

Deployment Plan (Dağıtım Planı)

Proje Adı: TripGuide

Takım: TrueGuiders

Tarih: 25 Mart 2025

List of Contributors (Katkıda Bulunanlar)

- İrem Keser
- Efe Selim Sürekli
- Zeynep Ay
- Elif Sema Küçük
- Saadet Cansu Baktıroğlu

Table of Contents (İçindekiler)

- Task Matrix (Görev Matrisi)
- Deployment Plan (Dağıtım Planı)
 - 2.1 Deployment Overview (Dağıtım Genel Bakışı)
 - 2.2 Deployment Process (Dağıtım Süreci)
 - 2.3 Configuration Plan (Yapılandırma Planı)

1. Task Matrix (Görev Matrisi)

Task Description (Görev Tanımı)	İrem Keser	Efe Selim Sürekli	Elif Sema Küçük	Zeynep Ay	Saadet Cansu Baktıroğlu
Deployment Overview	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Deployment Process	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Configuration Plan		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>

2. Deployment Plan

2.1 Deployment Overview

Demo sunumu için proje, backend'in yerel makinede (Localhost) barındırıldığı ve istemci (Frontend) ile veritabanının aynı ortamda çalıştığı hibrit bir düzende yapılandırılmıştır. Bu yaklaşım, sunum sırasındaki olası ağ gecikmelerini önlemek ve maksimum stabiliteyi sağlamak için tercih edilmiştir.

- **Tools Used (Kullanılan Araçlar):**
 - **Backend:** Java Spring Boot (Apache Tomcat üzerinde çalışır), Maven bağımlılık yönetimi ve JUNIT test kütüphanesi ile.
 - **Database:** PostgreSQL, pgAdmin yönetim arayüzü ile.
 - **Frontend:** HTML5, CSS3, JavaScript (Modern Web Tarayıcısı üzerinden servis edilir).
 - **IDE/Editor:** VS Code veya IntelliJ IDEA.
- **Environment (Ortam):**
 - **Development:** Windows 10/11 veya macOS üzerinde yerel geliştirme ortamı.
 - **Communication:** Frontend ve Backend arasındaki iletişim RESTful API çağrıları (HTTP istekleri) üzerinden yönetilir.

2.2 Deployment Process (Buraya tekrar bakılacak)

Projenin demo ortamında ayağa kaldırılması için aşağıdaki adımlar izlenmiştir:

1. **Database Deployment:**
 - PostgreSQL servisi yerel makinede başlatılır.
 - pgAdmin veya terminal kullanılarak `trip_guide_db` adında boş bir veritabanı oluşturulur.
 - Spring Boot uygulaması başlatıldığında, JPA/Hibernate yapılandırması sayesinde Java entity sınıfları baz alınarak tablolar otomatik olarak oluşturulur ve demo verileri sisteme yüklenir.
2. **Backend Deployment:**
 - Terminal veya komut satırında proje kök dizinine gidilir.
 - Bağımlılıkların temizlenmesi ve indirilmesi için `mvn clean install` komutu çalıştırılır.
 - Uygulamayı başlatmak için `mvn spring-boot:run` komutu çalıştırılır.
 - Konsolda `Started TripGuideApplication` logu görüldüğünde sunucunun `http://localhost:8080` adresinde aktif olduğu doğrulanır.
3. **Frontend Deployment:**
 - Eğer Frontend statik dosyaları Spring Boot içerisindeyse ekstra işlem gerekmez.
 - Ayı bir modül ise, Frontend klasöründe terminal açılarak VS Code "Live Server" başlatılır veya `index.html` tarayıcıda açılır.

- Uygulamanın <http://localhost:63342> (veya ilgili port) üzerinden erişilebilir olduğu kontrol edilir.
4. **Integration Testing:**
- Tarayıcı geliştirici araçları (F12 > Network) açılarak API isteklerinin başarılı (Status 200) olduğu doğrulanır.
 - Postman veya Swagger UI (<http://localhost:8080/swagger-ui.html>) kullanılarak uç noktaların yanıt verdiği teyit edilir.

2.3 Configuration Plan

Backend Configurations:

- `application.properties` dosyası, yerel veritabanı kimlik bilgileri ve port ayarları ile yapılandırılmıştır:
 - `server.port=8080`
 - `spring.datasource.url=jdbc:postgresql://localhost:5432/trip_guide_db`
- CORS (Cross-Origin Resource Sharing) politikaları, Frontend'in çalıştığı porta izin verecek şekilde `@CrossOrigin` anotasyonları ile ayarlanmıştır.

Frontend Configurations:

- API çağrıları için temel URL, global bir sabit olarak tanımlanmıştır: `const API_BASE_URL = "http://localhost:8080/api";`
- Debug mod, geliştirme sırasında konsol loglarını görmek için aktif tutulmuştur.

Version Control:

- Tüm değişiklikler GitHub deposuna açık ve anlaşılır commit mesajları ile yüklenmiştir.
- Proje, `main` branch üzerinden sunuma hazırlanmıştır.