

Test ve Kalite Güvencesi Nedir?

Çalışma Kağıdı

Test, yazılımın belirli gereksinimleri karşılayıp karşılamadığını doğrulamak için yapılan bir inceleme iken, kalite güvencesi ise yazılımın geliştirme süreci boyunca yüksek kalitede olmasını sağlamaya odaklanan daha geniş bir yaklaşımdır.

Sorular

- Aşağıdakilerden hangisi Kalite Güvencesi'nin (QA) temel amacıdır?
 - A) Sadece hataları bulmak
 - B) Yazılımın geliştirme süreci boyunca kalitesini sağlamak
 - C) Kodun performansını optimize etmek
 - D) Kullanıcı arayüzünü güzelleştirmek
- Hangi test türü, yazılımın en küçük parçalarını (fonksiyonlar, metotlar) test eder?
 - A) Sistem Testi
 - B) Entegrasyon Testi
 - C) Birim Testi
 - D) Kabul Testi
- Kullanıcıların yazılımı kabul etmeden önceki son test aşaması genellikle hangisidir?
 - A) Birim Testi
 - B) Entegrasyon Testi
 - C) Sistem Testi
 - D) Kullanıcı Kabul Testi (UAT)
- Bir web uygulamasının kullanıcı kayıt formunu test etme örneği nedir?
- Performans testi nasıl yapılır?
- Güvenlik testi kapsamında hangi kontroller yapılır?
- Tanımla: Test Nedir?
- Tanımla: Kalite Güvencesi (QA) Nedir?
- Tanımla: Birim Testi (Unit Test)

Cevap Anahtarı

1. B) Yazılımın geliştirme süreci boyunca kalitesini sağlamak — Kalite Güvencesi, sadece hataları bulmaktan ziyade, geliştirme süreçlerini iyileştirerek baştan itibaren yüksek kalitede bir ürün elde etmeyi hedefler.
2. C) Birim Testi — Birim testi, yazılımın en küçük, bağımsız olarak test edilebilen bileşenlerine odaklanır.
3. D) Kullanıcı Kabul Testi (UAT) — Kullanıcı Kabul Testi (UAT), son kullanıcıların veya temsilcilerinin yazılımı gerçek dünya senaryolarında test ederek kabul edip etmeme kararı verdiği aşamadır.
4. 1. Geçerli verilerle kayıt olmayı dene. 2. Geçersiz verilerle (e-posta formatı hatalı, şifre çok kısa vb.) kayıt olmayı dene. 3. Alanları boş bırakarak kayıt olmayı dene. 4. Kayıt sonrası hata mesajlarının doğruluğunu kontrol et.
5. 1. Uygulamaya aynı anda çok sayıda kullanıcı yüklemesi yap. 2. Uygulamanın yanıt sürelerini ve kaynak kullanımını (CPU, bellek) izle. 3. Belirlenen eşik değerlerinin aşıp aşılmadığını kontrol et.
6. 1. SQL Injection, XSS gibi bilinen zafiyetlere karşı test et. 2. Yetkilendirme ve kimlik doğrulama mekanizmalarını kontrol et. 3. Hassas verilerin şifrelenip şifrelenmediğini doğrula.
7. Yazılımın belirli gereksinimleri karşılayıp karşılamadığını doğrulamak için yapılan bir inceleme.
8. Yazılımın geliştirme süreci boyunca yüksek kalitede olmasını sağlamaya odaklanan daha geniş bir yaklaşım.
9. Yazılımın en küçük test edilebilir parçaları olan fonksiyon veya metotların bağımsız olarak test edilmesi.

Bounlu

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