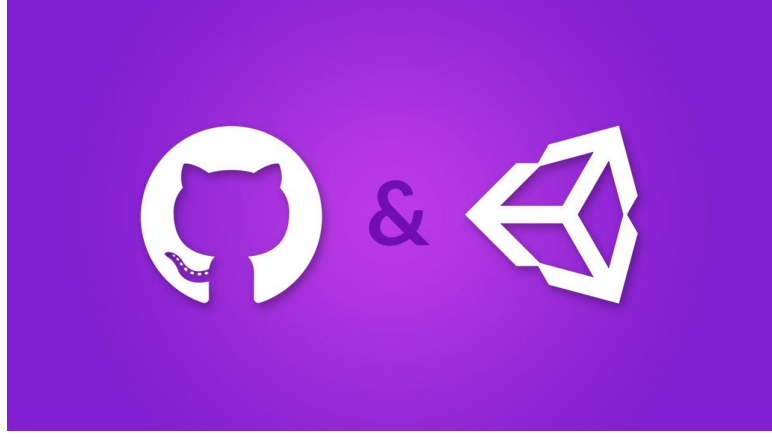




UNITY-GITHUB KULLANIMI

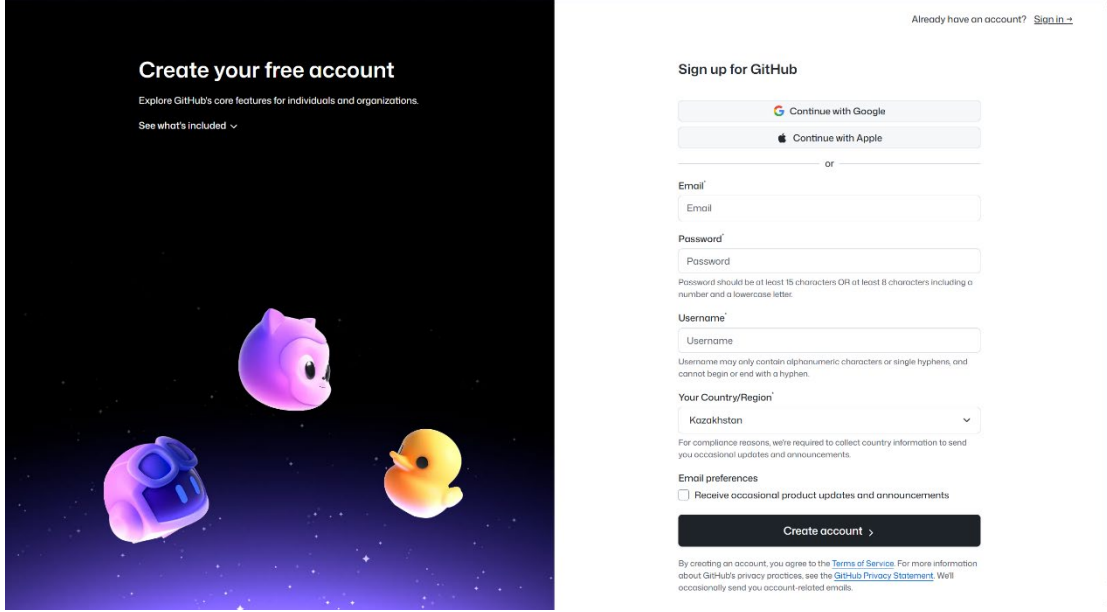
UNITY-GITHUB ENTEGRASYONU

EMRE ERSOY | 3B OYUN TASARIMI | 16.10.2025

GITHUB'A ÜYE OLMAK

1) <https://github.com>

2) Linke tıklayın ve githuba gidin. Sitenin sağ üst kısmındaki Sign up butonuna tıklayın.



Already have an account? [Sign in](#)

Sign up for GitHub

[Continue with Google](#)

[Continue with Apple](#)

or

Email

Password

Password should be at least 15 characters OR at least 8 characters including a number and a lowercase letter.

Username

Username may only contain alphanumeric characters or single hyphen, and cannot begin or end with a hyphen.

Your Country/Region

For compliance reasons, we're required to collect country information to send you occasional updates and announcements.

Email preferences

☐ Receive occasional product updates and announcements

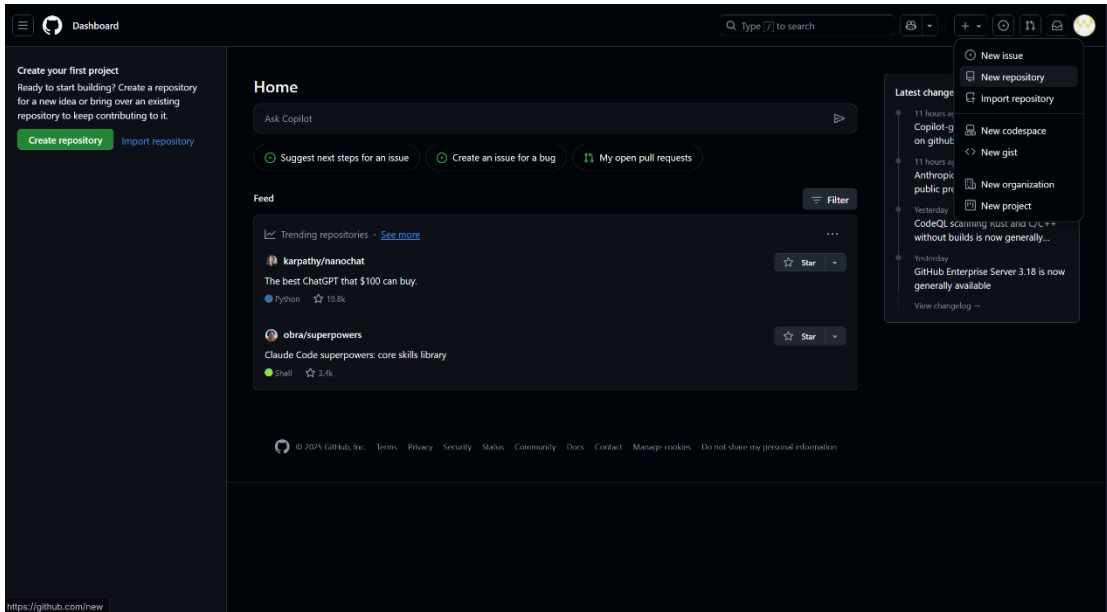
[Create account](#)

By creating an account, you agree to the [Terms of Service](#). For more information about GitHub's privacy practices, see the [GitHub Privacy Statement](#). We'll occasionally send you account-related emails.

REPOSITORY OLUŞTURMA

1) Sitenin sağ üstünde kısmında bulunan “+” işaretine tıklayın.

2) Açılan pencereden “New Repository” seçeneğine tıklayın.



3)Açılan sayfada;

Repository'nin sahibini

Adını

Açıklamasını

Görünürlük ayalarını

README dosyasını aktifleştirme deaktifleştirme

Gitignore dosyası oluşturma

Lisans ekleme ayarlarını bulabilirsiniz.

Şimdilik görünürlük ayarını "Private" yapalım. Sonradan grup arkadaşlarımızın görmesini sağlayacağız.

Create a new repository
Repositories contain a project's files and version history. Have a project elsewhere? [Import a repository](#)
Required fields are marked with an asterisk (*).

1 General

Owner * WhoIsStanK / Repository name * Repo Adi
✓ Your new repository will be created as Repo-Adi.
The repository name can only contain ASCII letters, digits, and the characters -, ., and _.

Great repository names are short and memorable. How about [laughing-disco](#)?

Description
Açıklama
8 / 350 characters

2 Configuration

Choose visibility *
Choose who can see and commit to this repository

Public Public
Private

Public
Anyone on the internet can see this repository. You choose who can commit.

Private
You choose who can see and commit to this repository.

Add README
READMEs can be used as longer descriptions. [About READMEs](#)

Add .gitignore
.gitignore tells git which files not to track. [About repository files](#)

Add license
Licenses explain how others can use your code. [About licenses](#)

No license

Create repository

4)README ekleme ayarlarını buradan yapabilirsiniz. README açıklama bölümünün yetersiz kalacağı durumlarda uzun açıklamalar yazacaksanız kullanabileceğiniz bir bölümdür.

Add README
READMEs can be used as longer descriptions. [About READMEs](#)

On ☒

5)Gitignore ayarlarını bu bölümden yapabilirsiniz. Templateler arasından Unity'i seçelim. Gitignore otomatik olarak çok fazla kullanmayacağınız dosyaları githuba aktarmamanızı sağlar.

Create a new repository

Repositories contain a project's files and version history. Have a project elsewhere? [Import a repository](#).

Required fields are marked with an asterisk (*).

1

General

Owner *

WholsStanX

Repository name *

Repo Adi

✓ Your new repository will be created as Repo-Adi.
The repository name can only contain ASCII letters, digits, and the character hyphen (-).

Great repository names are short and memorable. How about [laughing-disco](#)?

Description

Acıklama

8 / 350 characters

2

Configuration

Choose visibility *

Choose who can see and commit to this repository

Add README

READMEs can be used as longer descriptions. [About READMEs](#)

Add .gitignore

.gitignore tells git which files not to track. [About ignoring files](#)

Unity

Add license

Licenses explain how others can use your code. [About licenses](#)

No license

Create repository

6)Lisans ayarlarını buradan yapabilirsiniz. Lisans yaptığınız projenin yazdığınız kodların lisanslanmasını sağlar . Diğer insanların ne amaçla kodlarınızı, projenizi kullanacağını izinlerini buradan ayarlayabilirsiniz. Lisans türlerini araştırarak öğrenebilirsiniz.

Create a new repository

Repositories contain a project's files and version history. Have a project elsewhere? [Import a repository](#).

Required fields are marked with an asterisk (*).

1 General

Owner * WholsStanX / **Repository name *** Repo Adi

✓ Your new repository will be created as Repo-Adi.
The repository name can only contain ASCII letters, digits, and the character hyphen (-).

Great repository names are short and memorable. How about [laughing-disco](#)?

Description

Aciklama

8 / 350 characters

2 Configuration

Choose visibility *
Choose who can see and commit to this repository

Add README
READMEs can be used as longer descriptions. [About READMEs](#)

Add .gitignore
.gitignore tells git which files not to track. [About ignoring files](#)

Add license
Licenses explain how others can use your code. [About licenses](#)

Choose a license

Search licenses

✓ No license

Apache License 2.0

GNU General Public License v3.0

MIT License

BSD 2-Clause "Simplified" License

BSD 3-Clause "New" or "Revised" License

Boost Software License 1.0

Creative Commons Zero v1.0 Universal

Eclipse Public License 2.0

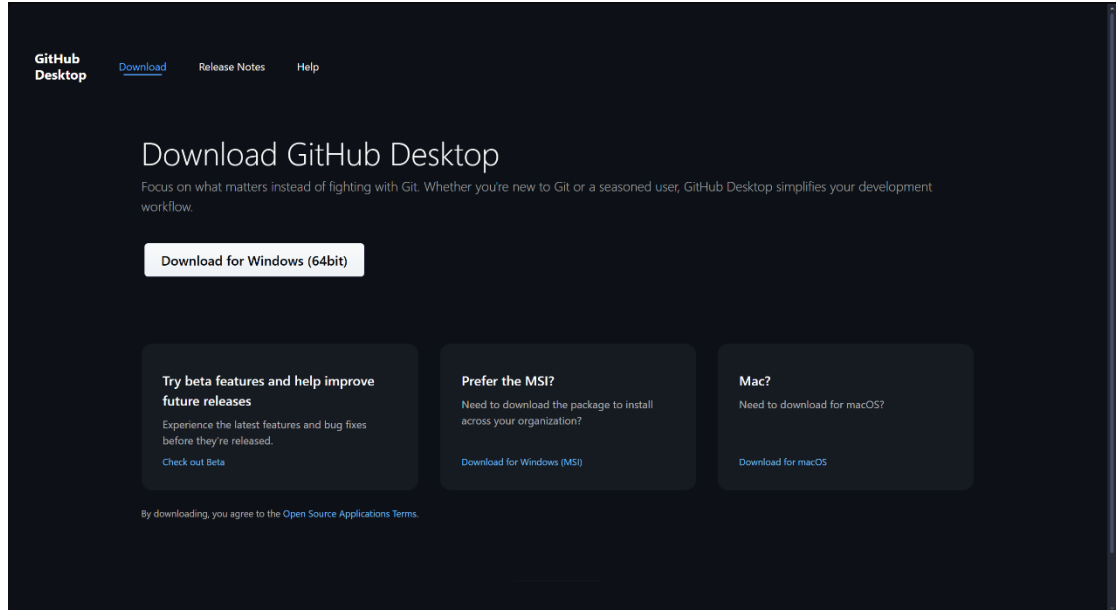
GNU Affero General Public License v3.0

No license

Create repository

GITHUB DESKTOP KURULUMU

- 1) <https://desktop.github.com/download>
- 2)Bağlantıya tıklayarak indirme sayfasına gidin.
- 3)Download for Windows(64bit) seçeneğine tıklayarak indirmeyi başlatın. (Mac için sağ aşağıda bulunan “Mac?” Bölümünden devam edin.)



4)İndirme işlemi bittikten sonra “Sign in with GitHub.com” butonuna tıklayıp site yönlendirmesiyle giriş yapın.



Welcome to GitHub Desktop

GitHub Desktop is a seamless way to contribute to projects on GitHub and GitHub Enterprise Server. Sign in below to get started with your existing projects.

New to GitHub? [Create your free account.](#)

[Sign in with GitHub.com](#) [Sign in to Enterprise](#)

Username or email address

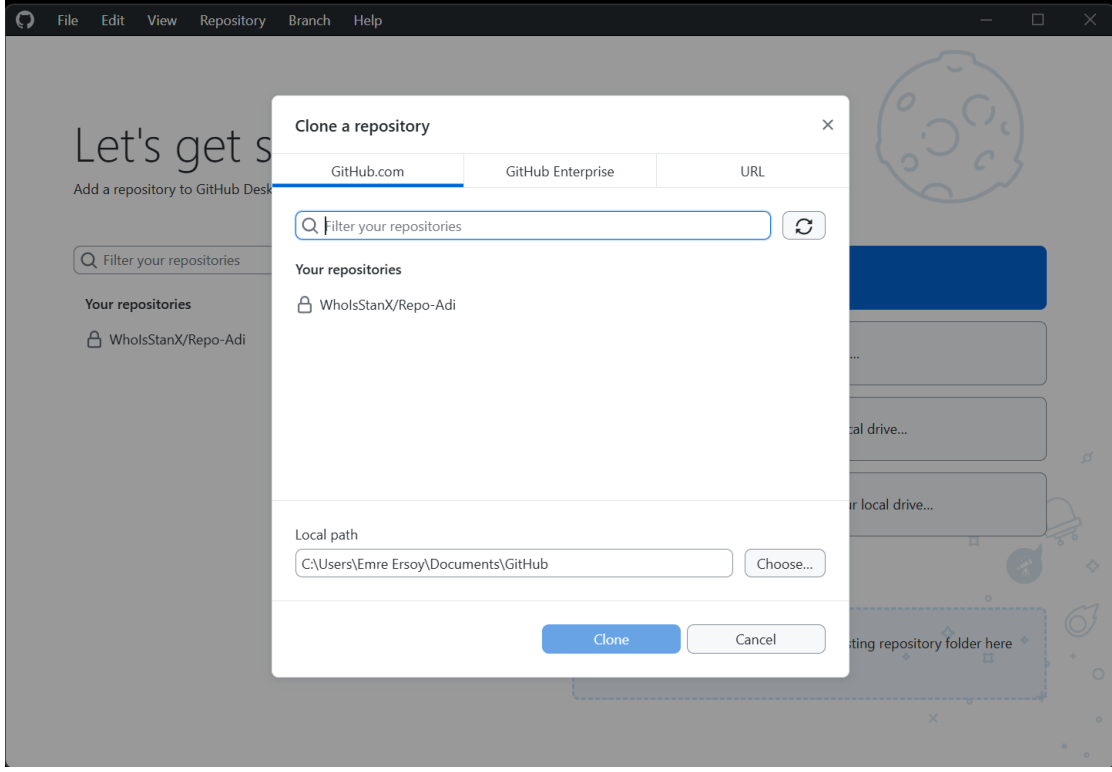
Password

[Sign in](#) [Forgot password](#)

[Skip this step](#)

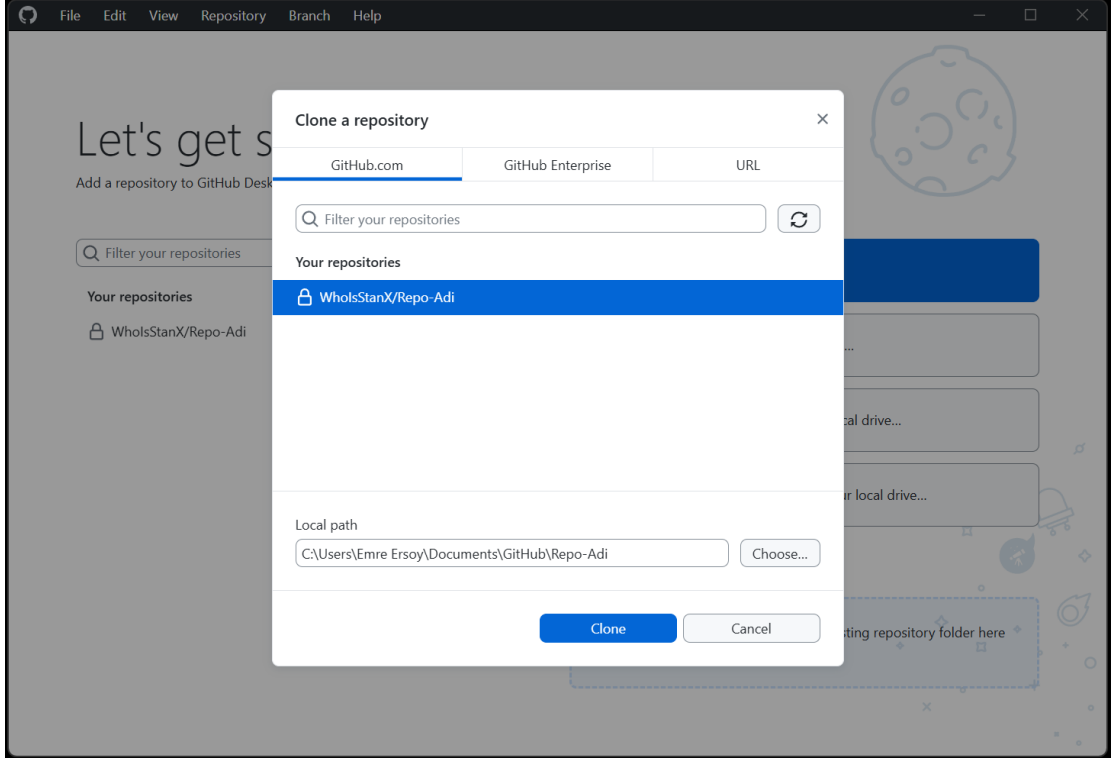


5)Giriş yaptıktan sonra “Clone a Repository from the Internet...” seçeneğine tıklayın.



6)Site üzerinden oluşturduğunuz Repository seçin.

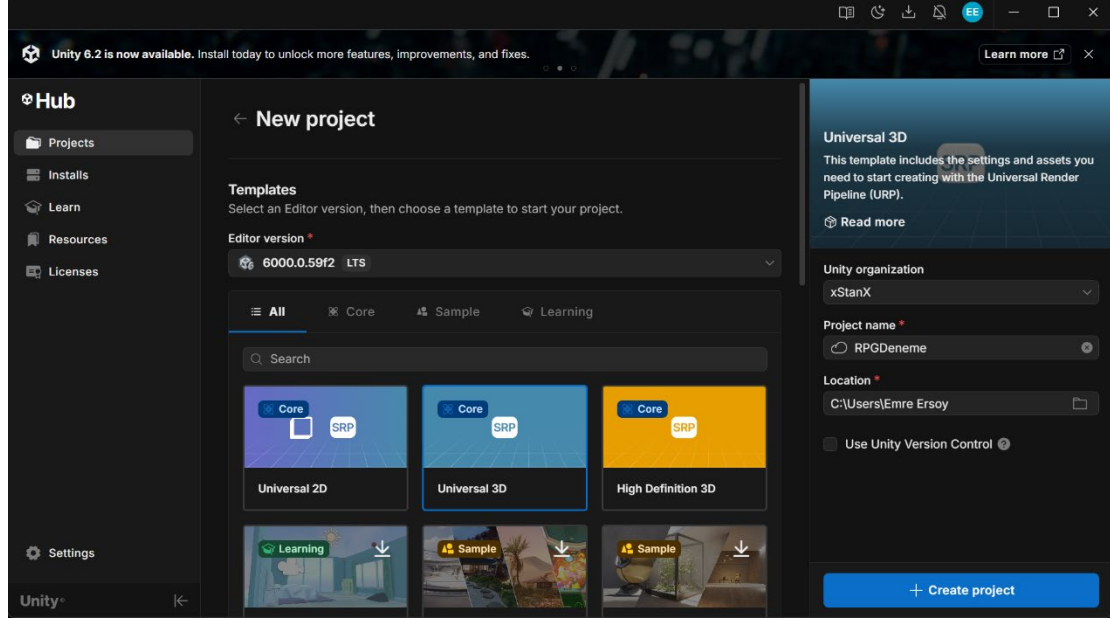
7)Local path seçeneğinin yanındaki “Choose...” kısmından oluşturulacak dosyanın lokasyonunu seçebilirsiniz.



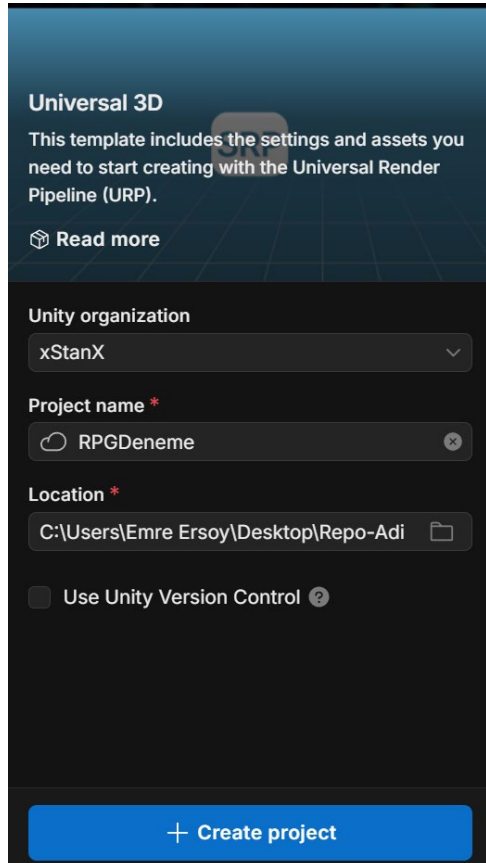
UNITY PROJE KURULUMU

1)Yapacağınız projeye uygun Unity template'ini seçin.

2)Porjenizi isimlendirin.

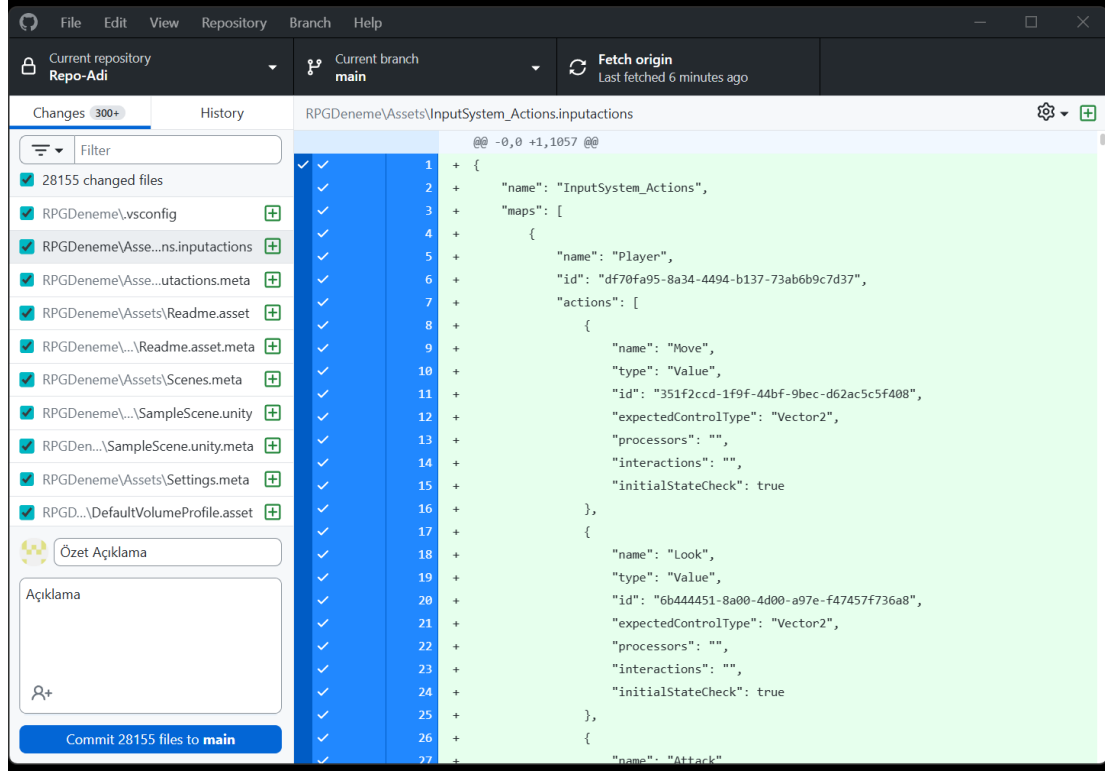


3)Location bölümünden proje lokasyonunu github tarafından oluşturulan dosyayı seçin.



4)Unity projesi açıldıktan sonra “GitHub Desktop” üzerinden dosya içindeki değişiklikleri anlık olarak görebileceksiniz.

5) “GitHub Desktop” üzerinden proje üzerinde yaptığınız değişiklikleri “Özet Açıklama” ve “Açıklama” şeklinde diğer grup arkadaşlarınız ve kendinizin daha kolay hatırlaması için yorum ekleyebiliyorsunuz.



6)Unity projenizi oluşturduğunuz dosyaya gidin.

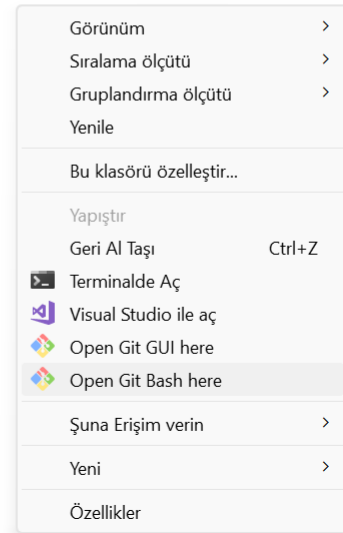
7)İçerisinde .gitignore adlı dosya göreceksiniz. Bu dosyayı Assets,Library,Logs vb. dosyaların bulunduğu kısma atacaksınız.

8).gitignore gereksiz dosyaların GitHub üzerinden senkron olmasını engeller. Süreci hızlandırır.

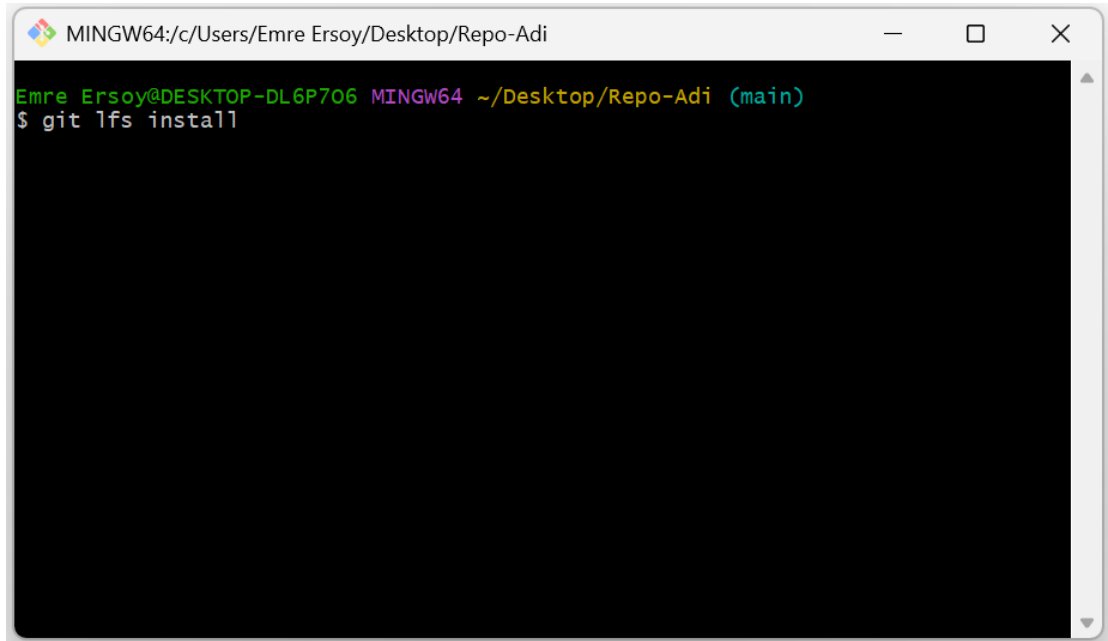
Ad	Değiştirme tarihi	Tür	Boyut
RPGDeneme	16.10.2025 07:30	Dosya klasörü	
README.md	16.10.2025 07:27	MD Dosyası	1 KB
.gitignore	16.10.2025 07:27	txtfile	3 KB

9)100MB upload sınırı hatasını almamak için GitLFS kurulumu yapacağız. GitLFS projenizdeki büyük dosyaların saklanması sağlar. Bunun için Unity projemizin olduğu dosyaya gidiyoruz. Boş bir alana sağ tık yapıyoruz.

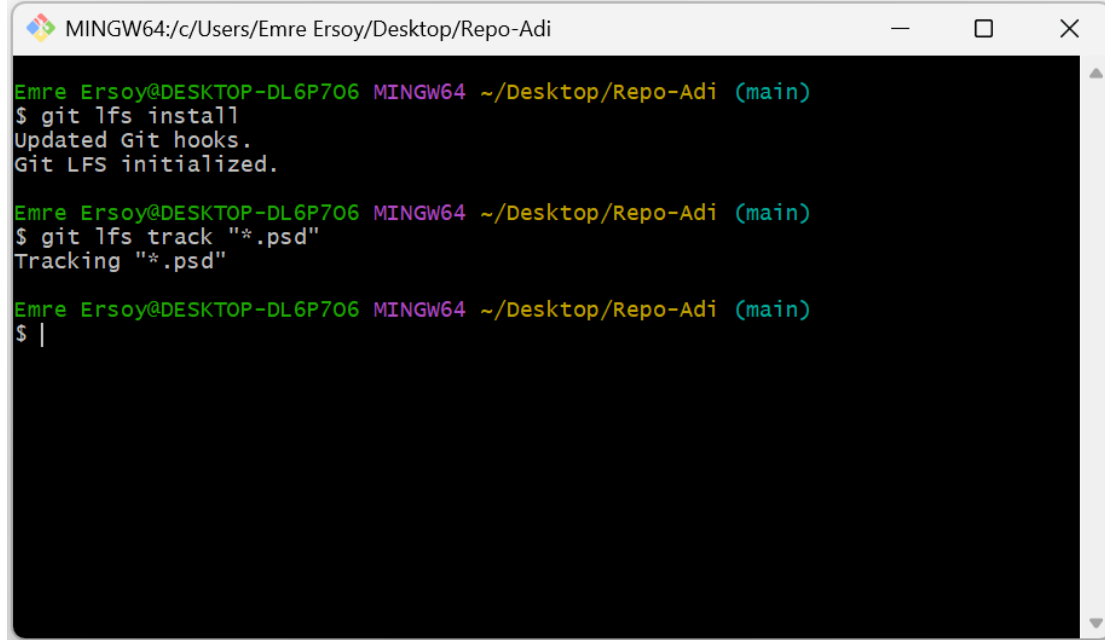
Ad	Değiştirme tarihi	Tür	Boyut
.git	16.10.2025 07:32	Dosya klasörü	
RPGDeneme	16.10.2025 07:47	Dosya klasörü	
README.md	16.10.2025 07:27	MD Dosyası	1 KB



10)Açılan pencereden “Open Git Bash here” seçeneğine tıklıyoruz



11) MAC kullananlar 12. Aşamadan devam edecek. Açılan pencere “git lfs install” yazıp ENTER’a basıyoruz.

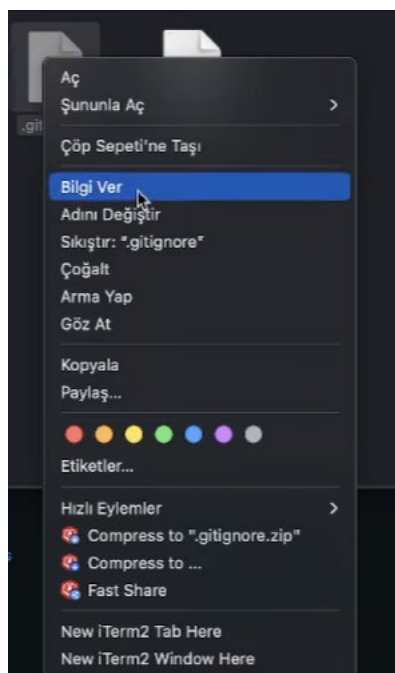


```
MINGW64:/c/Users/Emre Ersoy/Desktop/Repo-Adi
Emre Ersoy@DESKTOP-DL6P706 MINGW64 ~/Desktop/Repo-Adi (main)
$ git lfs install
Updated Git hooks.
Git LFS initialized.

Emre Ersoy@DESKTOP-DL6P706 MINGW64 ~/Desktop/Repo-Adi (main)
$ git lfs track "*.psd"
Tracking "*.psd"

Emre Ersoy@DESKTOP-DL6P706 MINGW64 ~/Desktop/Repo-Adi (main)
$ |
```

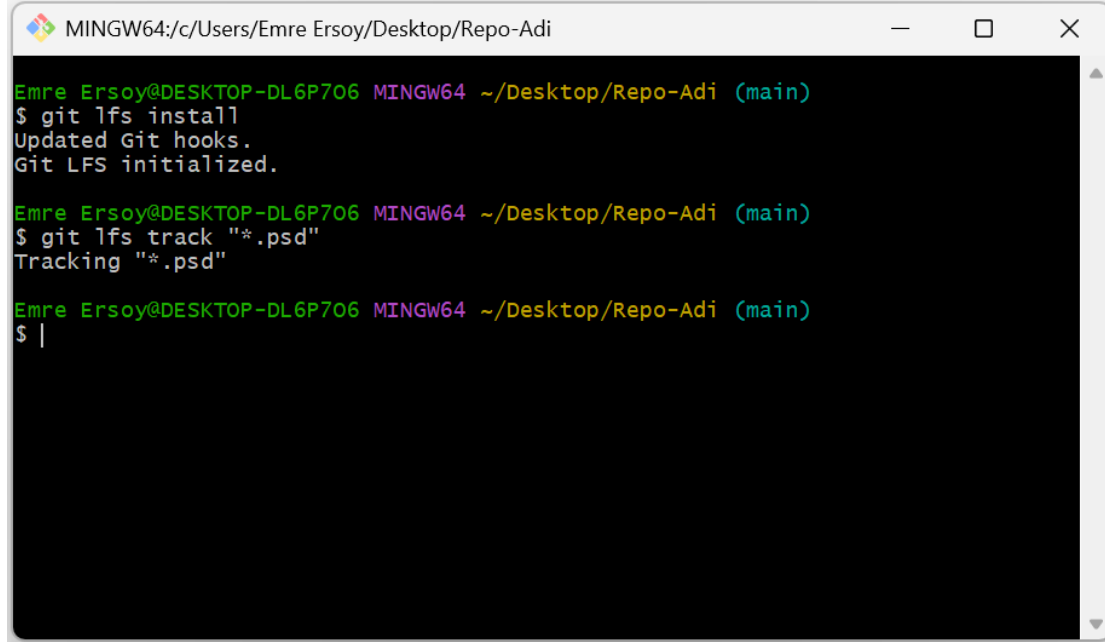
12)Unity proje dosyamızdaki herhangi bir dosyaya sağ tıklayıp “Bilgi Ver” seçeneğine tıklıyoruz. Ardından açılan pencerede “Yer” kısmında dosyamızın bulunduğu konumun adresini seçip sağ tıklayıp “Yol Adı Olarak Kopyala” seçeneğini seçiyoruz. Daha sonra önceden açtığımız GitLFS penceresine dönüp “cd” yazıp yer kısmından kopyaladığımız adresi yapıştırıyoruz CTRL-V. ENTER basıp devam ediyoruz. Sonraki aşama olarak “git lfs install” yazıp ENTER’a basıyoruz.





```
-zsh
Last login: Sat Oct 21 13:30:25 on ttys000
(base) orkun.hacilar@orkun-mbp ~ % cd /Users/orkun.hacilar/Documents/GitHub/Chess40
(base) orkun.hacilar@orkun-mbp Chess40 % git lfs install
zsh: command not found: git
(base) orkun.hacilar@orkun-mbp Chess40 % git lfs install
Updated Git hooks.
Git LFS initialized.
(base) orkun.hacilar@orkun-mbp Chess40 %
```

13) Aynı pencere üzerinden istediğimiz uzantılı dosyaları GitLFS üzerinden depolayabiliyoruz. Örnek olarak Photoshop uzantılı dosyaları otomatik olarak depolamasını sağlayalım. Pencere içine git lfs track "*.psd" yazıp ENTER'a basalım. Artık .psd uzantılı dosyalar otomatik olarak GitLFS üzerinden depolanacaktır.



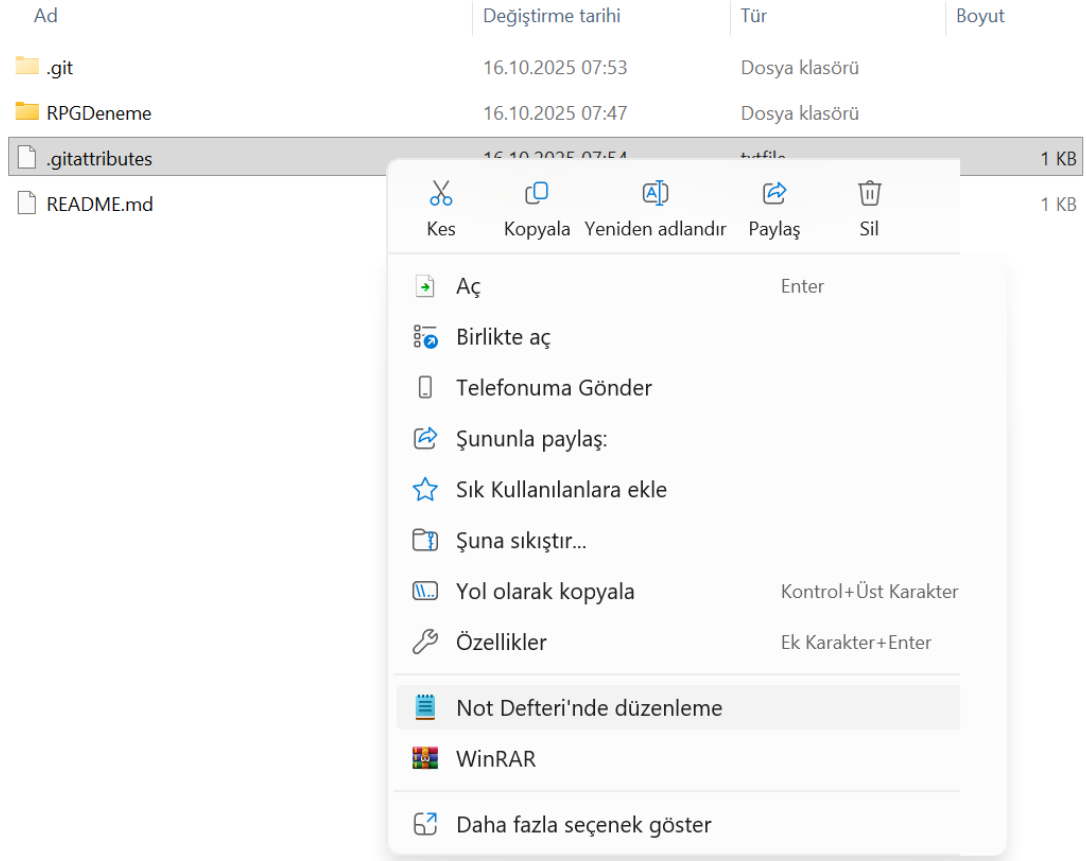
```
MINGW64:/c/Users/Emre Ersoy/Desktop/Repo-Adi
Emre Ersoy@DESKTOP-DL6P706 MINGW64 ~/Desktop/Repo-Adi (main)
$ git lfs install
Updated Git hooks.
Git LFS initialized.

Emre Ersoy@DESKTOP-DL6P706 MINGW64 ~/Desktop/Repo-Adi (main)
$ git lfs track "*.psd"
Tracking "*.psd"

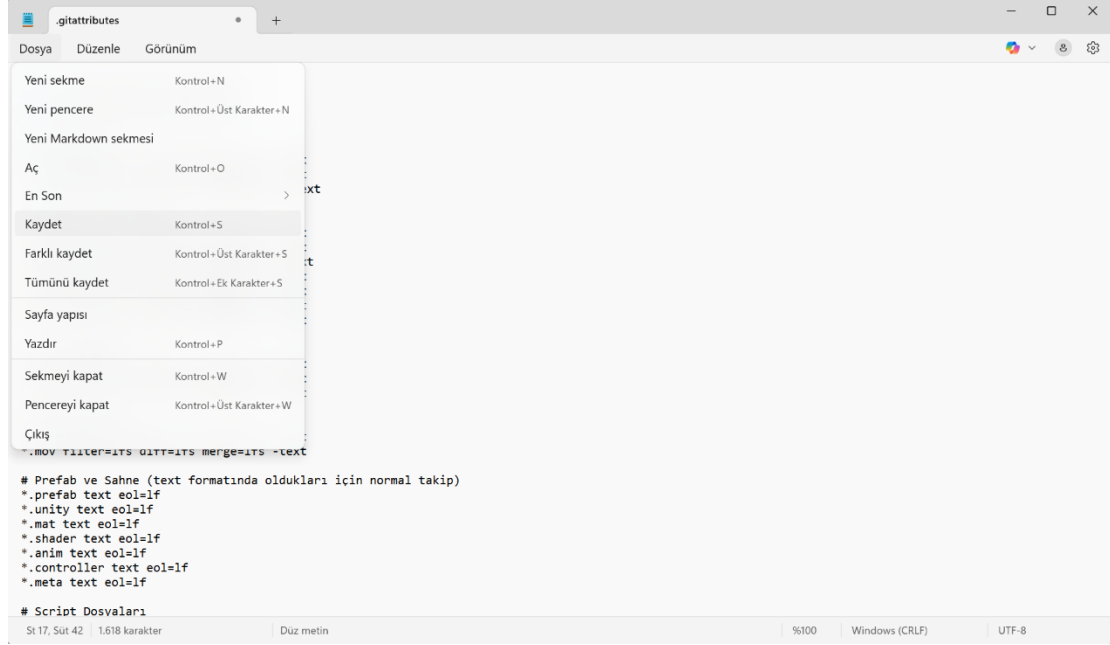
Emre Ersoy@DESKTOP-DL6P706 MINGW64 ~/Desktop/Repo-Adi (main)
$ |
```

14) Her bir uzantıyı tek tek eklemekle uğraşmamanız için Unity projelerinde çoğunlukla kullanılan uzantıları sizin için düzenledim. Bu dosyayı sınıf whatsapp grubuna atıyorum.

15)Bu dosyayı kullanmak için Unity proje dosyamıza geri dönüyoruz. Dosyaların içinde .gitattributes adlı bir dosya oluşmuş olacaktır. Bu dosyaya sağ tıklayıp “Not Defteri’nde düzenleme” seçeneğine tıklıyoruz.

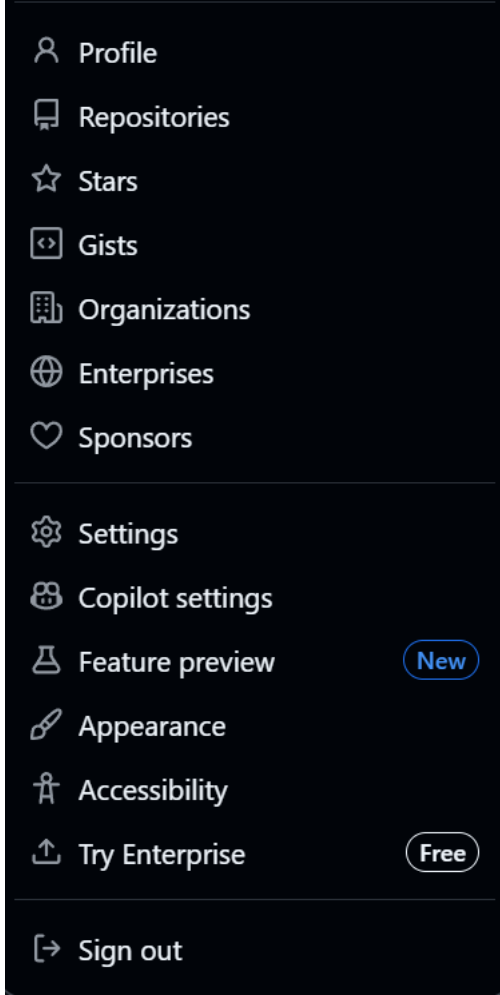


16) Açılan pencerede olan her şeyi silebilirsiniz. Sonrasında benim size verdiğim not defterinde yazılı olan her şeyi kopyalayıp buraya yapıştırıyorsunuz. Sonrasında sol üstteki “Dosya” bölümüne tıklayıp “Kaydet” seçeneğine basarak dosyayı kaydederek bu aşamayı da tamamlamış oluyoruz.



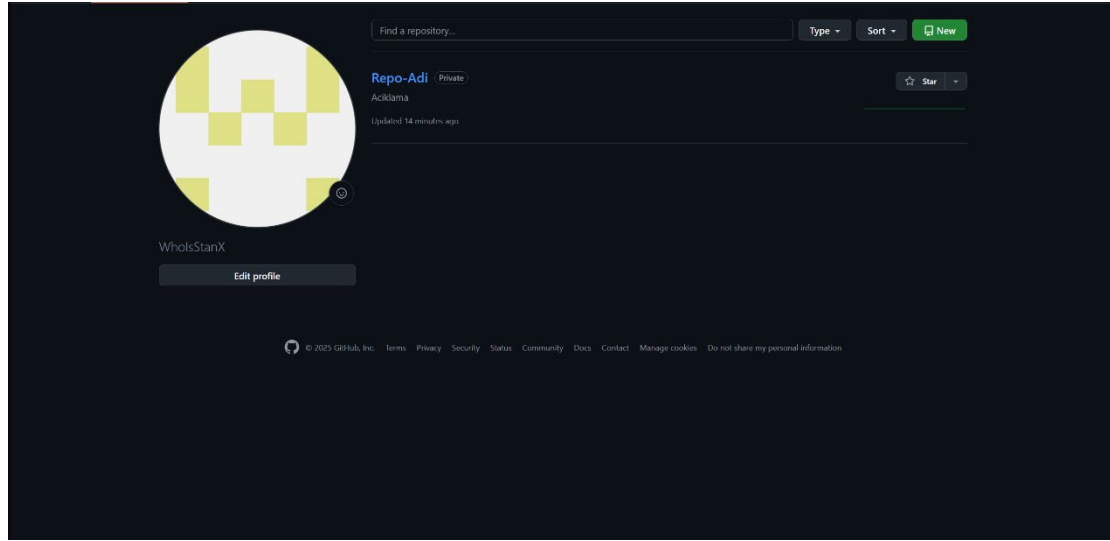
GITHUB ÜZERİNDEN GRUPÇA ÇALIŞMAK

- 1) <https://github.com> siteye geri dönün.
- 2) Sağ üstteki profil fotoğrafınıza tıklayın.
- 3) Açılan pencereden “Repositories” bölümüne tıklayın.

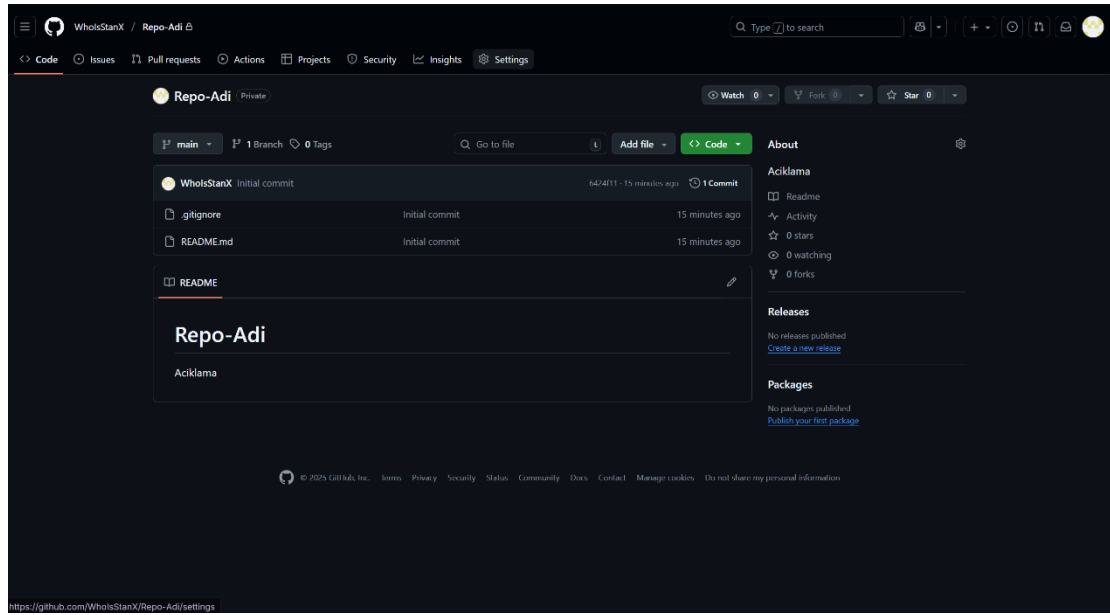


- 4) Önceden oluşturduğunuz “Repository”ler burada gözükecektir.

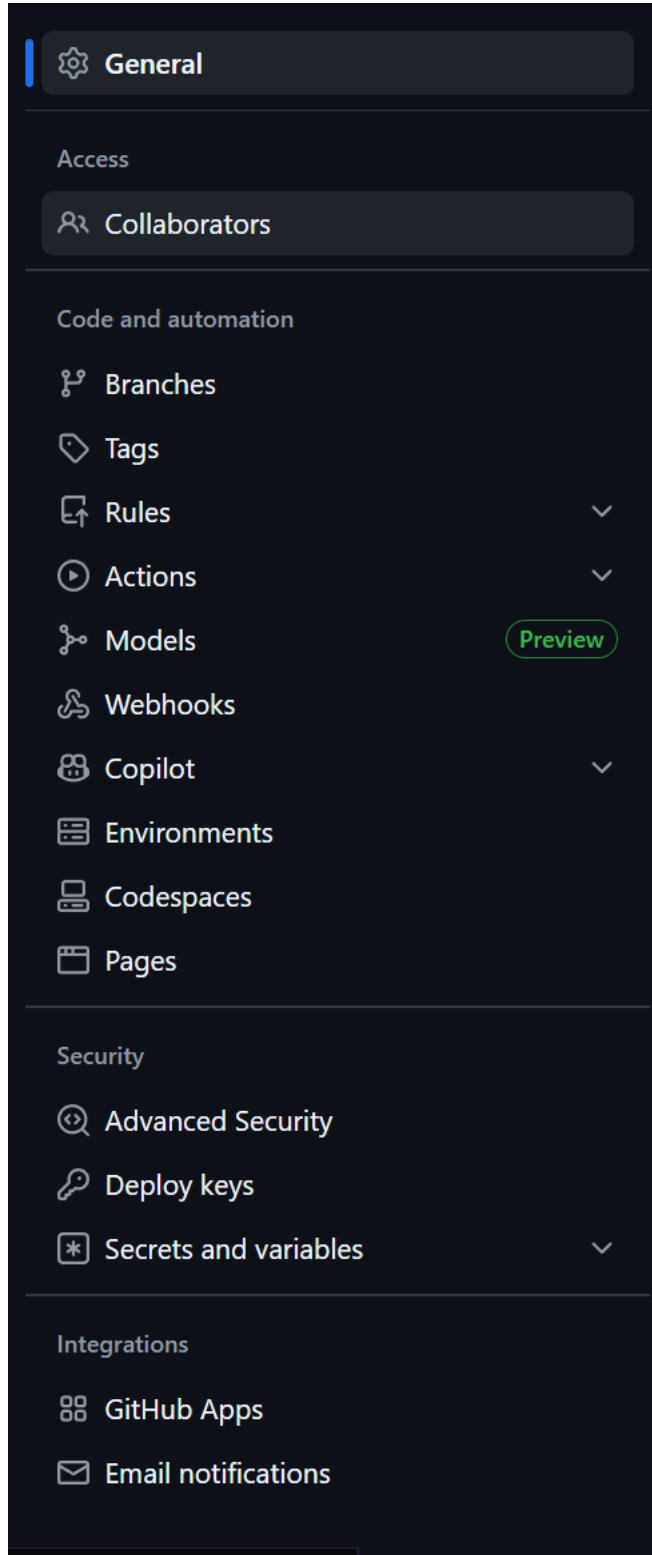
5)Grupça çalışmak istediğiniz “Repository”e tıklayın.



6)Açılan sayfada “Settings” bölümüne tıklayın.



7)Açılan sayfada sol kısımda “Access” bölümünden “Collaborators” sekmesine tıklayın.



8)Açılan sayfada “Manage Access” kısmından “Add People” kısmına tıklayın.

Collaborators and teams

**Private repository**
Only those with access to this repository can view it

Manage visibility

Direct access

0 collaborators have access to this repository. Only you can contribute to this repository.

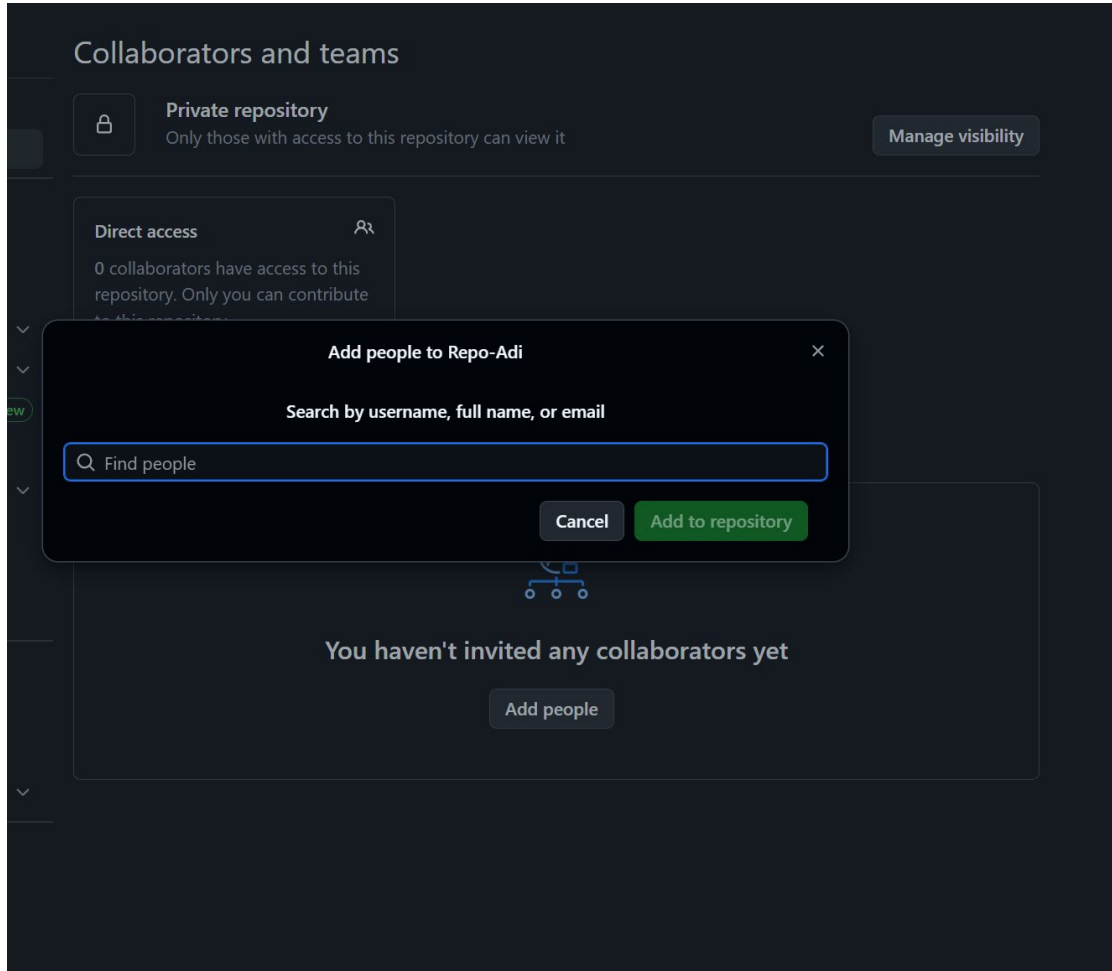
Manage access



You haven't invited any collaborators yet

Add people

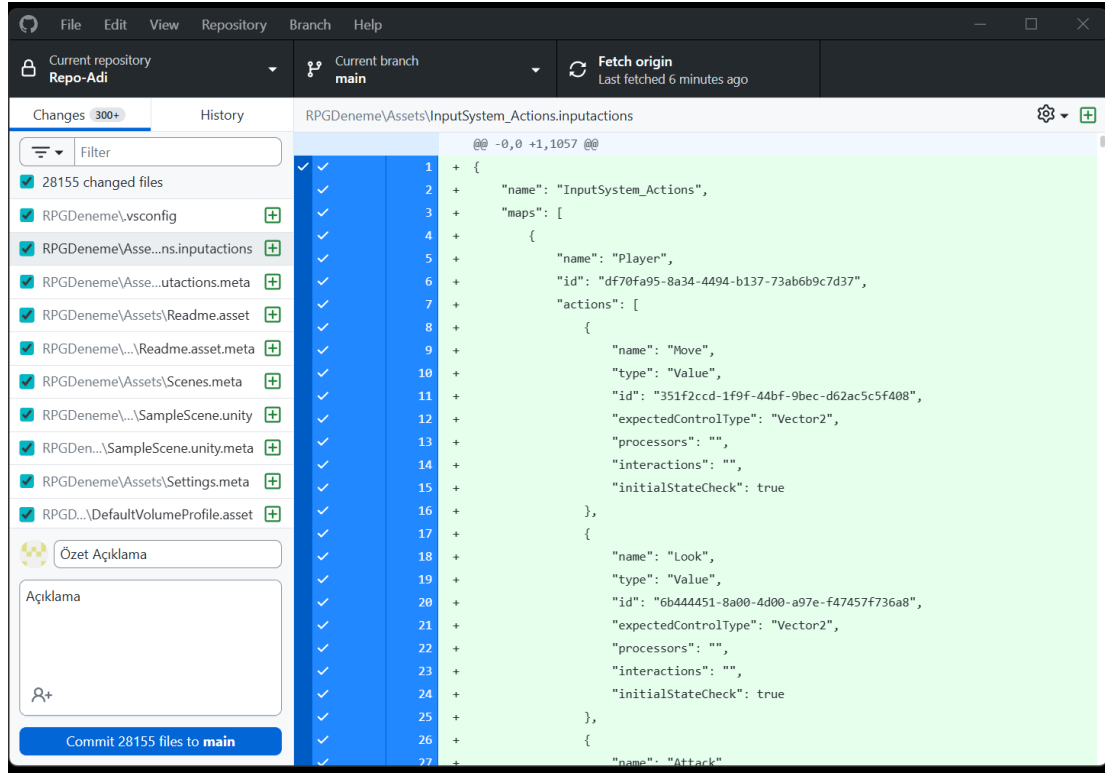
9) Projeye davet etmek istediğiniz grup arkadaşınızı aratın ve seçin.

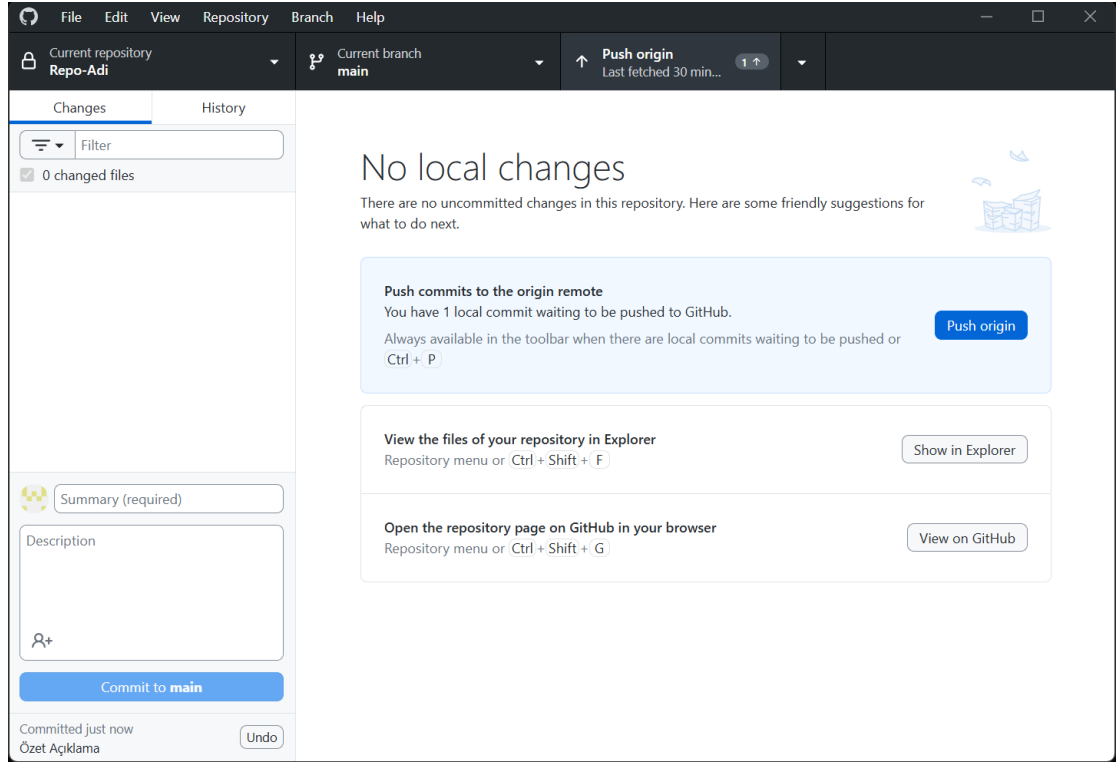


10) Repoya davet edilen grup arkadaşınız daveti GitHub'a kayıt olduğu mail üzerinden kabul ediyor.

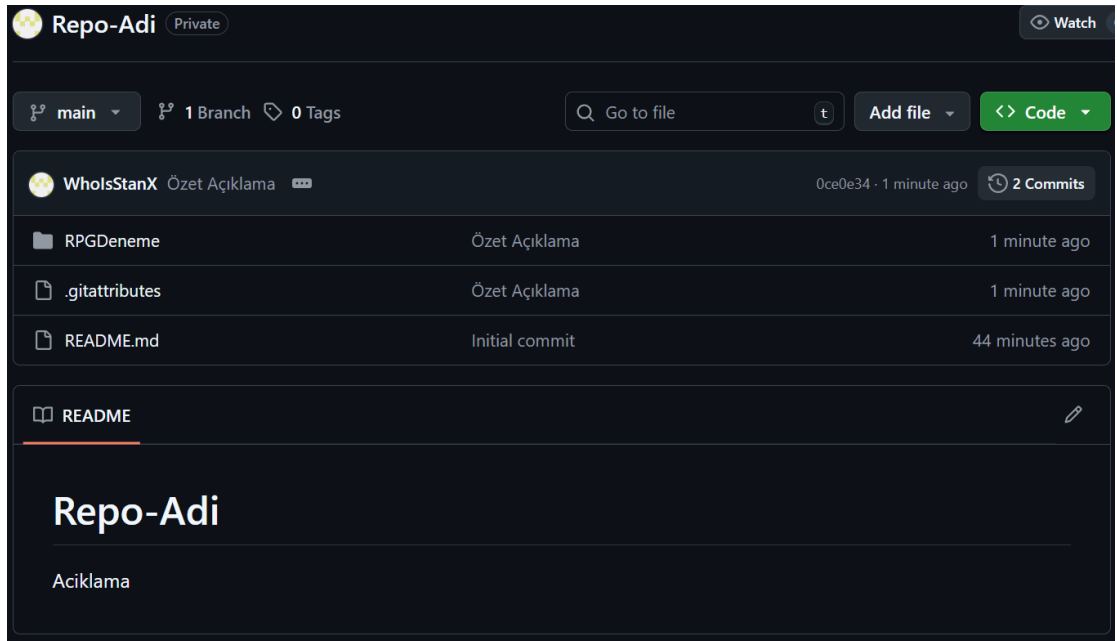
PROJE DOSYALARINI GİTHUBA GÖNDERMEK-PUSHLAMAK

1)GitHub Desktop uygulaması üzerinden sol alt kısımdan kısa bir başlık ve kısa açıklamalarımızı ekledikten sonra sağ üst köşeden “Fetch origin” bölümüne tıklıyoruz. Biraz bekledikten sonra aynı kısım “Push origin” e dönüyor ve oraya tıklıyoruz. Dosya değişikliklerine göre bu aşamanın uzunluğu değişiklik gösteriyor.





2)GitHub.com'a dönerek dosya değişikliklerini arkadaşlarınızın hangi dosyayı eklerken hangi açıklamalarla bu değişiklikleri kaydettiklerini "Repositories" kısmına girip "Commits" kısmından görebilirsiniz.



Commits

main

All users

All time

Commits on Oct 16, 2025

Özet Açıklama

WhoIsStanX committed 2 minutes ago

0ce0e34



Initial commit

WhoIsStanX authored 45 minutes ago

Verified

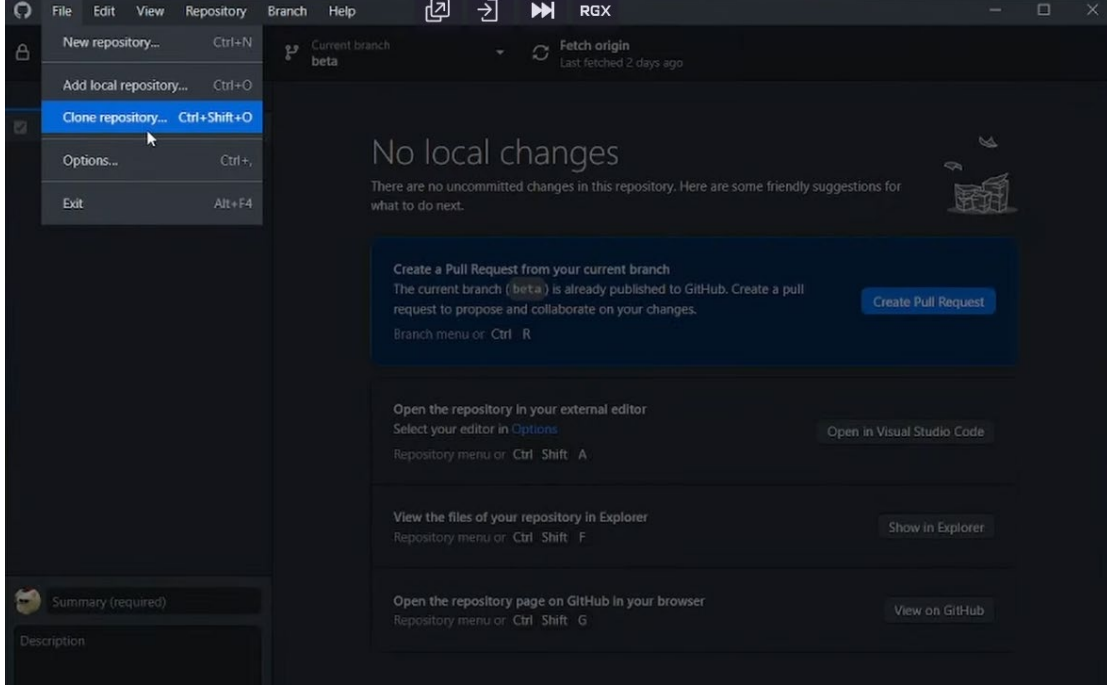
6424f11



DİĞER GRUP ÜYELERİNİN YAPMASI GEREKENLER

1)GitHub Desktop'a giriyoruz.

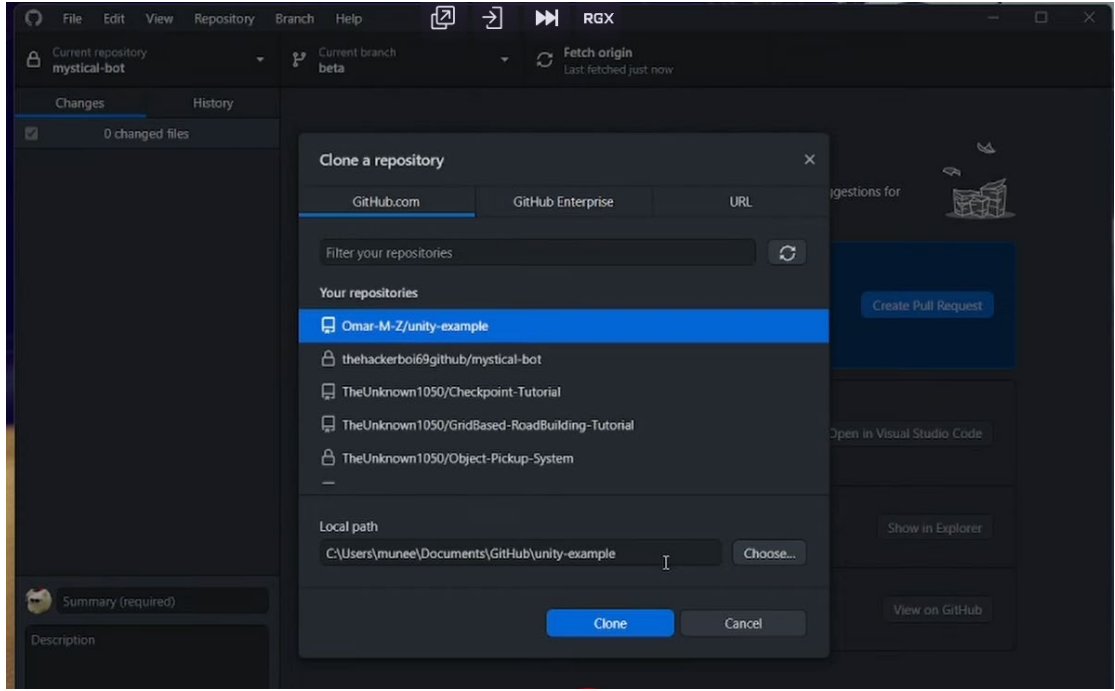
2)Sol üst kısımdan “File” a tıklıyoruz. Açılan pencereden “Clone repository...” kısmına tıklıyoruz.



3)Açılan pencereden “GitHub.com” kısmını seçip. Grup liderinin sizi davet ettiği Repository’i seçiyorsunuz.

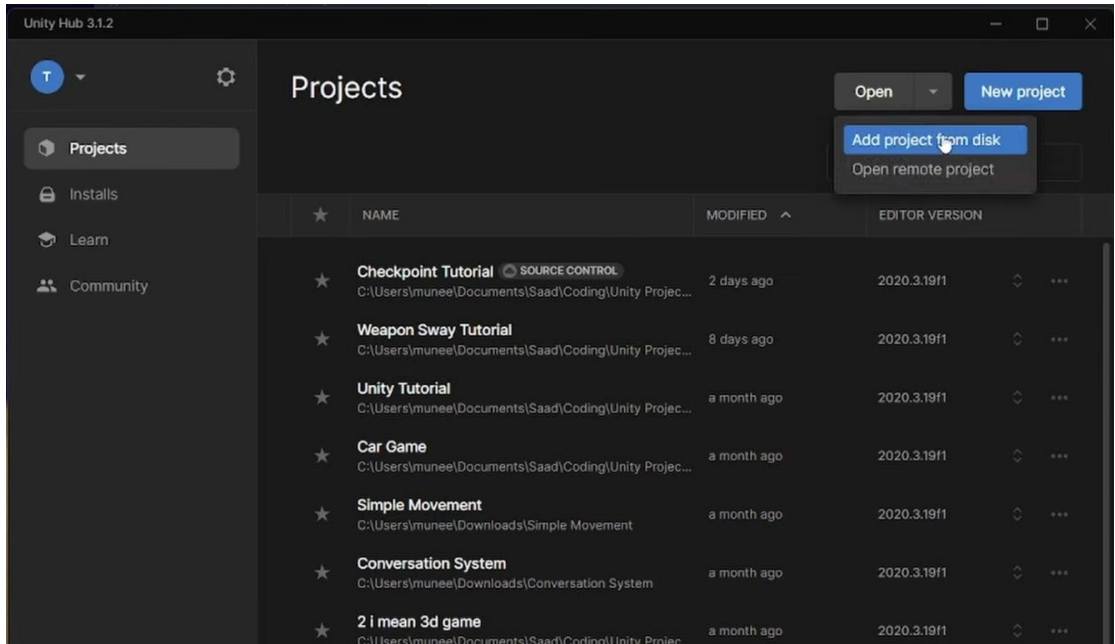
4)Alt kısımdaki “Local path” bölümünden proje liderinin paylaştığı dosyaları kendi bilgisayarınızda nereye klonlamak istediğinizi seçiyorsunuz.

5)”Clone” butonuna tıklayarak paylaşılan dosyayı klonlamayı başlatıyorsunuz.

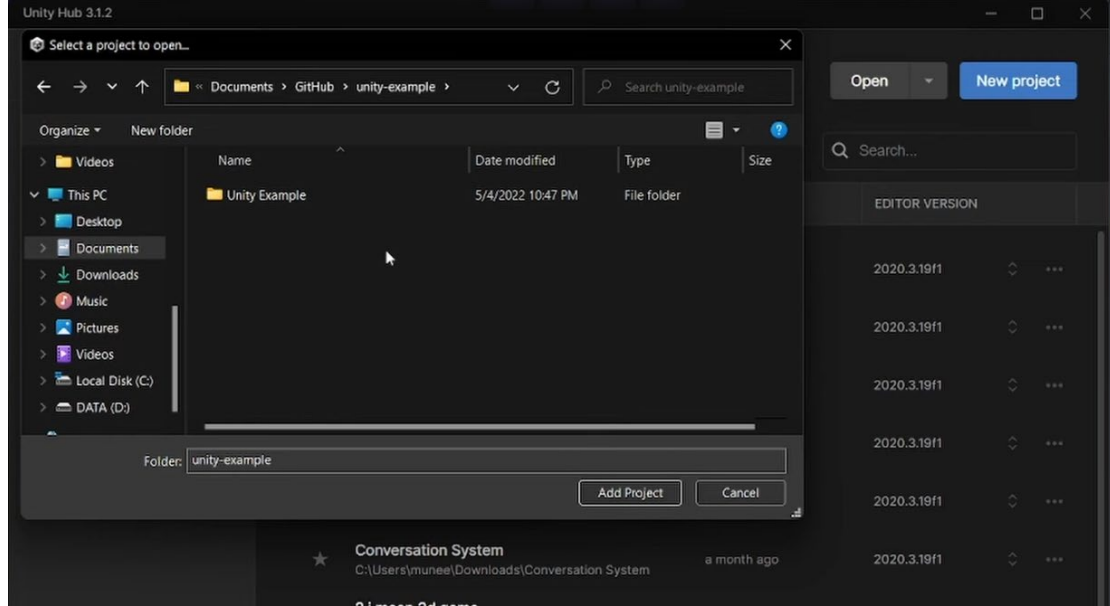


PAYLAŞILAN UNITY PROJESİNİ UNITY HUB'A EKLEMEK

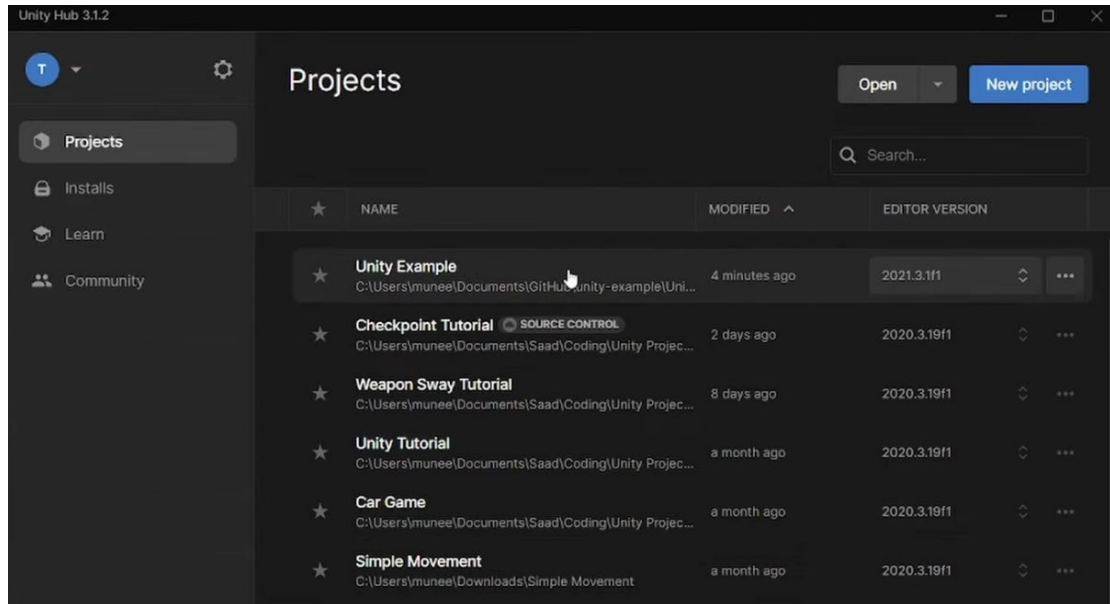
- 1)UnityHub'ı açın.
- 2)Sol kısımdan “Projects” kısmına tıklayın.
- 3)Sağ üst kısımda bulunan “Open” butonunun yanındaki aşağı ok a tıklayın.
- 4)”Add Project from disk” seçeneğine tıklayın.



5)GitHub üzerinden klonladığınız dosyayı seçin ve “Add Project” butonuna tıklayın.(Hata vermemesi için Assets,Library vb. dosyaların bulunduğu kısımdan bir önceki dosya yolunu seçin ve add project’e tıklayın.)

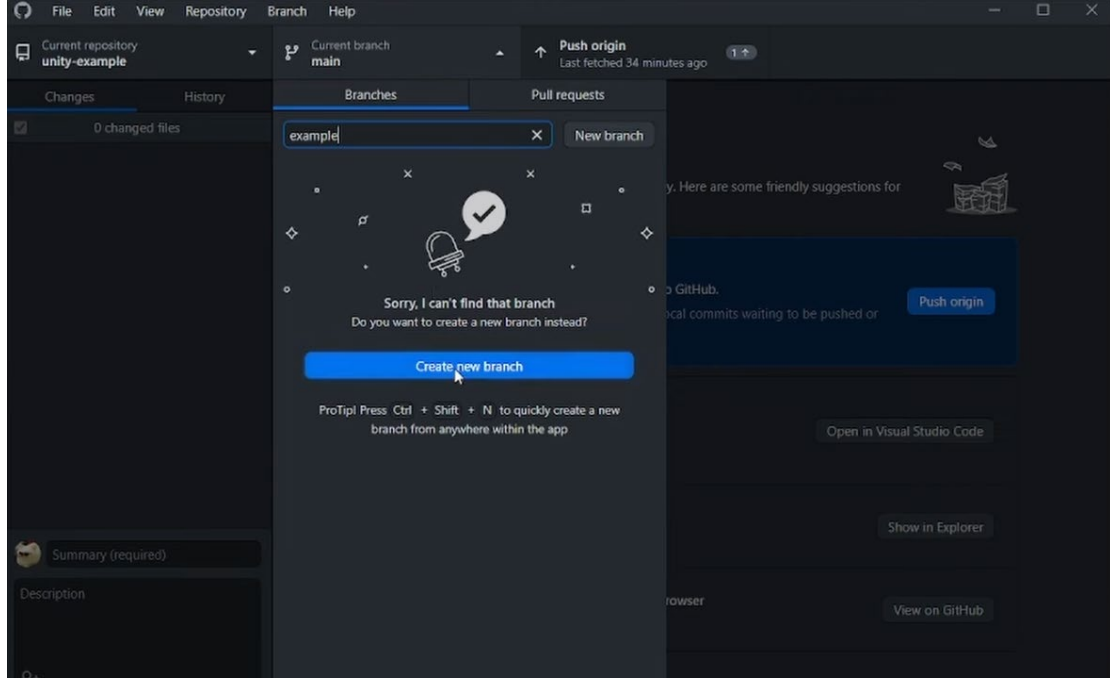


6)Seçtiğiniz dosya “Projects” kısmına gelecektir. Tıklayıp açabilirsiniz.



ÖNEMLİ NOKTALAR

1)Projenizde çalışan kısımları bozmamak ve daha temiz bir çalışma ortamı yaratmak için her üye kendi “Branch” ini oluşturup çalışabilir. Branchler özel bir çalışma ortamı oluşturmaya benzer. Örnek olarak grup arkadaşınız yeni bir mekanik deneyecek. Bu mekanik hatalı çalışıyor veya projedeki herhangi farklı bir şeyi bozuyor. Bu hatalar farklı branch üzerinde olduğu için projenin bağlı olduğu “Main Branch” i bozmuyor. Yeni mekanığın stabil bir şekilde çalıştığı durumda mekanığın yapıldığı branchi main branch ile eşleştirebilirsiniz.

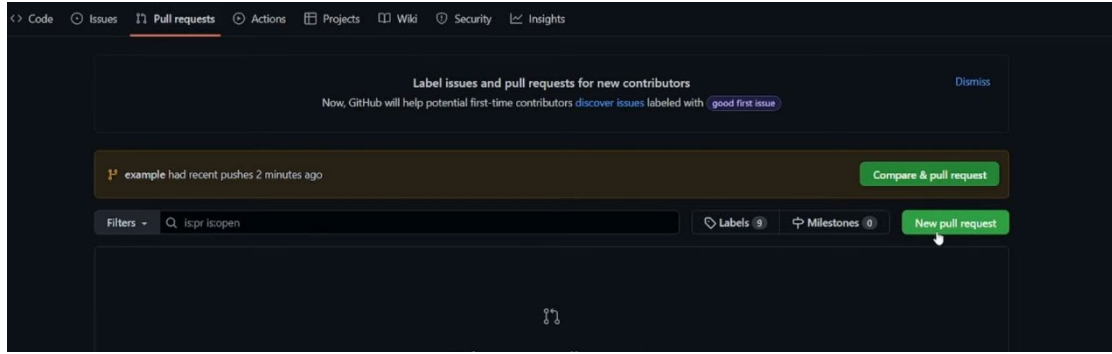


2)GitHub’den bağımsız fiziksel yedekler de almanızı tavsiye ederim. Projenizin stabil çalıştığı durumlarda USB, harici harddisk gibi araçları kullanıp yaşanabilecek kötü durumlar için en garanti yedeği tutabilirsiniz.

BRANCH EŞLEŞTİRME

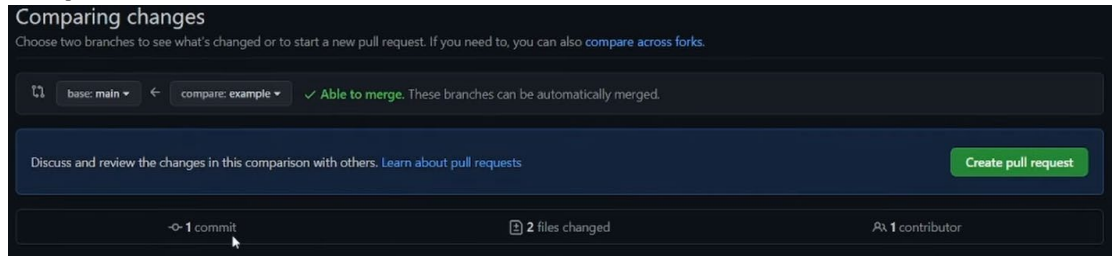
- 1)github.com’a girin.
- 2)Repository’nize girin.
- 3)Üst kısımda “Pull requests” kısmına tıklayın.

4) "New pull request" kısmına tıklayın.

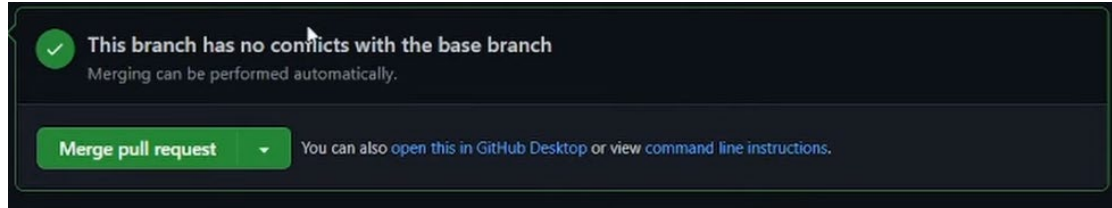


5) Hangi branchleri eşleştireceğinizi seçin.

6) "Crate pull request" kısmına tıklayın. Başlık ve açıklama ekleyin. Crate pull request'e tıklayın.



7) "Merge pull request" kısmına başlık ve açıklama ekleyip "Confirm merge" e tıklayın.



8) Eşleştirme tamamlandı. İsterseniz sonradan açtığınız branchi silebilirsiniz.

