



İZMİR BAKIRÇAY ÜNİVERSİTESİ MİMARLIKVE MUHENDİSLİK FAKÜLTESİ BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

BİL 102 YAZILIM MÜHENDİSLİĞİ TEMELLERİ ÖDEVİ
yazılım geliştirme modelleri

19 MART 2023

210601015 MEHMET ŞİRİN AKIN

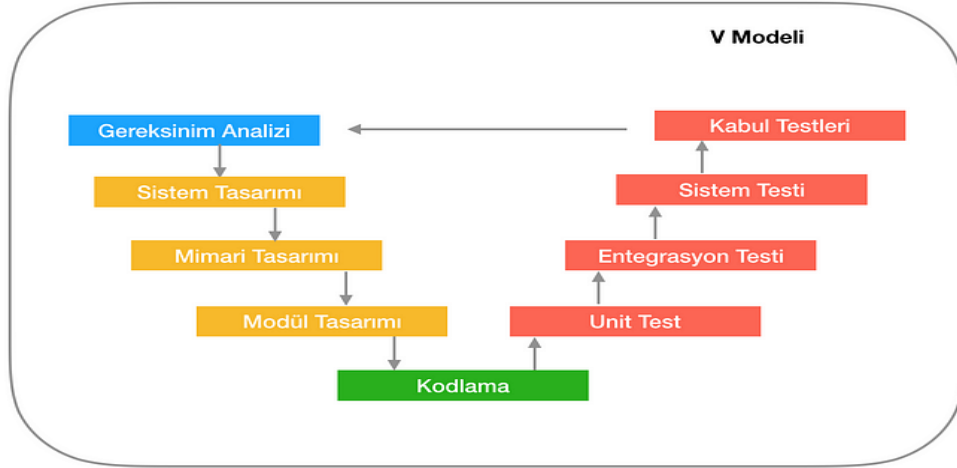
<https://github.com/MehmetsirinAKIN>

<https://medium.com/@msrnakin4949>

<https://www.linkedin.com/in/mehmet-%C5%9Firin-akin-809ba526a/>

1. V Modeli:

Bu model, gereksinimleri belirleme, tasarım, kodlama, test etme ve dağıtım gibi yazılım geliştirme sürecinin aşamalarını sırayla ve çift yönlü bir şekilde ele alır. Her aşamanın karşılıklı bir test aşaması vardır. Bu model, yazılım geliştirme sürecinin planlamasını ve yönetimini kolaylaştırır.



V Modeli: Avantajları:

- Sıkı bir belgelendirme süreci sayesinde, projenin erken aşamalarında hataların tespit edilmesi daha olasıdır.
- Geleneksel bir model olması nedeniyle, proje planlaması daha kolaydır.
- Projenin her aşaması için kesin ve önceden belirlenmiş bir hedefi vardır.

Dezavantajları:

- Model, değişiklikleri ve müşteri geri bildirimlerini kabul etmekte zorlanabilir.
- Her aşama tamamlandıktan sonra geri dönülmesi mümkün değildir.
- Model, yazılım geliştirme sürecini sıkı bir şekilde takip etmek için yüksek düzeyde disiplin gerektirir.

2. Spiral Model:

Bu model, risk yönetimi odaklıdır. Proje geliştirme süreci, bir spiralin dönüşleri gibi aşamalı olarak ilerler. Her dönüş, bir önceki dönüşte alınan geri bildirimlerin analiziyle bir sonraki aşamaya hazırlanır. Bu model, özellikle büyük ve karmaşık projeler için uygundur.



Spiral Model: Avantajları:

- Risk analizi ve yönetimi, projenin her aşamasında yapılacağından, riskler daha erken tespit edilip, daha iyi yönetilir.
- Değişen gereksinimlere uyum sağlamak daha kolaydır.
- Prototipler sayesinde, müşteri geri bildirimleri daha erken alınır.

Dezavantajları:

- Model, prototip üretimi nedeniyle maliyetli olabilir.
- Sürekli tekrarlanan döngüler, projenin gecikmesine neden olabilir.
- Model, risklerin doğru bir şekilde yönetilmesi için yüksek düzeyde tecrübe gerektirir.

3. Artımlı ve Yinelemeli Model:

Bu model, geliştirme sürecinin aşamalarını parçalara ayırır ve her parçanın tamamlanması için bir veya daha fazla yineleme kullanır.

Her yineleme, işlevsellik ekler ve önceki yinelemelerden elde edilen geri bildirimlerle geliştirilir. Bu model, projenin esnekliğini ve müşteri odaklılığını artırır.



Artımlı ve Yinelemeli Model: Avantajları:

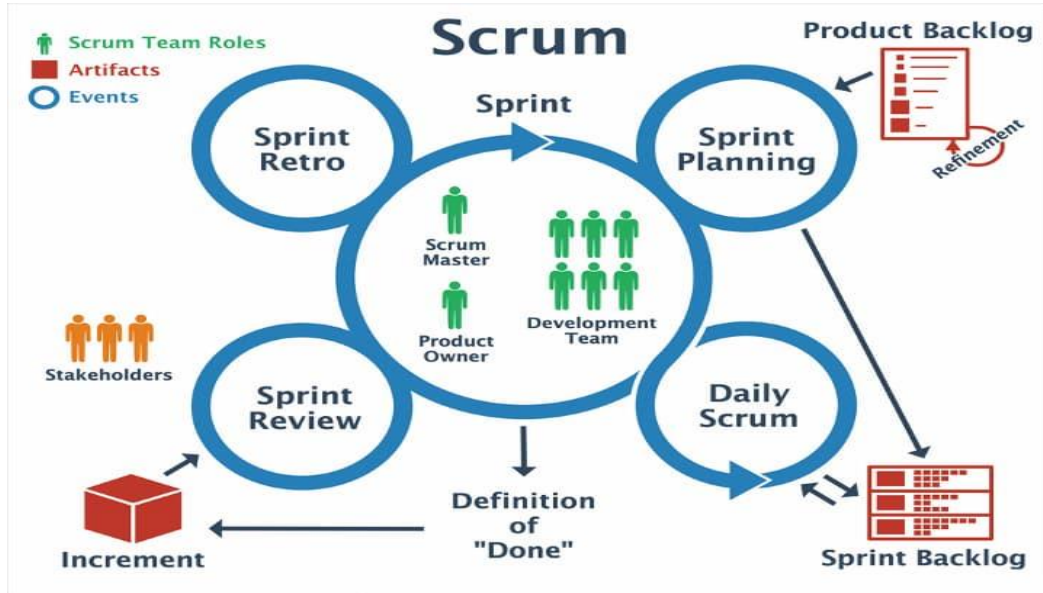
- Projenin her aşamasında müşteri geri bildirimleri alınarak, değişen gereksinimlere kolayca uyum sağlanır.
- Model, maliyetleri daha iyi kontrol etmeyi sağlar.
- Geliştirme süreci daha hızlı ve esnektir.

Dezavantajları:

- Her yinelemede yapılan testler, maliyet ve zaman açısından dezavantajlı olabilir.
- Her yinelemede proje yönetimindeki süreçlerin tekrarlanması nedeniyle, proje yönetimi zorlaşabilir.
- Model, tamamlanmamış işlerin ortaya çıkması riskini taşır.

4. Çevik Model ve Scrum Modeli:

Bu model, artımlı ve yinelemeli bir yaklaşıma dayanır. Proje, kısa ve yoğun iş döngüleri olan "sprint" adı verilen zaman dilimlerinde bölünür. Bu sprint'lerde, belirlenmiş bir dizi hedefe ulaşmak için ekipler birlikte çalışırlar. Çevik modelin bir türü olan Scrum, özellikle yazılım geliştirme ekipleri arasında popülerdir ve ürün geliştirme sürecinde müşteri işbirliği, adaptasyon ve öz-organizasyonun önemini vurgular.



Çevik ve Scrum Modeli: Avantajları:

- Müşteri odaklı bir yaklaşım sayesinde, müşteri gereksinimleri daha iyi anlaşılır ve değişen gereksinimlere kolayca uyum sağlanır.
- Model, ekip işbirliğini ve sürekli iletişimi teşvik eder.
- Sprint'ler sayesinde, projenin ilerlemesi takip edilir ve hızlı geri bildirimler alınır.

Dezavantajları:

- Model, yüksek düzeyde organizasyongerektirir, bu nedenle ekiplerin buna hazır olması gerekir.
- Çevik ve Scrum modeli, geleneksel yöntemlerden farklı olduğu için, bazı şirketler ve müşteriler için yeni ve yabancı olabilir.
- Model, belirli bir süreç ve rol tanımlarına ihtiyaç duyduğu için, bazı projeler için uygun olmayabilir.

SCRUM günümüzde neden popüler?

SCRUM, günümüzde yazılım geliştirme endüstrisinde popüler bir modeldir çünkü müşteri odaklı bir yaklaşım benimser. Bu model, müşteri gereksinimlerini anlamak ve değişen gereksinimlere hızlı bir şekilde uyum sağlamak için sprint'ler kullanır. Ayrıca, ekip işbirliği ve sürekli iletişimi teşvik eder, bu da proje yönetimi için önemlidir. SCRUM ayrıca, sürekli iyileştirme felsefesi sayesinde, projenin kalitesini artırmayı hedefler. Bu nedenlerden dolayı, SCRUM, esnek, hızlı, müşteri odaklı ve yüksek kaliteli yazılım geliştirme süreci için popüler bir seçenektir.

SCRUM günümüzde neden popüler? SCRUM, günümüzde yazılım geliştirme ekipleri arasında popüler bir model haline gelmiştir. Bunun nedeni, birkaç faktöre dayanmaktadır:

1. Esneklik: SCRUM, hızlı iş döngüleri ve sürekli geri bildirimler sağlayarak projenin esnek olmasına olanak tanır.
2. Müşteri Odaklılık: SCRUM, müşteri işbirliği ve müşteri gereksinimlerinin sürekli olarak göz önünde bulundurulması ile müşteri odaklılığı artırır.
3. Ekip İşbirliği: SCRUM, ekip üyeleri arasında işbirliğini ve öz-organizasyonu öne çıkarır, bu da ekibin daha verimli ve etkili olmasına yardımcı olur.
4. Hız: SCRUM, sprint'ler adı verilen kısa ve yoğun iş döngüleriyle birlikte projenin daha hızlı tamamlanmasını sağlar.
5. Adaptasyon: SCRUM, sürekli geri bildirimler ve sprint'ler aracılığıyla projenin değişen ihtiyaçlara ve koşullara uygun şekilde adapte edilmesine olanak tanır.

Bu faktörler, SCRUM'un yazılım geliştirme ekipleri arasında popüler olmasını sağlayan ana nedenlerdir.

SCRUM'un popülerliği aynı zamanda, geleneksel yazılım geliştirme modellerinin (örneğin Waterfall modeli) sınırlamalarına da bir tepki olarak ortaya çıkmıştır. Waterfall modeli gibi geleneksel modeller, projenin önceden belirlenen bir plana göre ilerlemesini öngörürler. Ancak, bu modellerin esneklikleri düşüktür ve değişen koşullara veya müşteri gereksinimlerine uyum sağlamakta zorlanırlar.

SCRUM, değişkenlik ve hızlı tepki verme becerisi ile diğer modellere kıyasla daha uygun olabilir. Bu nedenle, özellikle karmaşık ve değişken projelerde, SCRUM daha çok tercih edilmektedir.

SCRUM aynı zamanda, açık iletişim ve sürekli geri bildirimlere dayalı bir yaklaşımı benimser. Bu, projenin tüm paydaşları arasında daha iyi bir işbirliği sağlayarak, projenin başarısını artırabilir.

SCRUM, ürün sahibi, geliştirme ekibi ve SCRUM master gibi farklı rolleri tanımlar. Bu roller, projenin farklı yönlerine odaklanarak, projenin daha verimli bir şekilde yönetilmesini sağlar. Örneğin, ürün sahibi, müşteri gereksinimlerinin tanımlanmasında ve ürünün önceliklerinin belirlenmesinde sorumludur. Geliştirme ekibi ise, ürünün geliştirilmesi ve teslim edilmesi için çalışır. SCRUM master ise, ekip üyeleri arasında işbirliğini teşvik eder, engelleri ortadan kaldırır ve SCRUM süreçlerinin doğru bir şekilde uygulanmasını sağlar.

SCRUM'un bir diğer popüler özelliği, sprint'lerin kullanılmasıdır. Sprint'ler, birkaç haftalık kısa ve yoğun iş döngüleridir. Her sprint, önceden belirlenmiş bir hedefe odaklanır ve sprint sonunda tamamlanması gereken işleri içerir. Sprint'ler, projenin ilerlemesinin takip edilmesini ve sürekli geri bildirimlerin alınmasını sağlar.

Hangi projede hangi modeli kullanmalıyız?

Hangi modelin kullanılması gerektiği, projenin özelliklerine, gereksinimlerine ve müşteri ihtiyaçlarına bağlıdır. İşte her bir modelin hangi projeler için uygun olduğuna dair birkaç örnek:

1. V Modeli:

- Karmaşık ve büyük ölçekli projeler için uygundur.
- Müşteri gereksinimleri önceden belirlenmiş ve değişmeyecekse tercih edilebilir.
- Projelerin süreçlerinin ve sonuçlarının belirgin bir şekilde tanımlandığı askeri, savunma ve uzay endüstrileri gibi sektörlerde kullanılabilir.

2. Spiral Model:

- Yeni teknolojilerin kullanıldığı ve risklerin yüksek olduğu projelerde tercih edilebilir.
- Projenin çıktısı üzerindeki değişikliklerin beklenildiği projeler için uygundur.
- Ürünün prototiplerinin sık sık kontrol edildiği proje süreçleri için uygun bir modeldir.

3. Artımlı/Yinelemeli Model:

- Büyük projelerin bölümlere ayrıldığı, parçaların tek tek üretildiği ve test edildiği projeler için uygundur.
- Müşterinin gereksinimleri üzerinde çalışmanın ön planda olduğu projeler için uygundur.
- İşlevsellik ve tasarımın her bir aşamasının ayrı ayrı geliştirildiği projeler için uygundur.

4. Çevik ve Scrum Modeli:

- Küçük veya orta ölçekli projeler için uygun bir modeldir.

- Değişen müşteri gereksinimlerinin hızlı bir şekilde ele alınması gerektiği projeler için uygundur.
- Ekip üyeleri arasındaki işbirliğinin önemli olduğu projeler için uygundur.
- Projenin kalitesinin artırılması ve sürekli iyileştirme hedefi olan projeler için uygundur.

Özetle, hangi modelin kullanılacağı, projenin gereksinimlerine ve özelliklerine göre değişebilir. Projede ne tür bir çıktı elde etmek istediğinizi, müşteri gereksinimlerinin nasıl değiştiğini ve hangi endüstride çalıştığınızı göz önünde bulundurarak en uygun modeli seçmeniz gerekir.

V Modeli, Spiral Model, Artımlı/Yinelemeli Model ve Scrum Modeli'nin karşılaştırması yer almaktadır:

1. V Modeli:

- Lineer bir yapıya sahiptir ve işlemler belirli bir sıraya göre gerçekleştirilir.
- Teknik belgelerin, test dokümantasyonlarının ve izlenebilirlik matrislerinin hazırlanması önemlidir.
- Teknik ekiplerin birbiriyle iletişim kurması yerine, üretim ekibiyle test ekibi arasında bir ayrım vardır.
- Planlama, test, tasarım ve kodlama aşamaları ayrı ayrı gerçekleştirilir.
- Projenin sonucu belirli bir şekilde tahmin edilebilir ve riskler önceden belirlenir.

2. Spiral Model:

- Projeye özgü bir süreç takip edilir ve bu süreçte belirli aşamalar tekrar edilir.
- Riskler sürekli olarak değerlendirilir ve kontrol edilir.
- Projenin sonuçlarının oluşması için gereksinimlerin belirlenmesi, tasarım, geliştirme ve test aşamaları tekrarlanır.
- Müşteriye prototipler gösterilerek, geri bildirimler alınır ve değişiklikler yapılır.
- Proje esnek ve adaptif bir yapıya sahiptir.

3. Artımlı/Yinelemeli Model:

- Proje işlevselliği bölümlere ayrılır ve her bölüm tek tek geliştirilir.
- Her bir bölüm, kendine özgü bir döngü izler ve bu döngü tasarım, geliştirme ve test aşamalarını içerir.
- Müşteri geri bildirimleri ve gereksinimler sürekli olarak değerlendirilir ve işlevselliğe yeni özellikler eklenir.

- İterasyonlar, tüm proje boyunca devam eder ve her bir adımda, müşteri isteklerinin tam olarak karşılanması sağlanır.
- Proje esnek ve adaptif bir yapıya sahiptir.

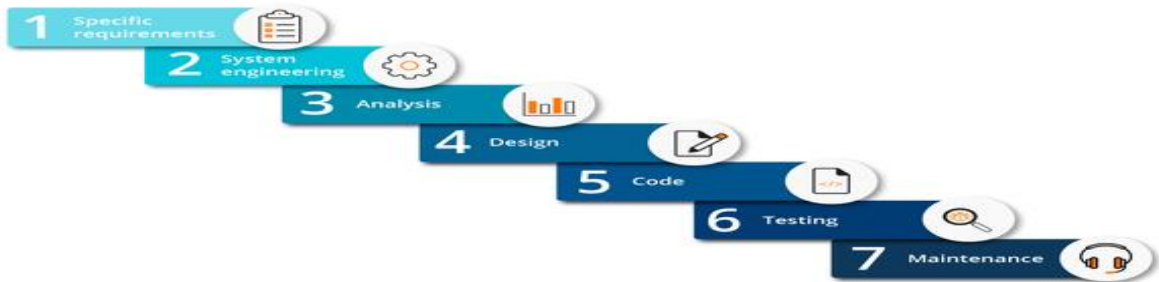
4. Çevik ve Scrum Modeli:

- Scrum, çevik geliştirme için bir çerçeve sunar ve ekip çalışması ve esneklik üzerine odaklanır.
- Proje işlevselliği küçük parçalara bölünür ve bu parçalar, belli bir zaman dilimi içinde geliştirilir.
- Müşteri sürekli olarak geri bildirimler sağlar ve gereksinimler değiştiğinde, ekip bu değişikliklere hızlı bir şekilde adapte olabilir.
- Çevik ve Scrum, esnek ve hızlı bir şekilde çalışma üzerine odaklanır, ancak proje risklerinin yönetimi ve belgelerin yönetimi konusunda yetersiz kalabilir.

Özetle, her bir modelin kendine özgü avantajları ve dezavantajları vardır. V Modeli, lineer bir yapıya sahip olması ve risklerin önceden belirlenmesi ile proje yönetiminde güvenilir bir seçenek olabilir, ancak esnekliği kısıtlayabilir. Spiral Model, risk yönetimi ve adaptifliği önemseyen bir yaklaşım sunar, ancak daha fazla kaynak gerektirebilir. Artımlı/Yinelemeli Model, müşteri geri bildirimlerini sürekli olarak dikkate alması ve esnekliği sağlaması ile avantajlıdır, ancak çok sayıda iterasyon nedeniyle zaman ve maliyet açısından yüksek olabilir. Çevik ve Scrum Modeli, hızlı bir şekilde işlevsellik sağlamak için idealdir, ancak risk yönetimi ve belgelerin yönetimi konusunda yetersiz kalabilir.

Hangi modelin kullanılacağı, projenin gereksinimlerine, risklere ve özelliklerine bağlıdır. Örneğin, küçük bir projede hız ve esneklik önemliyse, Scrum Modeli veya Artımlı/Yinelemeli Model uygun olabilir. Öte yandan, büyük ve karmaşık bir projede, risklerin önceden belirlenmesi ve belgelerin yönetimi gibi faktörler daha önemli olduğundan, V Modeli veya Spiral Model daha uygun olabilir.

Özetle, projenin gereksinimleri ve özellikleri dikkate alındığında, hangi modelin kullanılacağına karar vermek önemlidir ve bu karar, proje yönetimi için en uygun yöntemi sağlayabilir.



Approximation of the Waterfall Model's Phases