



ÇEVİK



TERİMLER



SÖZLÜĞÜ



IT GOVERNANCE TURKEY



**IT GOVERNANCE
TURKEY**

A

Acceptance Test Driven Development - Kabul Test Odaklı Geliştirme (ATDD)

Kabul Testi Odaklı Geliştirme (ATDD), kod yazmadan önce kabul testlerinin varlığından emin olmak için kullanıcı, geliştirici ve testçi arasında iş birliğini teşvik etmeyi amaçlamaktadır. Kabul Testi kullanıcının bakış açısından yazılır ve yazılımın nasıl çalışması gerektiğine dair bir gereksinim olarak işlev görür. “Kod yapılması isteneni yerine getiriyor mu?” sorusu ATDD’nin ardındaki ana sorudur.

Acceptance Testing - Kabul Testi

Son kullanıcıya dayanan test işlemidir. Kullanıcıların, uygulamayı “kabul” etmeden önce, söz konusu uygulamanın gereksinimlerini ne ölçüde karşılayıp karşılamadığını belirleyip, özetle isteği olmuşmu sorusunun cevabının alındığı süreçtir.

Antipattern

Kabul Testi Odaklı Geliştirme (ATDD), kod yazmadan önce kabul testlerinin varlığından emin olmak için kullanıcı, geliştirici ve testçi arasında iş birliğini teşvik etmeyi amaçlamaktadır. Kabul Testi kullanıcının bakış açısından yazılır ve yazılımın nasıl çalışması gerektiğine dair bir gereksinim olarak işlev görür. “Kod yapılması isteneni yerine getiriyor mu?” sorusu ATDD’nin ardındaki ana sorudur.

B

Backlog Grooming / Backlog Refinement - Ürün İş Listesi’nin Detaylandırılması, İşlerin Olgunlaştırılması

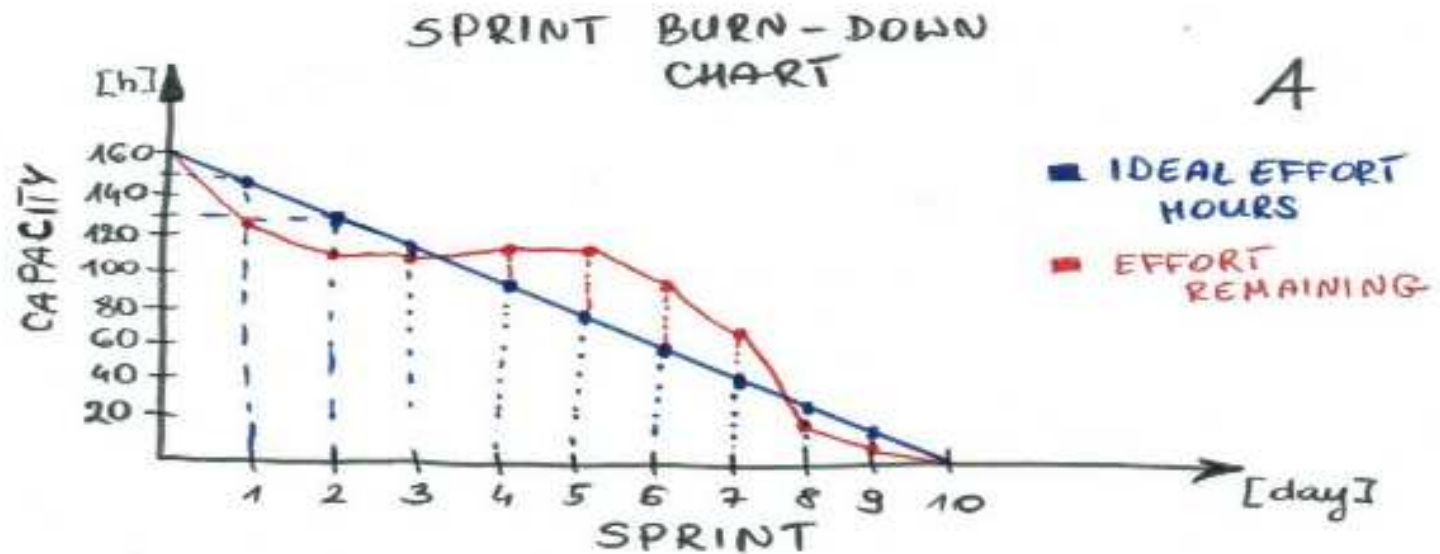
Product Backlog Refinement, Product Owner tarafından belirlenen User Story’lerin, Sprint’lere sığabilecek küçüklükte parçalanması, story point’lerinin veya eforlarının belirlenmesi işlemidir. Bu, geliştirme boyunca sürekli devam eden bir süreçtir. Scrum, bir Sprint boyunca, Development Team’in bu aktiviteye ayırdığı zamanın, sprint uzunluğunun %10’unu geçmemesinin verimli olduğunu söyler.

Behavior Driven Development (BDD) - Davranış Odaklı Geliştirme

Yazılım süreçlerinin daha test odaklı gitmesini sağlayan bir yaklaşımdır. BDD, Test Driven Development (TDD) gibi prensip olarak öncelikle test kodlarının yazılmasını daha sonrasında proje kodunun yazılmasını teşvik etmektedir. Behaviour Driven Development'in (BDD) Test Driver Development'dan (TDD) farkı, bir sistemin davranışlarını günlük konuşma dili şeklinde yazılmasıdır.

Burndown Chart - Burndown Grafiği

Sprint Burndown Chart (Sprint Burndown Grafiği), Sprint hedefine ulaşırken ne durumdayız, nasıl gidiyoruz sorularına en iyi cevap veren görsel araçtır.



Business Agility - İş Çevikliği

Değişen iş ortamında etkili bir şekilde performans gösterebilmek için kurum stratejisini, organizasyonunu ve operasyonlarını sürekli olarak gözden geçirip adapte edebilme yetkinliği ve esnekliğidir.

C

Continuous Delivery - Sürekli Teslimat

Başarılı olan bir yapıyı (build) bir ortama atmanın otomatik bir yoludur. Burada teslimat ile dağıtım arasında küçük bir nüans vardır. Teslimat manual olarak, dağıtım ise otomatik olarak yapılmaktadır. Sürekli teslimat düzgün bir şekilde uygulandığında, müşteriler her zaman standartlaştırılmış bir test sürecinden geçmiş hazır yapıya sahip olacaklardır.

Continuous Deployment - Sürekli Dağıtım

Sürekli dağıtım, bir kod satırı yazma ve bu kodu üretimdeki kullanıcılar için kullanılabilir hale getirme arasında geçen süreyi azaltmayı amaçlamaktadır. Ekip, çeşitli adımları otomatikleştiren ve uygulayan altyapıya güvenir, böylece her bir entegrasyon bu sürüm kriterlerini başarıyla karşıladıktan sonra canlı uygulama yeni kodla güncellenmektedir.

Continuous Integration - Sürekli Entegrasyon

Bu aşama, tüm DevOps yaşam döngüsünün kalbidir. Sürekli entegrasyon süreci geliştirme sonrasında otomatik olarak başlar. Bir sonraki aşamada yapılacak testlerin planlanması, ilk proje belgelerinde ihtiyaç duyulan sonucu üretme kodunun anlaşılması gibi birkaç adım içermektedir.

D

Daily Meeting - Günlük Scrum Toplantıları

Geliştirme Takımının faaliyetleri hakkında takım üyelerinin birbirlerine bilgi verdiği ve takımın önündeki 24 saat için bir plan oluşturduğu, 15 dakikayla sınırlı bir etkinliktir. 3 soruya cevap verilir; 1-Dün ne yaptım? 2- Bugün ne yapacağım ? 3- İşimi yapmaya engel bir durum var mı?

Engeller toplantılarda çözülmez. Örneğin; “Bugün şu işi yapacağım, bunu yapmam için, bir başka işin tamamlanmasına ihtiyacım var.” Bu toplantılar iletişimi iyileştirir, başka toplantılara olan ihtiyacı ortadan kaldırır, geliştirmenin önündeki engellerin tespit edilmesini sağlar, hızlı karar almayı teşvik eder ve Geliştirme Takımının bilgi seviyesini artırır. Bu etkinlik kilit bir gözlem ve adaptasyon toplantısıdır. Scrum Master Geliştirme Takımının toplantıyı yapmasını teşvik eder fakat Günlük Scrum’ı yürütmek Geliştirme Takımının sorumluluğudur. Scrum Master, Geliştirme Takımına Günlük Scrumı 15 dakikayla sınırlı tutmasını öğretir.



Definition of Done (DoD) - Teslim Tanımı

Product Backlog Item'ların (PBI) istenen kalitede teslim edilmesi için belirlenen aktiviteler listesi ve takım tarafından bir işin tamamlanmış olarak kabul edilmesi için yapılması gereken adımlardır. Scrum içerisinde Sprint'lerdeki hedef çalışır durumda canlıya atılmaya hazır yazılım teslimatları üretmektir. Bu anlamda DoD, canlıya atılmaya hazır bitmiş yazılımı üretmek için yapılması gereken standart aksiyonları belirlemelidir.

Definition of Ready (DoR) - Hazır Tanımı

İster/Talebin sprint planına alınması için gereken asgari tanımlardır. Takımın maddeyi sprint planına almak için yeterli olgunlukta görmesi gerekmektedir. Ayrıca, Sprint içerisinde maddenin "Bitti" durumuna gelmesinin riske atılmaması için de "Hazır" tanımının da belirlenmesi önemli bir adımdır.

DevOps

Bilgi Teknolojileri departmanı içerisinde bulunan iki temel birimi (Developers and Operations) Geliştiriciler(Yazılım Geliştiriciler, Yazılım Testçileri, vb.), Operasyon (Sistem Mimari ve Altyapı Ekipleri,Güvenlik ve Ağ ekipleri vb.) bir arada etkili bir iletişim içerisinde beraber çalışmalarını.DevOps'u aslında bir felsefe, yaklaşım veya bakış açısı olarak değerlendirebiliriz.Yazılım geliştiricilerin alışık olduğu Scrum, Agile, Kanban ve diğer yöntemler gibidir.

E

Epic - Epik

Bir sprint'e sığmayacak kadar büyük bir iş yüküdür. Epikler sprintlere dahil edilmezler. Ürün sahibi (Product Owner) tarafından belirlenmektedir. Epikler genellikle ilk ürün yol haritası sırasında tanımlanır ve konu anlaşıldıkça kullanıcı hikayelerine ayrıştırılır. Bir epik genel olarak en az iki sprint de tamamlanabilecek büyüklükte bir iş puanlamasına sahiptir.



Estimation - Tahmin

Belirli bir geliştirme görevini yerine getirmek için gereken çabanın değerlendirilmesidir; bu çoğunlukla süre cinsinden ifade edilir.

Exploratory Testing - Keşif Testi

İsminden de anlaşılacağı gibi araştırmacı test ya da keşif testi olarak bilinen bu teknik, black box test teknikleri grubunda yer almaktadır. Kod ile ilgili herhangi bir test içerisine sahip olmadan, ürünün girdileri ile çıktıları arasında bulunan ilişkiyi test etme eğilimindedir.

Extreme Programming - Uç Geliştirme

Yazılım geliştirme süreci boyunca son derece kaliteli olmak koşuluyla çalıştırılabilir kod üretmeye odaklanmış bir yazılım geliştirme metodolojisidir. Yazılım geliştirme sürecinin en temel, en önemli ve final çıktısı ya da ürünü çalıştırılabilir kod olduğundan, XP metodolojisi sürecin en başından itibaren çalıştırılabilir kodu sürecin merkezinde tutmaktadır.

F

Facilitation - Fasilitasyon

Belli bir grup insanı katılım (participation), konuyu sahiplenme (ownership) ve fikir üretmelerini sağlayacak süreçleri oluşturmak ve bu süreçlerin uygulanmasını sağlamak olarak tanımlanmaktadır.

G

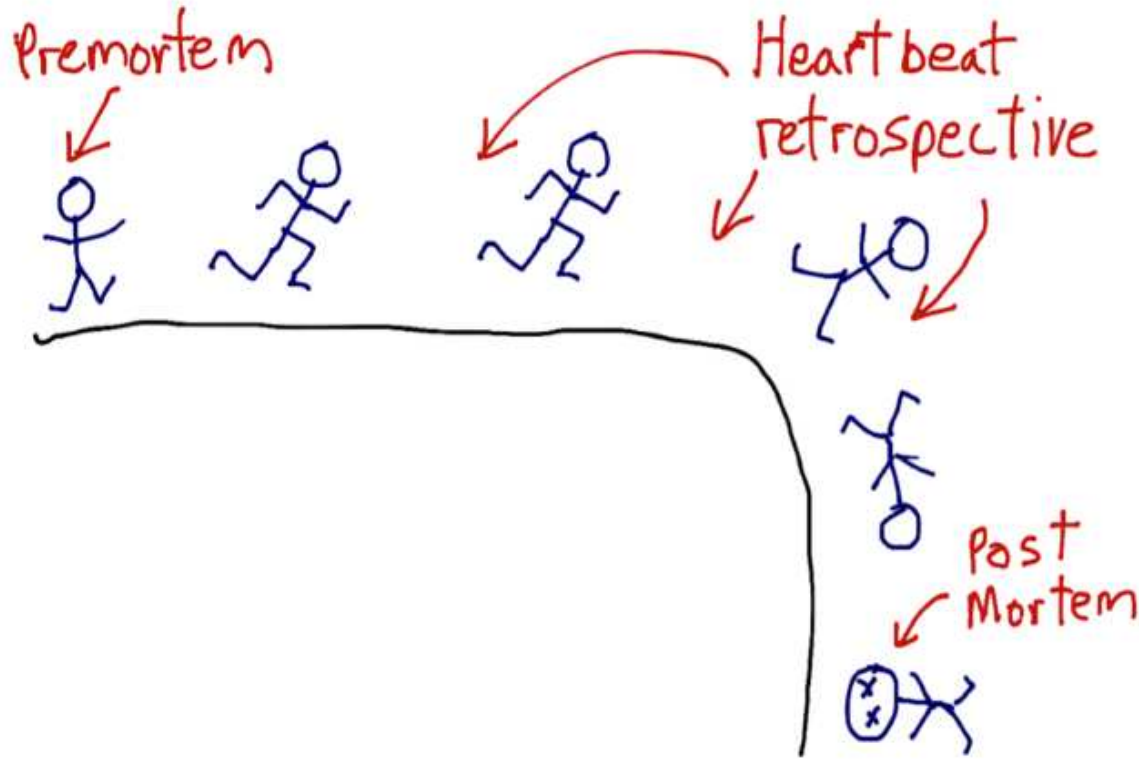
Given When Then

Girdi-işlem-çıkıtı (input-process-output) sürecine dayandırılmış test odaklı geliştirme (Test-Driven Development) şeklindedir. Test senaryoları yazılımın geliştirilmesi süresince koşturarak, programın daha az hata ile geliştirilmesi mantığını esas almıştır. Bu sayede, işlemin öncesindeki ve sonrasındaki durumlar net bir şekilde birbirinden ayrılarak tanımlanır ve müşteri isterinin ürün ile eşleşme durumunun kontrol maddeleri çıkartılmış olur.

H

Heartbeat Retrospective

Ekibin, bir önceki toplantıdan bu yana meydana gelen en önemli olayları yansıtmak ve iyileştirme fırsatlarını belirlemek için düzenli olarak toplanmasıdır.

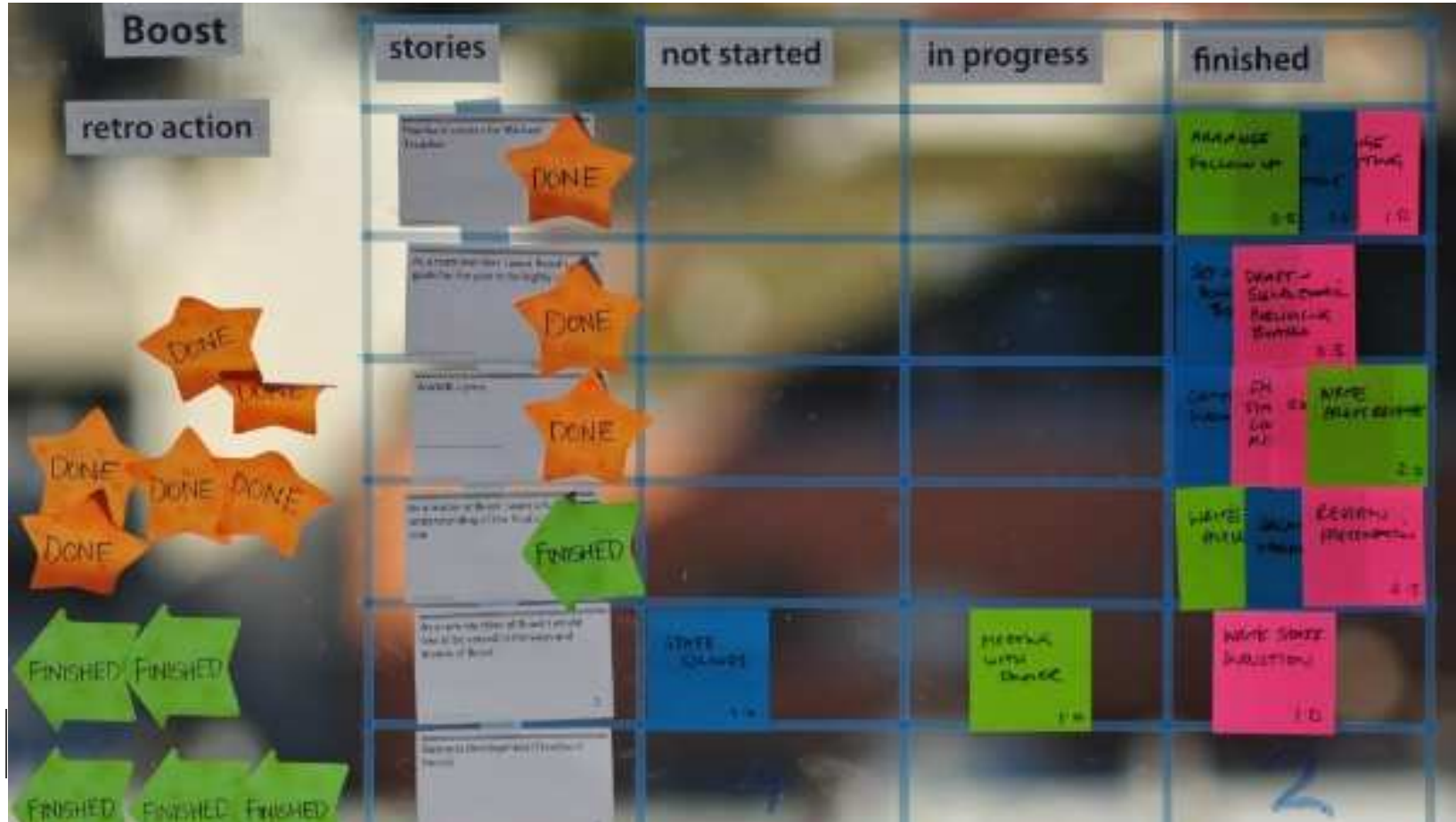


Incremental Development - Artımlı Geliştirme

Bir işlemin artan yönü, mevcut bir ürüne veya geliştirilmekte olan ürüne yeni özellikler ve diyagramlar eklemeyi mümkün kılar. Böylece yeni gereksinimler keşfedilir ve artan şekilde modellenir. Sürecin bu artımlı yönü, sistemin ek bölümlerinin geliştirilmesine devam etmeden önce bir sistemin parçalarının mümkün olduğunca eksiksiz bir şekilde oluşturulmasını sağlar.

Information Radiators - Bilgi Yayıncıları

Bir ekibin görünür bir yere yerleştirdiği bir dizi görsellerdir, böylece tüm ekip üyeleri en son bilgileri bir bakışta görebilir.



Integration - Entegrasyon

Bir proje ekibinin piyasaya sürülmeye uygun bir ürün sunması için gereken tüm çabaları ifade eder.

Increments - Çalışır Durumdaki Çözümün Son Hali

Her sprint sonunda bir ürün veya çalışabilir bir yazılım parçası çıkarılmalıdır. Scrum'da, müşteri için anlamlı ve fonksiyonel olan bu yazılıma "Increment" denir. Increment'in temel özelliği potansiyel olarak canlıya alınabilir (Potentially Releasable) olmasıdır, canlıya alınması gerekmez, amaç sadece müşteriye gerçek bir deneyim sunmaktır.

Initiative - Ortak Hedefe Doğru Giden Epik Koleksiyonu

Bir veya daha fazla değer akışını kapsayan büyük bir stratejik taleptir ve birden fazla ekip, ekip ve / veya uzmanın teslim etmesini gerektirir.

INVEST - Bağımsız, Tartışılabilir, Değerli, Tahmin Edilebilir, Küçük, Test Edilebilir

INVEST kısaltması, bir kullanıcı öyküsünün kalitesini değerlendirmek için kullanılan bir dizi ölçüt anlamına gelir. Hikaye bu kriterlerden birini karşılayamazsa, takım hikayeyi yeniden değerlendirmek isteyebilir.

I.N.V.E.S.T;

Independent (bağımsız): Bağımlılıklar azdır. Bu da planlamayı kolaylaştırır.

Negotiable (tartışılabilir): Üzerinde tartışılabilir. Böylece detaylandırılabilir.

Valuable (değerli): Müşteriye bir değer sağlar.

Estimatable (tahmin edilebilir) : Çok büyük veya anlaşılmaz olursa efor tahmini yapılamaz.

Small (küçük) : Bir haftadan daha kısa sürede takım tarafından gerçekleştirilebilir.

Testable (test edilebilir) : Kabul edilme kriteridir.

Iteration - İterasyon

Yapılmak istenen projeyi, küçük ve çalışan parçaları bir araya getirerek tamamlama işlemidir. Her bir "iterasyon" farklı mini projeler olarak düşünülebilir.

Iterative Development – Tekrarlı Geliştirme

Proje aşamalara bölünmez. Çalışabilecek, çok kısıtlı sayıda özelliği bulunan bir sürümü en kısa zamanda üretilir. Daha sonra ortaya çıkan hatalar, bazı eksiklikler ve bozukluklar giderilir. Daha da sonra bir önceki adımdan dolayı ortaya çıkan hatalar giderilir ve sırası gelen başka özellikler eklenir. Her yinelemede projeye katkısı bulunan bütün kişiler işlerini tümleştirirler. Böylelikle "Tümleştirme Cehennemi" (Integration Hell) denilen durumla karşılaşmaz.

K

Kanban

Kanban, iş yerinizdeki problemleri ortaya çıkarmaya ve Lean (yalın) yaklaşımı ile onları çözümlenize yardımcı olan bir metottur. Kanban 'ın Temel İlkeleri :

- Mevcut sistemi iyileştirme
- Değişimi kabul etme
- Rollere sadık kalma
- Liderliği motive etme

Kanban Board

Birden çok sütundan oluşan görsel bir iş akışı aracıdır. Her sütun, iş akışı sürecindeki farklı bir aşamayı temsil eder.



L

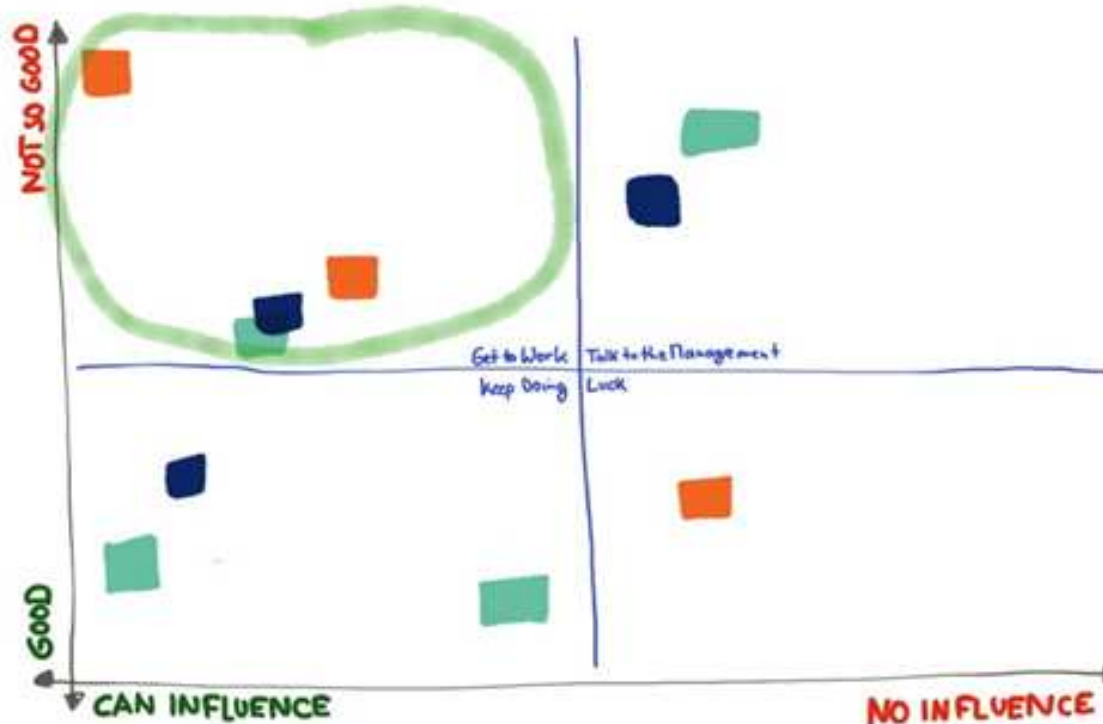
Lead Time

Lead Time bir talep geldiği andan itibaren o talep gerçekleştirilip teslim edene kadar geçen zamanı temsil etmektedir. Cycle Time'den farkı; Cycle Time o işin yapılmaya başlandığı andan itibaren bitene kadar geçen zamana karşılık gelmektedir. Diğer bir deyişle talebin geldiği andan itibaren Lead Time süresi çalışmaya başlar. Sonra belli bir süre geçer ve o iş yapılmaya başlanır. Bu andan itibaren de Cycle Time başlamış sayılır ve Cycle Time o iş üzerinde ne kadarlık bir süre çalışıldığını ölçmüş olur. O iş tamamlandığı an ise Cycle Time ve Lead Time tamamlanmış olur.

M

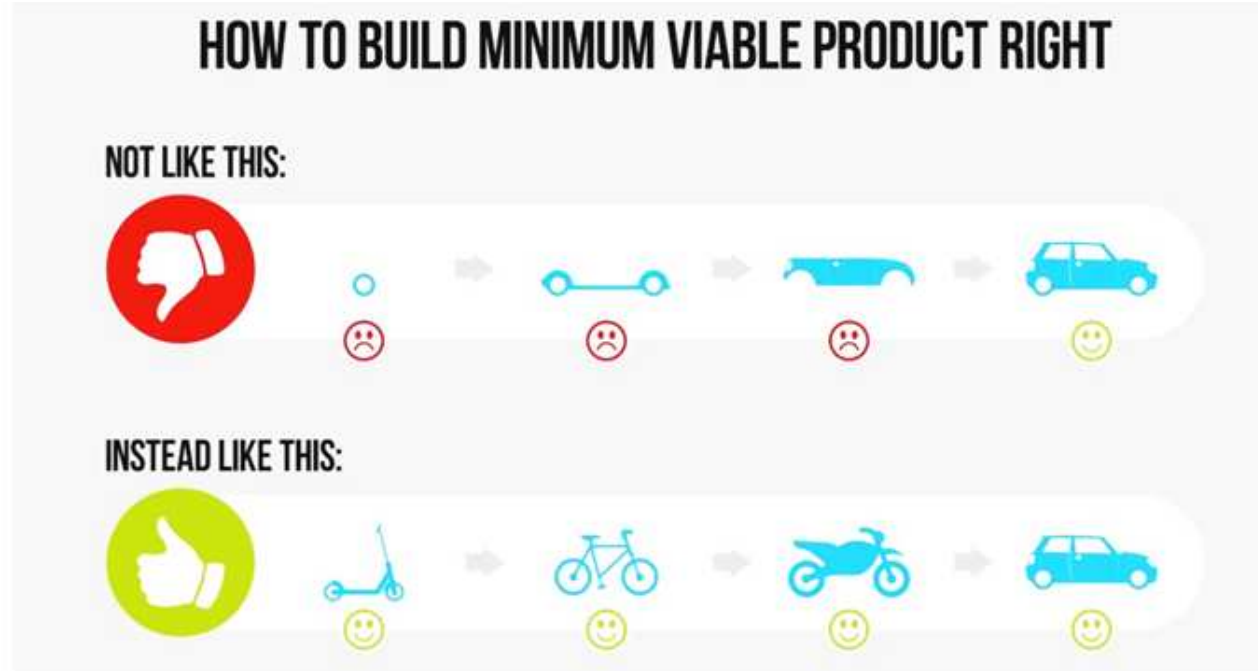
Milestone Retrospective

Bir ekibin belirli bir süre sonra veya projenin sonunda projenin önemli olaylarının ayrıntılı analizidir.



Minimum Viable Product (MVP)

Ana ürünün hızlıca hedef kitesinden feedback alabilmek ve ürüne pazarda bir ihtiyaç olup olmadığını anlamak için minimum efor ve fonksiyon setiyle ürünün hayata geçirildiği halidir. Ürünün odaklandığı alanda hedef kitlesine değer sunabilen en ufak yapıdır.



Mock Objects

Yazılan kodların test edildiği zamanlarda oluşturulan ya da test yapanların kodu test etmek için oluşturduğu sahte nesnelere.

N

Niko-niko Calendar - Duygu Durumu Takvimi

Takımın duygusal durumunun günlük seviyede izlendiği takvimdir. "Duygu Durumu Tahtası" olarak da adlandırılır.

SEPTEMBER 2011							
	1	2	3	4	5	6	7
BERND	😊	😬	🚫	🚫	😬	😊	😡
MARIKO	😊	😊	🚫	🚫	😡	😊	😊
ALEX	😊	😬	🚫	🚫	😊	😬	😡
MEIKE	😡	😊	🚫	🚫	😊	😊	😡
STEFAN	😊	😊	🚫	🚫	😬	😬	😊



Nonfunctional Requirements- Fonksiyonel Olmayan Gereksinimler

Yazılımın performans, kullanıcı arabirimi vb konulardaki özellikleridir. Bu gereksinimlerin başında kullanıcı arabirimi, performans ve güvenliğe ilişkin özellikler gelir.

P

Pair Programming - Eşli Kodlama

Bir kişi bilgisayar başında kod yazarken diğer kişi onu izleyerek gerekli yerlerde yönlendirmesidir. 2 pilotun bir uçağı kullanması gibi düşünülebilir. Birisi daldığı ya da hata yaptığı anda diğer pilot duruma müdahale ederek durumu kurtarır.

Planning Poker - Scrum Poker

Yazılım geliştirme aşamasında, hedef belirlemek için kullanılan bir oyunlaştırma tekniğidir. Oyunun amacı oyuna katılan ve geliştirmede söz sahibi olacak uzmanların belirli hedeflerde anlaşmalarını sağlamaktır. Bu açıdan Delfi metoduna benzemektedir. Planlama pokeri gayet basittir, katılımcılara öncelikle bir problemin tanımı yapılır ve katılımcıların tahminde bulunmaları ve bu tahmin için kartlardan birisini tercih etmesi istenir.

Product Backlog - Ürün İş Listesi

Proje için gerekli olan gereksinimler listesidir. Proje sonunda "Ne üretilmek isteniyor?" sorusuna cevap aranır. Product owner tarafından müşteriden gereksinimler alınır, öncelik sırasına göre sıralanır. Product owner, değişen ihtiyaçlara göre product backlog'a ekleme veya çıkarma yapabilir. Böylece değişim, projenin her aşamasında projeye kolayca entegre edilebilir olur.

Product Owner - Ürün Sahibi

Yazılım geliştirme takımı tarafından ortaya çıkartılan ürünlerin mevcut değerini maksimuma çıkarmaktan sorumlu kişidir ve tam zamanlı olarak çalışılması gerekmektedir.



R

Refactoring - Kod Düzenleme

Yazılan bir kodun zamanla değiştirilerek optimize edilmesi, kodu daha anlaşılır hale getirilmesidir.

Release - Sürüm

Sprint içerisinde tamamlanan işlerin yayına alınması işlemine verilen isimdir.

Retrospective

Sprint bitirildikten sonra tüm ekibin bir araya gelerek genel bir değerlendirme yapmasıdır. Bu toplantıların 3 temel sütunu bulunur.

- **Stop** : Ekip üyeleri bundan sonra neyi yapmayı bırakacağı hakkında kendi fikirlerini iletir.
- **Continue** : Ekip üyeleri neyi yapmaya devam etmeleri gerektiği hakkında kendi düşüncelerini paylaşır.
- **Start** : Ekip üyeleri artık neyi yapmaya başlamaları gerektiği konusunda fikirlerini aktarı

S

Scrum

Agile manifestosunu esas alan bir proje geliştirme sürecidir. Şeffaf, denetlenebilir ve revize edilebilir olması temel özellikleridir.

Scrum Master

Scrum kuralları, teori ve pratiklerini iyi bilen ve Scrum Takımı'nın bu kuralları uygulamasından sorumlu kişidir.

Scrum of Scrums

Grupları 5-10 kişilik Agile takımlara ayırmak suretiyle Scrum'ı büyük gruplara ölçeklendirme tekniğidir. Takımlar arasındaki bağımlılıkları tartışmak için her takımdan bir kişinin katıldığı senkronize toplantılar ile yürütülür.

Self Organization

Takımların çalışmalarını bağımsız bir şekilde organize ettiği örgütsel ilkedir. Takımlar, takım dışındaki kişiler tarafından yönlendirilmek yerine, işlerini en iyi nasıl gerçekleştireceklerini kendileri seçerler.

Sign Up for Tasks

Agile ekip üyeleri yöneticileri tarafından atanmış işler yerine çalışacakları işleri kendilerinin seçmesidir.

Sprint Backlog

Sprint Backlog ise bir sprintte yapılacak işlerin yer aldığı ve Product Backlog'daki öncelik sırasına göre, tüm ekibin beraber oluşturduğu bir listedir. Sprint Backlog, işlerin puanlarına ve takımın hızına (velocity) göre oluşturulur.

Story Mapping - Ürün Deneyim Haritası

Büyük resmi gösterecek ve tüm paydaşları aynı sayfada birleştirmeye yarayan bir araçtır. Deneyim haritalarının amacı tüm paydaşları aynı sayfaya getirmektir, bir görünürlük sağlama aracı olarak kullanılması gerekmektedir.



Task Board – Görev Panosu

En temel şekli "To Do", "In Progress" ve "Done" olmak üzere üç sütuna ayrılmıştır. Kartlar, o görevin geçerli durumunu yansıtacak şekilde sütunlara yerleştirilir.



Test Driven Development (TDD) - Test Odaklı Geliştirme

Kodun yazılmadan önce testlerinin yapıldığı ardından bu testleri geçecek kodun yazıldığı bir yazılım geliştirme yöntemidir.

Three C's - 3C İlkesi

Kullanıcı hikayeleri, kullanıcı ya da müşteriye özgü geliştireceğimiz işlevi (fonksiyon) tanımlamak için kullanabileceğimiz pratik bir araçtır. Ron Jeffries'e göre 3 temel özelliğe (3C — Card, Conversation, Confirmation) sahiptir:

- **Kart** : Sunulan işlevin tanımını ifade eder. Kapsamlı bir açıklamadan ziyade, işlevi temsil eden (represent) kısa bir tanımdır.
- **Konuşma** : Kullanıcı hikayeleri talep edilen işlevle ilgili detay içermez. Bu eksiklik, iletişimin en etkili yolu olan "yüz-yüze" iletişimle giderilir. Ürün sahibi ve geliştirme ekibi Sprint süresince sıklıkla bir araya gelerek kullanıcı hikayesi üzerinde çalışır ve birlikte detaylandırılırlar.
- **Teyit** : Her kullanıcı hikayesi, işin tamamlanması için koşul olan kullanıcı kabul kriterlerini de içermektedir. Kriterler, talepten beklenen küçük şartlar olarak değerlendirilebilir.



Three Amigos- Business,Development,Test

3 farklı şapkaya sahip (rol, pozisyon, yetkinlik vs olarak) kişilerin bir araya gelmesidir. Farklı bakış açılarını bir araya getirmenin, bir parçayı oluşturan öğeleri her alandan değerlendirebilmek için en etkili tekniklerden birisidir. Önemli olan 3 farklı perspektifin bir arada olmasıdır. Bir çıktı üretebilmek için gerekli olabilecek şapkalar değerlendirildiğinde,

- **WHAT – Business:** Hangi işin yapılacağı, hangi sorunun çözüleceği bilgisi
- **HOW – Geliştirme:** Yapılacak işin teknik olarak nasıl yapılması gerektiği, sorunun nasıl çözülmesi gerektiği bilgisi
- **QUALITY – Test:** Üretilen çözümün etkisinin ne olacağı bilgisidir.

U

Ubiquitous Language

Projede kullanılan her servisin domainde bir karşılığı olması gerekmektedir. Böylelikle projede yer alan herkes bu ortak dili konuşabilir birbirini anlayabilir hale gelmektedir.

Unit Testing - Birim Testi

Unit Test, bir yazılımın en küçük test edilebilir bölümlerinin, tek tek ve bağımsız olarak doğru çalışması için incelendiği bir yazılım geliştirme sürecidir. Unit Test yazılım testinin ilk seviyesidir ve entegrasyon testinden önce gelir. Unit Testleri geliştiriciler kendileri yazar ve yürütürler.

Usability Testing - Kullanılabilirlik Testi

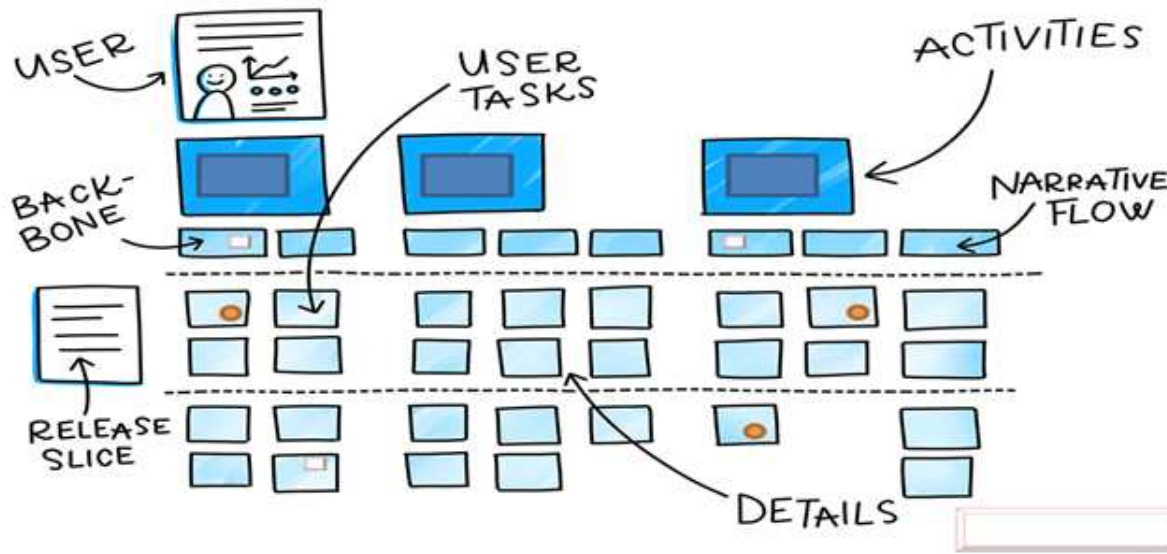
Kullanılabilirliğinin değerlendirilmesi istenen arayüzün belirlenen hedef kitlede belirlenen görevlerin yaptırılması ve bu süreçte kullanıcıdan verimlilik, etkililik ve memnuniyet değerlerinin alınmasıdır. Kullanıcı testleri 5 bileşenden oluşur;

- **Öğrenilebilirlik:** Kullanıcılar ilk defa kullandıkları arayüzde, basit işlemleri kolaylıkla tamamlayabiliyor mu?
- **Verimlilik:** Kullanıcılar verilen taskleri hızlı tamamlayabiliyor mu?
- **Hatırlanabilirlik:** Kullanıcılar ilk kullanımdan sonra yazılımı kullanmadan geçen süre sonrasında yazılımı hala verimli şekilde kullanabiliyor mu?
- **Hatalar:** Kullanıcı ne kadar hata yapıyor, yaptığı hatalar ne kadar ciddi ya da kullanıcı hatalarını kolayca düzeltebiliyor mu?
- **Memnuniyet:** Kullanıcılar yazılımdan memnun kaldırlar mı?

User Stories - Kullanıcı Hikayeleri

Son kullanıcı tarafından kullanılacak bir özelliğin(feature), yine bu kullanıcı tarafından en basit ve anlaşılır şekilde açıklanmasıdır. Elbette bu açıklamayı gerçekleştirirken aşağıda yer alan 3 soruya cevap vermesi gerekiyor.

- Talep eden kullanıcı kim?
- Talep edilen ihtiyaç nedir?
- Neden bu talebe ihtiyaç var?

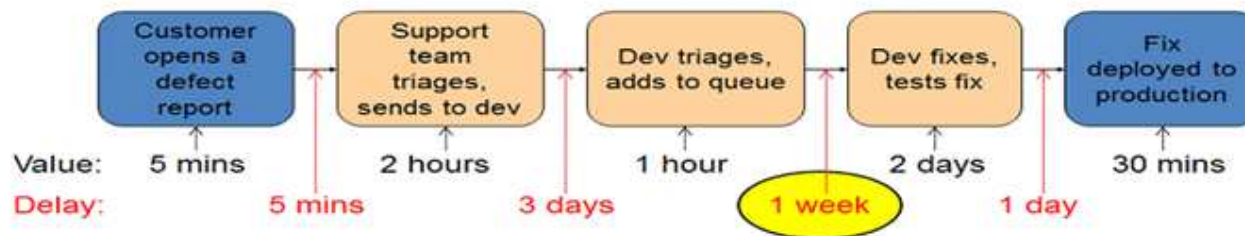


V

Value Stream Mapping - Değer Akışı Haritalandırma

Ürün ve hizmet oluşumundaki süreçlerin bütünü ele alarak değer yaratan ve yaratmayan tüm faaliyetleri mevcut durum haritalandırması ile belirleyip israf yaratan faaliyetleri minimize etmeyi amaçlayan bir yalın üretim metodudur.

A Simple Value Stream Map



Velocity

Velocity, son 3 ya da 4 sprintin ortalama rakamlarıdır. Backlog itemlerinin ne zaman ulaşacağını tahmin etmek için kullanılır.



**IT GOVERNANCE
TURKEY**