

İçerik Özeti:

Proje yapmak ve proje yaptırmak, dinamiklerin hızlı geliştiği, devrimin giderek yükseldiği, yazılım teknolojilerine olan yatırımlarının artmaya başladığı bu dönemde, işlerinizi hızlıca yön verebilecek, esnek planlama yapabilecek şekilde yazılım ürünü nasıl geliştirilir konusunda bilgi sahibi olunması gerekmektedir.

Eğer yazılım uygulamaları geliştiriyorsanız çevik yazılımlar ile ilgili uygulamalı bir eğitim almanız iş hayatınıza önemli katkılar sağlayacaktır.

Bu eğitimde çevik yazılım gruplarının nasıl çalıştıklarını ve teknik olarak bu süreçleri işletmek için gerekli olan bilgilere sahip olacaksınız.

SCRUM karmaşık ve geniş kapsamlı ürün gereksinimleri olan, kapsamında değişiklik olacağı düşünülen ve dolayısıyla belirsizlik ve belli bir miktar risk içeren projelerde sıkça tercih edilen bir yazılım geliştirme modelidir.

Bu yöntem yazılım geliştiricilere rahatlıkla kullanabilecekleri, ufak bir takım ile bile rahatça uygulayabilecek bir üretim imkânı sağlayacaktır.

Yazılım geliştirme dünyasında 2000 yılından sonra aktif olarak kullanılmaya başlayan ve başarılı sonuçları ile dikkatleri üzerine çeken SCRUM; Google, Microsoft, Huawei ve Apple gibi dünya devleri tarafından benimsenmiş bir metodolojidir. Bu eğitim, İstanbul'da büyük şirketlerle birlikte zaman içinde küçük veya orta ölçekli şirketler ve hatta kamu kurumları tarafından da tercih edilen bir yazılım ve sistem mühendisliği eğitimidir.

Eğitmen, 13 yıllık yazılım geliştirme tecrübesine ve yazılım mühendisliği konusunda doktora ve yüksek lisans derecelerine sahip, 2000 yılından bu yana birçok projede çevik metodolojilerini kullanan, Huawei Ar-Ge biriminin çevik dönüşüm danışmanı olan, konu üzerinde yetkin bir kişidir.

Bu eğitim sayesinde, eğitmenin pratik deneyiminden de faydalanabilecek, gerçek sorunlarla defalarca karşılaşmış bir uygulayıcı tarafından bilgilendirilmiş olacaksınız.

- Agile Metodolojiler ile ilgili bilgilendirme (XP, SCRUM, KANBAN),
- Agile Disiplini, Agile Uygulama Gereklilikleri,
- SCRUM Metodolojisi ile ürün Yönetimi,
- SCRUM'ın rolleri ve sorumlulukları, ritüeller, kurallar, kontrol çalışmaları ve iyileştirmeler

hakkında bilgi sahibi olunması.

- Çevik Metotlar
- Çevik Manifesto
- Extreme Programming (Uç Programlama)
- Yazılım Geliştirme ile ilgili pratikler hakkında pratik bilgiler
- Jira kullanarak SCRUM uygulaması
- Balsamiq aracı ile hızlı prototipleme

En yaygın AGILE yöntemler:

Extreme Programming

Scrum

Feature Driven Development

Lean Software Development

Kanban

Incremental Development Strategy

Çevik Manifestolar (olmazsa olmaz):

Süreçlerden ve araçlardan ziyade bireyler ve etkileşime

Kapsamlı dökümantasyondan ziyade çalışan yazılıma

Sözleşme pazarlıklarından ziyade müşteri iş birliğine

Plana bağlı kalmaktan ziyade değişime karşılık vermeye önem verilmelidir.

İlkeler:

Erken ve devamlı teslim sağlanması

Değişen gereksinimlerin kabulü

Rekabet avantajı sunulması

Çalışan yazılım üretilmesi

Sadelik

XP - Extreme Programming (Uç Programlama)

Dökümantasyonun bir kısmı iletişim aktiviteleri ile değiştirilir.

Kaynak kod ve test odaklı.

Müşteri gereksinimleri çok sık değişir.

Yazılım ürünleri daha kolay değiştirilebilir olmalı.

İletişim-Geri besleme-Basitlik-Cesaret

Life Circle

Klasik yönteme kıyasla değişim maliyeti

12 Pratik:

Planlama oyunu - kısa aralıklı sürümler - basit tasarım - test etme – refactoring - eşli programlama -kolektif mülkiyet - sürekli entegrasyon - müşteri ile birlikte üretim - sürdürülebilir hız – metafor -kodlama standardı

Çeviklik, değişime hızlı tepki verme yeteneğidir. Çevik yazılım geliştirmek aslında buradaki on iki pratiği ve manifestoyu uygulamaktır.

Kısıtlar

Müşteri gelişme ortamında olabilir mi?

Ekip üyeleri aynı ortamda olabilir mi? Uzaktan çalışma uygun olmayabilir.

Eşit bilgi düzeyinde takım elemanları bulmak ne kadar mümkün?

Timeboxing çalışma yöntemi.

Storyboard (JIRA + Gerçek dünyada)

Pair programming (Navigator ve Driver)

SCRUM

En yüksek iş değerini en kısa zamanda sunmaya odaklanmamızı sağlayan çevik bir süreç.

Sprint Iteration

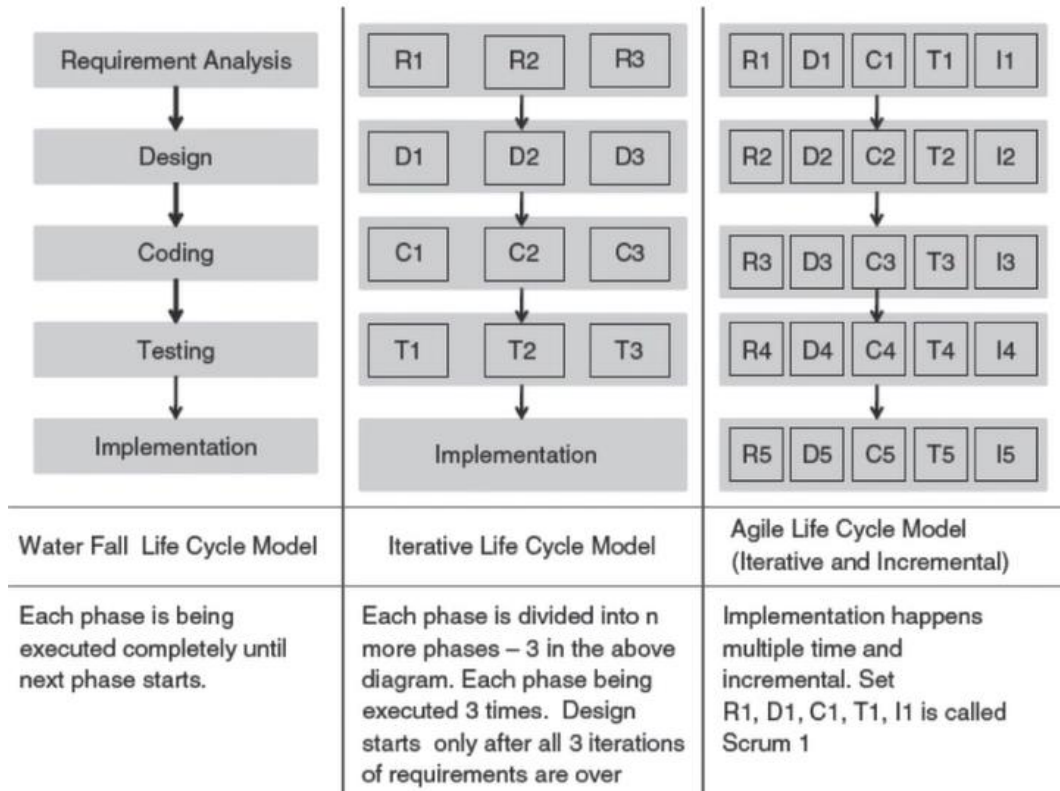
Scrum dezavantajı yazılı dökümanın azlığı (ancak çalışan kodun varlığı söz konusudur avantaj olarak) ve sorumluluk ihlali olasılığıdır.

Kritik projelerde Agile kullanılması dökümantasyon problemi açısından uygun olmayabilir.

Kişi sayısı arttıkça Agile kullanımı zorlaşır.

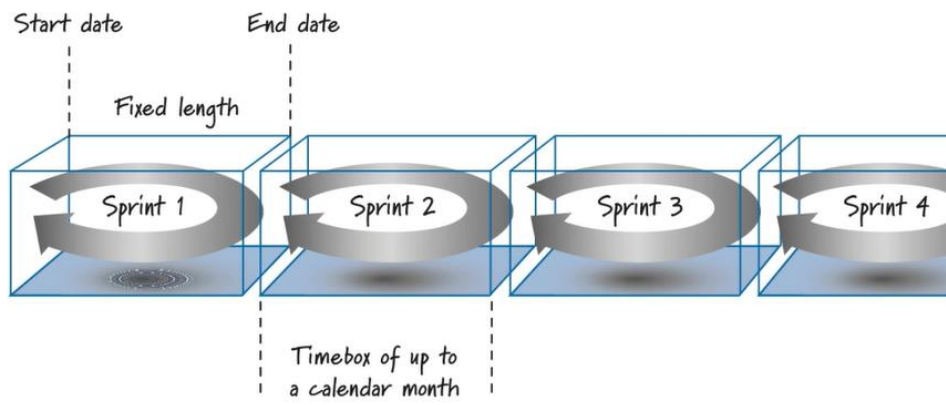
Sürekli değişmeyen gereksinimlere sahip çalışmalarda Agile kullanılmasına gerek duyulmaz.

Waterfall vs Iterative vs Agile



Agile geliřtirmede her iterasyon sonucunda alıřan bir řey elde edilmelidir.

Scrum



Product Owner

Scrum Master

Scrum Takımı

Daily Scrum (Ayakta 15 dk)