

Contenido

○	Apartado 1: Crear cuenta en GitHub y crear un repositorio.....	2
○	○ Paso 1: Instalamos Git y Creamos el primer repositorio en local.....	2
○	○ Paso 2: Creamos una cuenta en GitHub y creamos un repositorio	5
	1. Creamos una cuenta en GitHub e iniciamos sesión:	5
	2. Accedemos a nuestra cuenta y creamos un repositorio:	6
	3. Conectamos el repositorio de GitHub con nuestro repositorio local:.....	8

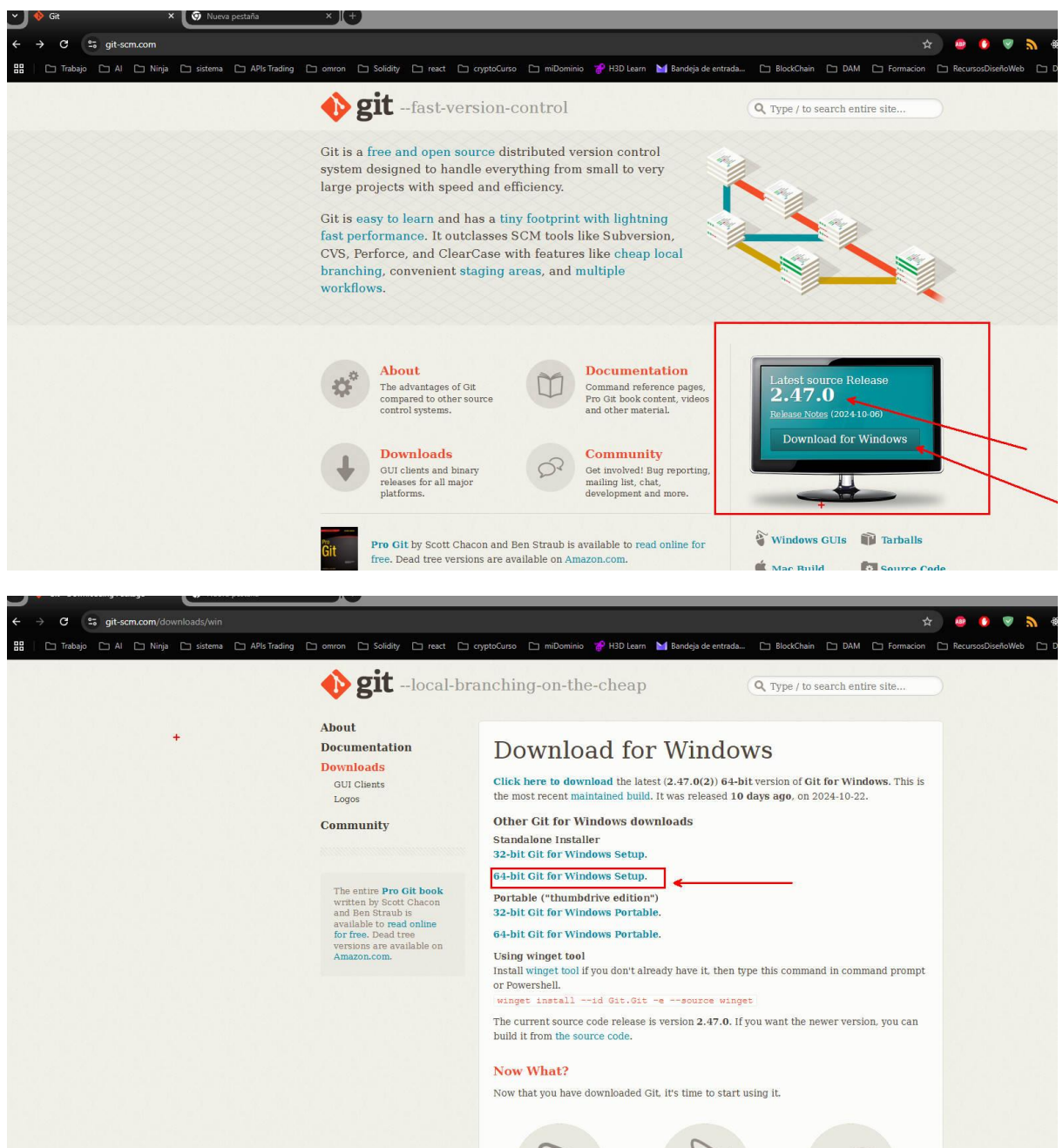
○ Apartado 1: Crear cuenta en GitHub y crear un repositorio.

Comenzamos el proceso para llegar a la creación del repositorio en GitHub:

https://github.com/AntonioRRZ/Ruiz_Rodriguez_Antonio_PIA01_Tarea

○ Paso 1: Instalamos Git y Creamos el primer repositorio en local.

Comenzamos la instalación de **Git** descargándonos el instalador desde su **web oficial** para el sistema operativo que estemos utilizando:

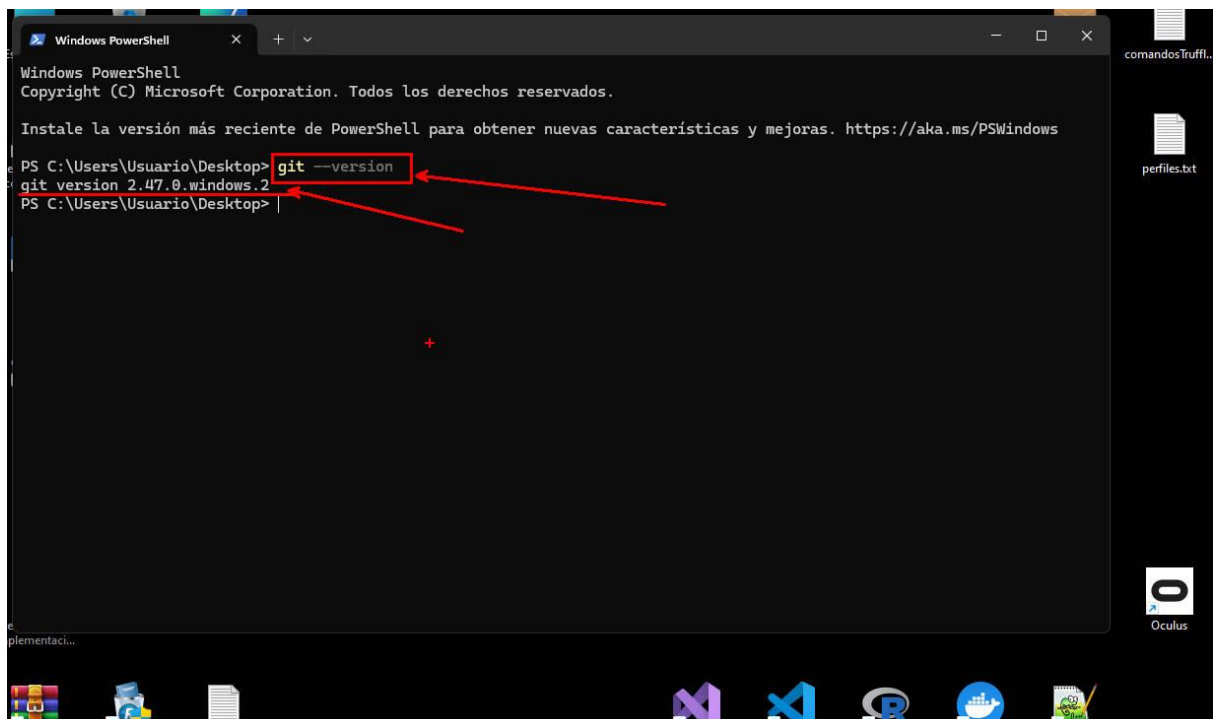


Programación de Inteligencia Artificial Unidad 1. Apartado 1

La instalación de **Git**, puede llegar a ser muy compleja, pero un usuario medio que no necesite características especiales de configuración tiene el 100% de la funcionalidad con la instalación estándar:



Una vez terminada la ejecución del instalador, comprobamos que ya tenemos disponible **Git** en nuestro ordenador haciendo “Git --versión”, si todo ha ido bien, por consola veremos la versión de Git que tenemos instalada:



Programación de Inteligencia Artificial Unidad 1. Apartado 1

Antes de crear el primer repositorio realizamos las primeras configuraciones básicas, como definir las variables globales “user” y “email” con las que Git identificará nuestros repositorios:

- Git config --global user.name <nombre usuario>
- Git config --global user.email <correo electrónico>

```
PS C:\Users\Usuario\source\repos> git config --global user.name "Antonio Ruiz Rodriguez"
PS C:\Users\Usuario\source\repos> git config --global user.email "antonioruizr@protonmail.com"
PS C:\Users\Usuario\source\repos>
```

Ahora ya podemos crear nuestro primer repositorio local con Git:

1. Abrimos un CMD, ventana de comandos o consola y navegamos hasta colocarnos en la ruta a la carpeta donde queremos crear el repositorio y ejecutamos el comando **git init**:

```
Windows PowerShell
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

Instale la versión más reciente de PowerShell para obtener nuevas características y mejoras. https://aka.ms/PSWindows

PS C:\Users\Usuario\Desktop\Ciclo\Programacion de InteligenciaArtificial\Modulo 1\Tarea\instalacion_Git_GitHub> git init
Initialized empty Git repository in C:/Users/Usuario/Desktop/Ciclo/Programacion de InteligenciaArtificial/Modulo 1/Tarea/instalacion_Git_GitHub/.git/
PS C:\Users\Usuario\Desktop\Ciclo\Programacion de InteligenciaArtificial\Modulo 1\Tarea\instalacion_Git_GitHub>
```

Con este comando ya hemos inicializado nuestro repositorio, pero aún no estamos haciendo seguimiento a ningún fichero. Si hacemos **git status** la aplicación nos informará del estado de los ficheros y de si estamos o no haciéndoles seguimiento:

```
Windows PowerShell
PS C:\Users\Usuario\Desktop\Ciclo\Programacion de InteligenciaArtificial\Modulo 1\Tarea\instalacion_Git_GitHub> git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
  creacionRepositorioGitHub.docx
  creacionRepositorioGitHub.pdf

nothing added to commit but untracked files present (use "git add" to track)
PS C:\Users\Usuario\Desktop\Ciclo\Programacion de InteligenciaArtificial\Modulo 1\Tarea\instalacion_Git_GitHub>
```

2. Agregamos los archivos al repositorio: Haciendo **git add** . con el carácter “.” es indicamos a **git** que queremos que prepare todos los ficheros de la carpeta y con **git commit** añadimos finalmente los ficheros al repositorio. El parámetro -m, nos permite añadir alguna descripción:

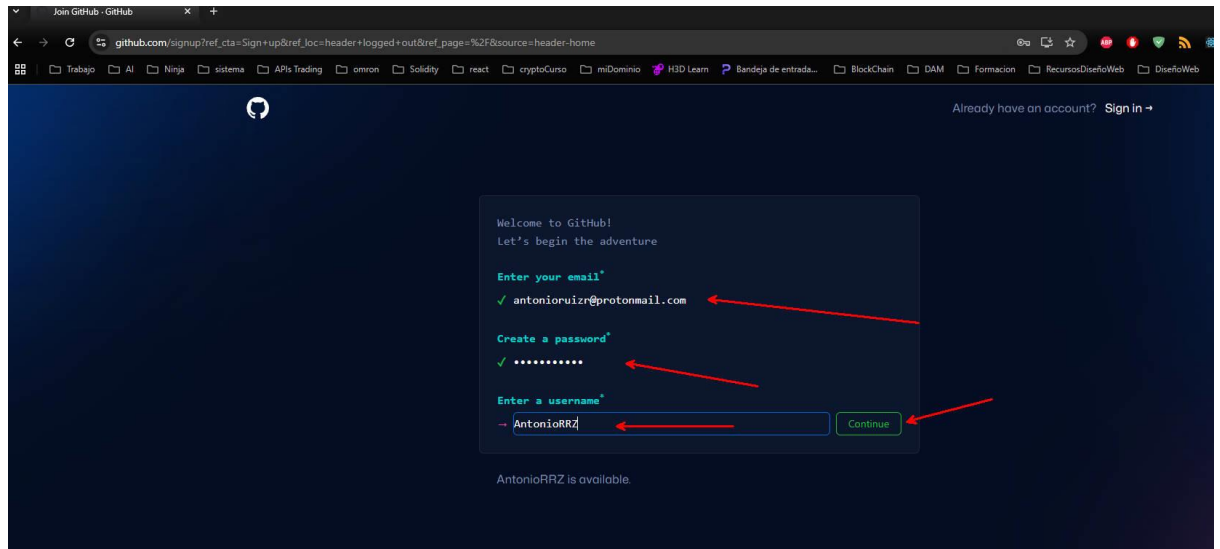
```
PS C:\Users\Usuario\Desktop\Ciclo\Programacion de InteligenciaArtificial\Modulo 1\Tarea\instalacion_Git_GitHub> git add .
PS C:\Users\Usuario\Desktop\Ciclo\Programacion de InteligenciaArtificial\Modulo 1\Tarea\instalacion_Git_GitHub> git commit -m "Primer Repositorio"
[master (root-commit) ec763bf] Primer Repositorio
2 files changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 creacionRepositorioGitHub.docx
 create mode 100644 creacionRepositorioGitHub.pdf
PS C:\Users\Usuario\Desktop\Ciclo\Programacion de InteligenciaArtificial\Modulo 1\Tarea\instalacion_Git_GitHub> git status
On branch master
nothing to commit, working tree clean
PS C:\Users\Usuario\Desktop\Ciclo\Programacion de InteligenciaArtificial\Modulo 1\Tarea\instalacion_Git_GitHub>
```

Programación de Inteligencia Artificial Unidad 1. Apartado 1

○ Paso 2: Creamos una cuenta en GitHub y creamos un repositorio

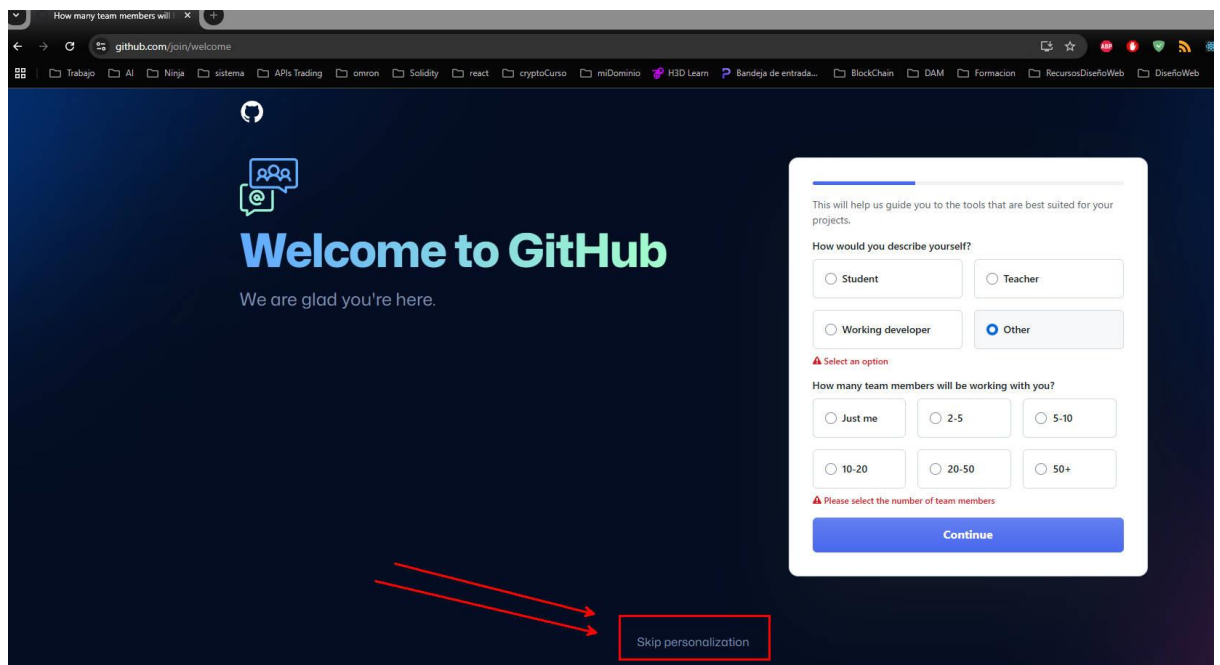
1. Creamos una cuenta en GitHub e iniciamos sesión:

Nos colocamos en la página oficial GitHub.com y creamos una cuenta:



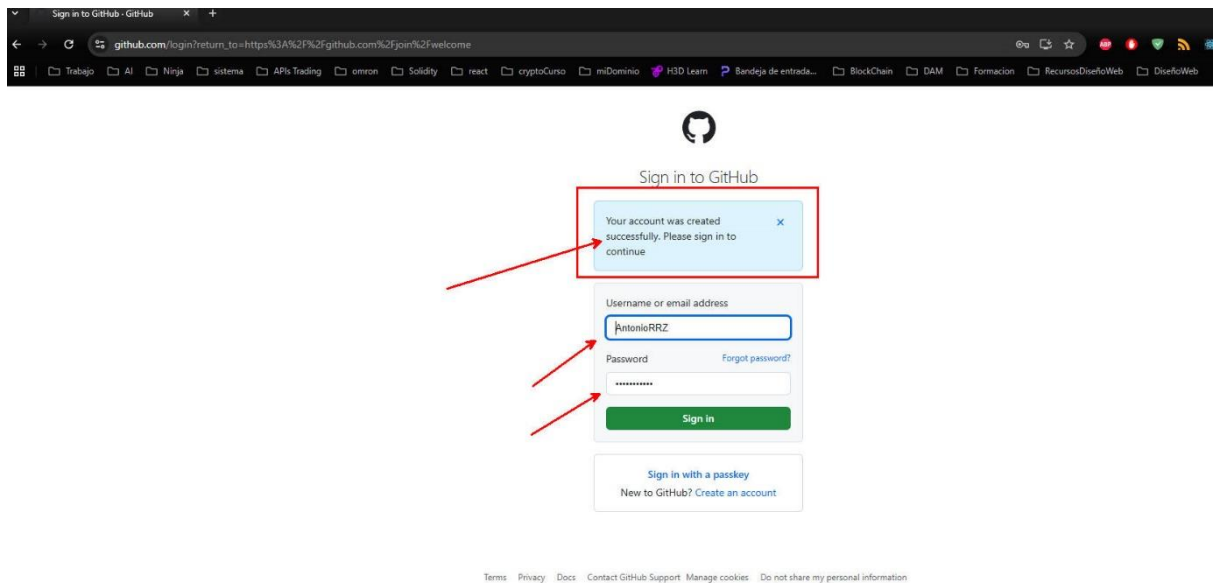
The screenshot shows the GitHub sign-up page. The form is titled "Welcome to GitHub! Let's begin the adventure". It has three input fields: "Enter your email*", "Create a password*", and "Enter a username*". The email field contains "antonioruizr@protonmail.com", the password field contains "*****", and the username field contains "AntonioRRZ". A "Continue" button is to the right of the username field. Red arrows point to each of these three input fields and the "Continue" button. Below the form, it says "AntonioRRZ is available."

Después de introducir nuestros datos y pasar las verificaciones de seguridad, saltamos la personalización e iniciamos sesión en nuestra nueva cuenta:



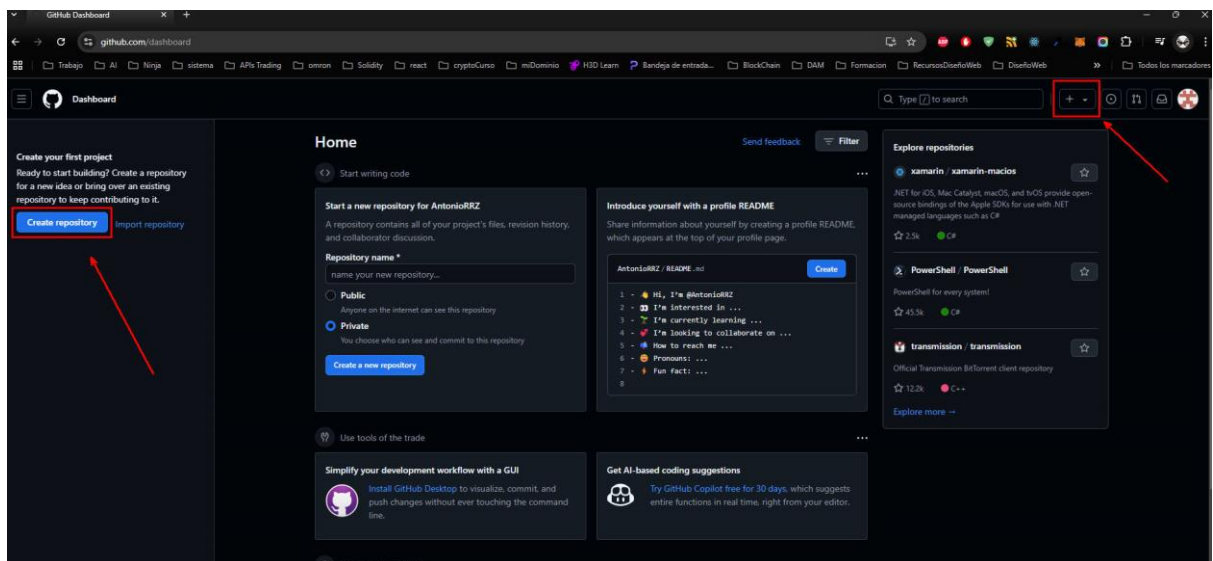
The screenshot shows the GitHub welcome page. On the left, it says "Welcome to GitHub" and "We are glad you're here." On the right, there is a survey titled "How would you describe yourself?" with options: "Student", "Teacher", "Working developer", and "Other". Below that, it asks "How many team members will be working with you?" with options: "Just me", "2-5", "5-10", "10-20", "20-50", and "50+". A "Continue" button is at the bottom of the survey. At the bottom of the page, there is a "Skip personalization" button, which is highlighted with a red box and red arrows pointing to it.

Programación de Inteligencia Artificial Unidad 1. Apartado 1



2. Accedemos a nuestra cuenta y creamos un repositorio:

Nuestra cuenta tiene este aspecto, para crear un repositorio solo tenemos que ir a “Create repository” o al botón “+” en la esquina superior derecha:



En la pantalla para crear un repositorio nos pedirán los siguientes datos:

- Nombre:** Es obligatorio darle un nombre a nuestro repositorio y deberá ser único dentro de nuestra cuenta.
- Descripción:** No es obligatorio, pero servirá a otros usuarios y a nosotros mismos a identificar el contenido del repositorio.
- Público o Privado:** Es obligatorio, determina la “visibilidad” del repositorio y dependerá del uso y la metodología que estemos siguiendo.

- d) **Fichero READ.Me:** Es común que un repositorio valla acompañado de un fichero que defina y explique con más detalle las partes del repositorio y su contenido. No es obligatorio
- e) **Fichero gitignore:** Este fichero contiene una lista de ficheros que no se subirán al repositorio, bien porque son archivos de configuración o datos que deben generarse una vez descargado y que harían el repositorio demasiado pesado innecesariamente. No es obligatorio.
- f) **Licencia:** Podemos añadir una licencia que determine las condiciones de uso del contenido de nuestro repositorio. No es obligatorio.

The screenshot shows the 'Create a new repository' page on GitHub. The form includes the following sections:

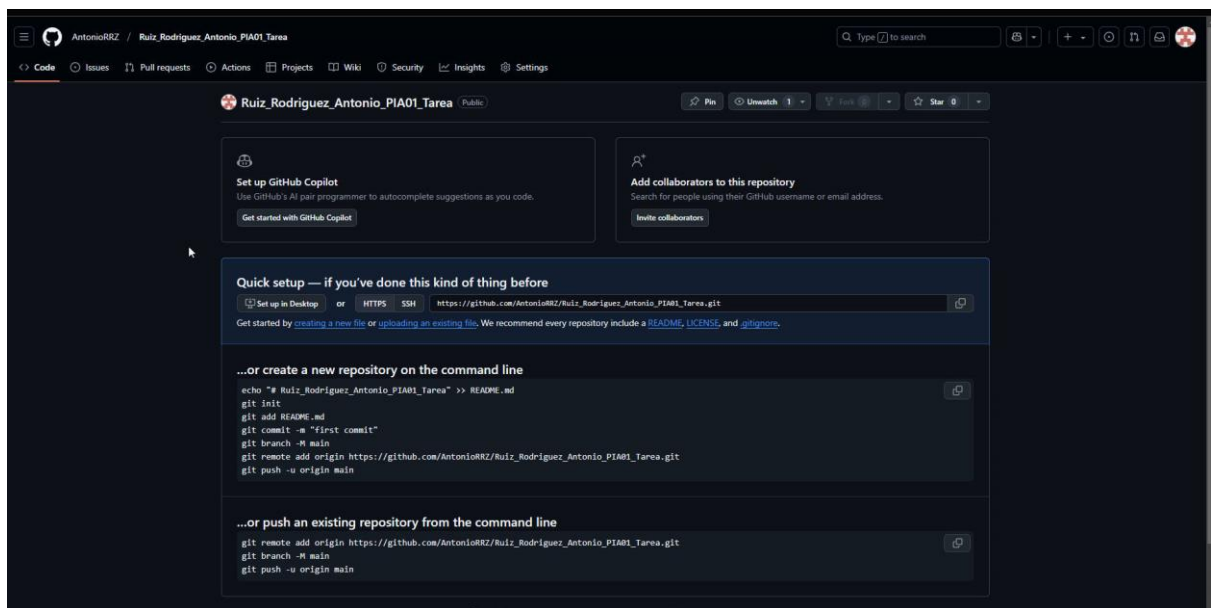
- Owner ***: A dropdown menu showing 'AntonioRRZ'.
- Repository name ***: A text input field containing 'Ruiz_Rodriguez_Antonio_J'. A red arrow points to this field.
- Description (optional)**: A text input field.
- Public/Private**: Two radio buttons. The 'Public' option is selected, and a red arrow points to it. The 'Private' option is also visible.
- Initialize this repository with:**
 - ☐ Add a README file
 - Add .gitignore**: A dropdown menu showing 'None'.
 - Choose a license**: A dropdown menu showing 'None'.
- Create repository**: A blue button at the bottom right, with a red arrow pointing to it.

Una vez definido nuestro repositorio, pulsando en “Create repository”, Github lo creará y lo dejará listo para empezar a añadir contenido.

3. Conectamos el repositorio de GitHub con nuestro repositorio local:

Una vez creado el repositorio, GitHub nos da las instrucciones para conectar el repositorio en la nube, con nuestro repositorio local:

- La URL donde se encuentra el repositorio que acabamos de crear. La línea de comando completa que debemos usar en nuestro equipo local para conectar con el repositorio remoto, ofreciéndonos dos alternativas, por https o ssh, en nuestro caso usaremos la conexión https:
git remote add origin
`https://github.com/AntonioRRZ/Ruiz_Rodriguez_Antonio_PIA01_Tarea.git`
- Dos alternativas, la primera en el caso de que **no** hayamos creado el repositorio en local, en cuyo caso nos da las líneas de comando necesarias para crearlo y subir el contenido al repositorio remoto.
- Si ya tenemos un repositorio local creado, no dice las líneas de comando necesarias para subir el contenido



En nuestro caso, ya habíamos creado un repositorio local con lo que usamos la segunda opción. Después de introducir las líneas de comando por consola, GitHub no pide autenticarnos. Si la autenticación es correcta, Git sube el contenido de nuestro repositorio al GitHub, teniendo en cuenta que solo habrá subido el contenido al que previamente hayamos hecho un *commit* :

- ✓ `git branch -M main`
- ✓ `git push -u origin main`

Programación de Inteligencia Artificial Unidad 1. Apartado 1

Una vez realizado todo el proceso, vemos en GitHub el contenido del repositorio que acabamos de subir

```
on branch master
no commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    .vscode/
    Apartado_2_Problema_1.ipynb
    Apartado_2_Problema_2.ipynb
    Apartado_2_Problema_3.ipynb
    creacionRepositorioGitHub.pdf
    data_description.txt
    sample_submission.csv
    test.csv
    train.csv

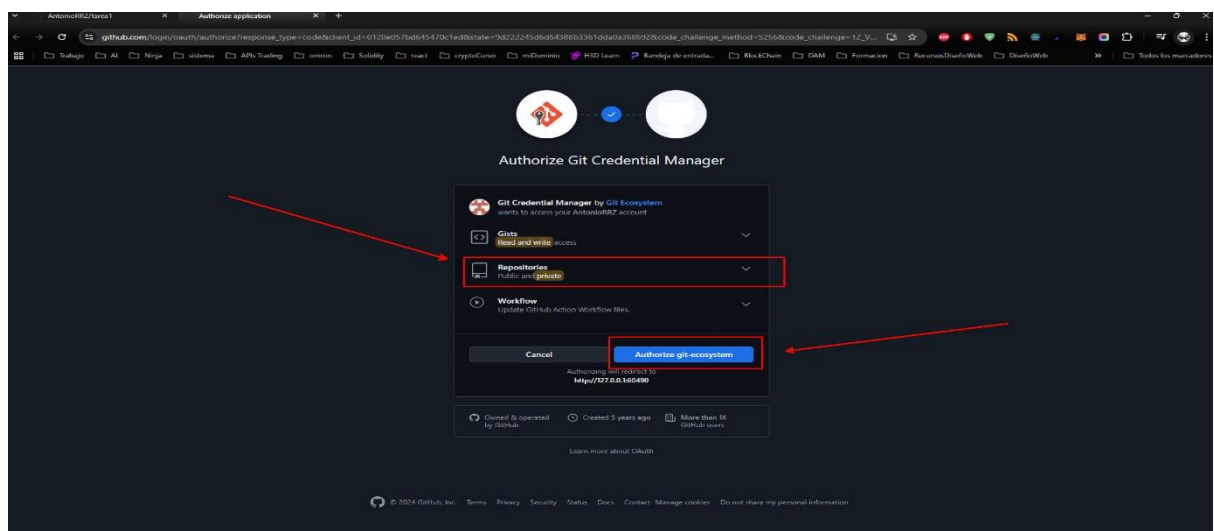
nothing added to commit but untracked files present (use "git add" to track)

isuario@DESKTOP-FN38BDH MINGW64 ~/Desktop/Ciclo/03-Programacion de Inteligencia Artificial/Modulo 1/Tarea/Ruiz_Rodriguez_Antonio_PIA01_Tarea (master)
$ git add .
warning: in the working copy of 'Apartado_2_Problema_1.ipynb', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'Apartado_2_Problema_2.ipynb', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'Apartado_2_Problema_3.ipynb', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'data_description.txt', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'sample_submission.csv', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'test.csv', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'train.csv', LF will be replaced by CRLF the next time Git touches it

isuario@DESKTOP-FN38BDH MINGW64 ~/Desktop/Ciclo/03-Programacion de Inteligencia Artificial/Modulo 1/Tarea/Ruiz_Rodriguez_Antonio_PIA01_Tarea (master)
$ git commit -m "Subimos la Documentación de la tarea"
[master (root-commit) abe5a2] Subimos la Documentación de la tarea
9 files changed, 5175 insertions(+)
create mode 100644 .vscode/settings.json
create mode 100644 Apartado_2_Problema_1.ipynb
create mode 100644 Apartado_2_Problema_2.ipynb
create mode 100644 Apartado_2_Problema_3.ipynb
create mode 100644 creacionRepositorioGitHub.pdf
create mode 100644 data_description.txt
create mode 100644 sample_submission.csv
create mode 100644 test.csv
create mode 100644 train.csv

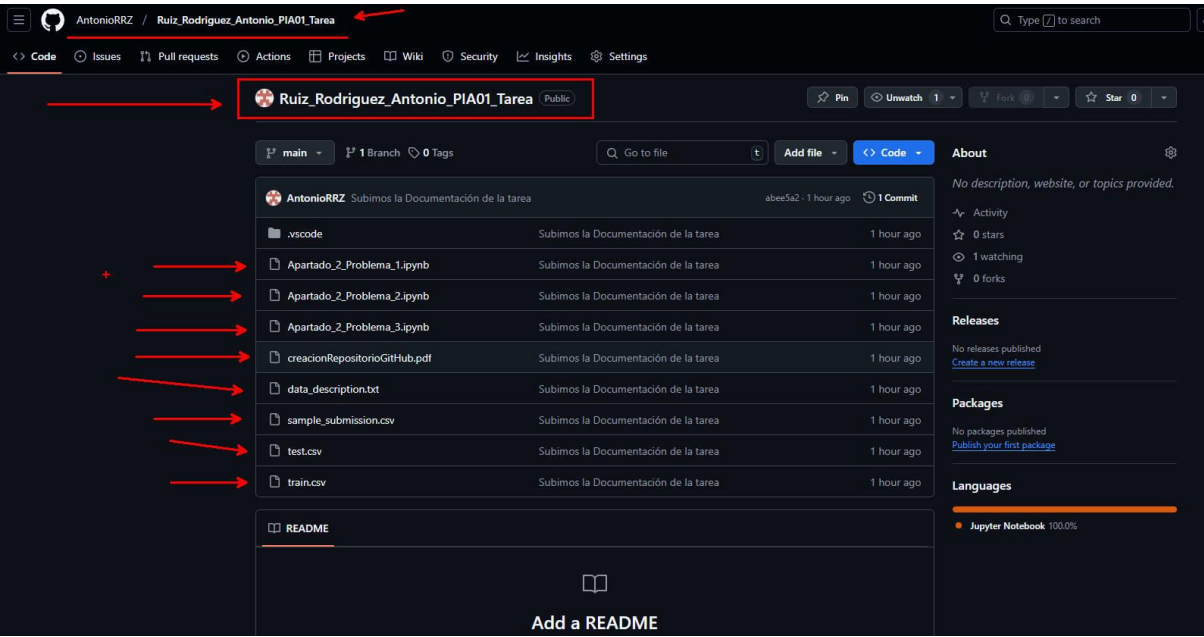
isuario@DESKTOP-FN38BDH MINGW64 ~/Desktop/Ciclo/03-Programacion de Inteligencia Artificial/Modulo 1/Tarea/Ruiz_Rodriguez_Antonio_PIA01_Tarea (master)
$ git remote add origin https://github.com/AntonioRRZ/Ruiz_Rodriguez_Antonio_PIA01_Tarea.git
git branch -M main
git push -u origin main
Enumerating objects: 12, done.
Counting objects: 100% (12/12), done.
Delta compression using up to 32 threads
Compressing objects: 100% (11/11), done.
Writing objects: 100% (12/12), 237.50 KiB | 6.25 MiB/s, done.
Total 12 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), done.
To https://github.com/AntonioRRZ/Ruiz_Rodriguez_Antonio_PIA01_Tarea.git
 * [new branch] main -> main
branch 'main' set up to track 'origin/main'.

isuario@DESKTOP-FN38BDH MINGW64 ~/Desktop/Ciclo/03-Programacion de Inteligencia Artificial/Modulo 1/Tarea/Ruiz_Rodriguez_Antonio_PIA01_Tarea (main)
```



Ruiz_Rodriguez_Antonio_PIA01_Tarea

Programación de Inteligencia Artificial Unidad 1. Apartado 1



Ruiz_Rodriguez_Antonio_PIA01_Tarea