

*Instituto Tecnológico y de Estudios Superiores de Monterrey*

## ***Guía de repositorio***



*Diego Alejandro Michel Castro / A01641907*

*Omar Arias Zepeda / A00830966*

*José Oswaldo Sobrevilla Vázquez / A01412742*

*Liga al repositorio: [https://github.com/DiegoMichel14/Multiagentes\\_Equipo4](https://github.com/DiegoMichel14/Multiagentes_Equipo4)*

***Modelación de sistemas multiagentes con gráficas computacionales***

***Docentes:***

*Raúl Ramírez*

*Iván Dounce*

*Guadalajara*

*14 de Marzo 2024*

## Especificaciones

El proyecto desarrollado se compone de 2 secciones:

- Codificación en Python
- Modelado y codificación en unity

Todos los avances que construyeron la solución se encuentran en el siguiente repositorio:

*[https://github.com/DiegoMichel14/Multiagentes\\_Equipo4](https://github.com/DiegoMichel14/Multiagentes_Equipo4)*

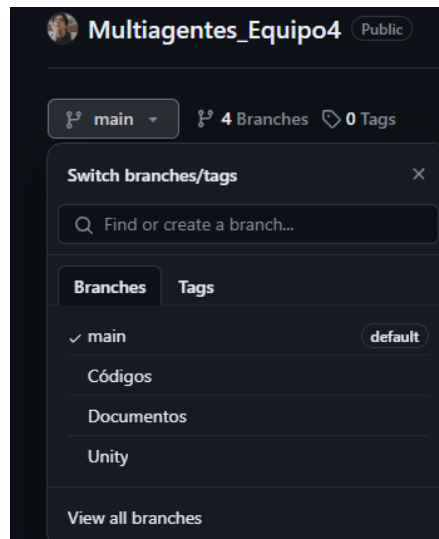
Este documento tiene la finalidad de guiar al lector/usuario en la instalación correcta para poder ejecutar la solución a nuestro reto de forma local.

Previo a la descripción de los pasos se debe asegurar que se tenga instalado Python en el sistema junto con cualquier editor de código como visual studio, visual studio code o IDEs propios del lenguaje como pycharm o anaconda, una vez dentro del código agentes.py ejecutar el siguiente comando en terminal para el importe del framework agentpy: “pip install agentpy”, de igual manera es necesario tener conocimiento básico sobre la página web de unity engine.


## Guía para el repositorio

Debido a un problema de almacenamiento requerido y poca organización del primer repositorio se decidió crear uno nuevo con todos los contenidos organizados por secciones.

Se cuenta con 3 ramas principales que tienen un nombre descriptivo sobre sus contenidos, la rama principal únicamente cuenta con un archivo readme con evidencias de comunicaciones entre nuestro proyecto en código y el motor gráfico unity.

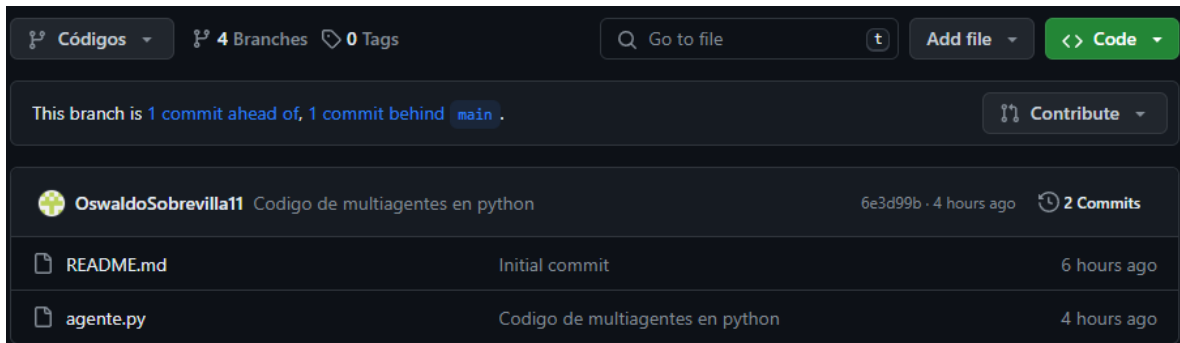


- Branch “Documentos”

 <b>DiegoMichel14</b> Tercera entrega sobre la solución del proyecto 			39a04dc · 5 hours ago	 <b>4 Commits</b>
 M5.- Revisión de avance 2_Equipo4.pdf	Segunda publicación de avances del proyecto			5 hours ago
 M5.- Revisión de avance 3_Equipo 4.pdf	Tercera entrega sobre la solución del proyecto			5 hours ago
 README.md	Initial commit			6 hours ago
 REVISIÓN 1 EQUIPO 4.pdf	Primer avance del proyecto			5 hours ago

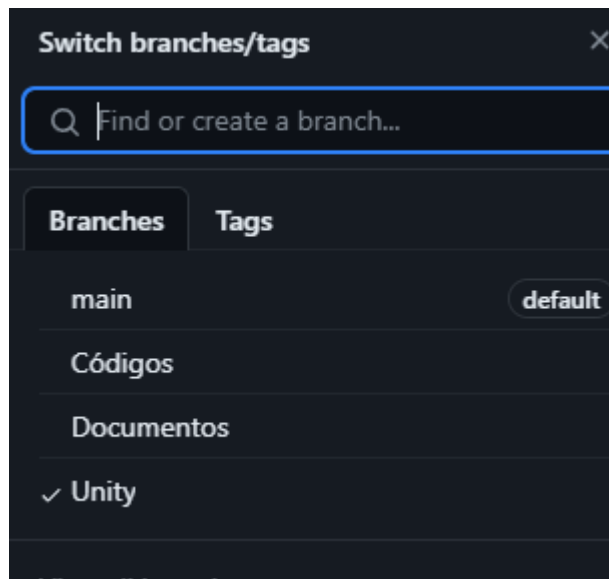
Como parte de esta unidad de formación se nos solicitó publicar evidencias del desarrollo progresivo del reto, por lo que dividimos los avances en diferentes documentos etiquetados por número de avance y fecha, dentro de esos documentos en formato PDF NO se encuentra ningún ejecutable. En esta rama están redactados los avances por escrito de nuestra solución.

- Branch “Códigos”



En esta rama se encuentran códigos independientes a unity utilizados en nuestro equipo, el único necesario fue la implementación y codificación en Python de los agentes.

- Branch “Unity”



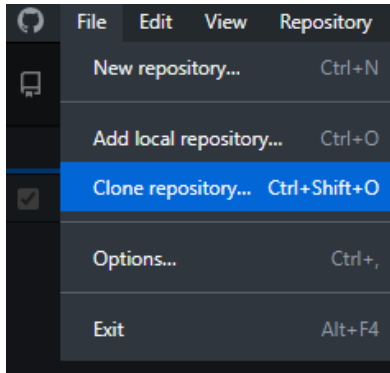
En esta rama se encuentra la carpeta del proyecto en unity, en dicha carpeta se pueden consultar de manera individual los archivos que permiten la ejecución del proyecto (una gran mayoría de archivos de la carpeta son propios del motor gráfico y no fueron redactados por los estudiantes).

## Proceso de instalación para la ejecución del proyecto

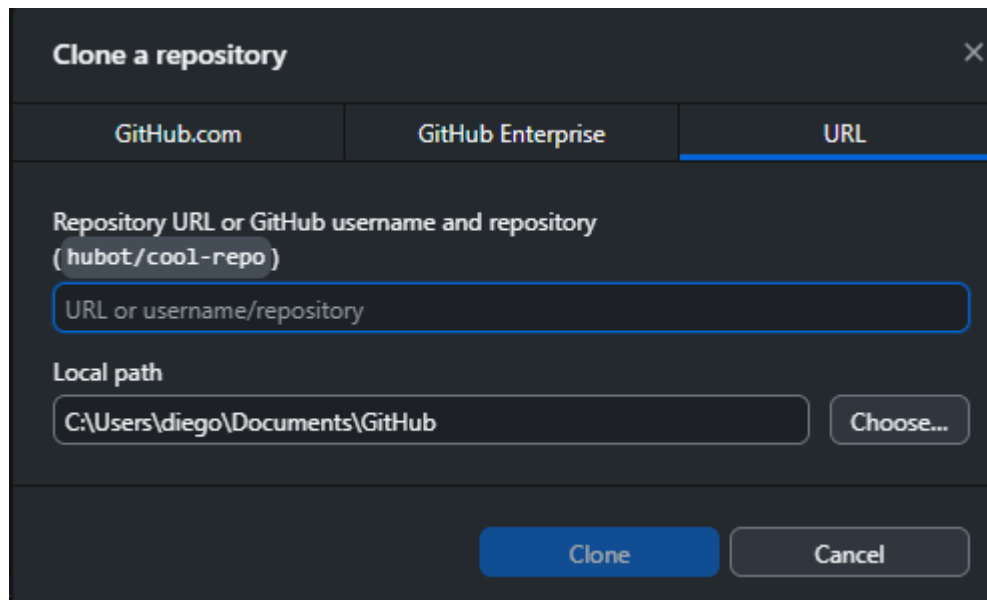
Hay diversas formas de clonar un repositorio remoto a local, en esta guía se detallará por medio de github desktop.

Primero se debe contar con una cuenta en github para poder consultar el repositorio remoto, con dicha cuenta se debe descargar github desktop y acceder de forma regular.

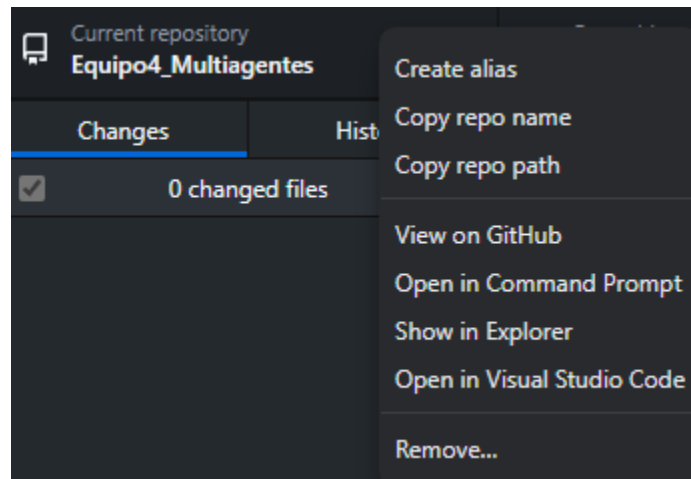
Dentro de github desktop se debe seleccionar la opción de clonar repositorio.



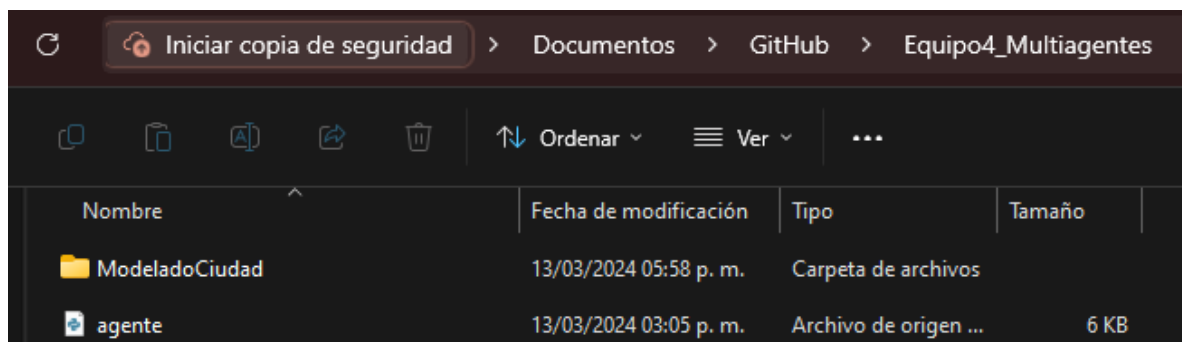
Una vez seleccionada dicha opción nos dirigimos a clonar por medio de un url e ingresamos el link del repositorio



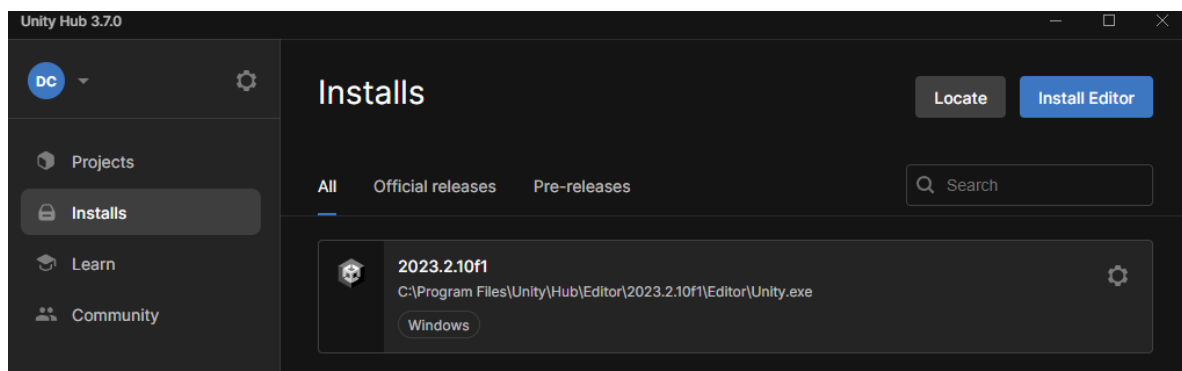
Una vez terminada la clonación colocamos el ratón sobre el nombre del repositorio clonado y presionamos click derecho para desplegar las siguientes opciones:



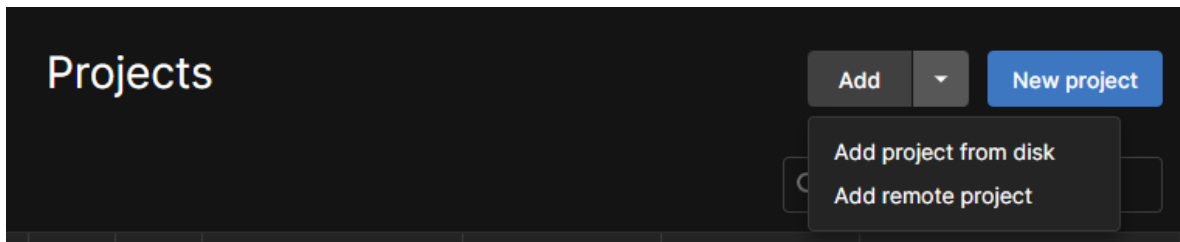
Seleccionamos “Show in Explorer” para que nos dirija a la carpeta local donde se clonaron los archivos, debe aparecer lo siguiente:



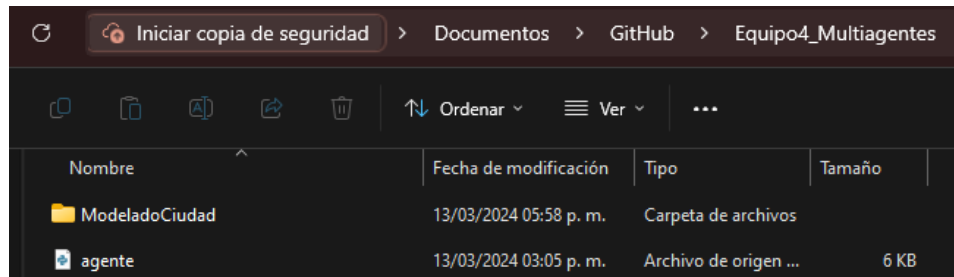
Posterior a la correcta clonación del repositorio se debe descargar el motor unity desde su página oficial, preferentemente instalar la versión 2023.2.10f1 ya que en esa se desarrolló el proyecto.



Con la versión instalada en el equipo nos dirigimos a la sección de proyectos para poder añadir un proyecto desde el disco local



Al seleccionar esta opción nos abrirá el explorador de archivos y seleccionaremos la carpeta del proyecto que fue clonada desde el repositorio remoto.



## Proceso de ejecución del proyecto

Con el proyecto abierto en unity presionamos el botón de play y después el botón de pausa, esto comenzará la conexión y hará que unity espere los datos que mandará el código de Python, posteriormente ejecutamos el código adjunto para el funcionamiento del proyecto agente.py que viene junto a la carpeta, tras ejecutarse el código de Python volvemos a unity y presionamos nuevamente el botón de pausa y se instanciarán los coches dando comienzo a la simulación.

