

Testin 7 İlkesi

1

İçindekiler

- ▶ Testin 7 İlkesi
- ▶ Testler hataların varlığını gösterebilir
- ▶ Kapsamlı test mümkün değildir
- ▶ Erken test
- ▶ Hata kümelenmesi
- ▶ Pesticide paradox
- ▶ Test bağımlılığı
- ▶ Hataların yokluğu yanılgısı

2

İlke 1 - Testler Hataların Varlığını Gösterebilir

- ▶ Testler, hataların varlığını ortaya çıkarmak için yapılır, hataların olmadığını ispatlamak için değil.
- ▶ Testler, yazılımın hatasız olduğunu kanıtlamaz, sadece mevcut hataları ortaya çıkarabilir.
- ▶ Örnek: Bir sistemde yapılan testler sonucu bulunan hatalar.

3

İlke 2 - Kapsamlı Test Mümkün Değildir

- ▶ Yazılımın tüm kombinasyonlarını test etmek mümkün değildir.
- ▶ Sınırsız sayıda giriş kombinasyonu, yol ve durum olduğundan dolayı hepsini test etmek imkansızdır.
- ▶ Örnek: Bir yazılımın olası tüm senaryolarının test edilmesinin zorluğu.

4

İlke 3 - Erken Test

- ▶ Test faaliyetleri mümkün olan en erken safhada başlatılmalıdır.
- ▶ Erken test, hataların daha erken tespit edilmesini ve düzeltilmesini sağlar, bu da maliyetleri düşürür.
- ▶ Örnek: Yazılım geliştirme sürecinin başında yapılan testlerin avantajları.

5

İlke 4 - Hata Kümelenmesi

- ▶ Hatalar genellikle belirli modüllerde veya bileşenlerde yoğunlaşır.
- ▶ Yazılımın bazı bölümleri diğerlerine göre daha fazla hataya meyillidir.
- ▶ Örnek: Bir projede en çok hata bulunan modüllerin analizi.

6

İlke 5 - Pesticide Paradox

- ▶ Aynı testlerin sürekli tekrarlanması aynı hataları bulur.
- ▶ Testlerin periyodik olarak gözden geçirilip güncellenmesi gerekmektedir.
- ▶ Örnek: Düzenli olarak güncellenmeyen testlerin etkinliğinin azalması.

7

İlke 6 - Test Bağımlılığı

- ▶ Testler, sistemin bağlamına bağımlıdır.
- ▶ Farklı sistemler, farklı test stratejileri gerektirir.
- ▶ Örnek: Web uygulaması ve mobil uygulama test stratejilerinin farklılıkları.

8

İlke 7 - Hataların Yokluğu Yanılgısı

- Hata bulunamaması, yazılımın kullanım amacına uygun ve hatasız olduğu anlamına gelmez.
- Yazılımın gereksinimlere uygun çalışıp çalışmadığının doğrulanması gerekmektedir.
- Örnek: Kullanıcı gereksinimlerini karşılamayan hatasız bir yazılım.