CEN429 GÃ¼venli Programlama Hafta-2

GeliÅŸtirme OrtamÄ± GÃ¼venliÄŸi ve YazÄ±lÄ±m GeliÅŸtirme SÃ¼reÃ§leri

Yazar: Dr. Ã–ÄŸr. Ãœyesi UÄŸur CORUH

# CEN429 GÃ¼venli Programlama

## Hafta-2

#### GeliÅŸtirme OrtamÄ± GÃ¼venliÄŸi ve YazÄ±lÄ±m GeliÅŸtirme SÃ¼reÃ§leri

Ä°ndir

* [PDF](pandoc_cen429-week-2.tr_doc.pdf)
* [DOC](pandoc_cen429-week-2.tr_word.docx)
* [SLIDE](cen429-week-2.tr_slide.pdf)
* [PPTX](cen429-week-2.tr_slide.pptx)

### Outline

* GeliÅŸtirme OrtamÄ± GÃ¼venliÄŸi ve YazÄ±lÄ±m GeliÅŸtirme SÃ¼reÃ§leri
* YazÄ±lÄ±m GeliÅŸtirme SÃ¼reci
  + YazÄ±lÄ±m GeliÅŸtirme AkÄ±ÅŸÄ±
  + KonfigÃ¼rasyon Sabitleme
  + DeÄŸiÅŸiklik BaÅŸlatma ve SÄ±nÄ±flandÄ±rma
  + DeÄŸiÅŸikliÄŸi Onaylama ve YayÄ±nlama
* YazÄ±lÄ±m GeliÅŸtirme OrtamlarÄ±
  + GeliÅŸtirme OrtamÄ± GÃ¼venliÄŸi
  + SÃ¼rÃ¼m Kontrol Sistemleri
  + Kaynak Kod Sunucu GÃ¼venliÄŸi
  + Sunucu OdasÄ± ve GeliÅŸtirme BilgisayarlarÄ± GÃ¼venliÄŸi

## **YazÄ±lÄ±m GeliÅŸtirme SÃ¼reci ve DeÄŸiÅŸiklik YÃ¶netimi (Software Development Flow and Change Management)**

### 1. **YazÄ±lÄ±m GeliÅŸtirme AkÄ±ÅŸÄ± (Flow)**

#### Teorik AÃ§Ä±klama:

YazÄ±lÄ±m geliÅŸtirme sÃ¼reÃ§leri, belirli akÄ±ÅŸlarla kontrol edilmelidir. DeÄŸiÅŸikliklerin doÄŸru ÅŸekilde yÃ¶netilmesi, projenin baÅŸarÄ±lÄ± bir ÅŸekilde devam etmesini saÄŸlar. Bu akÄ±ÅŸ genellikle sÃ¼rÃ¼m kontrol sistemleri, teknik ekipler ve proje yÃ¶netim sÃ¼reÃ§lerini iÃ§erir.

#### Uygulama:

* **Uygulama:** Basit bir yazÄ±lÄ±m projesi baÅŸlatÄ±n ve deÄŸiÅŸiklik taleplerini (RFC) nasÄ±l yÃ¶neteceÄŸinizi gÃ¶steren bir sÃ¼reÃ§ oluÅŸturun. Her adÄ±mda bir onay mekanizmasÄ± kurun ve sÃ¼rÃ¼m kontrol sistemi Ã¼zerinden projeyi yÃ¶netin.

### 2. **KonfigÃ¼rasyon Sabitleme (Baseline the Configuration)**

#### Teorik AÃ§Ä±klama:

KonfigÃ¼rasyon sabitleme, bir Ã¼rÃ¼n veya sistemin belirli bir sÃ¼rÃ¼mÃ¼nÃ¼ belirleyerek, bu sÃ¼rÃ¼mden itibaren yapÄ±lacak tÃ¼m deÄŸiÅŸikliklerin izlenebilir olmasÄ±nÄ± saÄŸlar. Bu, geliÅŸtirme ve deÄŸiÅŸiklik yÃ¶netimi sÃ¼reÃ§lerinin temel adÄ±mlarÄ±ndan biridir.

#### Uygulama:

* **Uygulama:** Bir GIT deposu oluÅŸturun ve ilk yapÄ±landÄ±rma ayarlarÄ±nÄ± sabitleyin. Bu sÃ¼rÃ¼mdeki tÃ¼m deÄŸiÅŸikliklerin izlenmesini saÄŸlayacak bir yapÄ± kurun.

### 3. **DeÄŸiÅŸiklik BaÅŸlatma (Initiate the Change)**

#### Teorik AÃ§Ä±klama:

DeÄŸiÅŸiklik talepleri (RFC), projeye eklenecek yeni Ã¶zellikler veya dÃ¼zeltilecek hatalar iÃ§in yapÄ±lÄ±r. Bu sÃ¼reÃ§, geliÅŸtirme baÅŸlamadan Ã¶nce tÃ¼m gereksinimlerin belirlendiÄŸi ve teknik toplantÄ±larla deÄŸerlendirildiÄŸi bir adÄ±mdÄ±r.

#### Uygulama:

* **Uygulama:** Bir deÄŸiÅŸiklik talebi (RFC) oluÅŸturun ve bu talebin proje ekibine nasÄ±l iletildiÄŸini simÃ¼le edin. ToplantÄ±lar ve teknik incelemeler Ã¼zerinden karar verildiÄŸini gÃ¶sterecek bir senaryo oluÅŸturun.

### 4. **DeÄŸiÅŸikliÄŸi SÄ±nÄ±flandÄ±rma (Classify the Change)**

#### Teorik AÃ§Ä±klama:

DeÄŸiÅŸiklik talepleri, maliyet, sÃ¼re ve teknik gereksinimler aÃ§Ä±sÄ±ndan sÄ±nÄ±flandÄ±rÄ±lÄ±r. EÄŸer finansal veya teknik bir engel yoksa, Ã¼rÃ¼n sahibi (product owner) bu talebin geliÅŸtirilmesi iÃ§in teknik ekibe onay verir.

#### Uygulama:

* **Uygulama:** Bir deÄŸiÅŸiklik talebini inceleyin ve bu talebin nasÄ±l sÄ±nÄ±flandÄ±rÄ±ldÄ±ÄŸÄ±nÄ±, hangi koÅŸullara gÃ¶re onaylandÄ±ÄŸÄ±nÄ± gÃ¶sterecek bir sÃ¼reci uygulamalÄ± olarak yÃ¶netin.

### 5. **DeÄŸiÅŸikliÄŸi Onaylama ve Planlama (Approve and Schedule the Change)**

#### Teorik AÃ§Ä±klama:

GeliÅŸtirme sÃ¼recine baÅŸlamadan Ã¶nce, deÄŸiÅŸiklik talebi onaylanÄ±r ve bir proje planÄ± oluÅŸturulur. Bu plan, sprintler ve gÃ¶rev daÄŸÄ±lÄ±mÄ±nÄ± iÃ§erir.

#### Uygulama:

* **Uygulama:** Bir sprint planlama toplantÄ±sÄ± dÃ¼zenleyin ve deÄŸiÅŸiklik talebine uygun olarak gÃ¶revleri belirleyin. Planlama araÃ§larÄ± (Jira, Trello vb.) kullanarak bu sÃ¼reci organize edin.

### 6. **DeÄŸiÅŸikliÄŸi YayÄ±nlama (Release the Change)**

#### Teorik AÃ§Ä±klama:

GeliÅŸtirilen deÄŸiÅŸiklik, testler tamamlandÄ±ktan sonra Ã¼retim ortamÄ±na gÃ¶nderilir. Bu adÄ±mda deÄŸiÅŸikliÄŸin baÅŸarÄ±lÄ± bir ÅŸekilde uygulandÄ±ÄŸÄ±ndan emin olunur.

#### Uygulama:

* **Uygulama:** GeliÅŸtirilen bir deÄŸiÅŸikliÄŸi sÃ¼rÃ¼m kontrol sisteminden Ã§ekin ve Ã¼retim ortamÄ±na yayÄ±nlayÄ±n. YayÄ±n sÃ¼recindeki adÄ±mlarÄ± ve test sonuÃ§larÄ±nÄ± kaydedin.

### 7. **DeÄŸiÅŸikliÄŸi DoÄŸrulama ve Ä°nceleme (Validate and Review the Change)**

#### Teorik AÃ§Ä±klama:

DeÄŸiÅŸiklik yayÄ±nlandÄ±ktan sonra, deÄŸiÅŸikliÄŸin doÄŸru bir ÅŸekilde uygulandÄ±ÄŸÄ± ve beklentilere uygun olduÄŸu doÄŸrulanÄ±r. Teknik ve kullanÄ±cÄ± incelemeleri yapÄ±lÄ±r.

#### Uygulama:

* **Uygulama:** YayÄ±nlanan bir deÄŸiÅŸikliÄŸi test edin ve kullanÄ±cÄ± geri bildirimlerini toplayÄ±n. DeÄŸiÅŸikliÄŸin beklentilere uygun olup olmadÄ±ÄŸÄ±nÄ± kontrol edin.

## **YazÄ±lÄ±m GeliÅŸtirme OrtamlarÄ± ve SÃ¼rÃ¼m Kontrol Sistemleri (Software Development Environments and Source Code Version Control System)**

### 1. **YazÄ±lÄ±m GeliÅŸtirme OrtamlarÄ± (Development Environments)**

#### Teorik AÃ§Ä±klama:

YazÄ±lÄ±m geliÅŸtirme, farklÄ± ortamlar arasÄ±nda gerÃ§ekleÅŸtirilir: geliÅŸtirme, test ve Ã¼retim ortamlarÄ±. Her ortam, farklÄ± gÃ¼venlik Ã¶nlemleri ve konfigÃ¼rasyonlar gerektirir.

#### Uygulama:

* **Uygulama:** GeliÅŸtirme ve test ortamlarÄ±nÄ± kurun. Her ortam iÃ§in farklÄ± gÃ¼venlik yapÄ±landÄ±rmalarÄ±nÄ± gÃ¶sterecek bir uygulama geliÅŸtirin.

### 2. **SÃ¼rÃ¼m Kontrol Sistemleri (Version Control Systems)**

#### Teorik AÃ§Ä±klama:

SÃ¼rÃ¼m kontrol sistemleri (Git, SVN vb.), yazÄ±lÄ±m geliÅŸtirme sÃ¼reÃ§lerinin takip edilmesi ve deÄŸiÅŸikliklerin geri alÄ±nabilmesi iÃ§in kullanÄ±lÄ±r. Her deÄŸiÅŸiklik kaydedilir ve versiyonlar arasÄ±nda geÃ§iÅŸ yapÄ±labilir.

#### Uygulama:

* **Uygulama:** GIT kullanarak bir yazÄ±lÄ±m geliÅŸtirme sÃ¼recini yÃ¶netin. FarklÄ± dallar (branch) arasÄ±nda geÃ§iÅŸ yapÄ±n ve bir deÄŸiÅŸikliÄŸi geri alÄ±n.

### 3. **GeliÅŸtirme Sitesi ve Kaynak Kod Sunucu GÃ¼venliÄŸi (Development Site and Source Code Server Security)**

#### Teorik AÃ§Ä±klama:

GeliÅŸtirme ortamÄ±nÄ±n fiziksel ve dijital gÃ¼venliÄŸi Ã§ok Ã¶nemlidir. Kaynak kod sunucularÄ± ve izleme sistemlerinin korunmasÄ±, yazÄ±lÄ±mÄ±n bÃ¼tÃ¼nlÃ¼ÄŸÃ¼nÃ¼ saÄŸlar.

#### Uygulama:

* **Uygulama:** Bir geliÅŸtirme ortamÄ±nda, kaynak kod sunucusunun nasÄ±l gÃ¼venlik altÄ±na alÄ±nacaÄŸÄ±nÄ± gÃ¶sterin. Åžifreleme ve eriÅŸim kontrol sistemlerini uygulamalÄ± olarak yapÄ±landÄ±rÄ±n.

### 4. **Sunucu OdasÄ± ve GeliÅŸtirme BilgisayarlarÄ± GÃ¼venliÄŸi (Development Office and Server Room Security)**

#### Teorik AÃ§Ä±klama:

Sunucu odalarÄ± ve geliÅŸtirme bilgisayarlarÄ±, yazÄ±lÄ±mÄ±n gÃ¼venliÄŸini saÄŸlamak iÃ§in gÃ¼venlik Ã¶nlemleriyle korunmalÄ±dÄ±r. EriÅŸim kontrolleri, ÅŸifreleme ve fiziksel gÃ¼venlik bu sÃ¼reÃ§lerin bir parÃ§asÄ±dÄ±r.

#### Uygulama:

* **Uygulama:** Bir sunucu odasÄ±nÄ±n eriÅŸim kontrollerini simÃ¼le edin. GeliÅŸtirme bilgisayarlarÄ±nda gÃ¼venlik yazÄ±lÄ±mlarÄ±nÄ± yapÄ±landÄ±rÄ±n ve olasÄ± saldÄ±rÄ±lara karÅŸÄ± Ã¶nlemler alÄ±n.

## **HaftanÄ±n Ã–zeti ve Gelecek Hafta**

### Bu Hafta:

* **YazÄ±lÄ±m GeliÅŸtirme AkÄ±ÅŸÄ± ve DeÄŸiÅŸiklik YÃ¶netimi**
* **KonfigÃ¼rasyon Sabitleme ve DeÄŸiÅŸiklik Onaylama**
* **GeliÅŸtirme OrtamlarÄ± ve SÃ¼rÃ¼m Kontrol Sistemleri**
* **Fiziksel ve Dijital GÃ¼venlik**

### Gelecek Hafta:

* **Veri GÃ¼venliÄŸi ve Kriptografi**
* **GÃ¼venli Ä°letiÅŸim ve Anahtar YÃ¶netimi**