

**DT/NT : DT/NT - CUMARTESİ**  
**LESSON : STLÇ**  
**SUBJECT : BUG YAŞAM DÖNGÜSÜ**

**BATCH:189**  
**30.09.2023**



**TECHPRO**  
EDUCATION



techproeducation.com



+1 (585) 304 29 59





# Önceki Dersimizde..



Boundary Value Analysis

Sınır Değerleri Analizi



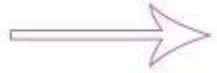
Equivalence Class Partitioning

Denklik Paylarına Ayırma Tekniği



Decision Table based testing

Karar Tablosu



State Transition

Durum Geçiş Tablosu



Use Case testing

Kullanım Durumları Testi



# BUG YAŞAM DÖNGÜSÜ

## **STLC**



# Temel Kavramlar

## Error:

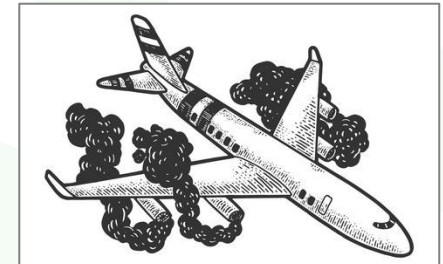
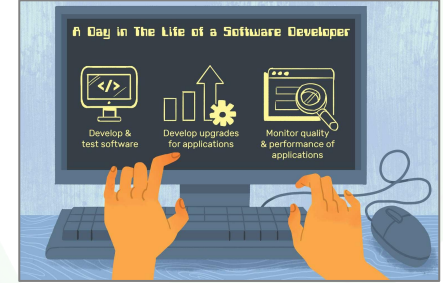
Kodun hatalı yazılması sonucu ortaya çıkan hatadır. Uygulamada beklenen sonuçların alınamaması veya istenmeyen sonuçların elde edilmesidir.

## Bug / Defect:

Uygulamada, gereksinimlere uygun olmayan herhangi bir davranışı ifade eder. Yazılımın hatalı çalışmasına ve beklenmeyen sonuçlara yol açar.

## Failure:

Sistemin belirli bir test veya kullanım durumunda işlevselliğini yerine getirememesidir. Bir error'un ortaya çıkardığı sonuçtur.





# Temel Kavramlar

Developer kod yazarken  
bir **hata** yapar



Bu hata uygulamada  
bir **bug** ortaya çıkarır



Bu bug, uygulama  
kullanılırken bir  
**başarısızlığa** sebep olur.



# Temel Kavramlar

## BUG mı, DEFECT mi

**Defect :** Uygulamanın **geliştirme aşamasında** ortaya çıkan kusurlara denir. Bunu developerda bulabilir , testerda. Önemli olan hangi aşamada bulunmuş olmasıdır

**Bug:** Uygulama **tamamlandıktan sonra** bulunan hatalardır. Bu hataları testerlarda bulabilir yada son kullanıcı da bulabilir.





# Bug Yaşam Döngüsü

Amaç, tespit edilen kusuru raporlamak, ilgili kişilere atamak ve hata düzeltme sürecini sistematik ve verimli hale getirmektir.

Bir bug bulunduğunda;

- Bug olduğundan emin olmak gerekir.
- Developer durumdan haberdar edilir.
- Developer bunun bug olmadığını, bug olduğunu ya da önemsiz bir durum olduğunu ifade edebilir.
- Developer'ın dönüşüne göre hareket edilir.



# Bug Yaşam Döngüsü

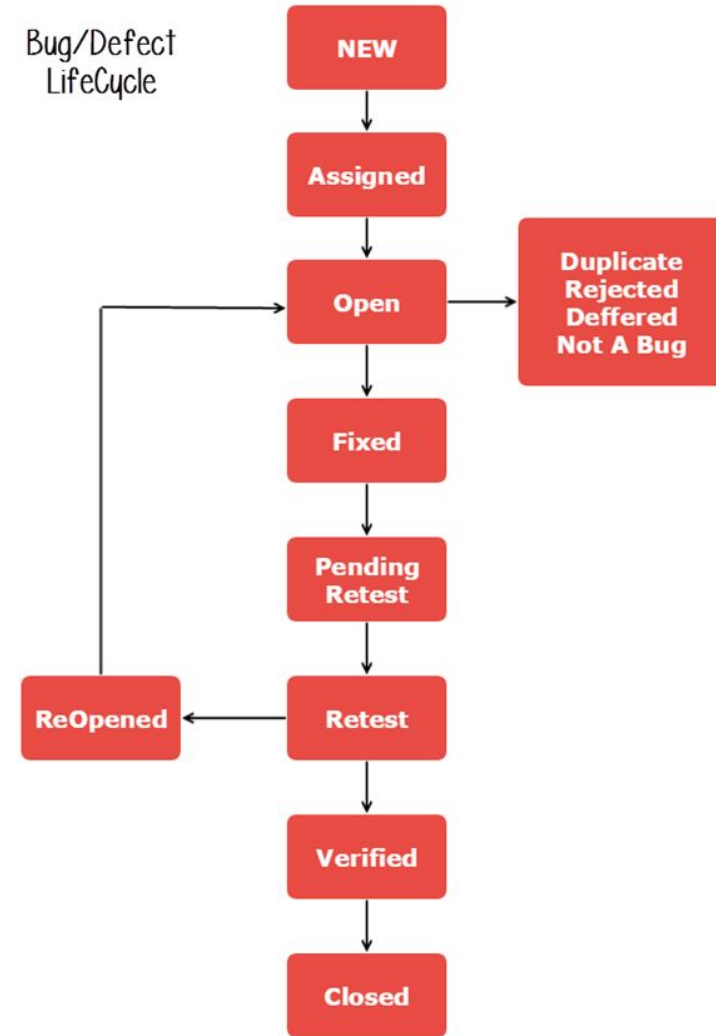
**New (Yeni)** : Bir hata ilk kez yayınlandığında, durum Yeni olur. (Jira, Alm, VersionOne gibi boardlarda..)

**Assigned (Atandı)** : Hata post edilince, test lead veya Proje lead, ilgili Developer'a "atanmış" olarak atayacaktır. (Meşru bir hata olup olmadığı ile alakalı Developer'ın kesinlikle bir girdisi olacaktır.)

**Open (Açıldı)** : Developer bu aşamada düzeltmek için hatayı analiz eder ve üzerinde çalışır.

**Fixed (Onarıldı)** : Developer gerekli kod güncellemelerini yaptığı ve hatanın düzeltilip düzeltilmediğini doğruladığında, hata durumunu "Fixed" olarak değiştirebilir ve hata test ekibine aktarılır.

Bug/Defect  
LifeCycle







# Bug Yaşam Döngüsü

**Pending retest (Tekrar Test İçin Bekliyor):** Arızayı giderdikten sonra Developer, yeniden test için bu özel functionalityyi deploy eder. Burada testler test aşamasında beklemede olur. Ve durum tekrar test etmeyi gerektirir.

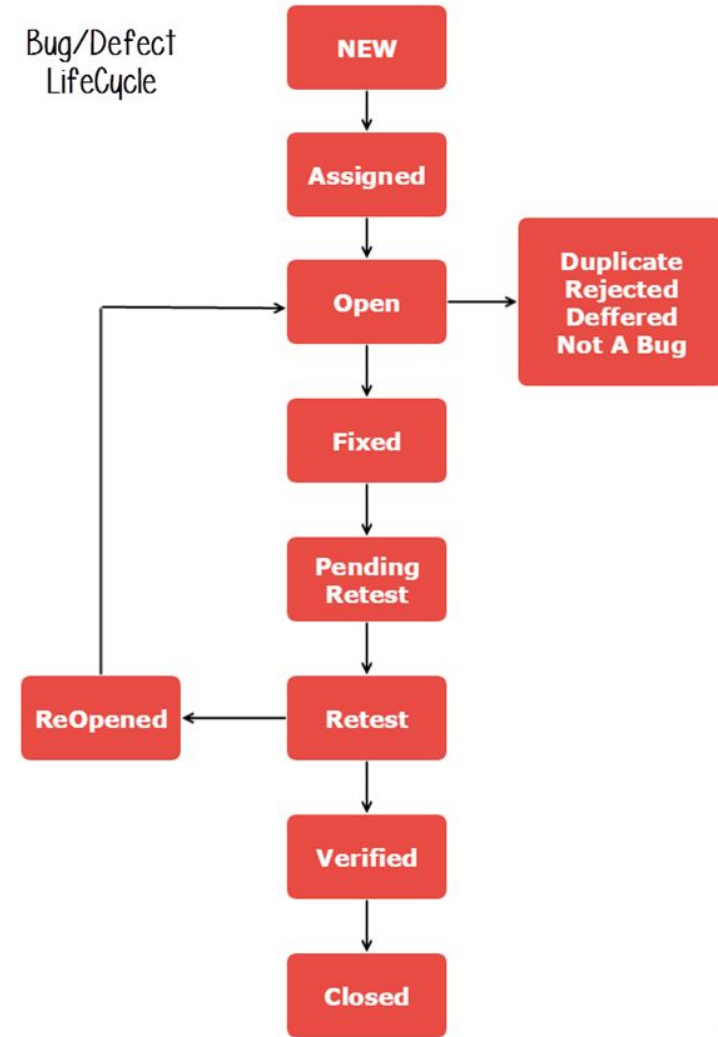
**Retest (Tekrar Test):** Bu aşamada, tester işlevselliği farklı test verileriyle tekrar test eder.

**Verified (Onaylandı):** Tester doğrulama yaptıktan ve hata yazılımda görünmediğinde, hatanın düzeltildiğini onaylar ve durumu "Doğrulandı" olarak değiştirir.

**Reopen (Tekrar Açıldı):** Hata Developer tarafından düzeltildikten sonra bile hata hala mevcutsa, tester durumu "Yeniden Açıldı" olarak yapar. Bu yüzden hata aynı süreçten bir kez daha geçer.

**Closed (Kapandı):** Hata giderildikten sonra tester tarafından test edilir. Tester yazılımda defect bulunmadığından emin olursa, hatanın durumunu "Kapalı" olarak değiştirir. Bu, hatanın düzeltildiği, test edildiği ve onaylandığı anlamına gelir.

Bug/Defect  
LifeCycle





# Bug Yaşam Döngüsü

Bug Ticket oluşturulduğunda Developer'ların cevapları:

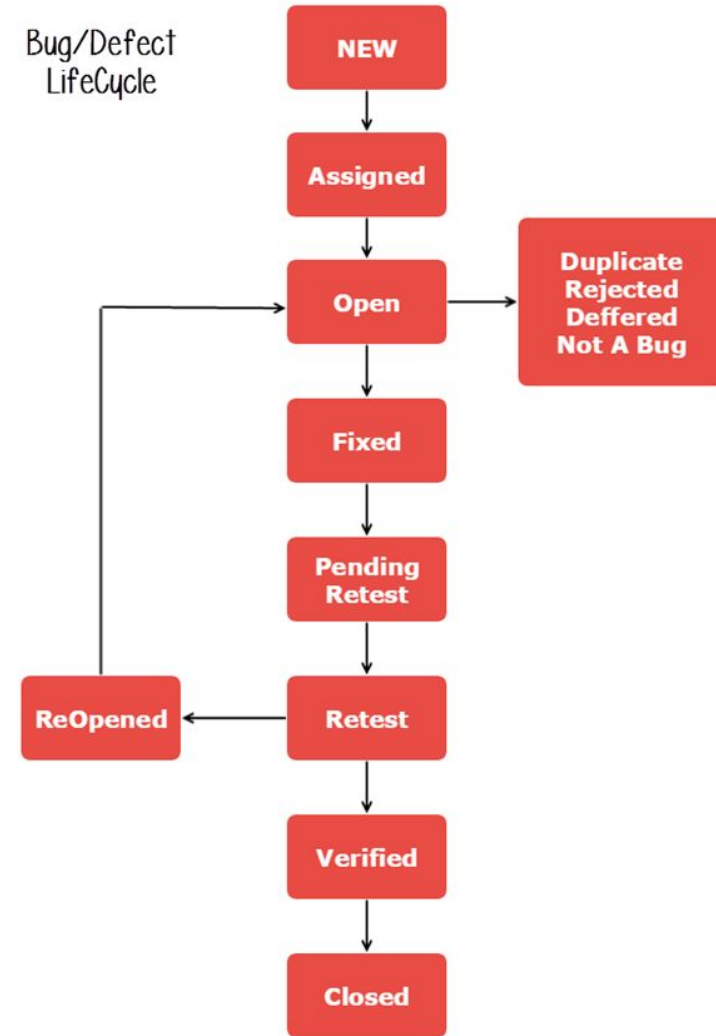
**Rejected (Reddedildi):** Kusur Developer tarafından meşru bir kusur olarak kabul edilmezse Developer tarafından **Reddedildi** olarak işaretlenir.

**Duplicate (Tekrar):** Developer hatayı diğer defectlerle aynı bulursa, hata durumu Developer tarafından **Çift** olarak değiştirilir.

**Deferred (Ertelendi):** Developer, hatanın çok önemli bir öncelik olmadığını ve sonraki sürümlerde giderilebileceğini düşünüyorsa, hatanın durumunu **Ertelendi** olarak değiştirebilir.

**Not a Bug (Bug Değil) :** Kusurun uygulamanın işlevselliği üzerinde bir etkisi yoksa, kusurun durumu **Bug Değil** olarak değiştirilir.

Bug/Defect  
LifeCycle





# Bug Yaşam Döngüsü

## Bug Raporu (Bug Report)

Hata raporlama işlemi proje yönetim aracı üzerinde gerçekleştirilebilirken, aşağıdaki başlıklardan oluşan bir dokümanda oluşturulabilir. Bu doküman sayesinde geliştirici hatayı daha rahat anlayabilir.



**Başlık (Title) :** Hata başlığı diğer hatalarda ayırt etmek için yazılır. Hata başlığı kısa ve öz olmalıdır. Örnek vermek gerekirse *“Kullanıcı tanımlama ekranında kaydet butonuna basıldığında hata veriyor.”*

**Atama (Assigned To) :** Bulunan hatanın hangi geliştiriciye veya takım liderine atanacağı bu alandan seçilmelidir.

**Öncelik (Priority) :** **Yazılım Test Planı’n**da hatalar önceliklerine göre derecelendirilmelidir. Bulunan hataların öncelik değerleri plana göre doğru şekilde girilmelidir.

**Önem Derecesi (Severity) :** Bulunan hatalar önem derecesine göre gruplandırılmalıdır. İlgili geliştirici hataları önem derecesine göre çözmelidir.

**Hata Kaynağı (Root Cause) :** Bulunan hatanın kaynağı belirtilmelidir. Bu bilgi test faaliyetleri sonucunda ölçüm ve analiz için kullanılabilir.



# Bug Yaşam Döngüsü

## Bug Raporu (Bug Report)

**Adımlar (Repro Steps) :** Bulunan hatanın ilgili geliştirici tarafından tekrar simüle edilebilmesi için hatanın adım adım yazılması gerekmektedir. Bunun için aşağıdaki örneği verebiliriz:

- Kullanıcı tanımlama sayfasına gidilir.
  - Ekranda yer alan tüm zorunlu alanlar doldurulur.
  - Kaydet komutu verilir.
- Beklenen Sonuç:** Kullanıcı tanımlama işleminin başarıyla tamamlandığına dair bilgilendirme mesajı verilmesi gerekir.
  - Gerçekleşen Sonuç:** NullPointerException hata mesajı alındı.

**Ekler (Attachments) :** Bulunan hatanın ekran görüntüsünü veya video kaydını alarak bu alana yüklenmesi test ekibinin hatayı geliştirme ekibine daha kolay aktarabilmesini sağlayacaktır.

	A	B	C
1	Category	Label	Value
2	Bug ID	ID number	#123
3		Name	CART - Unable to add new item to my cart
4		Reporter	Mike A
5		Submit Date	03/04/16
6	Bug overview	Summary	When my cart contains one item, I am unable to add a second item via the add to cart button on a product page
7		URL	www.example.com/product/abc
8		Screenshot	www.example.com/screenshot123
9	Environment	Platform	Macintosh
10		Operating System	OS X 10.12.0
11		Browser	Chrome 53
12	Bug details	Steps to reproduce	add one item to cart > go to product abc via the search bar > add new item to cart via "add to cart" button (see screenshot) > go to cart
13		Expected result	The cart should contain 2 items
14		Actual result	The cart contains only 1 item
15		Description	/
16	Bug tracking	Severity	Major
17		Assigned to	/
18		Priority	High
19	Notes	Notes	/



# Bug Yaşam Döngüsü

Örnek Bir Bug Raporu