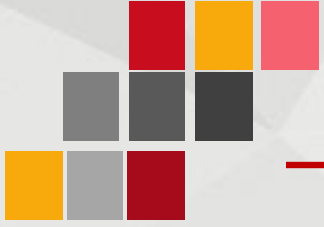
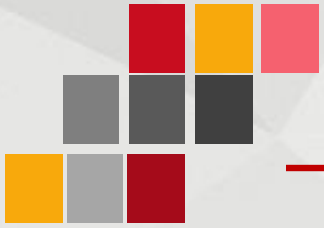


Yazılım Mühendisliği Süreç Modelleri



KONULAR

- ✓ Yazılım Yaşam Döngüsü (Çekirdek Süreçler)
- ✓ Belirtim Yöntemleri –Süreç Modelleri
- ✓ Yazılım Süreç Modelleri
- ✓ Gelişigüzel Model
- ✓ Barok Model
- ✓ Çağlayan Modeli
- ✓ V Modeli
- ✓ Spiral Model
- ✓ Evrimsel Geliştirme Süreç Modeli
- ✓ Artırımsal Geliştirme Süreç Modeli
- ✓ Araştırma Tabanlı Süreç Modeli
- ✓ Metodolojiler



Yazılım Yaşam Döngüsü

1. Planlama

Personel ve donanım gereksinimlerinin çıkarıldığı, fizibilite çalışmasının yapıldığı ve proje planının oluşturulduğu aşamadır.

2. Analiz

Sistem gereksinimlerinin ve işlevlerinin ayrıntılı olarak çıkarıldığı aşama. Var olan işler incelenir, temel sorunlar ortaya çıkarılır.

mantıksal; önerilen sistemin yapısı anlatılır,

3. Tasarım

Belirlenen gereksinimlere yanıt verecek yazılım sisteminin temel yapısının oluşturulduğu aşamadır.

mantıksal; önerilen sistemin yapısı anlatılır,

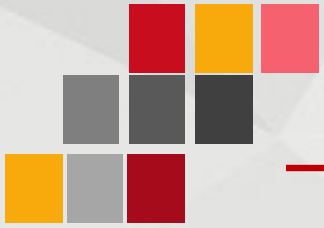
fiziksel; yazılımı içeren bileşenler ve bunların ayrıntıları.

4. Gerçekleştirim

Kodlama, test etme ve kurulum çalışmalarının yapıldığı aşamadır.

5. Bakım

Hata giderme ve yeni eklentiler yapma aşaması.



Belirtim Yöntemleri

Bir çekirdek sürece ilişkin fonksiyonları yerine getirmek veya çekirdek süreçler arası geçişlerin belirtilmesinde kullanılan yöntemler, **Belirtim Yöntemleri** olarak adlandırılır.

Süreç Akışı İçin Kullanılan Belirtim Yöntemleri

Süreçler arası ilişkilerin ve iletişimin gösterildiği yöntemler (**Veri Akış Şemaları, Yapısal Şemalar, Nesne/Sınıf Şemaları**).

Süreç Tanımlama Yöntemleri

Süreçlerin iç işleyişini göstermek için kullanılan yöntemler (**Düz Metin, Algoritma, Karar Tabloları, Karar Ağaçları, Anlatım Dili**).

Veri Tanımlama Yöntemleri

Süreçler tarafından kullanılan verilerin tanımlanması için kullanılan yöntemler (**Nesne İlişki Modeli, Veri Tabanı Tabloları, Veri Sözlüğü**).

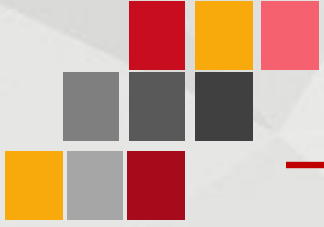


Yazılım Süreç Modelleri

- ✓ Yazılım yaşam döngüsündeki çekirdek süreçlerin geliştirme aşamasında hangi sırada uygulanacağını tanımlayan modellere **Süreç Modelleri** denir.
- ✓ Yazılım üretim işinin yapıma düzenine ilişkin rehberlerdir.

Süreçlere ilişkin ayrıntılarla ya da süreçler arası ilişkilerle ilgilenmezler.

1. Gelişigüzel Model
2. Barok Modeli
3. Çağlayan (Şelale) Modeli
4. V Modeli
5. Spiral Model
6. Evrimsel Model
7. Artırımsal Model
8. Araştırma Tabanlı Model



Gelişigüzel Model

- ✓ Herhangi bir model ya da yöntem yok.
- ✓ Geliştiren kişiye bağlı.
- ✓ İzlenebilirliği ve bakımı oldukça zor.
- ✓ Genellikle tek kişilik üretim ortamı.
- ✓ Karmaşık olmayan yazılım sistemleri.



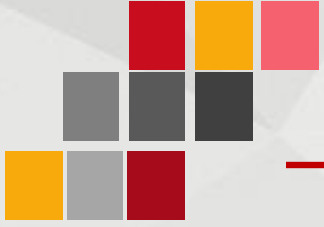
Barok Model

- ✓ Analiz
- ✓ Tasarım
- ✓ Kodlama
- ✓ Modül Testleri
- ✓ Altsistem Testleri
- ✓ Sistem Testi
- ✓ Belgeleme
- ✓ Kurulum

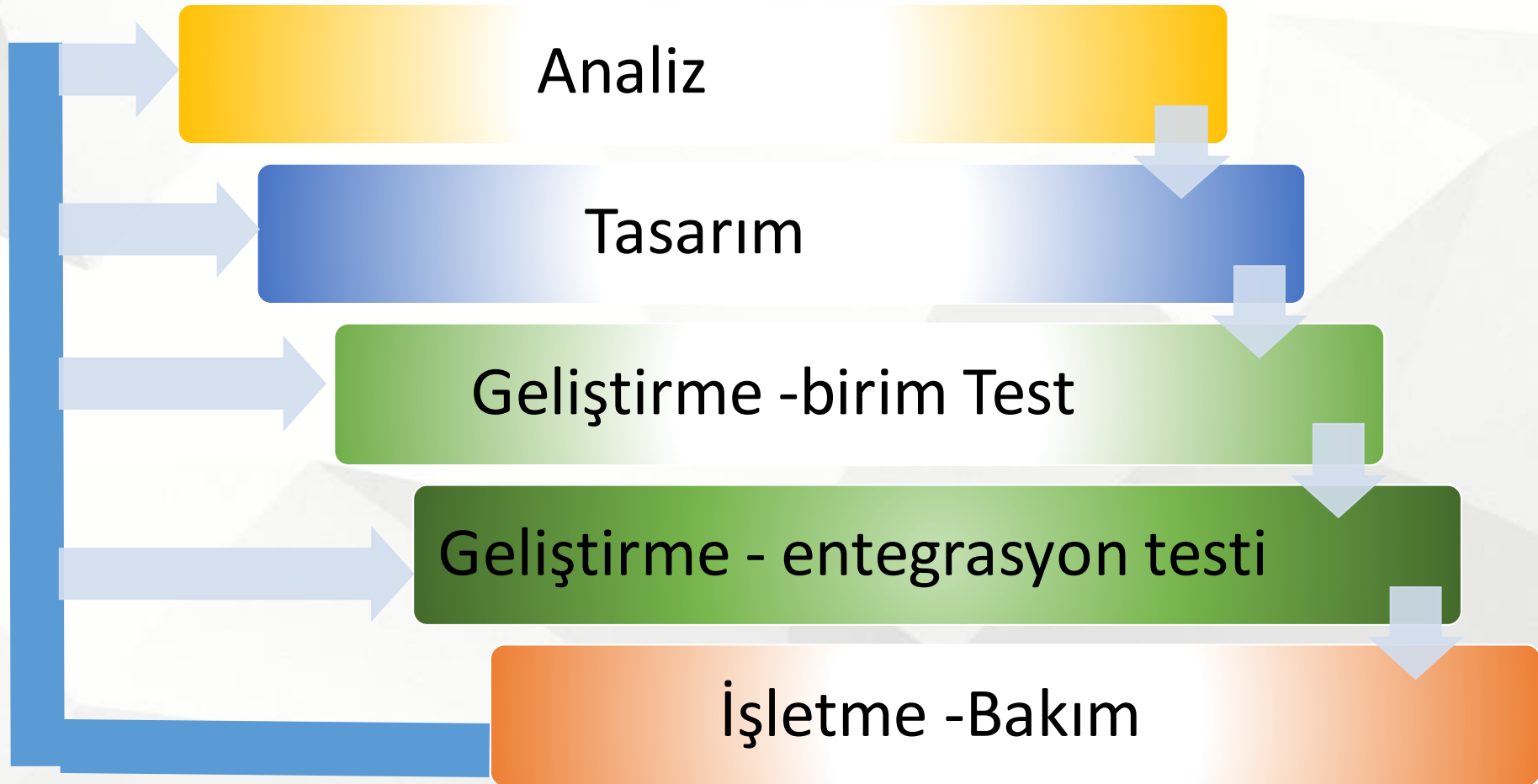
Yaşam döngüsü temel adımlarının doğrusal bir şekilde geliştirildiği model.

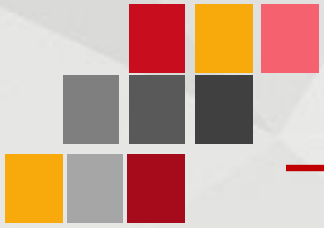
Belgelemeyi ayrı bir süreç olarak ele alır, ve yazılımın geliştirilmesi ve testinden sonra yapılmasının öngörür.

Aşamalar arası geri dönüşlerin nasıl yapılacağı tanımlı değil.



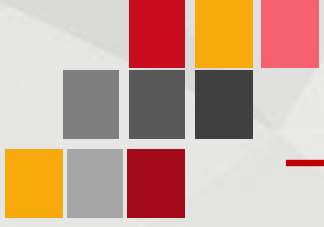
Çağlayan Modeli





Çağlayan Modeli

- ✓ Yaşam döngüsü temel adımları baştan sona en az bir kez izleyerek gerçekleştirilir.
- ✓ İyi tanımlı projeler ve üretimi az zaman gerektiren yazılım projeleri için uygun bir modeldir.
- ✓ Geleneksel model olarak da bilinen bu modelin kullanımı günümüzde giderek azalmaktadır.
- ✓ Barok modelin aksine belgeleme işlevini ayrı bir aşama olarak ele almaz ve üretimin doğal bir parçası olarak görür.
- ✓ Barok modele göre geri dönüşler iyi tanımlanmıştır.
- ✓ Yazılım tanımlamada belirsizlik yok (ya da az) ise ve yazılım üretimi çok zaman almayacak ise uygun bir süreç modelidir.

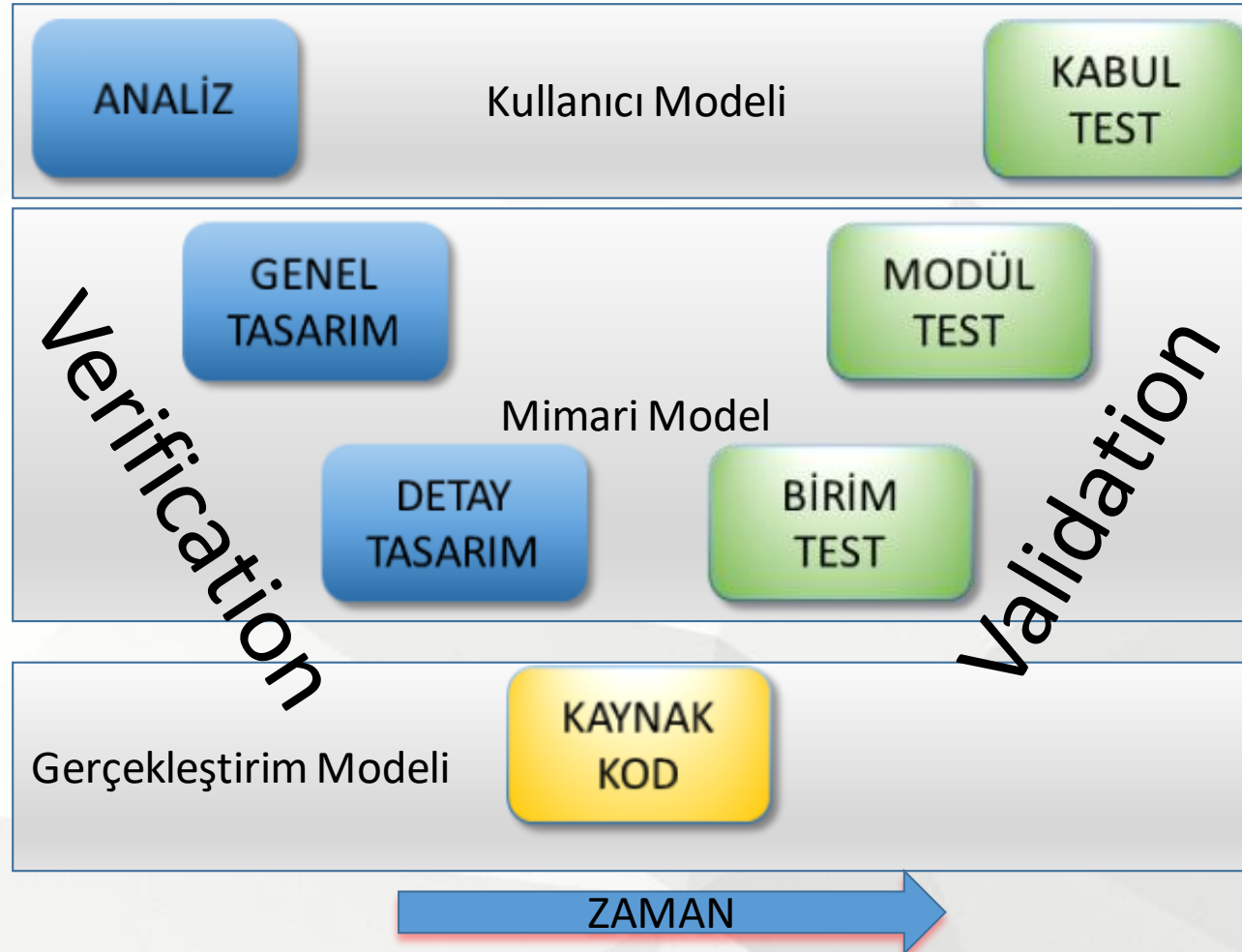


Çağlayan Modeli

- ✓ Gerçek yaşamdaki projeler genelde yineleme gerektirir.
- ✓ Genelde yazılımın kullanıcıya ulaşma zamanı uzundur.
- ✓ Gereksinim tanımlamaları çoğu kez net bir şekilde yapılamadığından dolayı, yanlışların düzeltilme ve eksiklerin giderilme maliyetleri yüksektir.
- ✓ Yazılım üretim ekipleri bir an önce program yazma, çalıştırma ve sonucu görme eğiliminde olduklarından, bu model ile yapılan üretimlerde ekip mutsuzlaşmakta ve kod yazma dışında kalan (ve iş yükünün %80'ini içeren) kesime önem vermemektedirler.
- ✓ Yöneticilerin ürünü görme süresinin uzun oluşu, projenin bitmeyeceği ve sürekli gider merkezi haline geldiği düşüncesini yaygınlaştırmaktadır



V Süreç Modeli





V Süreç Modeli

Sol taraf üretim, sağ taraf sınama işlemleridir.

V süreç modelinin temel çıktıları;

Kullanıcı Modeli

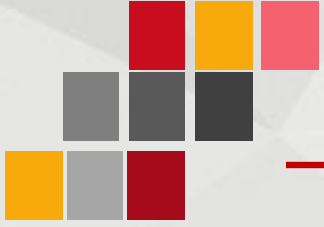
Geliştirme sürecinin kullanıcı ile olan ilişkileri tanımlanmakta ve sistemin nasıl kabul edileceğine ilişkin sınama belirtimleri ve planları ortaya çıkarılmaktadır.

Mimari Model

Sistem tasarımı ve oluşacak altsistem ile tüm sistemin sınama işlemlerine ilişkin işlevler.

Gerçekleştirim Modeli

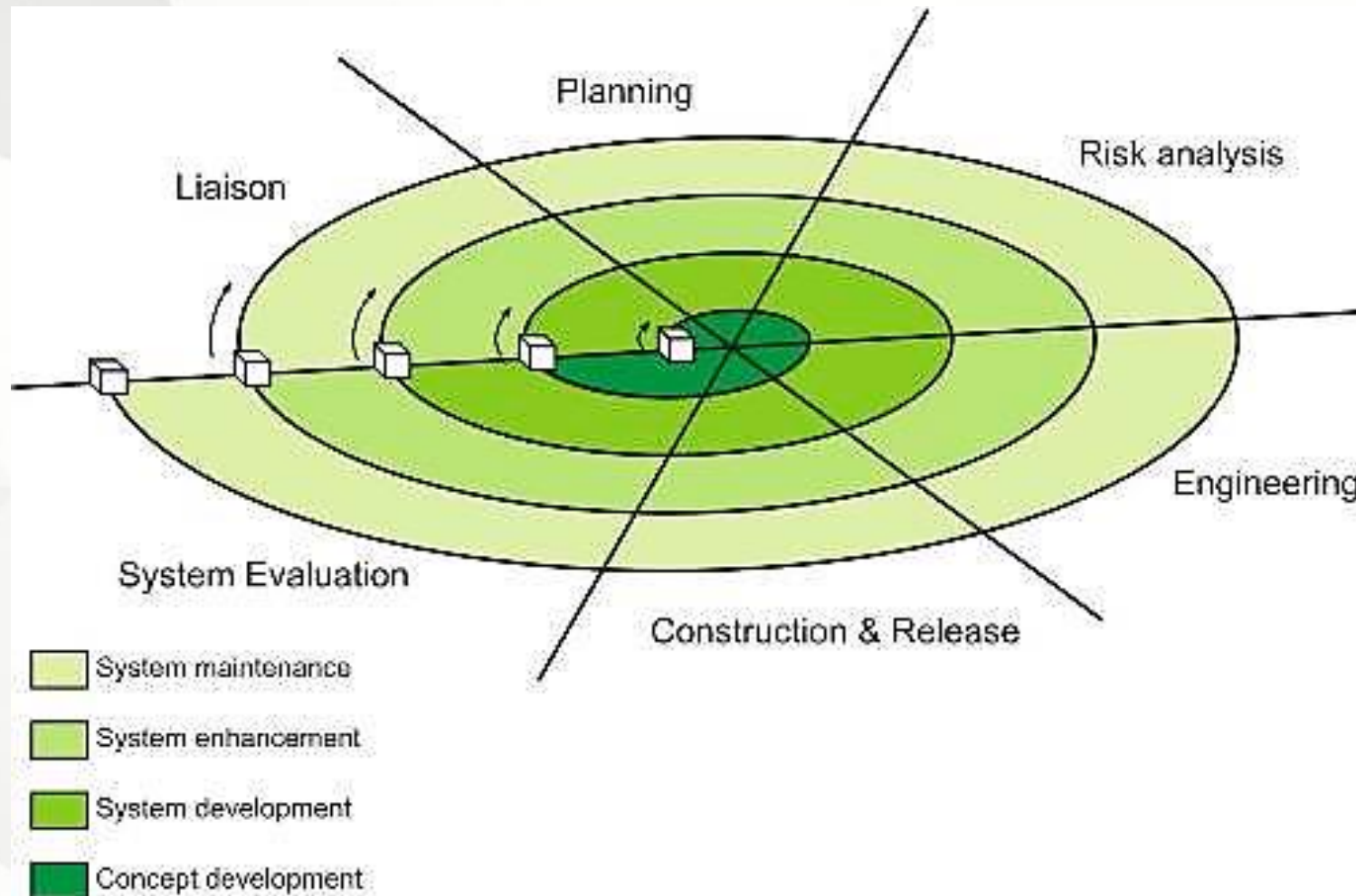
Yazılım modüllerinin kodlanması ve sınanmasına ilişkin fonksiyonlar.



V Süreç Modeli

- ✓ Belirsizliklerin az, iş tanımlarının belirgin olduğu BT projeleri için uygun bir modeldir.
- ✓ Model, kullanıcının projeye katkısını arttırmaktadır.
- ✓ BT projesinin iki aşamalı olarak ihale edilmesi için oldukça uygundur:
 - İlk ihalede kullanıcı modeli hedeflenerek, iş analizi ve kabul sınamalarının tanımları yapılmakta,
 - İkinci ihalede ise ilkinde elde edilmiş olan kullanıcı modeli tasarlanıp, gerçekleştirilmektedir.

Spiral Model



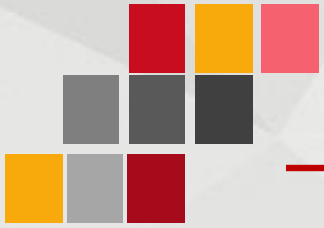


Spiral Model

Tasarımı doğrusal bir süreç olarak gören diğer modellerin aksine, bu model spiral bir süreç olarak görür. Bu, yineleyici tasarım döngülerini genişleyen bir spiral olarak temsil ederek yapılır.

Genellikle iç çevrimler, gereksinim tanımının rafine edilmesi için prototipleme ile birlikte ihtiyaç analizinin erken evresini ve dış spiraller yazılım tasarımını aşamalı olarak temsil eder.

Her helezonda, tasarım çabalarını ve bu yineleme için ilgili riski değerlendirmek için bir risk değerlendirme aşaması vardır. Her spiralin sonunda, mevcut spiralin gözden geçirilebilmesi ve bir sonraki aşamanın planlanabilmesi için gözden geçirme aşaması vardır.



Spiral Model

Her tasarım sarmalının altı ana faaliyeti altı temel görevle temsil edilmektedir:

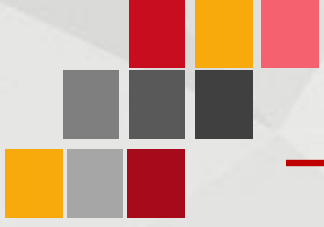
1. Müşteri İletişimi
2. Planlama
3. Risk Analizi
4. Yazılım Tasarımı
5. Üretim-dağıtım
6. Müşteri onayı

Avantajları

1. Risk analizi yapmaktadır.
2. Bu yazılım tasarım modeli, büyük yazılım projelerini tasarlamak ve yönetmek için daha uygundur.

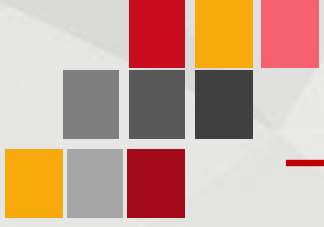
Dezavantajları

1. Risk analizi yüksek uzmanlık gerektirir.
2. Kullanması pahalı model
3. Küçük projeler için uygun değildir.

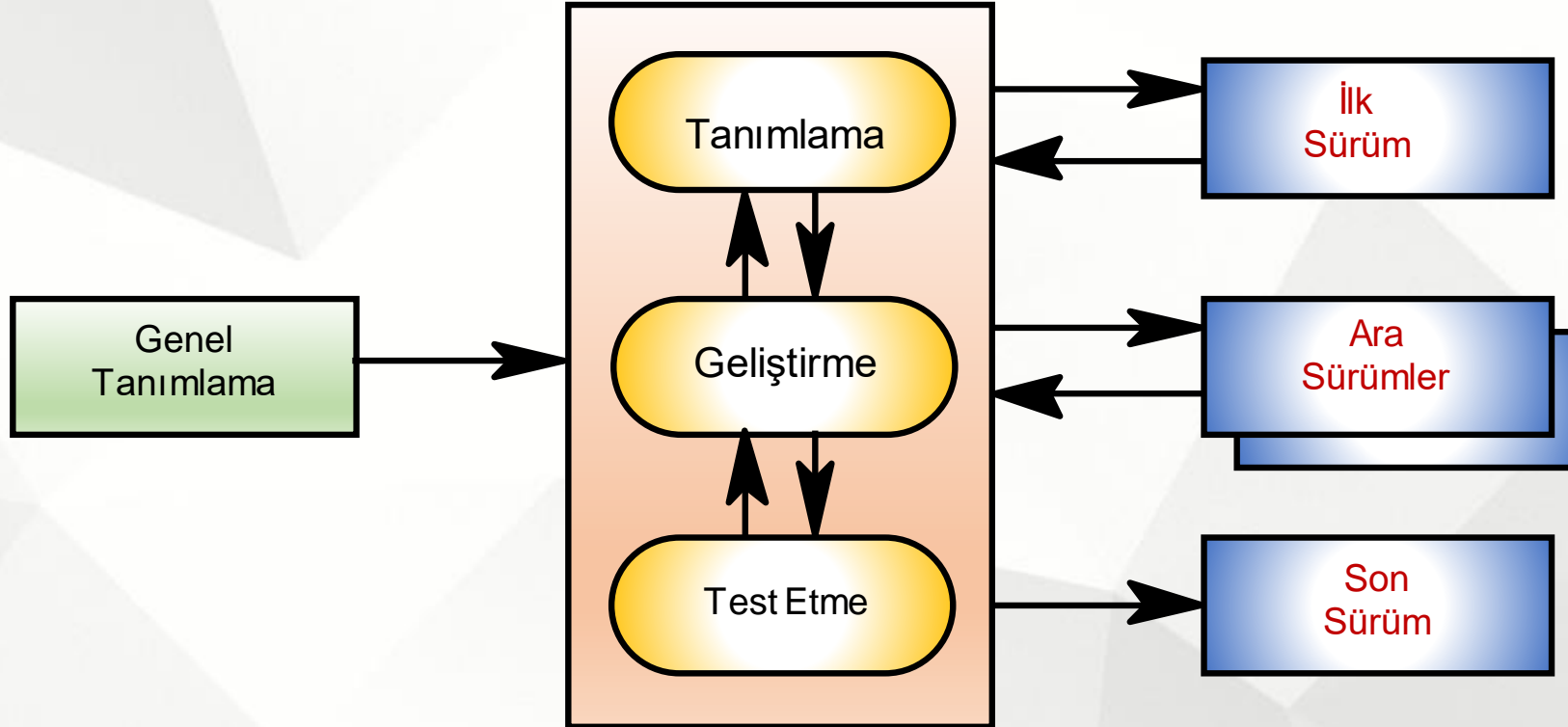


Evrimsel Geliştirme Süreç Modeli

- İlk tam ölçekli modeldir.
- Coğrafik olarak geniş alana yayılmış, çok birimli organizasyonlar için önerilmektedir.
- Her aşamada üretilen ürünler, üretildikleri alan için tam işlevselliği içermektedirler.
- Pilot uygulama kullan, test et, güncelle diğer birimlere taşı.
- Modelin başarısı ilk evrimin başarısına bağlıdır



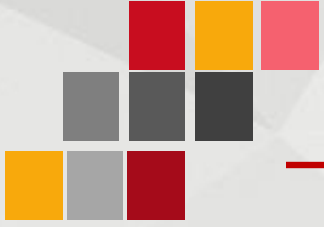
Evrimsel Geliştirme Süreç Modeli





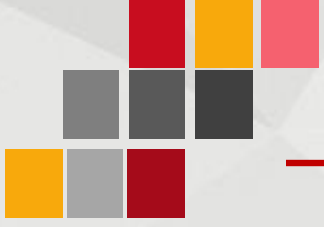
Evrimsel Geliştirme Süreç Modeli

- ✓ Çok birimli banka uygulamaları.
- ✓ Önce sistem geliştirilir ve Şube-1'e yüklenir.
- ✓ Daha sonra aksaklıklar giderilerek geliştirilen sistem Şube-2'ye yüklenir.
- ✓ Daha sonra geliştirilen sistem Şube-3'e,.... yüklenir.
- ✓ Belirli aralıklarla eski şubelerdeki güncellemeler yapılır.



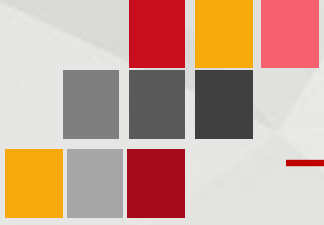
EKSİKLERİ

- ✓ Değişiklik denetimi
- ✓ Konfigürasyon Yönetimidir
 - Sürüm Yönetimi
 - Değişiklik Yönetimi
 - Kalite Yönetimi

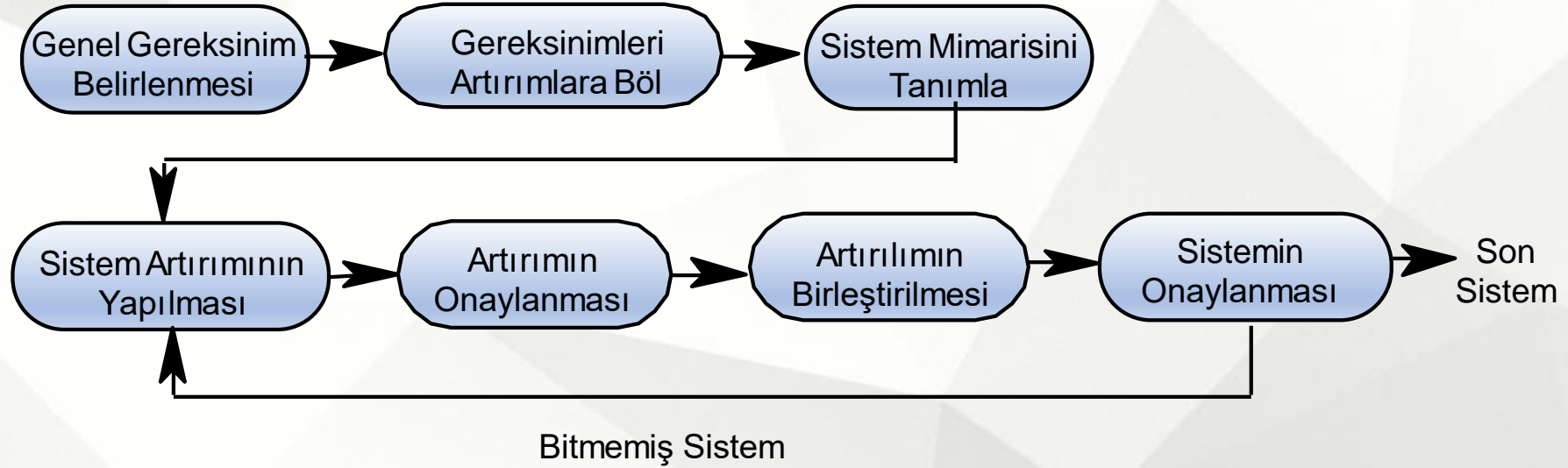


Artırımsal Geliştirme Süreç Modeli

- ✓ Üretilen her yazılım sürümü birbirini kapsayacak ve giderek artan sayıda işlev içerecek şekilde geliştirilir.
- ✓ Öğrencilerin bir dönem boyunca geliştirmeleri gereken bir programlama ödevinin 2 haftada bir gelişiminin izlenmesi (bitirme tezleri).
- ✓ Uzun zaman alabilecek ve sistemin eksik işlevlikle çalışabileceği türdeki projeler bu modele uygun olabilir.
- ✓ Bir taraftan kullanım, diğer taraftan üretim yapılır



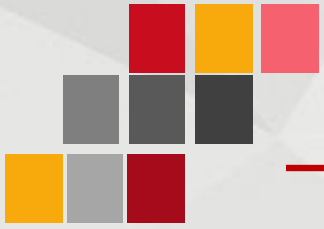
Artırımsal Geliştirme Süreç Modeli





Araştırma Tabanlı Süreç Modeli

- ✓ Yap-at prototipi olarak ta bilinir.
- ✓ Araştırma ortamları **bütünüyle belirsizlik** üzerine çalışan ortamlardır.
- ✓ Yapılan işlerden edinilecek sonuçlar belirgin değildir.
- ✓ Geliştirilen **yazılımlar genellikle sınırlı sayıda kullanılır** ve kullanım bittikten sonra işe yaramaz hale gelir ve atılır.
- ✓ Model-zaman-fiya kestirimi olmadığı için **sabit fiyat sözleşmelerinde uygun değildir.**



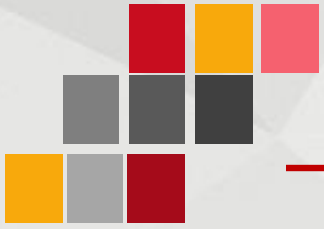
Metodolojiler

Metodoloji: Bir BT projesi ya da yazılım yaşam döngüsü aşamaları boyunca kullanılacak birbirleriyle uyumlu yöntemler bütünü.

Bir metodoloji,

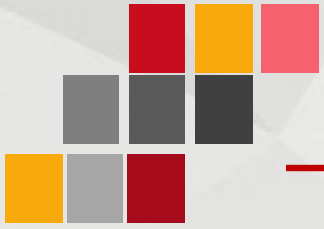
- bir süreç modelini ve
- belirli sayıda belirtim yöntemini içerir

Günümüzdeki metodolojiler; **çağlayan, V ya da spiral süreç modellerini** temel almaktadır.



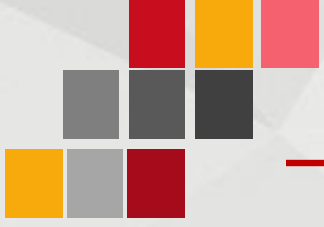
Bir Metodolojide Bulunması Gereken Temel Bileşenler

- ✓ Ayrıntılı bir süreç modeli
- ✓ Ayrıntılı süreç tanımları
- ✓ İyi tanımlı üretim yöntemleri
- ✓ Süreçler arası ara yüz tanımları
- ✓ Ayrıntılı girdi tanımları
- ✓ Ayrıntılı çıktı tanımları
- ✓ Proje yönetim modeli
- Konfigürasyon yönetim modeli
- Maliyet yönetim modeli
- Kalite yönetim modeli
- Risk yönetim modeli
- Değişiklik yönetim modeli
- Kullanıcı arayüz modeli
- Standartlar



Bir Metodolojide Bulunması Gereken Temel Bileşenler

- ✓ Metodoloji bileşenleri ile ilgili olarak bağımsız kuruluş (IEEE, ISO, vs.) ve kişiler tarafından geliştirilmiş çeşitli standartlar ve rehberler mevcuttur.
- ✓ Kullanılan süreç modelleri ve belirtim yöntemleri zaman içinde değiştiği için standart ve rehberler de sürekli güncellenmektedir.



Yourdon Yapısal Sistem Tasarım Metodolojisi

Aşama	Kullanılan Yöntem ve Araçlar	Ne için Kullanıldığı	Çıktı
Planlama	Veri Akış Şemaları, Süreç Belirtilimleri, Görüşme, Maliyet Kestirim Yöntemi, Proje Yönetim Araçları	Süreç İnceleme Kaynak Kestirimi Proje Yönetimi	Proje Planı
Analiz	Veri Akış Şemaları, Süreç Belirtilimleri, Görüşme, Nesne ilişki şemaları Veri	Süreç Analizi Veri Analizi	Sistem Analiz Raporu
Analizden Tasarıma Geçiş	Akışa Dayalı Analiz, Süreç belirtilimlerinin program tasarım diline dönüştürülmesi, Nesne ilişki şemalarının veri tablosuna dönüştürülmesi	Başlangıç Tasarımı Ayrıntılı Tasarım Başlangıç Veri tasarımı	Başlangıç Tasarım Raporu
Tasarım	Yapısal Şemalar, Program Tasarım Dili, Veri Tabanı Tabloları	Genel Tasarım Ayrıntılı Tasarım Veri Tasarımı	Sistem Tasarım Raporu