BSM462
Yazılım Testi

Hafta - 9

Entegrasyon ve Sistem Testi

Dr. Öğr. Üyesi M. Fatih ADAK

fatihadak@sakarya.edu.tr

İçerik

- Entegrasyon testi
  - Örnek uygulama ve yaklaşımlar
    - Big Bang Yaklaşımı
    - Bottom-up Yaklaşımı
    - Top-down Yaklaşımı
  - Sandviç test stratejisi
- Sistem testi
  - Prensipleri
  - Test durumları
  - Fonksiyonel ihtiyaçlar
  - Fonksiyonel olmayan ihtiyaçlar
  - Sistem testi çeşitleri
    - Siyah kutu testi
    - Beyaz kutu testi
    - Gri kutu testi
  - Ağ aktivitesinin incelenmesi
  - Güvenlik testleri

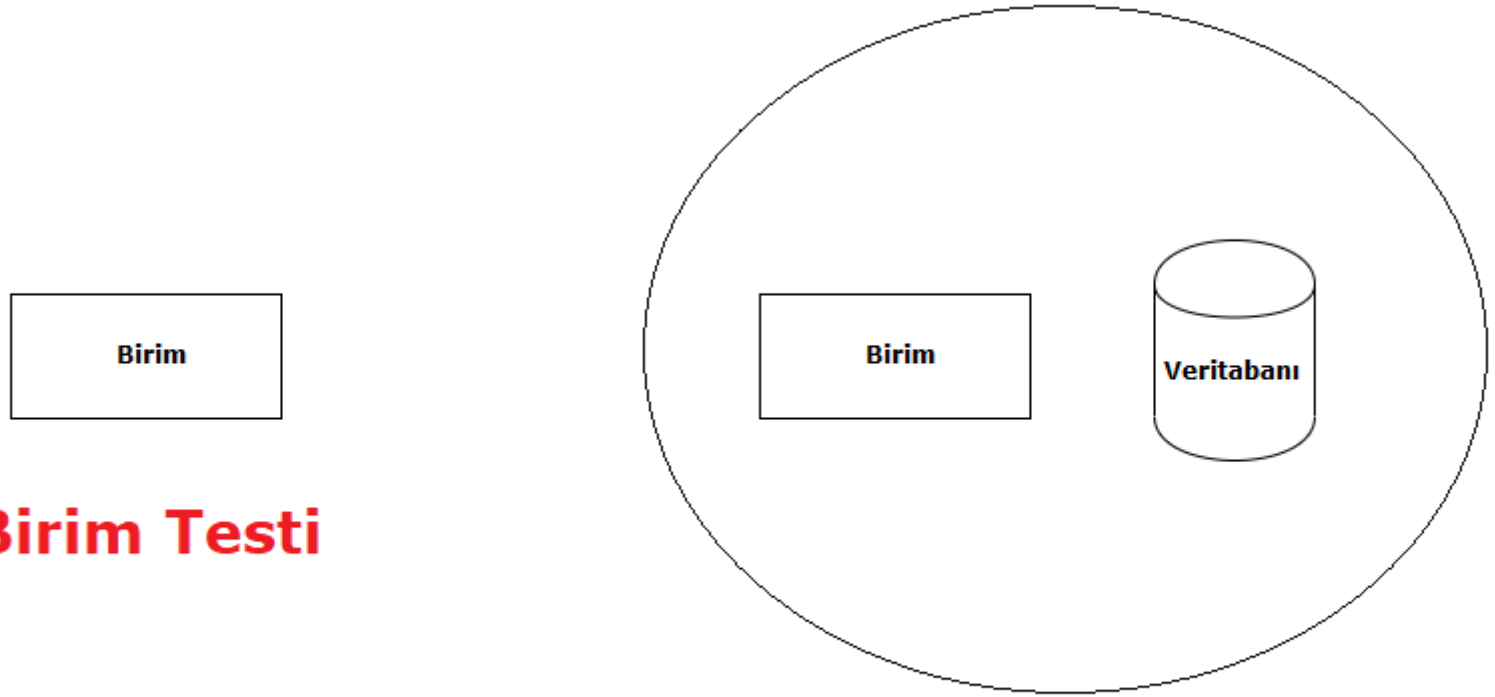
Entegrasyon Testi - Tanım

- Alt sistemlerde bulunan bütün arayüzlerin bir araya getirilip test edilmesi.
- Aralarındaki ilişkilerin incelenmesi
- Bir araya getirilen sistemde hangi alt sistemin önce bakılacağına karar verir.

Neden Entegrasyon Testi?

- Birim testleri sadece ilgili birimi test edebilir.
- Birçok hata ve başarısızlık alt sistemler bir araya getirildikten sonra ortaya çıkar.
- Entegrasyon testi olmadan sistem testi boşa zaman harcama olacaktır.
- Entegrasyon testinde tespit edilemeyen bir hata yazılımın piyasaya sürülmesinden sonra ortaya çıkacak ve yüklü maliyet getirecektir.

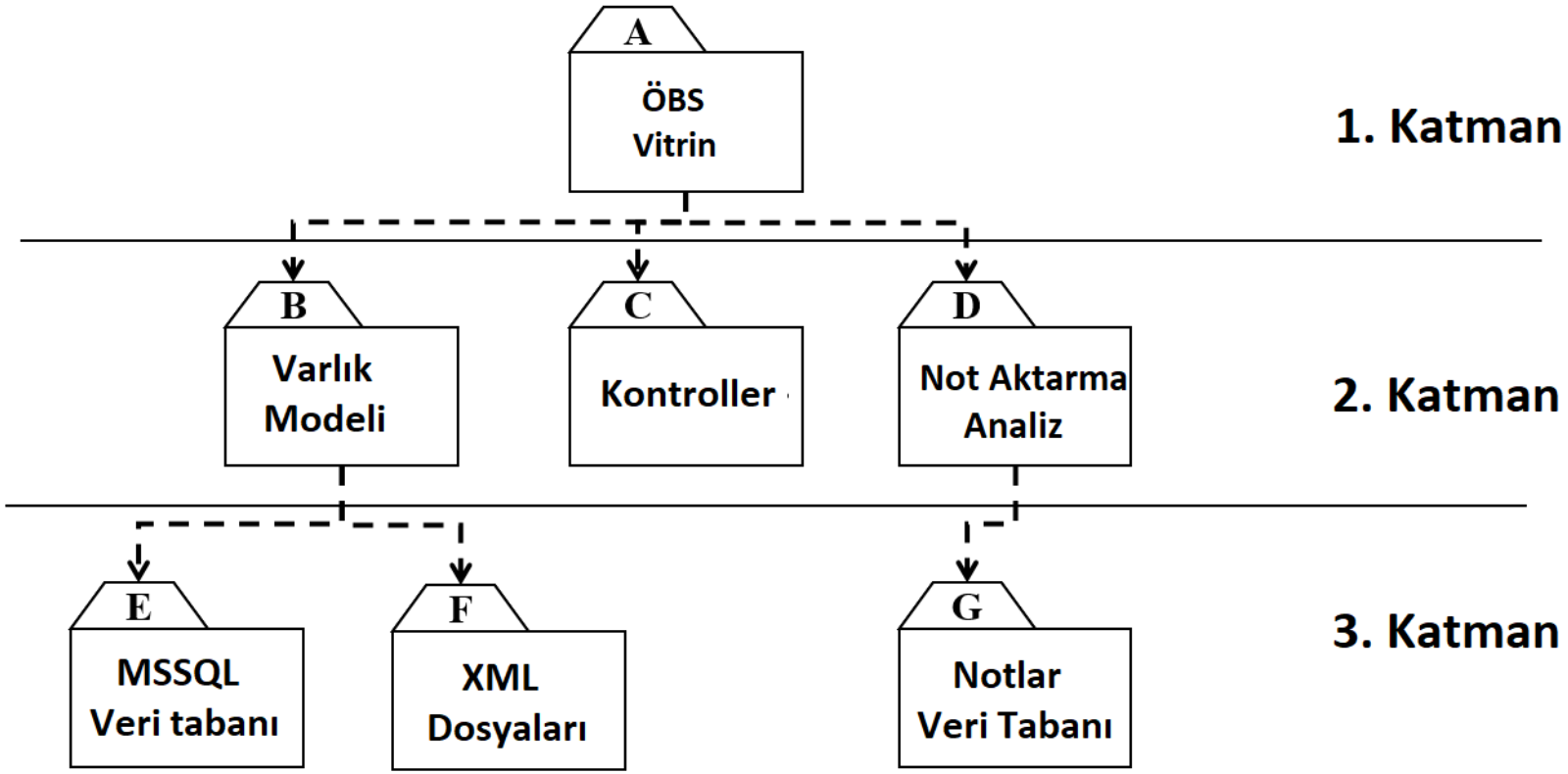
Entegrasyon Testi devam…



Birim Testi

Entegrasyon Testi

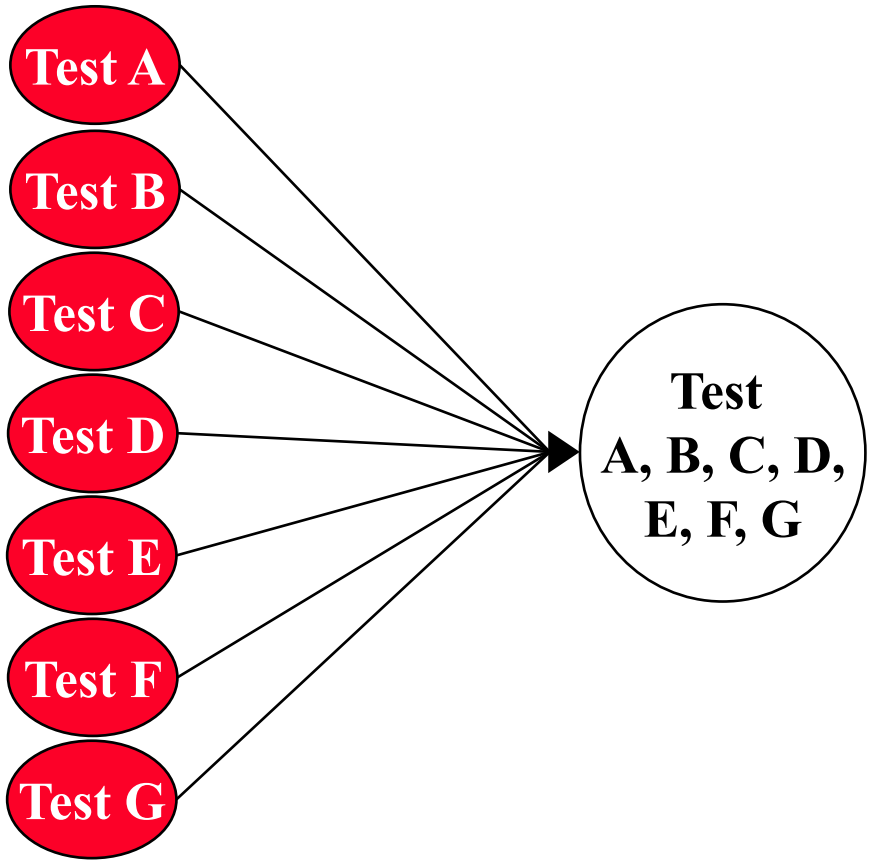
Örnek
Öğrenci Bilgi Sistemi

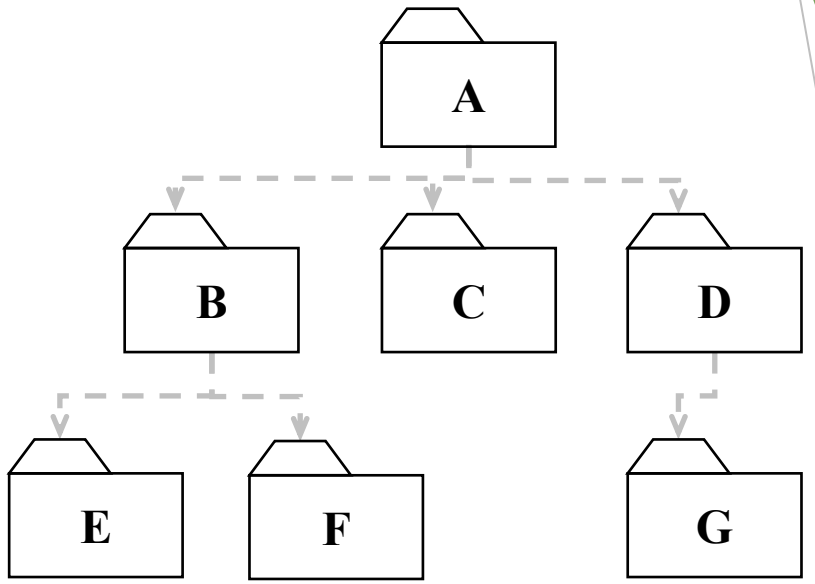


Örnek
Öğrenci Bilgi Sistemi

- Entegrasyon Test Yaklaşımları
  - Big Bang Yaklaşımı
  - Bottom-up Yaklaşımı
  - Top-down Yaklaşımı

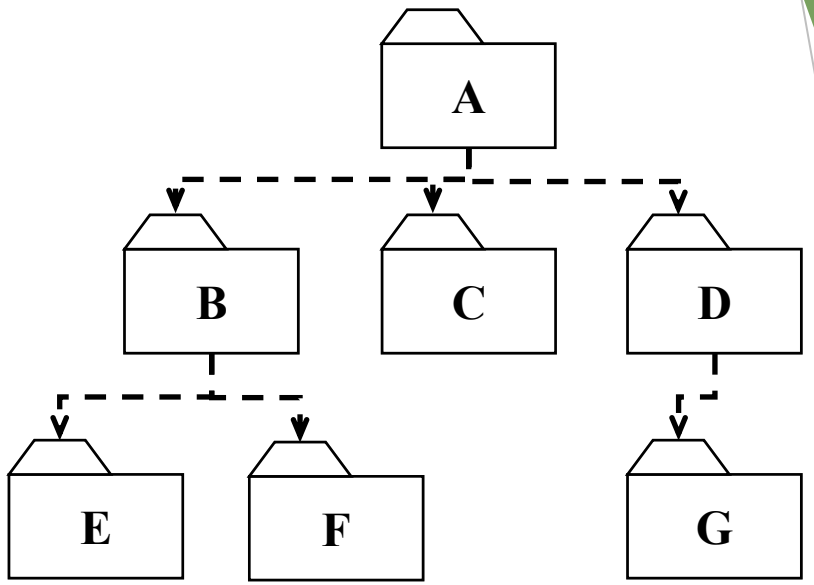
Big-Bang Yaklaşımı

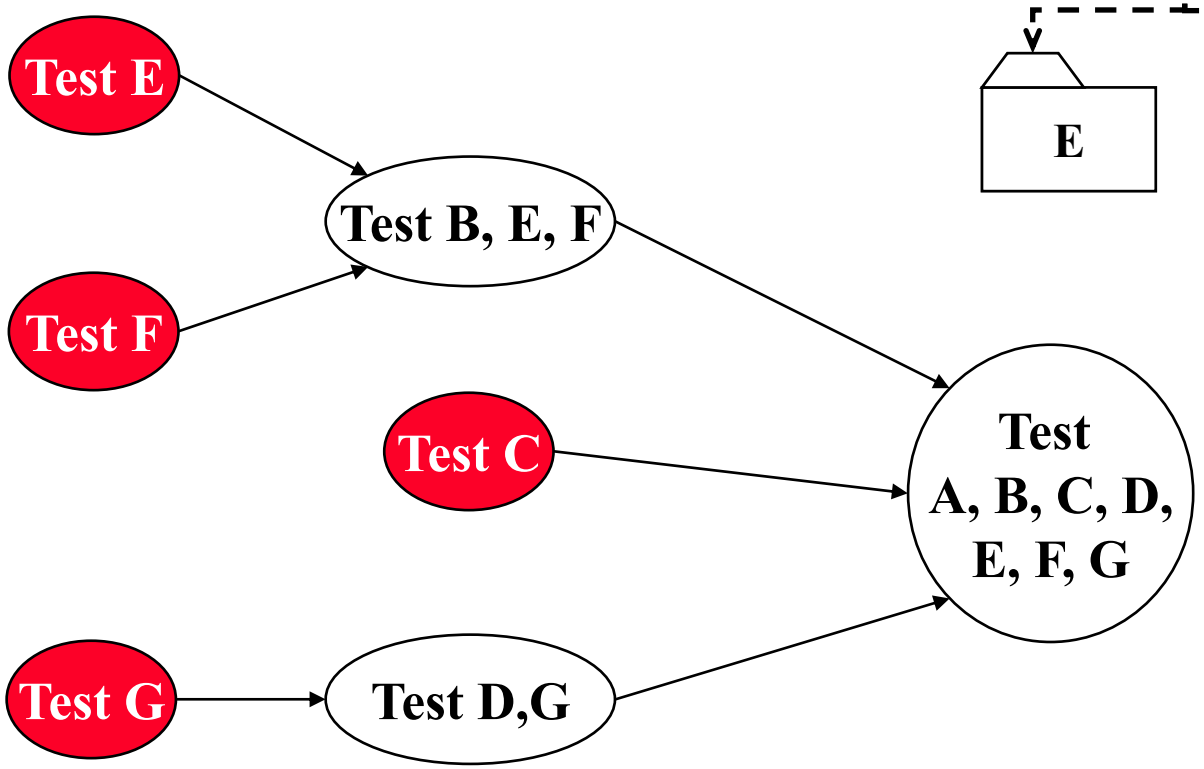




Önerilmez!!!

Bottom-up Yaklaşımı



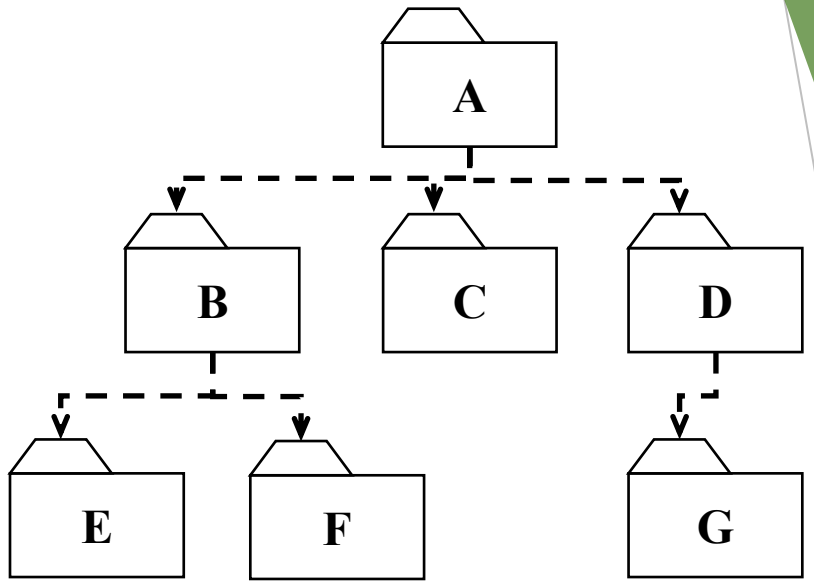


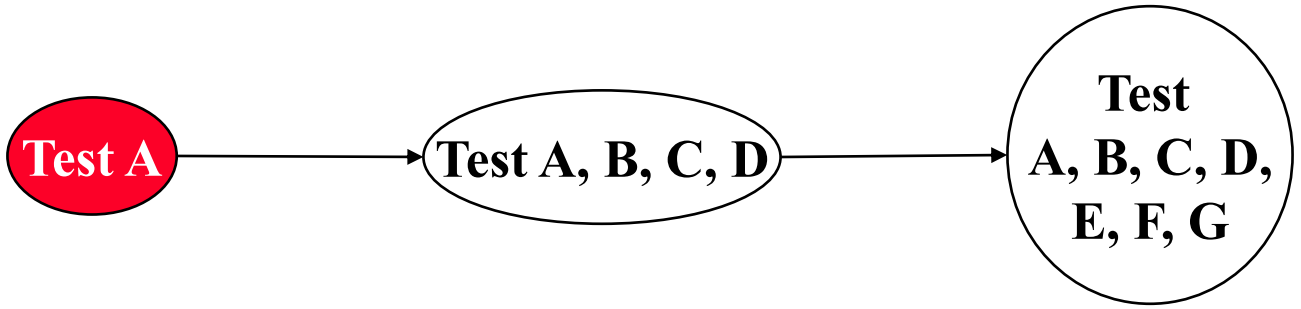
Bottom-up Yaklaşımının Avantajları

- Nesne yönelimli sistemlerin,
- Gerçek zamanlı sistemlerin,
- Sıkı performans gereksinimi olan sistemlerin,

Entegrasyon testinde kullanışlıdır.

Top-down Yaklaşımı





1. Katman **1 ve 2. Katman** **Tüm Katmanlar**

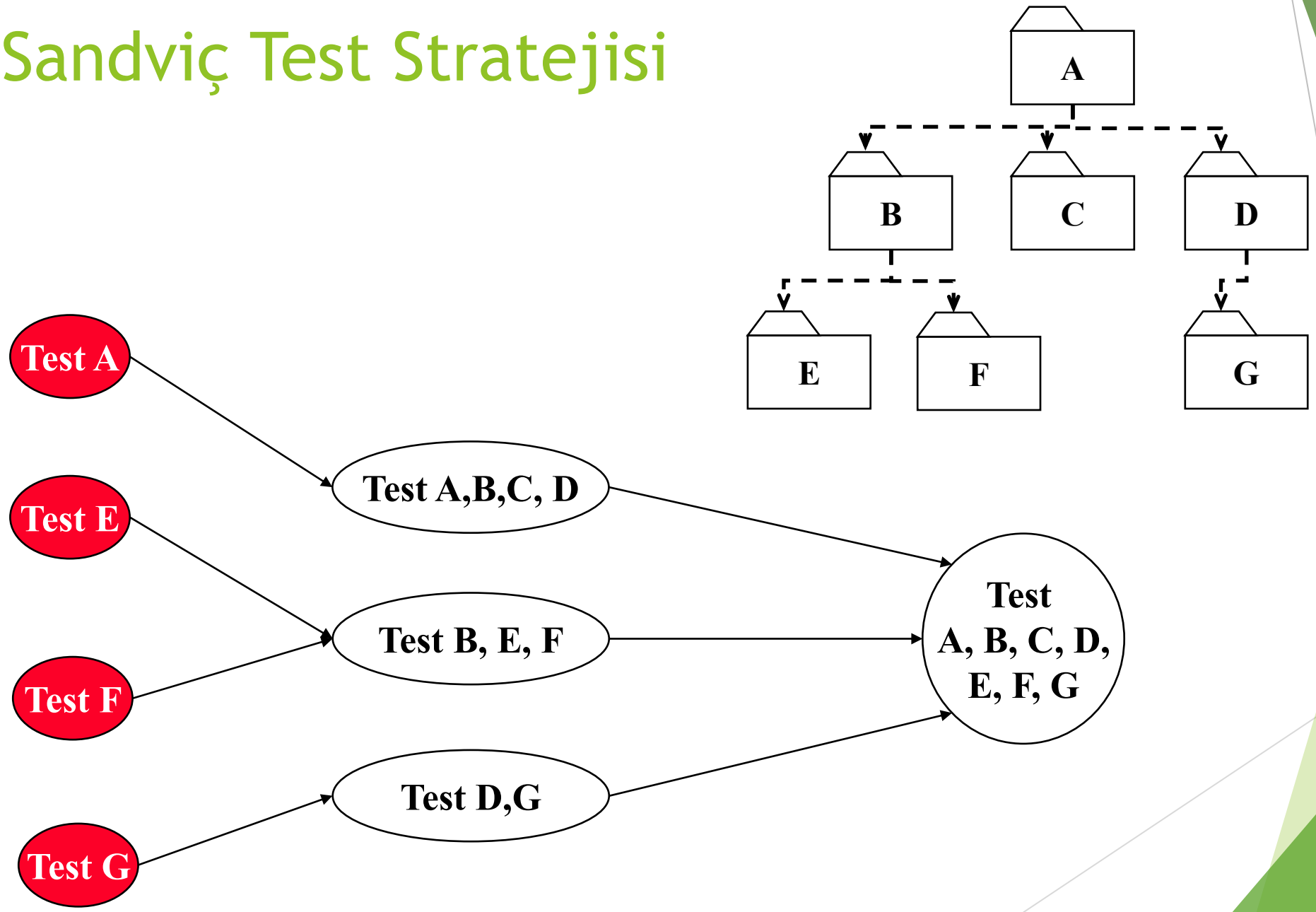
Top-down Yaklaşımının Dezavantajları

- Avantajından çok dezavantajı olan bir sistemdir.
- Entegrasyon oldukça zordur.
- Alt sistem ihtiyacı olduğu durumlarda gerçek test yapılamaz. Belki mock nesneleri olabilir.
- Bazı arayüzler ayrı test edilemeyecektir.

Sandviç Test Stratejisi

- Top-down ile Bottom-up birleştirilir.
- 3 katmanı vardır.
  - Ortadaki hedef katman
  - Hedefin üstündeki katman
  - Hedefin altındaki katman

Sandviç Test Stratejisi



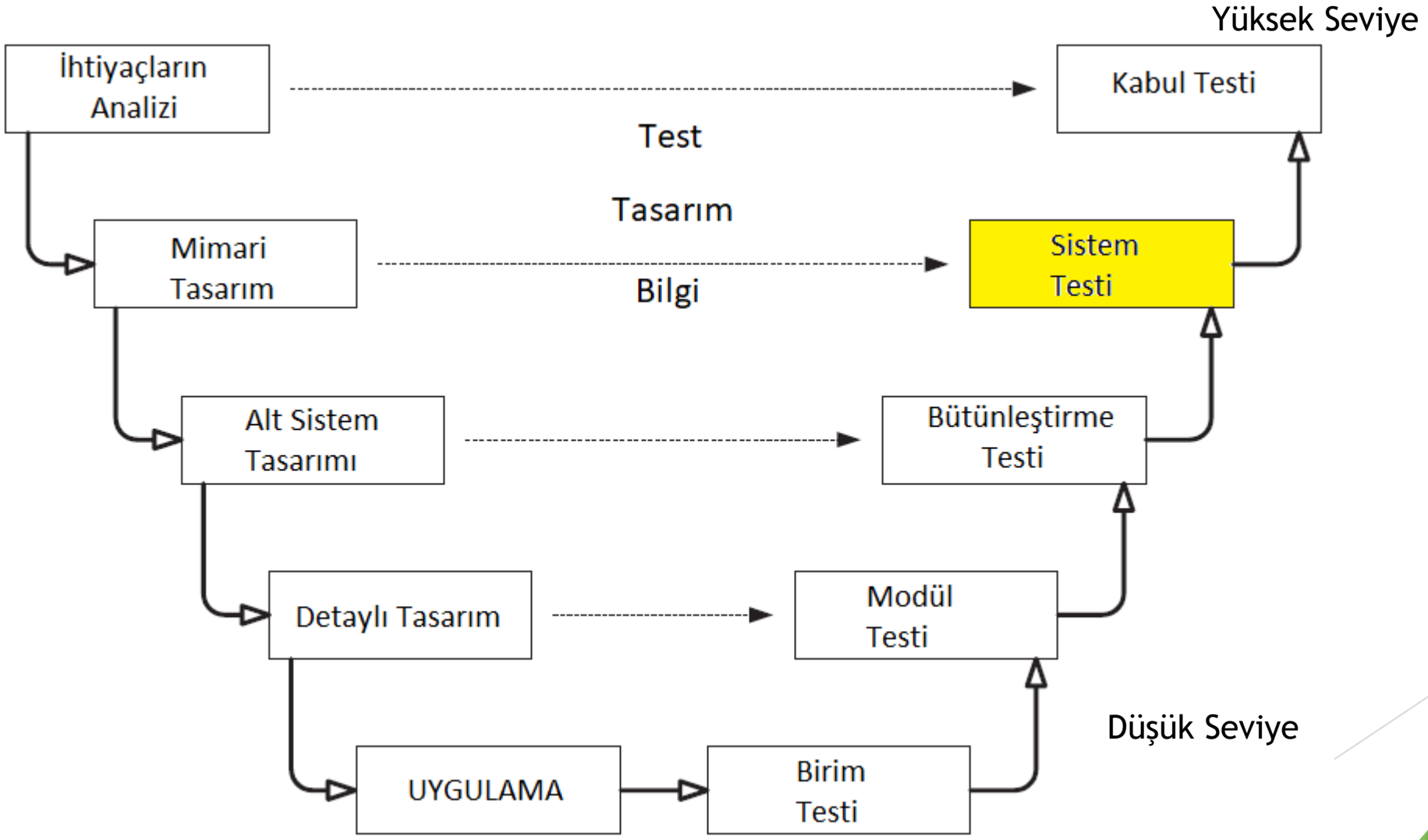
Sandviç Test Stratejisi Avantajları

- Üst ve Alt katmanlar paralel işletilebilir.
- Bağımlı olan alt sistemler birlikte çalıştırılabilir.

Sistem Testi - Tanım

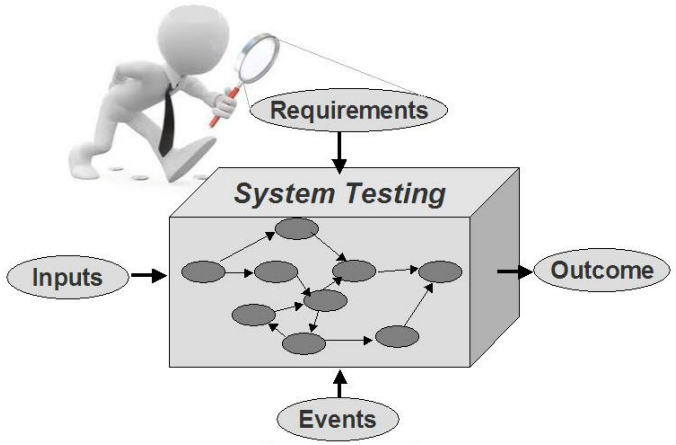
- Yazılımın davranışlarının test edildiği aşamadır.
- Yazılımda aranan özellikler gerçekleştirilmiş mi bakıldığı testtir.
- Çevre birimlerin dış uygulamaların da dahil edilip bir bütün olarak çalışmasının test edilmesidir.
- Birbiri arasındaki ilişkiler kontrol edilir.
- İş gereksinimlerinin ve uygulama mimarisinin test edildiği aşamadır.
- **Yazılımın bir bütün olarak test edildiği ilk seviyedir.**

Test Seviyelerindeki Yeri



System Testi Prensipleri

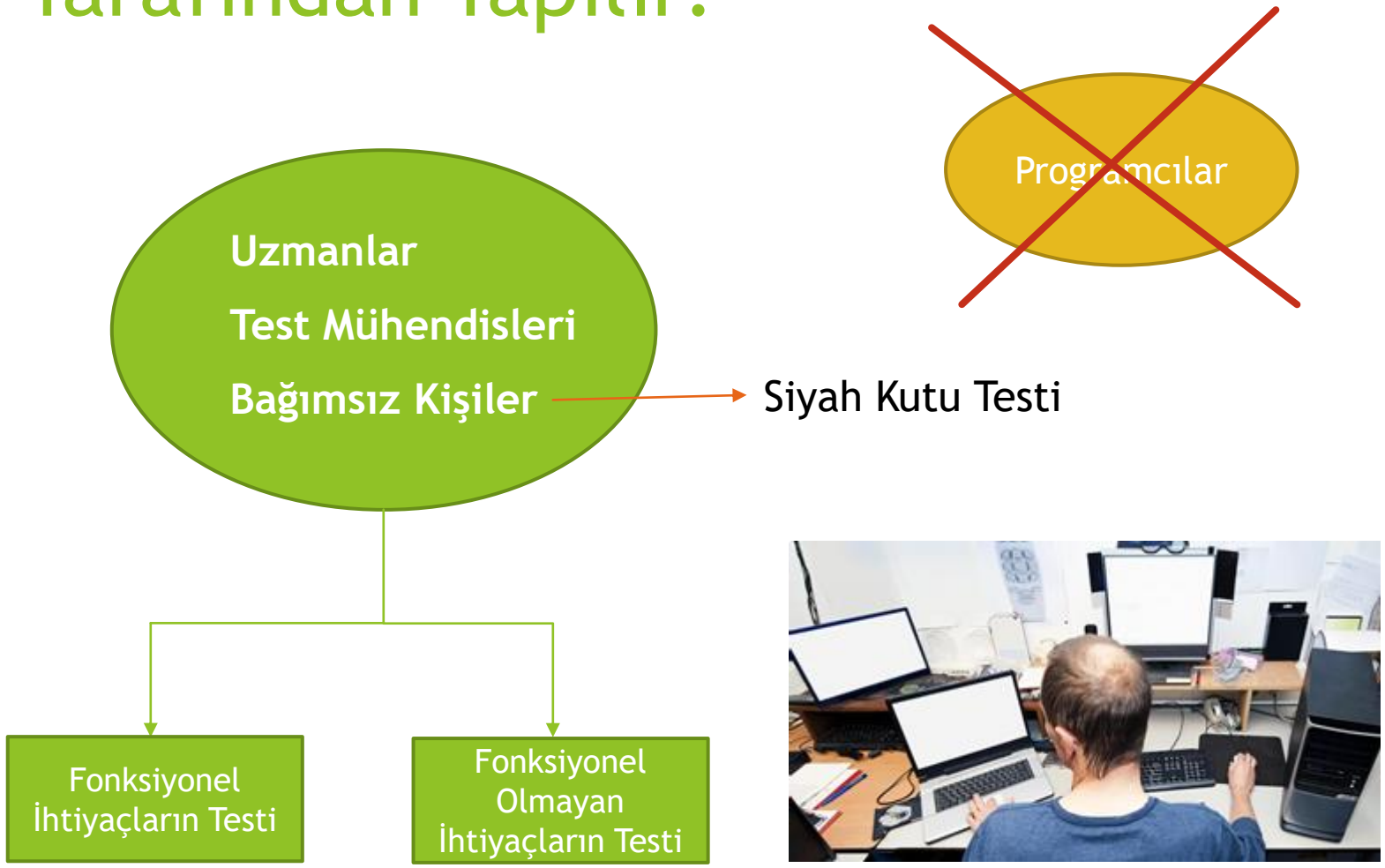
- Yeni olarak ifade edilebilecek hataları tanımlayabilir.
- Bu hatalar düzeltiliyor olabilir.
- Yeni sürümdeki bazı işlevlerin, eski sürüm ile halen aynı doğrultuda çalıştığını doğrulayabilir.



Bir Sistem Testi Durumu (case) Nasıl Yazılır?

- Birim testi yazmaya benzer.
- İki önemli adım yerine getirilmelidir.
  - Senaryoları ve durumları (use case) içermelidir.
  - Tüm ihtiyaç türlerini doğrulamalıdır.
    - Kullanıcı arayüzü (UI) ihtiyaçları
    - Fonksiyonel ihtiyaçlar
    - Fonksiyonel olmayan ihtiyaçlar
    - Performans durumları

Kim Tarafından Yapılır?

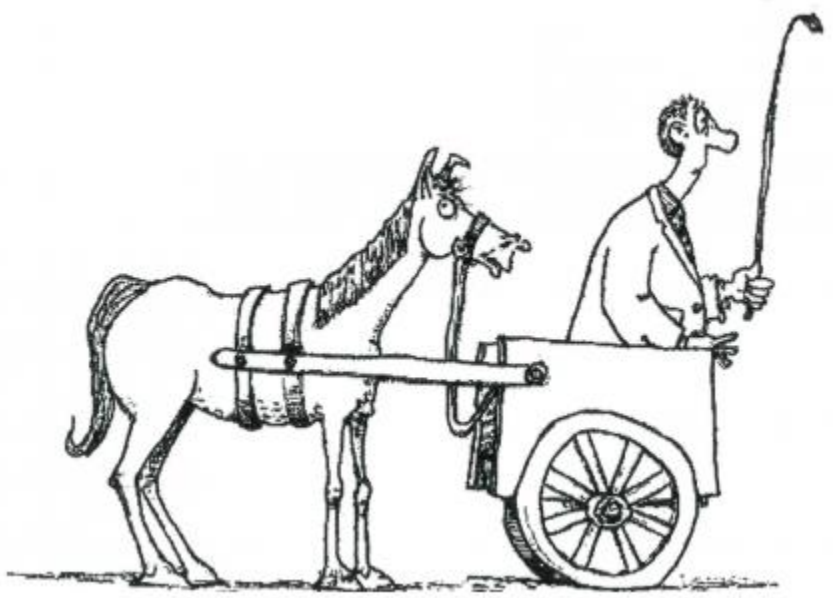


Fonksiyonel İhtiyaçlar

- İş kuralları
- İşlem düzenlemeleri, ayarlamalar, iptaller
- Yönetim fonksiyonları
- Yetkilendirme
- Yetkilendirme düzeyleri
- Dışsal arayüzler
- Sertifika ihtiyaçları
- Raporlama ihtiyaçları

Fonksiyonel Olmayan İhtiyaçlar

- Performans
- Ölçekleme
- Kapasite
- Erişilebilirlik
- Güvenirlik
- Güvenlik
- Onarılabilirlik
- Yönetilebilirlik
- Kullanılabilirlik



Fonksiyonel Olmayan Test Olarak Hacim Testi (Volume Testing)

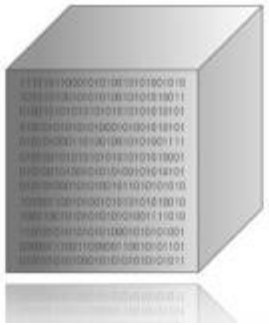
- Bu test veritabanı tarafından barındırılan veri hacmini kontrol etmek için yapılır.
- Aynı zamanda taşma testi (flood test) olarak ta adlandırılır. Programın çok yüksek veriye karşı performansının ölçülmesi.
- Çok ciddi veri işlemenin olmadığı ve gelecekte de olmayacağı yazılımlarda uygulanması gereksizdir.

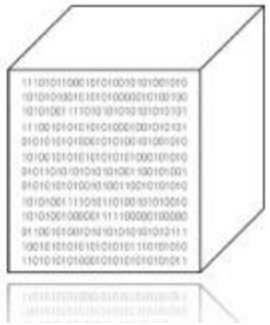
Hacim Testi ve Yük Testi Arasındaki Farklar

Hacim Testi (Volume Testing)	Yük Testi (Load Testing)
Veritabanı büyük veri hacmine karşı test edilir.	Kaynaklar farklı kullanıcı sayılarına göre test edilir.
Öncelikli bakış açısı veridir.	Öncelikli bakış açısı kullanıcıdır.
Veritabanı maks limitlere zorlanır.	Sunucu maks limitlere zorlanır.
Örnek: Büyük boyutlu bir dosyanın oluşturulması	Örnek: Çok fazla sayıda dosyanın oluşturulaması

Sistem Testi Çeşitleri

- Siyah Kutu Testi
- Beyaz Kutu Testi
- Gri Kutu Testi



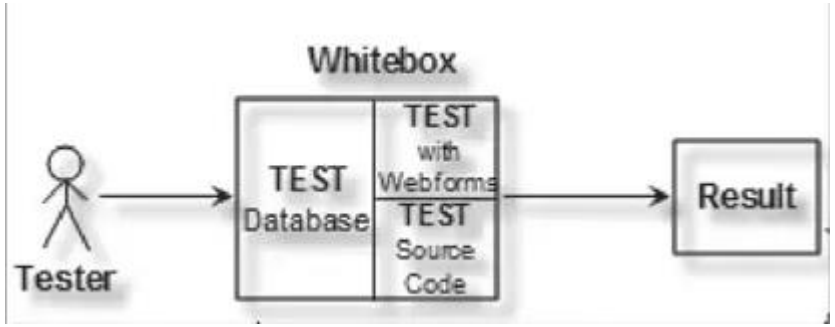


Siyah Kutu Testi

- Sistemin içerdiği detaylar ile ilgilenmeden sistemin dışarıdan test edilmesidir.
- Bulabileceği hata türleri
  - Eksik bırakılmış işlevler
  - Kullanılabilirlik problemleri
  - Performans problemleri
  - Eş zamanlılık ve zamanlama hataları
  - Başlatma ve sonlandırma hataları

Beyaz Kutu Testi

- Sistemin tasarımının ve iç detaylarının bilinerek test edilmesidir.
- Siyah kutu testine göre daha detaylı süreçler içerir.

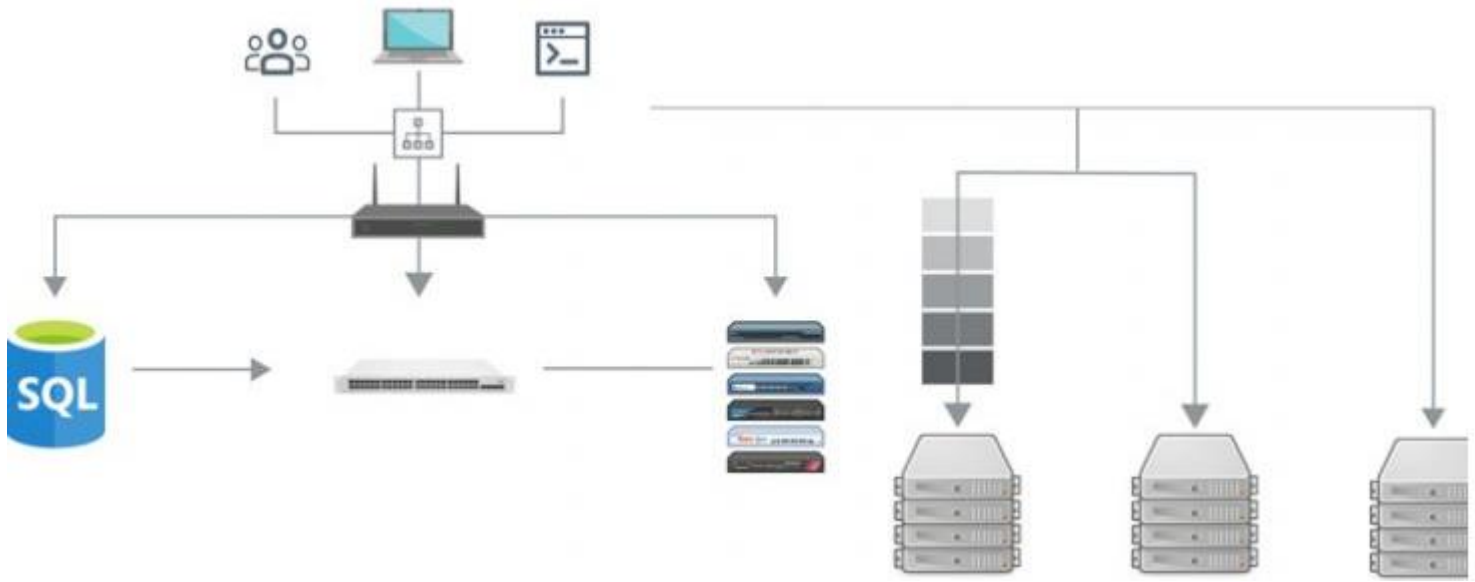


Gri Kutu Testi

- Beyaz ve Siyah kutu testinin bir araya getirilmiş halidir.
- Bu testin amacı sistemde uyumsuz yapı var mı? Kusur var mı? Kontrol edilmesidir.
- Bir yazılımda bir sistem testinin tamamlanabilmesi için hem siyah hem de beyaz kutu testinin yapılmış olması gerekir.
- Bu birlikte yapılan işleme de Gri Kutu Testi denir.

Ağ Aktivitesinin İncelenmesi

- Sistem testinin bir parçası ise yazılımı oluşturan parçalar arasında ağ aktivitesinin izlenmesidir.



Ağ Aktivitesinin İncelenmesi Örnek Uygulama

- Netpath ile Ağ Aktivitesinin izlenmesi

Güvenlik Testleri

- Bir yazılımın herhangi bir güvenlik açığından, tehlikeden, büyük bir kayba neden olabilecek risklerden arındırıldığından emin olmak için yapılan testtir.
- Bilgi kaybına neden olabilecek, yetkisiz kişiler tarafından girişe açık olan yerleri tespit etmek.

Güvenlik Testi Çeşitleri

- **Güvenlik Açığı Taraması:** Bazı yazılımlar ile yapılan açık var mı? taraması
- **Güvenlik Taraması**: Ağ ve sistem zayıflıklarından kaynaklanan risk var mı?
- **Sızma Testi**: Sisteme olası bir saldırıda içeri girilebiliyor mu?
- **Risk Değerlendirmesi**: Tespit edilen risklerin değerlendirildiği ve düşük orta yüksek şeklinde sınıflandırıldığı yer.
- **Güvenlik Denetimi**: Yazılımın iç muayenesi
- **Sosyal Hacking**: Siber saldırı metotları yerine sosyal metotlar ile gelebilecek saldırılara açık mı kontrol edilir.
- **Durum Değerlendirmesi**: Birçok değerlendirmeyi içinde barındıran genel değerlendirmedir.

Sistem Testi ile Entegrasyon Testinin Karşılaştırılması

Entegrasyon Testi	Sistem Testi
Arayüzler tarafından olaya bakar	İhtiyaçlar tarafından olaya bakar
Kod yapısının detayı vardır	Kod detayı bulunmaz
Düşük seviye bir testtir.	Yüksek seviye bir testtir
Test durumları bileşenler arasındaki arayüzün kullanılması için açık bir şekilde geliştirilmiştir.	Tüm sistem kontrollü bir ortamda yapılandırılmıştır.

Referanslar

- Naik, Kshirasagar, and Priyadarshi Tripathy. *Software testing and quality assurance: theory and practice*. John Wiley & Sons, 2011.
- Ammann, Paul, and Jeff Offutt. *Introduction to software testing*. Cambridge University Press, 2016.
- Padmini, C. "Beginners Guide To Software Testing." (2004).
- Archer, Clark, and Michael Stinson. *Object-Oriented Software Measures*. No. CMU/SEI-95-TR-002. CARNEGIE-MELLON UNIV PITTSBURGH PA SOFTWARE ENGINEERING INST, 1995.
- Pandey, Ajeet Kumar, and Neeraj Kumar Goyal. *Early Software Reliability Prediction*. Springer, India, 2015.