



# SDLC

## SOFTWARE DEVELOPMENT LIFE CYCLE

(Yazılım Geliştirme Yaşam Döngüsü)

4. Ders  
20.12.2022



# Bu derste neler öğreneceğiz?

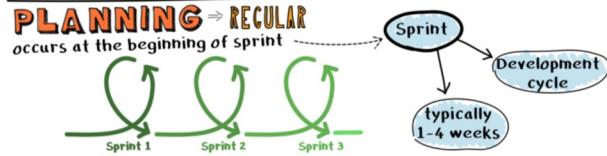
- 1- Epic, Feature, User Story**
- 2- Kahoot**



# SCRUM

VS

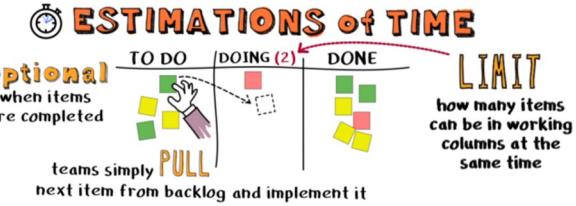
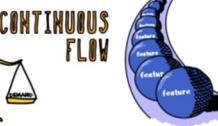
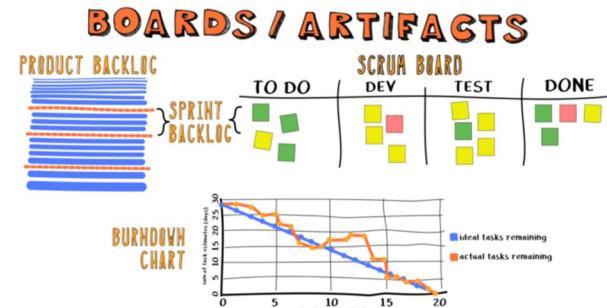
# KANBAN



**CHANGES TO WORK SCOPE**  
should wait for next sprint



**OWNERSHIP** → Product Owner  
**WHEN TO USE**  
small items - small value  
adding increments possible  
requirements in a good shape  
roadmap is clear  
more cross-dependent teams  
BOULDER + = ROCK + PEBBLE

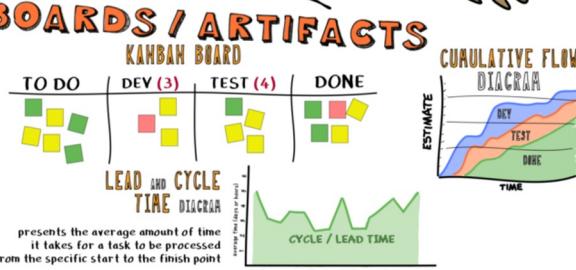
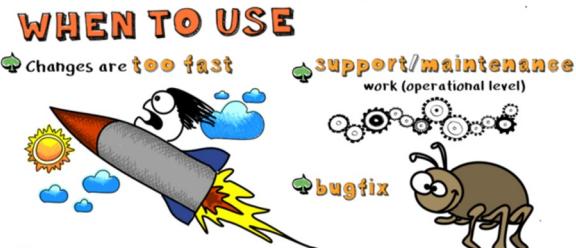


**CHANGES TO WORK SCOPE**  
added AS NEEDED

**ROLES AS NEEDED**

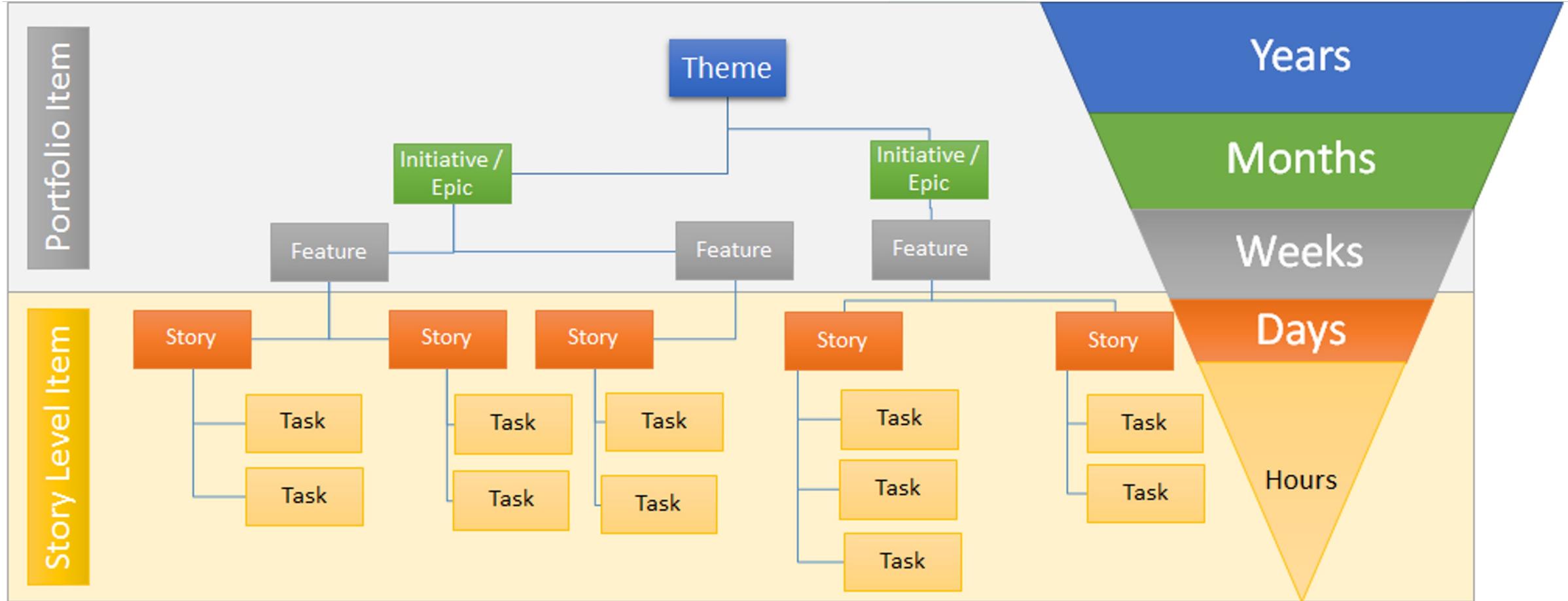
**MEETINGS NONE REQUIRED**

**OWNERSHIP DEPENDS** on defined roles and necessities  
who me?



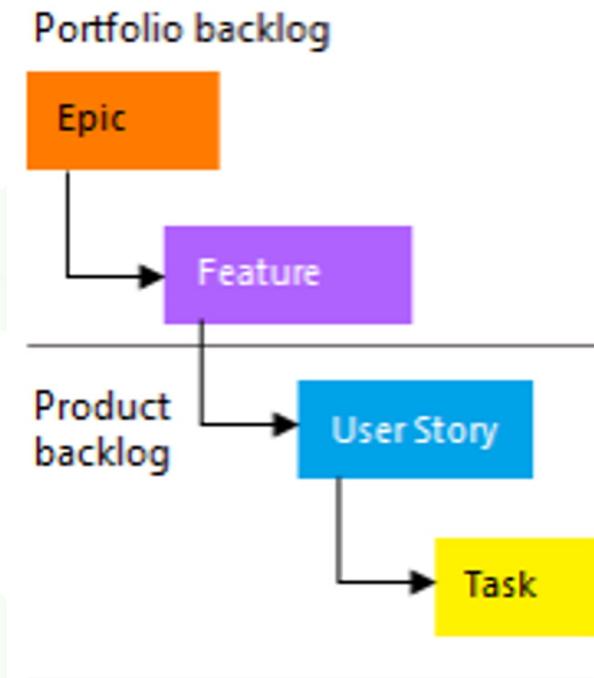
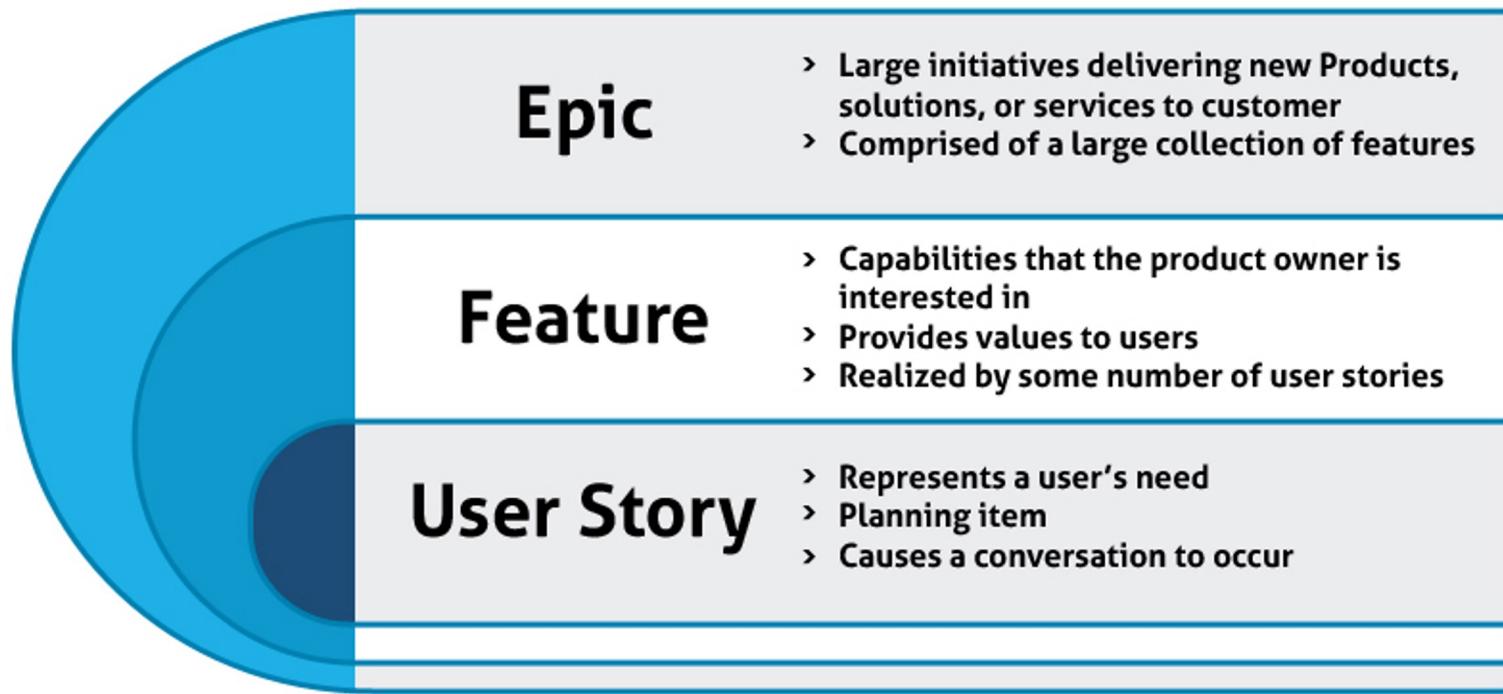


# EPICS , FEATURE , USER STORY , TASK





# EPICS , FEATURE , USER STORY , TASK





# EPIC , FEATURE NEDİR?

**EPIC** : Belirli bir vizyona ulaşabilmek için şirketlerin metrikler ile tanımlı, ulaşması gereken hedefleri vardır. Ekipler ise bu hedeflere ulaşabilmek için yeni girişimler/özellikler ile gelir ve sahip oldukları ürünü/ürünlerde yeni güncellemeler yayınlarlar. Epics, yapılması planlanan bir veya birden fazla yeni fonksiyonların yahut özelliklerin gruplanmasıdır.

Product ekibinde yapılan her şey son kullanıcıya sunulmaz veya yeni bir özellik olarak sonuçlanmaz. Bu yüzden özellikler(features) olarak değil epics olarak adlandırılırlar. Ayriyeten bir sprint'ten daha uzun süren çalışmalar da “epics” olarak adlandırılmaktadır.

**FEATURE** : Epicler içerisinde PO tarafından seçilerek Product Backlog'a eklenen, birden fazla user story içeren, birden fazla haftada tamamlanacak modüllerdir. Son kullanıcıya tamamlanmış ürün olarak ulaşır.

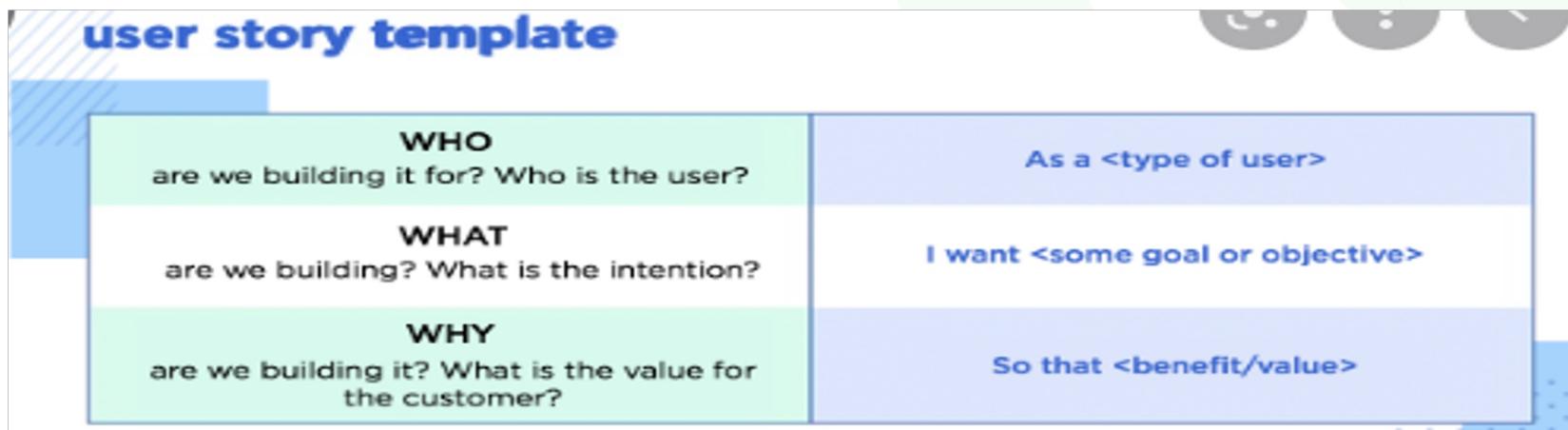


# USER STORY NEDİR?

User storyler 3 bölümden oluşur. Bunlar;

- Kullanıcının hikaye tanımı (Value statement)
- Kabul kriterleri (Acceptance criterias)
- Bitti tanımı (Definition of done)

User Stories, Agile metodolojisinin bir parçasıdır. Son kullanıcıya sunulacak herhangi bir özelliği veya ürünü açıklama yollarından biridir.





## USER STORY #1

### Hikâye Tanımı (Value Statement)

Ürün ekibi yetkilisi olarak, paket yenilemelerini arttırmabilmek için; üyelerimize paketlerinin bitmesine 1 ay kala “yenileme yapmak ister misiniz” mail gönderimi yapılmasını, mail şablonu içerisinde X, Y, Z paket detayları ve ilgili müşteri temsilcilerinin ad, soyad, iletişim bilgilerinin yer olmasını istiyorum.

### Kabul kriterleri (Acceptance criterias)

- Paket bitimine 1 ay kalan kullanıcılaraya mail gönderimlerinin başarıyla yapılması
- Üye – müşteri temsilcisi eşleşmesinin doğru yapılmış olması
- Gönderimlerin sadece mail gönderimine izin verilen kullanıcılaraya yapılması
- Mail opt-out kurgularının sağlıklı bir şekilde çalışması

### Bitti Tanımı (Definition of Done)

- Kod kontrol edildi (Code review)
- Birim testleri yapıldı (Unit tests)
- Kabul kriterleri karşılandı (Acceptance criterias)
- Product Owner(PO) story kabulü (PO accepts user story)



User Story ID	Description	Acceptance Criteria
US 0001	ATM kullanıcısı olarak, nakit çekebilmem için PIN kodumu girmek istiyorum	PIN dört basamaklı olmalıdır PIN özel karakterlere (sembollere) izin vermemelidir PIN'in 30 saniye içinde girilmesi gereklidir, aksi takdirde işlem iptal edilir
US 0002	Facebook login (Giriş sayfası) geçersiz kimlik bilgileriyle erişilmemelidir	Gecersiz kullanıcı adı ile erişim sağlanamaz Geçersiz şifre ile erişim sağlanamaz Geçersiz kullanıcı adı ve şifre ile erişim sağlanamaz

User Stories üç ortak özelliği barındırır:

- Kısa ve açıklayıcı olma:** Ürün ya da özelliğin kullanarak ne yapmak istendiğini kısaca anlatabilmenizi sağlar.
- Kullanıcı tipini belirleme:** Kullanıcı hakkında detaylı bilgi verilmese de, kullanıcının karakteristik özelliklerinin ürün ya da yazılımla ilişkisi vurgulanır.
- Problemi tanımlama:** Kullanıcı hikayeleri, ürünün ya da yazılımın çözmesi gereken problemleri ortaya çıkarır.



WHO

WHAT

WHY

As a **role**, I want **action** (so that **benefit**)

KİM?

NE?

NEDEN?

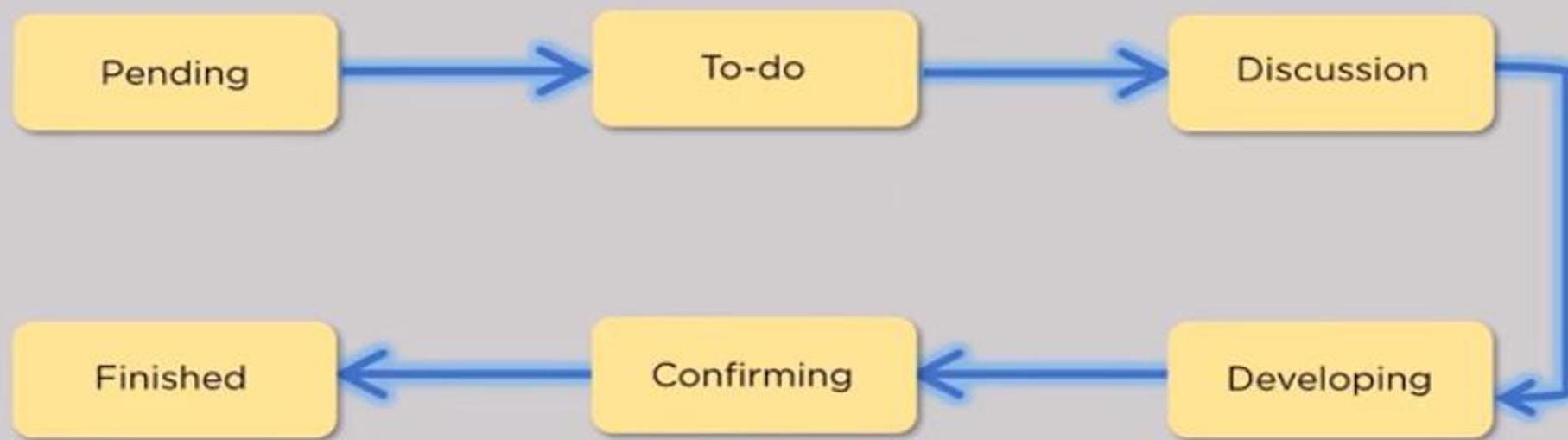
Rol olarak, **aksiyon** yapmak istiyorum, çünkü **yarar**.

- Bir kullanıcı olarak şifre oluştururken, güvenlik amacıyla, güçlü bir şifre oluşturmak istiyorum.
- Bir yönetici olarak, çalışanlarının bilgilerini, filtreleyerek daha çabuk ulaşmak istiyorum.
- Bir kullanıcı olarak birden fazla kişiye, daha hızlı olacağı için aynı anda mesaj göndermek istiyorum
- “Sıkça sinemaya giden bir izleyici olarak, yakınimdaki sinema salonlarında vizyona yeni girecek filmler hakkında bütün bilgilere telefonumdan ulaşabilmek, böylece her filmiin web sitesini ayrı ayrı kontrol etmeye ihtiyaç duymamak istiyorum.”

A	B	C	D	E
User_Story_ID	Description	Acceptance Criteria	Priority	Validation
1	US_001	System should allow any user to register with valid credentials validating the success message	High	UI and Backend Testing
2		There should be a valid SSN respecting the "-" where necessary, it should be 9 digits long		
3		There should be a valid name that contains chars and cannot be blank		
4		There should be a valid lastname that contains chars and it is a required field		
5		You can provide chars and digits to describe a valid address along with zip code		
6		User should provide 10 digit-long mobilephone number as a required field respecting the "-"		
7		User cannot use just digits, but can use any chars and digits along with them and of any length		
8		You should provide a valid email format that contains "@" sign and "." extensions		
9				
10	US_002	System should not allow anyone to register with invalid credentials seeing the message "If you want to sign in, you can try the default accounts:- Administrator (login="admin" and password="admin") - User (login="user" and password="user").	High	UI and Backend Testing
11		Any field on the registration page should not be left blank		
12		SSN number cannot be of any chars nor spec chars except "-"		
13		Mobilephone number cannot be of any chars nor spec chars except "-"		
14				
15	US_003	Registration page should restrict password usage to a secure and high level passcode	High	UI and Backend Testing
16		There should be at least 1 lowercase char for stronger password and see the level chart change accordingly		
17		There should be at least 1 uppercase char and see the level chart change accordingly		
18		There should be at least 1 digit and see the level chart change accordingly		
19		There should be at least 1 special char and see the level bar change accordingly		
20		There should be at least 7 chars for a stronger password		
21	US_004	Login page should accessible with valid credentials	High	UI and Backend Testing
22		There should be a valid username and password validating the success message to login		
23				
24	US_005	Login page should not be accessible with invalid credentials	High	UI and Backend Testing
25		User cannot login with invalid username validating the error message		
26		User cannot login with invalid password validating the error message		
27		User cannot login with invalid username and password validating the error message		
28		User with invalid credentials should be given an option to reset their password		
29		User should be given the option to navigate to registration page if they did not register yet		



## LIFECYCLE OF A USER STORY





# DEFINITION OF DONE (DoD) TANIMI

Development Team, kendilerinden tamamlanması beklenen işleri Product Owner'a sunmadan önce bir kalite-kontrol süreci gibi kontrol listesi hazırlar. Bu liste tüm User Story'ler için ortaktır ve biten tüm işler bu prosedürden geçmesi gereklidir. Buna kısaca "Done" veya "DoD — Definition of Done (Bitti Tanımı)" denir. Bu liste genellikle şu kontrolleri içerir:

- Unit testleri
- QA testleri
- User Story'deki tüm dökümanların güncellenmesi
- Kodun review edilmesi
- User Story kabul kriterlerinin karşılanması
- Product Owner onayı

DoD checklisti, Sprint veya Release sonunda belirli tamamlama kriterlerinin karşılanıp karşılanmadığını kontrol etmek için de kullanılmak üzere hazırlanabilir.



# DEFINITION OF DONE (DoD) TANIMI

## Definition of Done

2 / 5

Add a new item

- =  Functionality described in the story is implemented and tested by the QA. Technical tasks are accepted by the Component lead.
- =  Change has been deployed to the test server
- =  Unit tests are written for any new code
- =  The documentation is updated
- =  CHANGLOG entry is added for all services impacted during work on ticket.

Checked by Claribel Spring on Mon Mar  
25 2019 14:40:03 GMT+0100



# TAKIM KAPASİTESİ VE PUANLAMA

- User storyler sprint backlog'a alınırken takim tarafından puanlanır (**POINT**)
  - Sprint baslangic toplantisi veya refinement toplantısında User Story okunur ve development team'in tum uyeleri user story'e point verir
  - Verilen pointler kartlara yazılıarak gösterilir
  - Pointler Fibonacci Serisindeki sayılara gore verilir.  
1 1 2 3 5 8 13 21 34 .....
- Her üye niçin bu point'i verdigini açıklar ve team'in ortak karariyla user story'nin point'i belirlenir
- Team'deki developer ve QA sayisina gore takım (team) kapasitesi (capacity) belirlenir.  
1 point = 1 günlük çalışma yani 8 saat





# TAKIM KAPASİTESİ VE PUANLAMA

**Örnek:** 5 developer'in takım kapasitesi nedir?

$5 \times 1 = 5$  (5 developer'in günlük kapasitesi 5 point)

1 haftalık kapasitesi  $\Rightarrow 5 \times 5 = 25$  point

2 haftalık kapasitesi = 50 point

**Örnek:** 2 QA'in takım kapasitesi nedir?

$2 \times 1 = 2$  (2 developer'in günlük kapasitesi 2 point)

1 haftalık kapasitesi  $\Rightarrow 5 \times 2 = 10$  point

2 haftalık kapasitesi = 20 point

Development team'de 1 kişinin 2 haftalık sprintti yapacağı iş miktarı  $5+5=10$  point'tır.

