

Birim Test ve Test Odaklı Geliştirme Nedir?

Çalışma Kağıdı

Birim test, yazılımın en küçük test edilebilir parçalarının (fonksiyonlar, metotlar) doğru çalıştığını doğrulamaktır. Test Odaklı Geliştirme (TDD) ise, testlerin koddan önce yazıldığı bir geliştirme metodolojisidir.

Sorular

1. Aşağıdakilerden hangisi birim testin temel faydalarından biri DEĞİLDİR?

- A) Hata ayıklama süresini azaltır
- B) Kod kalitesini artırır
- C) Geliştirme sürecini yavaşlatır
- D) Refactoring'i kolaylaştırır

2. TDD'nin 'kırmızı' aşaması ne anlama gelir?

- A) Test başarıyla geçti
- B) Henüz yazılmamış kod nedeniyle test başarısız oldu
- C) Kod yeniden düzenleniyor
- D) Test ortamı hazırlanıyor

3. Birim testler genellikle kim tarafından yazılır?

- A) Test mühendisleri
- B) Son kullanıcılar
- C) Yazılım geliştiriciler
- D) Proje yöneticileri

4. Basit bir toplama fonksiyonu için birim testi nasıl yazılır?

5. TDD ile basit bir 'kullanıcı ekle' özelliği nasıl geliştirilir?

6. Tanımla: Birim Test Nedir?

7. Tanımla: Test Odaklı Geliştirme (TDD) Nedir?

8. Tanımla: Birim Testin Amacı Nedir?

Cevap Anahtarı

1. C) Geliştirme sürecini yavaşlatır — Birim testleri, hataları erken tespit ederek ve kodun doğruluğunu sağlayarak geliştirme sürecini hızlandırır, yavaşlatmaz.
2. B) Henüz yazılmamış kod nedeniyle test başarısız oldu — 'Kırmızı' aşama, yazılan testin henüz karşılığı olan üretim kodu olmadığı için başarısız olmasını ifade eder.
3. C) Yazılım geliştiriciler — Birim testleri, kodun işleyişini en iyi bilen yazılım geliştiriciler tarafından yazılır.
4. 1. Toplama fonksiyonunu tanımlayın. 2. Fonksiyonun beklenen çıktısını belirleyen bir test senaryosu yazın. 3. Testi çalıştırın ve beklenen sonucu alın. 4. Fonksiyonu, testin başarılı olmasını sağlayacak şekilde uygulayın.
5. 1. Başarısız olacak bir 'kullanıcı ekle' testi yazın (kırmızı). 2. Testin geçmesini sağlayacak en basit kodu yazın (yeşil). 3. Kodu iyileştirin ve yeniden düzenleyin (refactor). 4. Diğer özellikler için 1-3 adımlarını tekrarlayın.
6. Yazılımın en küçük, izole parçalarının (fonksiyon, metot) doğru çalıştığını doğrulayan otomatik testlerdir.
7. Testlerin, üretim kodundan önce yazıldığı bir yazılım geliştirme metodolojisi.
8. Kod hatalarını erken tespit etmek, kodun doğruluğunu garanti etmek ve refactoring'i kolaylaştırmaktır.

Bounlu

Tüm kartlar, adım adım çözümler ve AI hoca desteği Notek uygulamasında.
Sınav tarihlerini Promy otomatik hatırlatıcıya çevirir.