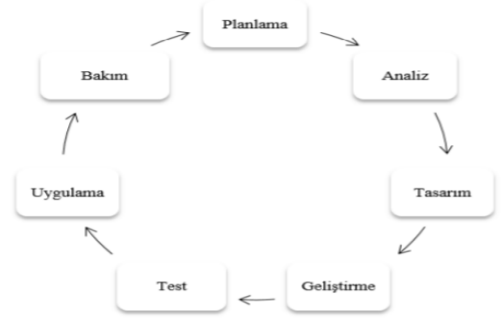


YAZILIM YAŞAM DÖNGÜSÜ VE MODELLERİ

Yazılım Yaşam Döngüsü Nedir?

Yazılım bir ürün değildir ancak ürün üretmek için kullanılır. Ürün üretmek için gerekli olan tüm basamakların içinde bulunduğu, problemi bulma sürecinden- problemi sonuca ulaştırma sürecine kadar gerekli olacak tüm adımların barındığı bir döngüdür yazılım yaşam döngüsü. Bu sürecin basamakları sırasıyla şu şekildedir;

- Planlama
- Analiz
- Tasarım
- Gerçekleştirme
- Test
- Uygulama
- Bakım



Yazılım yaşam döngüsünü neden kullanırız? Kullanmak zorunda mıyız?

Yazılım yaşam döngüsünü tabii ki kullanmak zorunda değiliz. Ancak aslında geliştirdiğimiz ya da geliştirmekte olduğumuz yazılımların çoğunda farkında olmasak bile bu döngüyü kullanıyoruz. İnsanoğlu, zamandan daha az kayıp vererek problemi çözme konusunda her geçen gün daha da ileriye gidiyor. Hayatımızda kaybedilen şeyler geri kazanılabiliyor -sağlık, para, şöhret vb.- ancak kaybedilen zaman bir daha asla geri kazanılmıyor. Biz de aslında tam bu yüzden yazılım yaşam döngüsünü kullanıyoruz. Peki yazılım yaşam döngüsünü hayatımızda nasıl kullanıyoruz? Şimdi de basamakların sırasıyla açıklamalarını yapalım.

1. PLANLAMA

Yazılıma başlama sürecinin ilk adımıdır. Bu adımda problem, gereksinim, maliyet, süre belirlenir. Çözüm için kaynak arayışına girilir. Yazılım ürününün hedef kitlesi belirlendikten sonra plan oluşturulur ve görev dağılımı yapılır.

2. ANALİZ

Planlama adımı bulanan çözümlerin doğruluğunu kontrol eder. Müşterinin isteğinin doğru anlaşılması önemlidir. Bu adımda bu kontroller yapılarak, yazılım sürecinin geri kalanında daha sağlıklı ilerlenmesi sağlanır.

3. TASARIM

Analiz adımıdan sonra, yazılımın modelinin oluşturulma sürecidir. Planlama ve analiz adımlarını iskelet oluşturmaya benzetirsek, tasarım adımı iskeletin boyanma sürecidir. UML, nesne modeli, akış şeması vb. araçlar kullanılabilir.

4. GELİŞTİRME

Bu adımda müşterinin istekleri en ön plandadır. Yazılım sürecinde, yazılım ile alakalı atılan adımlarda somut bir yapının oluşmaya başladığı adımdır.

5. TEST

Geliştirme adımı ortaya çıkan somut yazılım ürününün çalışıp, çalışmadığı kontrol edilir. Eğer çalışıyor ise doğru çalışıp çalışmadığına bakılır.

6. UYGULAMA

Yazılım ürününün müşteriye teslim edilme sürecindeki son adımıdır. Müşterinin, ürünü sıkıntısız kullanabilmesi için gerekli olan tüm bilgiler, müşteriye aktarılır. Ürün müşterinin istediği ortamda çalıştırılır ve teslimat yapılır.

7. BAKIM

İlerleyen süreçlerde müşterinin talebi doğrultusunda yazılım üstünde yapılan değişikliklerin yapılma sürecidir.

Yazılım Yaşam Döngüsü Modelleri Açıklamaları ve Kullanım Gerekçeleri

1) ŞELELE MODELİ

1970 yılında yayınlanan, yazılım yaşam döngü modellerinin ilk modelidir. Aşama aşama ilerler. Kısa süreli projelerde tercih edilir. Her aşama gözden geçirilmek zorundadır. Çünkü bir aşama atlandıktan sonra bir daha o aşamaya geri dönmek mümkün değildir.

Şelale Modeli Hangi Projelerde Kullanılır?

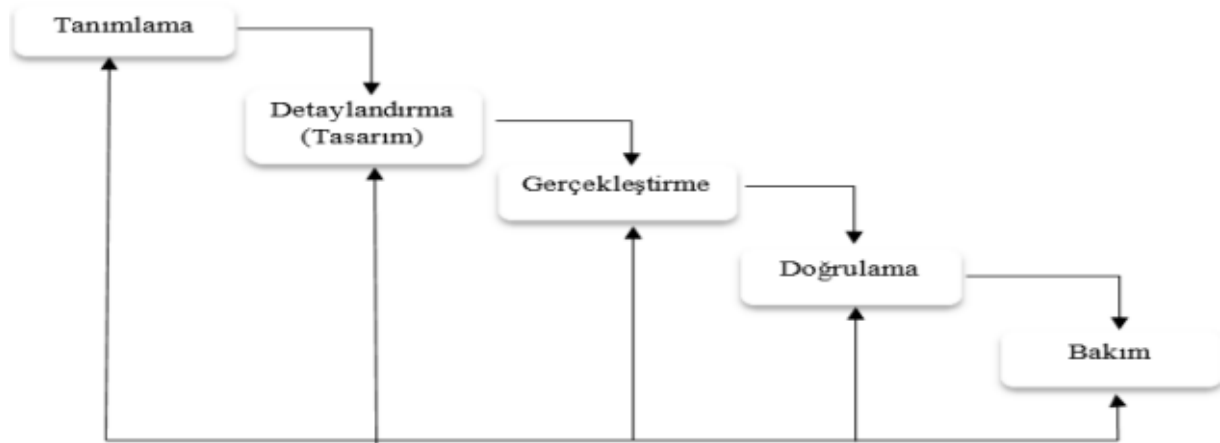
Soru işaretlerinin az, açıklamaların çok olduğu küçük projelerde tercih edilebilir.

Neden Şelale Modeli Kullanılır?

- Erken hata tespiti yapılmasına olanak sağlanır.
- Maliyet tahmini mümkün olduğundan dolayı, tahmin edilen maliyetin üstüne kolay kolay çıkmaz.
- Teknik açıdan yetersiz personellerin de kullanımına uygundur.

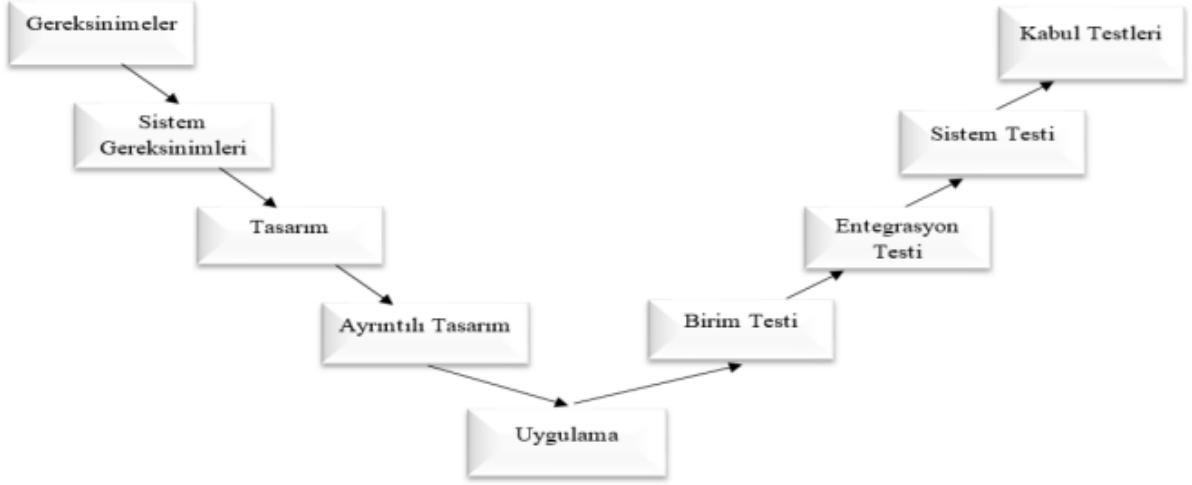
Neden Şelale Modeli Kullanılmaz?

- Adımların hiçbirinde müşteri olmadığı için, geri bildirim veremez.
- Zaman kaybı fazladır.
- Esnek değildir. Oysa yazılım ürünü esnek bir üründür.
- Adım sırasında fark edilmemiş hatalar, kolay kolay bulunamaz.



2) V MODELİ

Onaylama ve doğrulamanın olduğu model türüdür. Aşağıda bulunan görselde sol taraf üretim kısmını, sağ taraf ise test işlemlerini gösterir. Belirsizliklerin az olduğu makalelerde kullanılır. Kısa sürelidir.



V Modeli Hangi Projelerde Kullanılır?

Ürüne ait isteklerin açık ve net olduğu küçük/orta projelerde tercih edilebilir.

Neden V Modeli Kullanılır?

- Küçük projeler için iyi bir tercihtir.
- Basit, kolay ve anlaşılırdır.
- Yönetilmesi kolaydır.

Neden V Modeli Kullanılmaz?

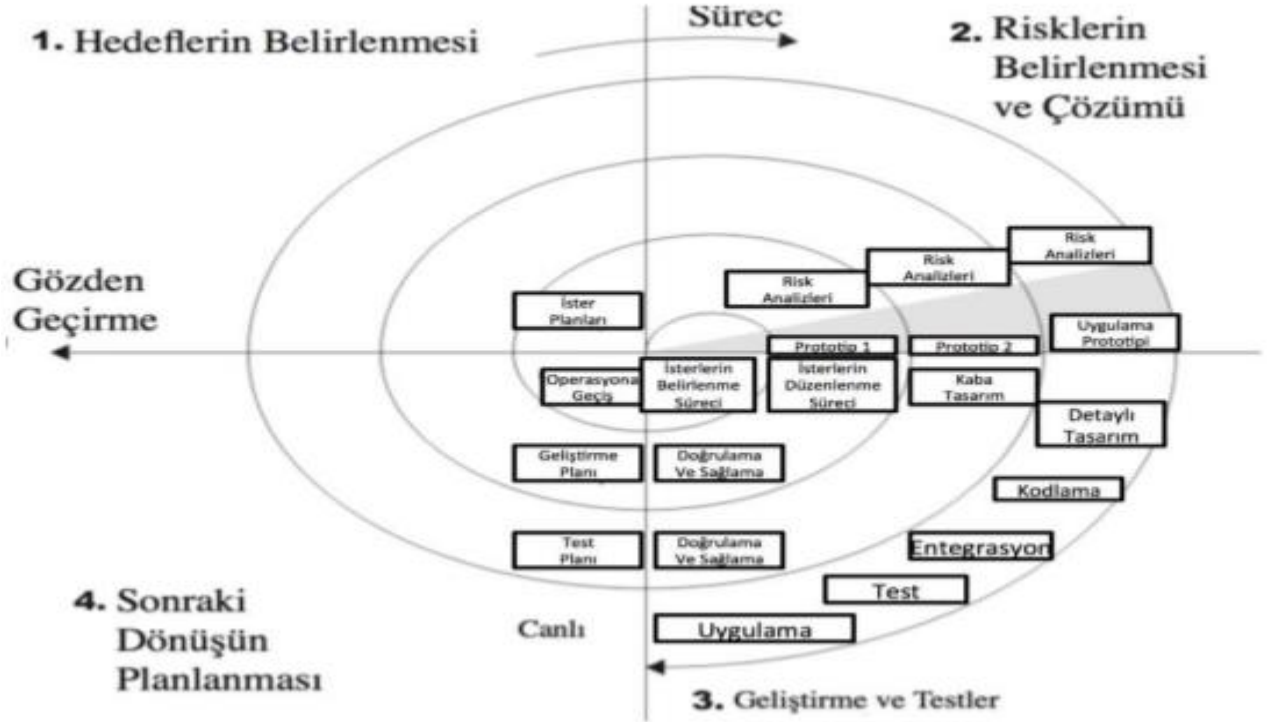
- Test aşamasında iken bulunan hatalarda geriye dönmek zordur.
- Döngü sonuna kadar, çalıştırılacak bir yazılım üretilmez.
- Uzun projeler için kötü bir tercihtir.

3) Spiral Model

Risk analizi ön plandadır. Prototipler oluşturularak ilerleme kaydedilir. Bu sayede hatalar erken giderilir. Risk değerlendirmesi; projeyi parçalara ayırarak yapılır. Bu sayede geliştirme süreci kolay değiştirilebilir. Her döngüye prototiple başlanır ve her döngü son bir kontrol ile bitirilir. Uzun projelerde kullanılır. Aşama aşama ilerleme kaydedilir.

- ✓ ANALİZ: Müşterinin isteği analiz edilir.
- ✓ TASARIM: Analiz aşamasında alınan bilgilerle taslak oluşturulur.
- ✓ 1. PROTOTİP: Müşterinin yazılım hakkında fikir sahibi olabilmesi için oluşturulur, basittir.

- ✓ 2.PROTOTİP: İlk prototipten dersler çıkarılır. Tekrardan analiz yapılır. Oluşturulan yeni tasarıma göre yazılım geliştirilir. Müşteri ile ortak hareket edilir. Boşa zaman kaybı ve fazla maliyetten kaçınılmış olunur.



Spiral Model Hangi Projelerde Kullanılır?

Uzun sürmesi beklenen, maliyetin azaltılması gereken büyük projelerde tercih edilebilir.

Neden Spiral Modeli Kullanılır?

- Müşteri odaklıdır.
- Değişirme işlemleri hızlıca yapılabilir.
- Risk yönetimi daha rahat ve kapsamlı yapılır.
- Fazla maddiyat ve gereksiz zaman kaybı önlenir.
- Uzun projeler için tercih edilebilir.

Neden Spiral Model Kullanılmaz?

- Yazılım ve süreç karmaşıktır.
- Yazılım sonu olmayabilir.
- Küçük projeler için maliyeti oldukça fazladır.

4) ARTIRIMLI MODEL

İlk önce basit ihtiyaçları karşılamakla işe başlanır. Bu şekilde ilerlenerek, süre sonunda tüm ihtiyaçları karşılayan yazılım ürünü ortaya çıkarılır. Döngüler küçük parçalar ayrılır. Her yenilenen döngü sonrasında, yazılıma yeni özellikler kazandırılır. Üretim ve kullanım eş zamanlıdır. Bu sayede kullanıcı her adımda geri bildirim yapabilir. Uzun projelerde kullanılır.



Artırmalı Model Hangi Projelerde Kullanılır?

Maliyetin düşürülmeye çalışıldığı, zamandan tasarruf etmek için uğraşılan büyük projelerde tercih edilebilir.

Neden Artırmalı Model Kullanılır?

- İlerleme ölçülebilir.
- Paralel geliştirme planlanabilir.
- Sonuçlar erken elde edilir.
- Gereksinimi değiştirmek için az maliyet gereklidir.
- Risk analizi daha iyi yapılabilir.
- Risk yönetimi daha kolaydır.

Neden Artırmalı Model Kullanılmaz?

- Kısa projeler için uygun değildir.
- Yönetim süreci karmaşıktır.
- Projenin sonu bilinmeyebilir.
- Yüksek vasıflı kaynaklar gerekebilir. ,

5) GELİŞİGÜZEL MODEL

Bu modelde tek kişilik çalışma ortamı vardır. 60'lı yıllarda bulunmuştur. Önceden belirlenen bir model yoktur. Bu sebepten dolayı da geliştiren kişiye bağlı olarak maliyet, süreç ve ilerleme aşamaları değişim gösterir.

6) BAROK MODELİ

Yazılım yaşam döngüsünün temel adımları bu modelde doğrudan kullanılır. Adımlar arası geri dönüşler belirli bir kural çerçevesinde yapılmaz. 70'li yıllarda bulunmuştur. Belgeleme süreci, yazılım geliştirme sürecinden ayrı tutulur. Bu sebeple günümüzde kullanılması uygun değildir.

7) KODLA & DÜZELT MODELİ

Direk olarak yazılım ürünü gerçekleştirmeye çalışılır. Genelde öğrencilerin kullandığı bu yöntemde istenilen yazılım ürünü gerçekleştirilene kadar geliştirme devam eder. Bakımı vardır ancak dokümantasyon olmadığı için çok zordur. Bakım sürecinden sonra emeklilik safhası da vardır.

Kodla & Düzelt Hangi Projelerde Kullanılır?

Genellikle öğrencilerin, verilen projelerde kullanıldığı bu model, küçük projelerde tercih edilebilir.

Neden Kodla & Düzelt Modeli Kullanılır?

- Yazılım ürünü gerçekleştirmek için kullanılan en kolay yoldur.

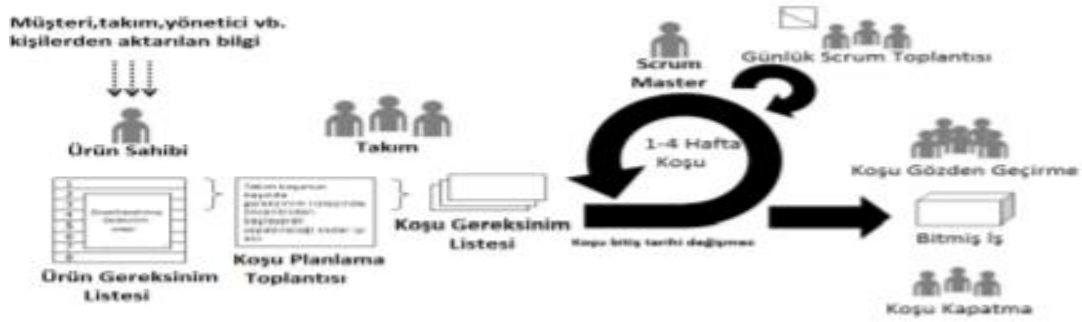
- Fazla tecrübe sahibi olmayan işletmelerde kullanılır.

Neden Kodla & Düzelt Modeli Kullanılmaz?

- Herhangi bir dokümantasyon olmadığı için bakım evresi çok zordur.
- Herhangi bir ürün değişiminde, çok masraflı hale gelebilir.

SCRUM

Scrum, ismini rugby sporunda kullanılan bir taktikten almıştır. 1990'larda bulunan bu model, karmaşık projeleri parçalara ayırır. Her bir parçaya sprint adı verilir. Her sprint ayrı ayrı geliştirilir. Ekip iletişiminin çok önemli olduğu SCRUM modellemesinde günlük olarak toplantılar düzenlenir. Ve bu toplantılara SCRUM Meeting adı verilir. Scrum modellemesi, sadece yazılım ürün geliştirmesinde değil, her alanda kullanılabilir. Üç temel kavramdan oluşur.



I. Roller

a. Uzman

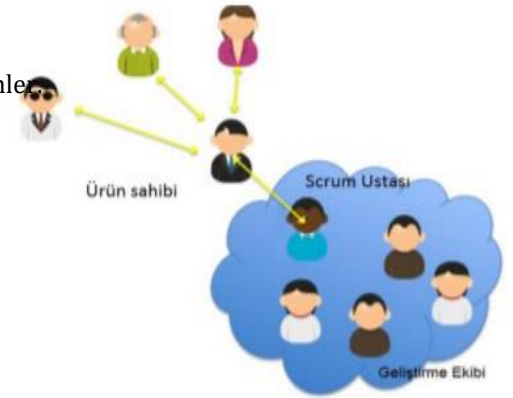
- Sürecin işlenmesini yönetir.
- Yüksek performanslı ekip kurmaya çalışır.
- Sorunları ortadan kaldırır.
- Ekibin dikkatini toplar, günlük toplantılar düzenler.
- İş takibi yapar.

b. Ürün Sahibi

- Ürünü tanımlar.
- Teslim tarihini belirler.
- Ürünün sonucundan sorumludur.
- Ürün özelliklerinde değiştirme hakkı vardır.

c. Geliştirme Ekibi

- Süre tahmini yaparlar.
- 5-9 kişiden oluşur.
- Hedef belirleyip, sprinte öyle başlarlar.
- Kendi kendilerini organize ederler.
- Sonuçları ürün sahibine sunarlar ve geri dönüş alırlar.



II. Toplantılar

a. Günlük Scrum Toplantıları

- Her gün yapılır.
- Katılım zorunludur.
- 3 soru yanıtlanmalıdır.
 - Dün ne yaptın?
 - Bugün ne yapacaksın?
 - İşleyişte herhangi bir sorun var mı?
- Toplantı süresi 15 dakika ile sınırlıdır.

b. Sprint Planlama Toplantıları

- i. Ekip üyeleri ve ürün sahibi katılmalıdır.
- ii. Toplantıdaki hedefler;
 - 1. Sprintin amacını belirlemek
 - a. 1-2 cümle içerebilir.
 - b. Açık ve sade olmalıdır.
 - 2. Sprint iş listesi hazırlamak
 - a. Ekibin söz verdiği işlerin ve bu işleri tamamlamak için gerekli görevlerin bulunduğu listedir.
- iii. Toplantı 2 saat sürmelidir.
- c. **Sprint Gözden Geçirme Toplantıları**
 - i. Önceki sprint hakkında tartışmalar yapılır.
 - ii. Sonraki sprinti daha verimli hale getirmek için gerekli görülenler listelenir.
 - iii. Başla, dur, devam et yaklaşımı ile çalışılır.
 - 1. **BAŞLA**= Ekip, nasıl üretken ve motive edilmiş şekilde işe başlar?
 - 2. **DUR**= Gereksiz işlerin yapılmaması, zaman kaybedilmemesi
 - 3. **DEVAM ET**=Ekibin motive, üretken şekilde işe devam etmesidir.

III. ARAÇLAR & BİLEŞENLER

- a. **Ürün Gereksinim Dokümanı**
 - i. Ürün için gerekli olan özellikler listesidir.
 - ii. Ürün sahibi liste sıralamasını değiştirebilir.
 - iii. Ürün geliştikçe, liste de gelişir.
 - iv. Tamamlanma süreci yoktur.
- b. **Sprint Dokümanı**
 - i. Ürün gereksiniminden elde edilen bilgilerden oluşur.
 - ii. Günlük scrum toplantılarında ilerlemenin anlaşılabilmesi için yeterli ayrıntıyı içeren plandır.
 - iii. Geliştirme ekibi tarafından değiştirilebilir. Değişime açıktır.
- c. **Sprint Kalan Zaman Grafiği**
 - i. Tamamlanan ve kalan görevlerin takip edilmesi için gereklidir.
 - ii. Günlük olarak güncellenmelidir.
 - iii. Ürüne ait istenen isteklerin, belirlenen süre içerisinde gerçekleşip gerçekleşmeyeceğine ait bilgiler bu grafik sayesinde daha net bir şekilde görülebilir.

SCRUM günümüzde neden popüler konumda?

- Müşteri direkt olarak geliştirme sürecine dahil ediliyor.
- Müşteri isteklerini, anlık olarak iletebiliyor ve bu sayede kolaylıkla değişim yapılabiliyor.
- Ekip içi iletişim, üst düzeyde tutulduğu için anlaşmazlıklar minimuma indiriliyor ve ortaya çıkan sorunlar kısa sürede çözülebiliyor.
- Büyük ve karışık projelerin tamamlanma süreçleri, scrum yöntemleri uygulanarak daha hızlı tamamlanabiliyor.

KAYNAKÇA

- Kamu Kurumlarında Yazılım Yaşam Döngülerinin Uygulanabilirliği Konulu Uzmanlı Tez Yazısı-SALİH SOYLU
 - https://webdosya.csb.gov.tr/db/cbs/icerikler/salihsoylu_tez_v10-20180925134450.pdf
- Agile ve Scrum Açıklama Yazıları-GÖKÇE DEMİR
 - <https://www.mshowto.org/agile-ve-scrum-nedir.html>
- Scrum Tanımlama Yazıları-CANER BULUT
 - <https://www.argenova.com.tr/scrum-nedir>
- Scrum Neden Popüler Konulu Yabancı Kaynaklar
 - <https://www.quora.com/Why-is-the-Scrum-process-so-popular-in-the-software-industry>
 - <https://www.techwell.com/2013/02/why-scrum-so-popular>
- Medium Ödev Yazıları-BERFİN KAÇAR
 - <https://medium.com/@brfn.kcr26/yazilim-gelistirme-ve-surec-modelleri-2131ea5f09b2>
- Medium Ödev Yazıları-ÖMER HARUN ÇETİN
 - <https://medium.com/@omerharuncetin/yazilim-yaşam-döngü-modelleri-543c7879a742>
- Medium Araştırma Yazıları-DENİZ KILINÇ
 - <https://medium.com/@denizkilinc/yazilim-yaşam-döngüsü-temel-aşamaları-software-development-life-cycle-core-processes-197a4b503696>
- Konu Hakkında İnternet Yazıları
 - <https://fikirjeneratoru.com/yazilim-proje-yonetimi-yontemleri/>
 - <https://furkanalniak.com/yazilim-muhendisligi-yazilim-surec-modelleri/>
 - <https://www.codex.com.tr/yazilim-gelistirme-modelleri>