

- Yazılım neden başarısız olur?
İnsanlar hata yaparlar, bu hatalar kodda, yazılımda, sistemde ya da dokümanda defect oluşturur. Defect olan kod çalıştırıldığında sistem beklenen fonksiyonları gerçekleştiremez ve başarısız olur. Ayrıca zaman kaybı ve gerekli kontrollerin yapılmaması da nedenler arasında sayılabilir.
- Testin rolü nedir?
İleriye dönük kodun geliştirilme masraflarını azaltmak,
Ürün çalıştırılmadan önce kalitesini ölçmek,
Yapılan hataların tekrarlanmasını önlemek,
Zaman ve maliyet tasarrufu yapmak,
Saygınlık kazanmak, şeklinde yazılım testi amaçları sıralanır.
- Kalite maliyetinin bileşenleri nelerdir?
Prevention cost, Appraisal cost, external and internal failure cost
- FMEA adımları nelerdir?
Risklerin Kategorize edilmesi, Risklerin ölçeklendirilmesi, Risklerin değerlerinin belirlenmesi.
Kategorileri: Fonksiyonelite, Yuk ve Kapasite, Güvenilirlik, Sürdürülebilirlik, Veri Kalitesi, Performans, Dokümantasyon, Arayüz, Entegrasyon
Ölçeklendirme:
Sisteme Göre: 1-Veri kaybı, 2-İşlevsellik Kaybı, 3- Giderilebilir İşlevsellik Kaybı, 4-Kısmi İşlevsellik Kaybı, 5-Kozmetik Riskler
Müşteriye Göre: 1-Acil, 2-Zorunlu, 3-Onemli, 4-Düzeltilmesi İyi Olacak, 5-İsteğe Bağlı
Gerçekleşme Olasılığına Göre: 1-Muhtemel, 2-Mümkün, 3-İhtimal Dahilinde
Değerlerinin Belirlenmesi: Hepsinin sırası toplanır, toplam ne kadar azsa o kadar önemlidir.
- Test hazırlığı nasıl yapılır?
Test yaşam döngüsünün ilk iki adımıdır.
- Testcase'de olması gerekenler?
Set of inputs, Execution conditions, Expected outputs
- Bir test uzmanında olması gereken özellikler nelerdir?
Süreci iyi bilmeli, metodolojisi olmalıdır, planlı olmalıdır, her iki profili de kapsamalıdır, net olmalıdır, araçları olmalıdır, test ortamı olmalıdır, süpürcü yaklaşmalıdır.
- Bir test uzmanının yapması/yapmaması gerekenler nelerdir?
Hatayı bulmalı, erken bulmalı ve çözülmediğinden emin olmalıdır.
- Temel test seviyeleri nelerdir?
Unit(Birim) Test: Ufak parçaların ve fonksiyonların testi
Modül Seviyesi: Birimlerin birbirleriyle etkileşimi testi
Sistem Seviyesi: Bütün yazılımın fonksiyonel ve görsel olarak uyumluluk testi
Kabul Seviyesi: Müşteri tarafından kabul testi
- Sık kullanılan test yöntemleri nelerdir?
Fonksiyonel Olan: Unit Test(Birim Testi), Integration test(Entegrasyon Testi), User Acceptance Testing(Kullanıcı Kabul Testi), Web Services Testing, Migration Testing, Maintenance Testing
Fonksiyonel Olmayan: Performance Testing, Load Test, Stress Test, Compatibility Test, Security Test, Usability test, Localization Test
Ayrıca Box Testleri
- Temel test prensipleri nelerdir?
Test defect in varlığını gösterir, exhaustive test mümkün değildir, erken test yapılmalıdır, defect kümelenmesi vardır, tarım ilacı(Pesticide) paradoksu, Test içerik bağımlıdır, hata yokluğu yanılgısı.
- Test yaşam döngüsü nedir?
Analiz -> Teknik Analiz -> Development -> Test -> UAT -> Production

Analiz, Teknik analiz: Musteri ve urun taninmali, risk anlizi yapilmali, test plani, use case ve senaryolar belirlenmeli

Development: Senaryolar ve riskler siniflandirilir, Unit testler yapilir

Test: Smok Testleri, Senaryo denemeleri, Hata raporlamalari, tekrarlamali testler yapilir.

UAT: Uygulamayi musteriye test ettirme, cikan hatalari raporlama, cozuldugunden emin olma.

- Oyun yazılımlarını test ederken, diğer yazılımlara göre farklı olan unsurlar neler olur?
Gorsellik(?)
- Web yazılımlarını test ederken, diğer yazılımlara göre farklı olan unsurlar neler olur?
Bir sayfanin hizli acilabilmesi, arama motorlarinin iyi ve hizli calismasi, iyi tasarlanmis bir web site ile google gibi arama motorlarinda one cikabilmesi, kullanicinin ilk iki saniyede kalmaya karar vermesi ardindan da 5 saniye icerisinde ne yapacagini anlamasi, istenilen sayfaya ulasimin kolayligi yani kullanilabilirlik testleri. Ayrica kullanicinin gozune hos gelebilmesi(80/20).
- Server yazılımlarını test ederken, diğer yazılımlara göre farklı olan unsurlar neler olur?
Guvencik ve sorgularin hizli calisabilmesi en onemli unsurlardir. Hiyerarsik yapi ve yapilan islemlerin iyi kaydedilmesi onemlidir. Veri kaybi riski cok onemli bir konudur.
- Tehlike ile risk arasındaki fark nedir?
- Bir yazılım için iç riskler ve dış riskler nelerdir?
- Bir yazılım testinde hiç hata çıkmazsa, bu durumun sebepleri neler olabilir?
Yanlis Test uygulama yada yeterince test yapmamis olmadir. Hatasiz sistem/uygulama olamaz.
- Bir yazılımın güvenli olması ile güvenilir olması arasındaki farklar nelerdir?
Security and reliability
- Gorilla ve exhausted testler arasındaki fark nedir?
Exhaustive Test: Test grubunun tüm girdi ve ön koşul kombinasyonlarını kapsadığı test yaklaşımı.
Gorilla Test: Ayni zamanda Frustrating testin olarak da bilinen gorilla testing, bir modulun belirli bir fonksiyonunun defalarca test edilmesi yontemidir.