Github Java Repoları Analizi

Vedat DOGAN

Özet

Kullanıcıdan alınan github linki ile repoyu klonlar, klonladığı yol ödev dosyasındaki doc adlı klasördür. Eğer bu klasör dolu ise içini boşaltır ve klonu buraya kaydeder. Eğer doc adlı klasör yoksa bunu oluşturur- kaydeder. Daha sonra bu klonu içindeki classları bulur ve belirli işlemlere tabi tutar. Bu işlemler Javadoc satır sayısı bulma, diğer yorum satırlarını bulma, kod satırlarını bulma, Loc, fonksiyon sayısı ve yorum sapma yüzdesi şeklindedir.

1. GELİŞTİRİLEN YAZILIM

Github’da yer alan Java dilinde yazılmış projelerinden, \*.java uzantılı dosyalar

üzerinde detaylı kod analizi yapan projedir. Projede geliştirilen yazılım, kullanıcı tarafından girilen Github deposunun URL’si üzerinden depoyu proje içindeki doc isimli dosyaya klonlamaktadır. Ana hedef, bu depolarda yer alan Java dosyalarını tarayarak sadece sınıf içeren dosyaları ayıklamak ve bu dosyalar üzerinde kod analizi gerçekleştirmek.

Analiz sürecinde , sınıf dosyaları belirli metrikler üzerinden değerlendirilir. Bu metrikler Javadoc satır sayısı bulma, diğer yorum satırlarını bulma, kod satırlarını bulma, Loc, fonksiyon sayısı ve yorum sapma yüzdesi şeklindedir.

Yazılımın geliştirilmesi sırasında Java programlama dili ve çeşitli kütüphaneler kullanılmıştır. Burada kaynak olarak mvnrepository.com kullandım. Bu sitedden gerekli dosyaları indirip projeme dahil ettim.Özellikle Github deposunu klonlama işlemi için JGit kütüphanesinden yararlandım. JGtin yanında SLFJ4-API ve SLFJ4-SİMPLE kütüphanelerini indirip projeme dahi ettim.

3 tane class oluşturdum. Birincisi Program classım, burası programı yürüttüğüm classım. Diger classım GitService classım klon işlemlerini gerçekleştirdiğim classım oldu. Burada JGit kütüphanesini kullandım. JavaDosyaAnaliz classımda ise klonladığım dosyaları analizini ve ekrana yazdırma işlemlerini yaptığım class oldu.

Proje de regex kullanımını ve mantığını ögrenmiş oldum. Kodum üzerinde küçük bir örnek aşağıdadır.

metin, ekran görüntüsü, yazı tipi, çizgi içeren bir resim

Açıklama otomatik olarak oluşturuldu

1. ÇIKTILAR

Farklı Java dosyaları üzerinden detaylı analizler yapabilme yeteneğine sahiptir. Bu analizler, dosyaların içeriğine göre değişiklik gösterir.

Standart Java Dosyası: Sınıflar, metodlar, Javadoc ve inline yorumlar içeren bir dosyada, yazılım beklenen tüm ölçütleri (yorum ve kod satır sayısı, LOC, fonksiyon sayısı, yorum sapma yüzdesi) doğru bir şekilde hesaplar.

Yorum İçermeyen Java Dosyası: Yalnızca kod içeren bir dosyada, yorum satır sayıları sıfır olarak raporlanır; ancak kod ve LOC değerleri hesaplanır. Yorum sapma yüzdesi, yorum eksikliği nedeniyle negatif bir değer alır.

Yoğun Yorum İçeren Java Dosyası: Ağırlıklı olarak yorum içeren bir dosyada, yüksek yorum satır sayıları ve düşük kod satır sayıları gözlemlenir. Yorum sapma yüzdesi, beklenenden yüksek çıkabilir.

Fonksiyon İçermeyen Java Dosyası: Yalnızca sınıf tanımı olan ve metod içermeyen dosyalarda, fonksiyon sayısı sıfır olarak belirtilir ve yorum sapma yüzdesi hesaplanamaz.

1. SONUÇ

Yazılan java dilindeki kodun yorum satırlarından anlaşılabilirliğini ölçebilir. Ama bu ne kadar yorum satırı varsa kod o kadar anlaşılır demek değildir. Bunun yanında kolayca klonlama işlemi yapmakta. Kod satır sayısı ve proje genişliği bakılabilir.

Referanslar

[1] https://mvnrepository.com/