**Dinamik Konfigürasyon Yönetim Sistemi Kullanım Dokümanı**

1. **Uygulamanın Amacı**
2. **Temel Özellikler**
3. **Sistem Gereksinimleri**
4. **Projeyi Çalıştırma**
5. **API Kullanım Klavuzu**
6. **Proje Yapısı**
7. **Geliştirme ve Genişletme**

**1. Uygulamanın Amacı**

Bu uygulama, farklı projeler için merkezi bir konfigürasyon yönetim sistemi sunar. Aşağıdaki özellikleri sağlar:

* Konfigürasyon ayarlarını veritabanında saklama ve yönetme.
* Belirli aralıklarla konfigürasyon verilerini dinamik olarak yenileme.
* Uygulamadan bağımsız olarak herhangi bir servis için kolayca konfigürasyon erişimi.
* İnaktif veya başka projelere ait ayarları filtreleme.
* Verilerin kullanıcı dostu bir arayüz üzerinden eklenmesi, güncellenmesi ve yönetilmesi.

**2. Temel Özellikler**

1. **Dinamik Veri Alma**:  
   Belirli bir anahtara (key) sahip konfigürasyon verisi uygulama adlarına göre getirilebilir.
2. **Merkezi Veri Yönetimi**:  
   API üzerinden konfigürasyon kayıtları listelenebilir, eklenebilir veya güncellenebilir.
3. **Depolama Sorunlarına Karşı Dayanıklılık**:  
   Veritabanı erişim sorunlarında son başarılı veriler kullanılabilir.
4. **Veritabanı Desteği**:  
   Proje varsayılan olarak SQL Server kullanır ancak istenirse farklı veri kaynaklarına genişletilebilir.
5. **Swagger UI Entegrasyonu**:  
   API uç noktaları kullanıcı dostu bir arayüzle test edilebilir.

**3. Sistem Gereksinimleri**

* **Geliştirme Ortamı**: Visual Studio 2022 veya .NET 8 uyumlu bir IDE.
* **.NET Sürümü**: .NET 8
* **Veritabanı**: Microsoft SQL Server (localdb desteği ile varsayılan olarak yapılandırılmıştır).
* **Paketler**:
  + Microsoft.EntityFrameworkCore.SqlServer
  + Microsoft.EntityFrameworkCore.Tools

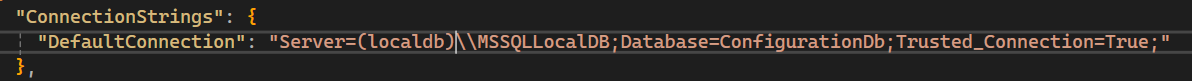
**4. Projeyi Çalıştırma**

**4.1. Adım 1: Projenin Açılması**

* Bilgisayarınıza Visual Studi0 2022 kurun.
* ConfigurationManagerAPI klasörü içerisinde ConfigurationManagerAPI.sln dosyasına tıklayarak projeyi açın.

**4.2. Adım 2: Veritabanının Hazırlanması**

1. appsettings.json dosyasındaki bağlantı dizesini (DefaultConnection) kendi veritabanınıza uygun şekilde ayarlayın:



1. Migration'ları ekleyip veritabanını oluşturun:

dotnet ef migrations add InitialCreate –-project ConfigurationManagerAPI

dotnet ef database update –-project ConfigurationManagerAPI

**4.3. Adım 3: Uygulamanın Çalıştırılması**

Projeyi Visual Studio 2022 de çalıştırın.

**4.4. Adım 4: Swagger UI Üzerinden Test**

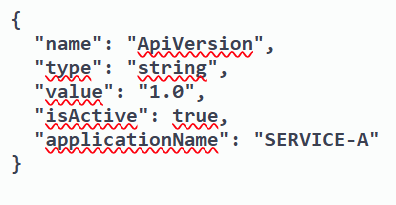
Uygulamayı çalıştırdıktan sonra tarayıcınızda şu URL'yi açarak Swagger UI'ye erişebilirsiniz:  
https://localhost:<port>/swagger  
Buradan API uç noktalarını kolayca test edebilirsiniz.

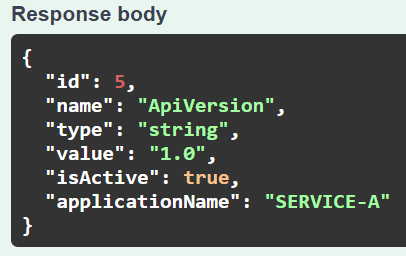
**5. API Kullanım Kılavuzu**

**5.1. Tüm Konfigürasyonları Listeleme**

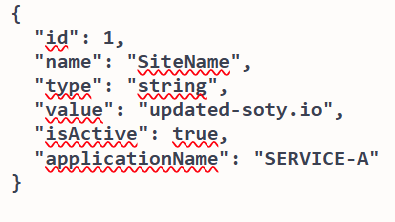
* **Endpoint**: GET /api/Configuration
* **Açıklama**: Tüm konfigürasyon verilerini döner. Uygulamaya özgü veriler için appName sorgu parametresi eklenebilir.
* **Parametreler**:
  + **appName (opsiyonel)**: Belirli bir uygulamaya ait kayıtları filtreler.
* **Örnek:**
  + GET /api/Configuration?appName=SERVICE-A

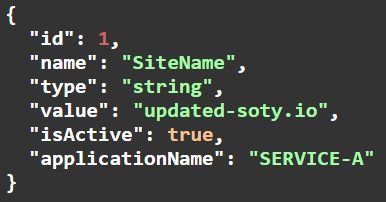
**5.2. Yeni Konfigürasyon Ekleme**

* **Endpoint**: POST /api/Configuration
* **Açıklama**: Yeni bir konfigürasyon kaydı ekler.
* **Body**
  + 
* **Başarılı Yanıt**



**5.3. Konfigürasyonu Güncelleme**

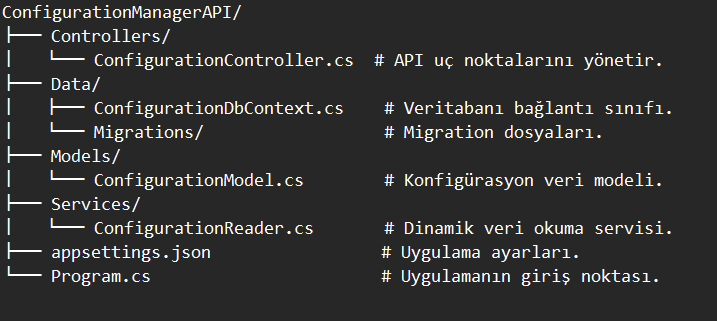
* **Endpoint**: PUT /api/Configuration/{id}
* **Açıklama**: Belirli bir ID'ye ait kaydı günceller.
* **Body**
  + 
* **Başarılı yanıt**



**5.4. Dinamik Konfigürasyon Değeri Alma**

* **Endpoint**: GET /api/Configuration/dynamic-value/{key}
* **Açıklama**: Uygulama adına ve anahtar adına göre aktif olan konfigürasyon değerini getirir.
* **Örnek:**
  + GET /api/Configuration/dynamic-value/SiteName
* Başarılı Yanıt
  + “**updated-soty.io”**

1. **Proje Yapısı**



1. **Geliştirme ve Genişletme**

* **Farklı Veri Kaynakları:** SQL Server dışında MongoDB, Redis veya dosya tabanlı sistemler entegre edilebilir.
* **Ön Bellek:** Daha yüksek performans için veriler Redis gibi önbellek sisteminde saklanabilir.
* **Rol Yetkilendirme:** Farklı kullanıcı rolleri ve yetkilendirme mekanizması eklenebilir.