

Map Tile Projesi Kurulum Kılavuzu ve Sistem Mimari Rehberi

Erenalp Özcan
ernlpzcn41@gmail.com

Özet—Bu belge, 2025 yılı staj dönemi kapsamında geliştirilen, harita tabanlı, çok katmanlı bir yazılım sisteminin kurulum ve yapısal detaylarını içermektedir. Sistem, offline harita indirim, görüntüleme, veri kaydı ve gerçek zamanlı durum takibi gibi fonksiyonları bir araya getiren masaüstü, web ve servis katmanlarından oluşur. Bu döküman yardımıyla tüm sistemin eksiksiz kurulumu ve işlevsel bağlantıları sağlanabilir.

Index Terms—.NET 8, Blazor, WPF, REST API, WebSocket, SQLite, Leaflet, Offline Harita, WebView2, JWT

I. GİRİŞ

Bu belge, **Map Tile Downloader**, **Map Tile API**, **Progress Notifier** ve **Map Tile Manager** olmak üzere toplam **dört uygulamadan oluşan** bir yazılım sistemini ele alır. Her bileşen kendi içerisinde bağımsız olarak çalışabilen, ancak entegre çalıştırıldığında offline harita yönetimi alanında etkili bir çözüm sunan parçalardır. **Amaç**, **kullanıcının belirlediği alanlardaki harita görsellerini indirip kaydetmesi, daha sonra bu verileri web arayüzü üzerinden yönetebilmesi ve ilerleme takibini anlık yapabilmesidir.**

II. SİSTEM MİMARISI

A. Katmanlar

- **Map Tile Downloader (WPF):** Harita alanı seçimi, zoom belirleme ve tile indirimini yapar.
- **Map Tile API (.NET Minimal API):** Harita meta verilerini SQLite üzerinde CRUD işlemleri ile yönetir.
- **Progress Notifier (Console/WebSocket):** Masaüstü uygulamasından gelen ilerleme bilgisini web arayüze aktarır.
- **Map Tile Manager (Blazor Server):** Haritaları listeleme, güncelleyen ve gösteren web tabanlı arayüzdür.

B. Veri Akışı

- 1) Kullanıcı masaüstü uygulamada alan seçer ve indirme başlatır.
- 2) Her tile indirildikçe ilerleme % bilgisi WebSocket ile Notifier'a iletilir.
- 3) Notifier bu bilgiyi Blazor uygulamasına aktarır.
- 4) Aynı anda indirilen haritanın meta verisi API'ye kaydedilir.
- 5) Web arayüzü üzerinden JWT ile giriş yapan kullanıcı bu haritaları listelebilir, silebilir, yeniden adlandırabilir ve haritada gösterebilir.

III. KURULUM GEREKSİNİMLERİ

A. Donanım

- 64-bit Windows 10 veya üzeri
- Minimum 8 GB RAM
- 4 çekirdekli işlemci
- 2 GB boş disk alanı

B. Yazılım

- Visual Studio 2022 veya üzeri
- .NET 8 SDK
- WebView2 Runtime
- Git (opsiyonel)

IV. PROJEYİ KURMA ADIMLARI

A. 1. Kodları Klonlama

```
git clone https://github.com/erozcann/MapTileSystem.git
```

B. 2. Projeleri Visual Studio ile Açma

- MapTileDownloader.sln (masaüstü)
- MapTileAPI (Minimal API)
- ProgressNotifier (WebSocket)
- MapTileManager (Blazor Web)

V. NUGET PAKETLERİ VE BAĞIMLILIKLAR

A. WPF Downloader

- Microsoft.Web.WebView2
- Microsoft.Data.Sqlite
- System.Net.WebSockets.Client

B. API (Minimal)

- Microsoft.AspNetCore.Authentication.JwtBearer
- Microsoft.Data.Sqlite
- Swashbuckle.AspNetCore (Swagger)

C. Web Dashboard (Blazor)

- Razor Components
- HttpClient + JWT Authorization header

VI. VERİTABANI YAPISI (SQLITE)

A. USERS Tablosu

- username: TEXT
- password: TEXT (SHA256 ile hashlenmiş)

B. DATA Tablosu

- map_name: TEXT
- date: TEXT
- min_zoom, max_zoom: INTEGER
- coord1_lat/lng, coord2_lat/lng: REAL
- folder_path: TEXT

VII. UYGULAMALARI ÇALIŞTIRMA ADIMLARI

A. Map Tile API

- JWT login/register endpointleri /login ve /register
- Swagger arayüzü: <https://localhost:7059/swagger>

B. Downloader (WPF)

- Harita alanı seçilir (WebView2 + Leaflet)
- Zoom seviyesi belirlenir
- % ilerleme barı gösterilir
- WebSocket ile ilerleme bildirimi gönderilir

C. Progress Notifier

- WebSocket portu: `ws://localhost:8181/ws`
- Gelen mesajları tüm istemcilere broadcast eder

D. Map Tile Manager (Web Arayüz)

- JWT ile login olunmadan /maps'e erişilemez
- REST API ile veri listelenir, güncellenir, silinir
- map.html ile offline tile görseli çizilir

VIII. DİKKAT EDİLMESİ GEREKENLER

- WebView2 eksikse harita alanı yüklenmez
- API token'sız erişim vermez
- map.html dosyası wwwroot altında olmalıdır
- Tile klasör yapısı `/z/x/y.png` şeklinde olmalıdır