

Table of Contents

UCH's Git-Flow	3
<i>Sıfırdan Repo Kurulumu</i>	3
<i>Dallar</i>	3
Ana Dallar	3
Özellik Dallarını	3
Sürüm Dallarını	4
Acil Yama Dallarını (Hotfix Branches)	5

UCH's Git-Flow

Projelerimde git'in gücünü bu iş akışı ile ortaya çıkarıyorum.

Amerikayı yeniden keşfetmeye gerek yok, Vincent Driessen'in [A successful Git branching model](#) mimarisini saf Git komutlarıyla git-flow eklentisi yüklemeyen kullanıyorum.

Sıfırdan Repo Kurulumu

```
# 1. Repoyu başlat
git init
# 2. Main dalının oluşabilmesi için ilk (boş) commit'i at.
git commit --allow-empty -m "<ORNEK_COMMIT_MESAJI>"
# 3. Develop dalını oluştur ve o dala geç.
git checkout -b develop
# 4. Uzak sunucuyu (GitHub, GitLab vb.) bağla.
git remote add origin <UZAK_SUNUCU_URL>
# 5. Her iki dalı da uzak sunucuya gönder.
git push -u origin main
git push -u origin develop
```

Dallar

Ana Dallar

Bu iki dal projenin kalbidir ve asla silinmezler. Üzerlerinde doğrudan düzenleme yapılmaz.

main	her zaman canlıdaki en stabil kodu temsil eder.
develop	Bir sonraki sürüm için hazırlanan en güncel geliştirme kodlarını içerir. Tüm yeni özellikler günün sonunda buraya merge edilir.

Özellik Dallar

Yeni bir özellik veya eklenti geliştirirken kullanılır.

develop'den dallanır, günün sonunda develop dalına birleşir.

İsimlendirme Kuralı: feature-<ÖZELLİK-ADI>

Komutlar

Yeni özellik dalı oluşturma ve geçiş yapma;

```
git checkout -b feature-<OZELLIK-ADI> develop
```

Özelliğin geliřtirmesi bittikten sonra, develop dalına birleřtirmek için;

```
git checkout develop
git merge --no-ff feature-<OZELLIK-ADI>
```

--no-ff bayrađı dalın geçmiřini korur ve commit'leri tek bir düğümde toplar. Çok önemlidir!)

Dalı silme ve güncel develop'u sunucuya göndermek için;

```
git branch -d feature-<OZELLIK-ADI>
git push origin develop
```

Sürüm Dalları

Yeni bir canlı sürüm (production release) hazırlığı için kullanılır. Yeni özellik eklenmez, sadece ufak hatalar giderilir (bugfix) ve versiyon (v1.2.0 vb.) numaraları güncellenir.

develop'den dallanır, günün sonunda main ve develop dalına birleřir.

İsimlendirme Kuralı: release-* (Örn: release-1.2.0)

Komutlar

Sürüm dalı oluřturma

```
git checkout -b release-<SEMANTIK-SÜRÜM-NUMARASI> develop
```

Projelerimde Semantik sürüm numaralandırma sistemini kullanıyorum, detay için;
<https://semver.org/lang/tr/>

Kalan ufak hataları kapatıp versiyon numarasını güncelledikten sonra ¹⁾ sürümü bitirip main'a birleřtirme ve etiketlemek için;

```
git checkout main
git merge --no-ff release-<SEMANTIK-SÜRÜM-NUMARASI>
git tag -a <SEMANTIK-SÜRÜM-NUMARASI> -m "Sürüm <SEMANTIK-SÜRÜM-NUMARASI> yayında"
```

Yapılan ufak hata düzeltmelerini develop'a da yansıtmak için;

```
git checkout develop
```

```
git merge --no-ff release-<SEMANTIK-SÜRÜM-NUMARASI> develop
```

Dalı silip ve tüm değişiklikleri (etiket ile birlikte) uzak sunucuya göndermek için;

```
git branch -d release-<SEMANTIK-SÜRÜM-NUMARASI>
git push origin main
git push origin develop
git push --tags
```

Acil Yama Dalları (Hotfix Branches)

Canlı ortamdaki (production) kritik bir hatayı çok acil çözmek gerektiğinde kullanılır. Sıradaki büyük sürümü bekleyemeyeceğiniz durumlar içindir.

main'den dallanır, günün sonunda main ve develop dalına birleşir.

İsimlendirme Kuralı: hotfix-* (Örn: hotfix-1.2.1)

Komutlar

Acil yama dalı oluşturmak için;

```
git checkout -b hotfix-<SEMANTIK-SÜRÜM-NUMARASI> main
```

Kritik hatayı düzelterip, versiyonu güncelledikten sonra commitleyip yamayı bitirip main'e birleştirmek için;

```
git checkout main
git merge --no-ff hotfix-<SEMANTIK-SÜRÜM-NUMARASI>
git tag -a <SEMANTIK-SÜRÜM-NUMARASI> -m "<ORNEK_COMMIT_MESAJI>"
```

Gelecek sürümlerde aynı hatanın tekrar etmemesi adına yamayı develop'a yansıtmak için;

```
git checkout develop
git merge --no-ff hotfix-<SEMANTIK-SÜRÜM-NUMARASI>
```

Dalı silip ve tüm değişiklikleri (etiket ile birlikte) uzak sunucuya göndermek için;

```
git branch -d hotfix-<SEMANTIK-SÜRÜM-NUMARASI>
git push origin main
git push origin develop
git push --tags
```

[UCH Viki](#)'den alınmıştır.

1)

Genelde projelerimde çeşitli scriptlerle semantik sürümü otomatik olarak arttırıyorum.

From:

<https://wiki.ulascemh.com/> - **UCH**

Permanent link:

<https://wiki.ulascemh.com/doku.php?id=tr:cs:devtools:git:flow>

Last update: **2026/04/05 12:35**

