler, buradan yeni çalışanlar ekleyebilir, mevcut çalışanların bilgilerini güncelleyebilir veya çalışanları kaldırabilir.

**Comments (Yorumlar) Sayfası:**

Bu sayfa, müşterilerin restoran hakkındaki yorumlarının yönetildiği bir alandır. Burada müşteri yorumlarının listesi, yorum yapan kullanıcı adı, yorum tarihi, ve yorum içeriği bulunabilir. Yöneticiler, bu yorumları görüntüleyebilir, yanıtlayabilir veya gerektiğinde silebilir.

**Reservation (Rezervasyon) Sayfası:**

Bu sayfa, restoran rezervasyonlarının yönetildiği bir alandır. Burada mevcut ve gelecekteki rezervasyonların listesi, rezervasyon tarihi, saat, masa numarası gibi bilgiler bulunabilir. Yöneticiler, bu sayfadan rezervasyonları görüntüleyebilir, düzenleyebilir, onaylayabilir veya iptal edebilir. Ayrıca, yeni rezervasyonlar oluşturabilir ve rezervasyonları tarih, saat, masa numarası gibi kriterlere göre filtreleme veya arama yapabilirler.