

Ministerul Educației și Cercetării al Republicii Moldova  
IP Colegiul “Iulia Hasdeu” din Cahul

Asistență pentru managementul proiectelor software

# **Tema 2: GIT - instalare**

## Și noțiuni de bază

Apareci Aurica  
Babaianu Savelie  
Grupa: AAW 2042

Git, unul dintre cele mai influente și utilizate Sisteme de Control al Versiunilor (SCV), a fost creat de către Linus Torvalds. Git a revoluționat gestionarea versiunilor și colaborarea în dezvoltarea software datorită distribuției descentralizate, ușurinței de utilizare și capacității de a face față proiectelor de orice dimensiune.

Astăzi, Git este folosit pe scară largă în industria IT, de la dezvoltarea open source la proiecte comerciale, și reprezintă o parte fundamentală a dezvoltării software moderne.



# git



# **Terminologie și vocabular**

## Repository

Un depozit Git este un spațiu în care Git stochează toate informațiile despre un proiect. Acesta include toate fișierele, directoarele și istoricul schimbărilor legate de proiect.

Figura 1.1

## Commit


Un commit reprezintă o imagine a stării actuale a proiectului. Fiecare commit este însoțit de o descriere care explică schimbările efectuate. Acestea sunt folosite pentru a urmări istoricul schimbărilor și pentru a reveni la stări anterioare ale proiectului.

Figura 1.2

## Branch

O linie separată de dezvoltare care permite lucrul asupra funcționalităților sau problemelor în izolare, fără a afecta ramura principală ("main"). Aceasta facilitează gestionarea proiectelor mai mari și colaborarea.


Figura 1.3

 ApaAura Merge pull request #1 from ApaAura/new-branch ...

37fff8e 2 weeks ago 24 commits

WebAppFirst	Added API for customer controler	last year
.gitattributes	Initial commit	last year
.gitignore	edited gitignore for database files	last year
NEW_TEXT.txt	Added new file	2 weeks ago
README.md	Update README.md	last year
WebAppFirst.sln	Initial Commit	last year

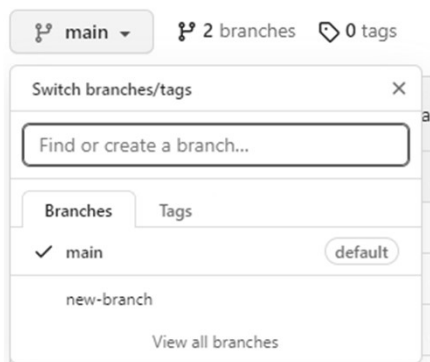
☰ README.md



# Learning-ASP.NET-MVC

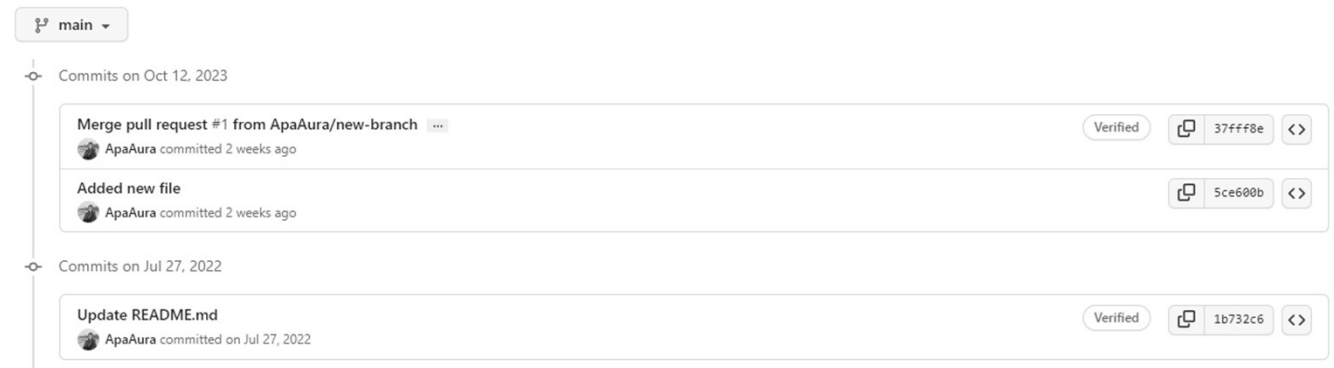
Simple Project made in ASP.Net MVC

**Figura 1.1**



**Figura 1.3**

## Commits



**Figura 1.2**

| **Clonare (Clone):** Clonarea unui depozit Git înseamnă a crea o copie locală a unui depozit existent, de obicei, un depozit remote.

| **Pull (Tragere):** Obținerea și combinarea schimbărilor dintr-o altă ramură sau depozit în ramura curentă. Acest proces vă permite să sincronizați proiectul local cu schimbările de pe un depozit remote sau să aduceți schimbări dintr-o ramură în alta.

| **Push (Împingere):** Încărcarea schimbărilor locale într-un depozit remote pentru a le partaja cu ceilalți dezvoltatori.

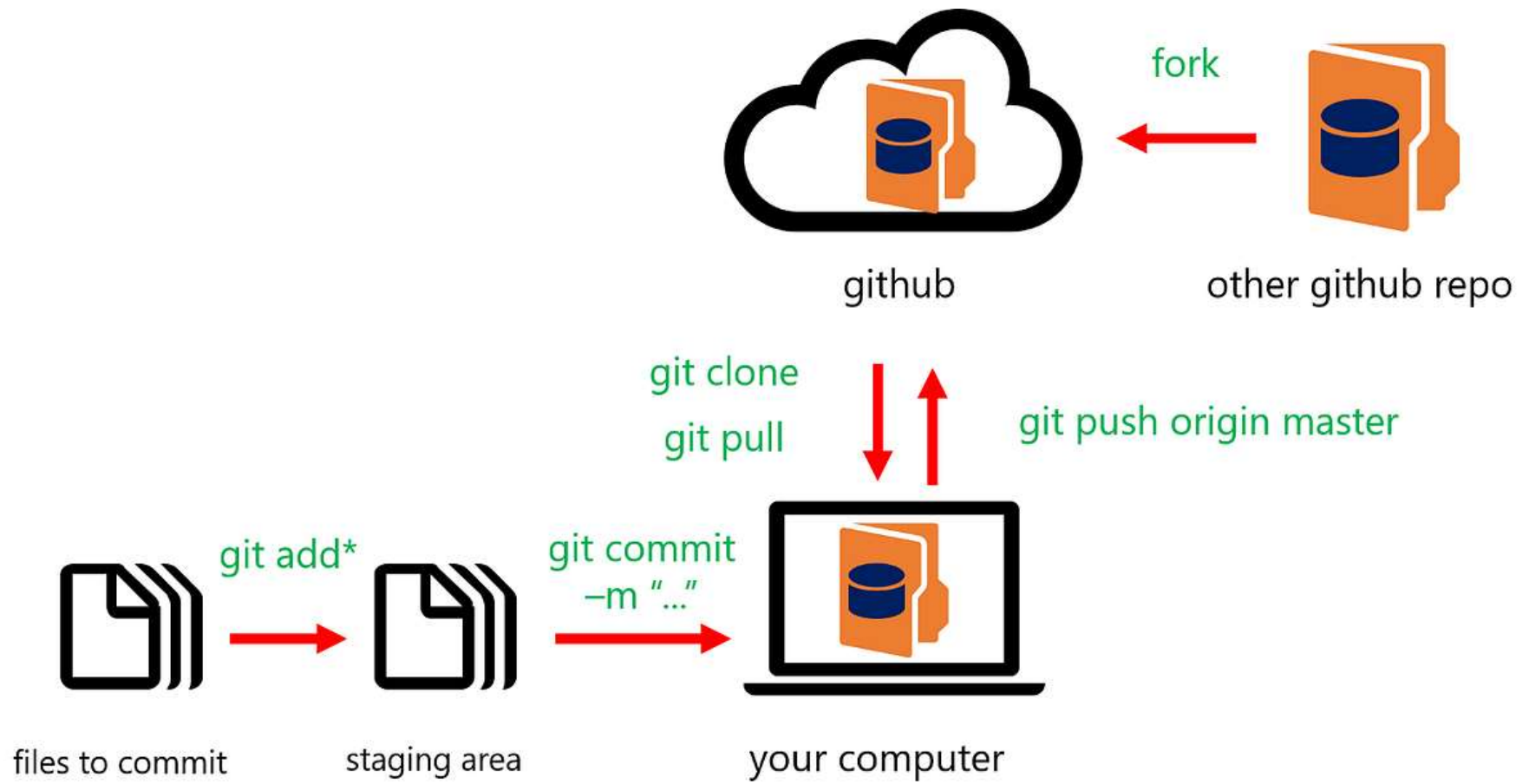
| **Conflict de îmbinare (Merge Conflict):** O situație în care două ramuri sau commit-uri conțin schimbări incompatibile în același fișier.

| **Rezolvare de conflicte (Conflict Resolution):** Procesul de soluționare a conflictelor de îmbinare pentru a permite finalizarea îmbinării cu succes.

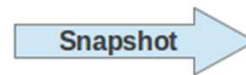
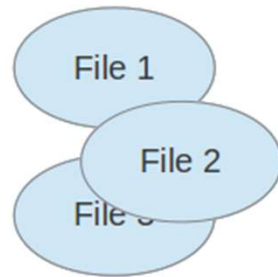
| **.gitignore:** Un fișier special în care specificați fișierele sau directoarele care nu trebuie urmărite sau incluse în depozit. Acesta este util pentru a exclude fișiere temporare, fișiere generate automat sau fișiere sensibile.

- | **Furcă (Fork):** Crearea unei copii a unui depozit remote în contul personal pentru a lucra independent asupra proiectului. Este folosit adesea pentru a contribui la proiectele open-source sau pentru a dezvolta o versiune separată a unui proiect.
- | **Pull Request (Cerere de tragere):** O solicitare de a aduce schimbările dintr-o ramură a unui depozit forked în depozitul original. Proprietarul depozitului original poate revizui schimbările și decide dacă le acceptă sau le respinge.
- | **CLI (Command Line Interface):** Interfața de linie de comandă (CLI) pentru Git este o modalitate de a interacționa cu Git folosind comenzi textuale introduse într-o fereastră de terminal sau linie de comandă.
- | **GUI (Graphical User Interface):** Interfața grafică pentru Git este o interfață cu utilizatorul bazată pe grafică, care oferă o experiență mai ușor de utilizat decât linia de comandă.

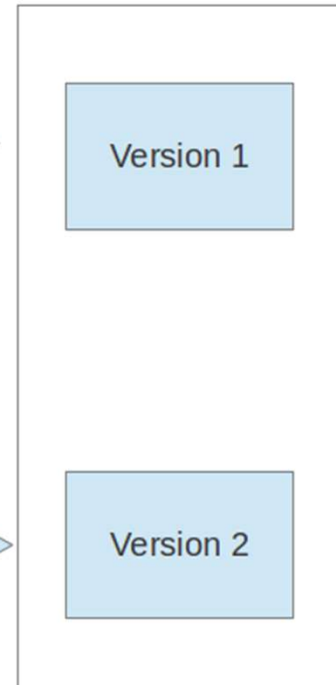




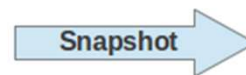
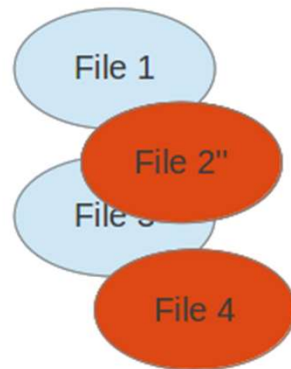
Initial state of files



Repository



File 2 changed, new File 4

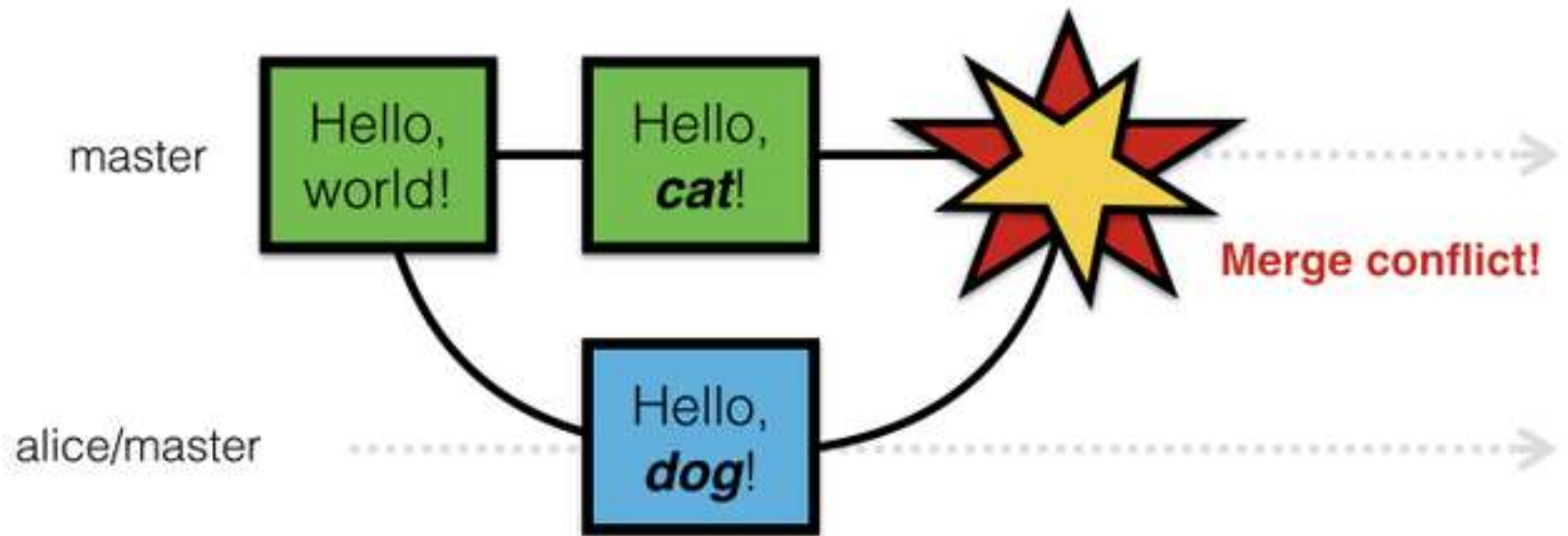


Unity Collab > bounce-tales-puzzle-platformer				
Search bounce-tales-puzzle-platformer				
<input type="checkbox"/> Name	Date modified	Type	Size	
.git	6/11/2022 4:08 PM	File folder		
.vs	6/10/2022 3:32 PM	File folder		
Assets	6/10/2022 3:32 PM	File folder		
Library	6/11/2022 4:08 PM	File folder		
Logs	6/10/2022 3:34 PM	File folder		
obj	6/10/2022 3:32 PM	File folder		
Packages	6/10/2022 3:32 PM	File folder		
ProjectSettings	6/10/2022 3:32 PM	File folder		
UserSettings	6/10/2022 3:32 PM	File folder		
.gitattributes	6/10/2022 3:32 PM	Text Document	1 KB	
.gitignore	6/11/2022 4:11 PM	Text Document	1 KB	
.vsconfig	6/10/2022 3:32 PM	VSCONFIG File	1 KB	
2D puzzle platformer.sln	6/10/2022 3:32 PM	Visual Studio Solution	1 KB	
Assembly-CSharp.csproj	6/10/2022 3:34 PM	Visual C# Project file	52 KB	
bounce-tales-puzzle-platformer.sln	6/10/2022 3:32 PM	Visual Studio Solution	1 KB	

User-generated  
files that need not  
be committed.



**.gitignore**

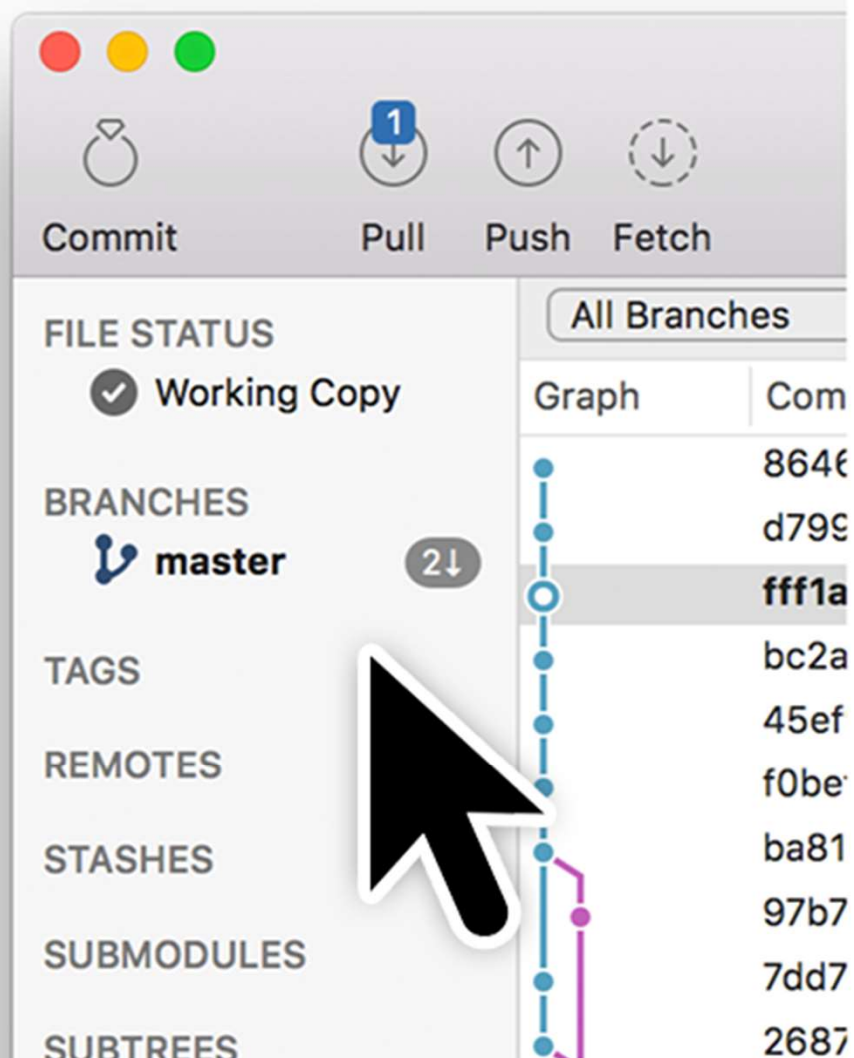




# git

GUI

CLI



The Git logo, which is a dark blue diamond shape containing a white branching diagram with two nodes and a connecting line, is positioned on the left side of the image.

# **Instalare GIT**

git

## Git poate fi instalat pe diferite platforme, cum ar fi Windows, Linux și MacOS.

Iată cum puteți instala Git pe fiecare dintre aceste sisteme de operare:

Resurse <https://git-scm.com/download/win>

### 1 Windows

Descărcați instalatorul Git de pe site-ul oficial Git și urmați instrucțiunile pentru instalare. Odată instalat, Git va fi disponibil în linia de comandă.

### 2 Linux

Pentru distribuțiile bazate pe Debian, utilizați comanda:  
``sudo apt-get install git``

Pentru distribuțiile bazate pe Red Hat, utilizați comanda:  
``sudo yum install git``

### 3 MacOS

Pe MacOS, cel mai ușor mod de a instala Git este utilizarea Homebrew. Dacă nu aveți Homebrew, instalați-l și apoi utilizați comanda:  
``brew install git``

Pentru a utiliza Git, este important să configurați numele și adresa de e-mail pentru a vă identifica. Puteți configura aceste informații global sau local, în funcție de nevoile dvs. Iată cum să configurați aceste informații.

## 1 Configurare globală

```
git config --global user.name "username"
```

```
git config --global user.email "email"
```

v

## 2 Configurare locală

```
git config user.name "username"
```

```
git config user.email "email"
```