

Görev Açıklaması

Bu görevde amaç, büyük dosyaların otomatik olarak küçük parçalara (chunk) ayrılması, bu parçaların farklı depolama sağlayıcılarına dağıtılması ve gerektiğinde birleştirilerek dosya bütünlüğünün korunmasının sağlandığı bir altyapıyı .NET Console Application olarak tasarlayıp geliştirmektir.

Görev, dosya yedekleme/dağıtık depolama gibi sistemlerde kullanılan temel yapı taşlarının soyutlanması ve genişletilebilir bir şekilde modellenmesini hedeflemektedir. Kullanıcının yüklediği dosyalar sistem tarafından parçalanacak, her parça farklı storage provider'a yazılacak, metadata veritabanında tutulacak ve yeniden birleştirme aşamasında orijinal dosya doğrulaması yapılacaktır.

Kabul Kriterleri

- Sistem, **tekli ve çoklu dosyaları** chunk'lara ayırabilmeli.
- Chunk'lama algoritması, **dosya boyutuna göre dinamik ve optimal** şekilde çalışmalı.
- Tüm chunk'lar farklı **IStorageProvider** implementasyonlarına (**FileSystemStorageProvider, DatabaseStorageProvider vb.**) dağıtılmalı.
- Chunk bilgileri ve ilişkili metadata **kalıcı olarak veritabanında saklanmalı**.
- Birleştirme işlemi, tüm chunk'ları ilgili storage'dan alarak orijinal dosyayı oluşturmalı ve **checksum doğrulaması** yapmalıdır.
- Kodlama, **.NET C# isimlendirme kurallarına, OOP prensiplerine ve SOLID** yapılarına uygun olmalıdır.
- **IStorageProvider** gibi temel bileşenler soyutlanmış olmalı, dependency injection (IoC) ile dışarıdan enjekte edilebilir yapıda olmalı.
- Uygulama içinde yapılan tüm adımlar **loglanmalıdır**.
- Sonuçtan çok **yaklaşım**, mimari yapı, kullanılan design pattern'ler değerlendirme kriteri olacaktır.
- Ekstra düşünülen özellikler olumlu puan getirecektir.
- Projenin **README** dosyasında, uygulamanın nasıl çalıştırılacağına, mimari tercihlere ve varsa ekstra özelliklere dair açıklayıcı notlar bulunmalıdır.

Teknik Notlar

- Geliştirme ortamı: **.NET Console Application** (.NET 8 veya 9)
- **IoC container** (Microsoft.Extensions.DependencyInjection) gerekli görüldüğü yerlerde kullanılmalı.
- Tüm bileşenler test edilebilir olacak şekilde **interface tabanlı soyutlanmalıdır**.
- **Veritabanı tercihi serbesttir**.
- Her dosya ve chunk işlemi sırasında loglama yapılmalı (**ILogger**, Serilog vb. olabilir).
- **Dosya bütünlüğü kontrolü** için checksum (örn: SHA256) kullanılmalı.

Teslimat ve Değerlendirme Süreci

- Geliştirme süreci, aday tarafından kişisel bir **Git** repository'si üzerinden yürütülmelidir. Repository **public** olmalıdır.
- İlgili çalışma mailde belirtilen tarihine kadar içerisinde görev linki ile beraber teslim edilmelidir.