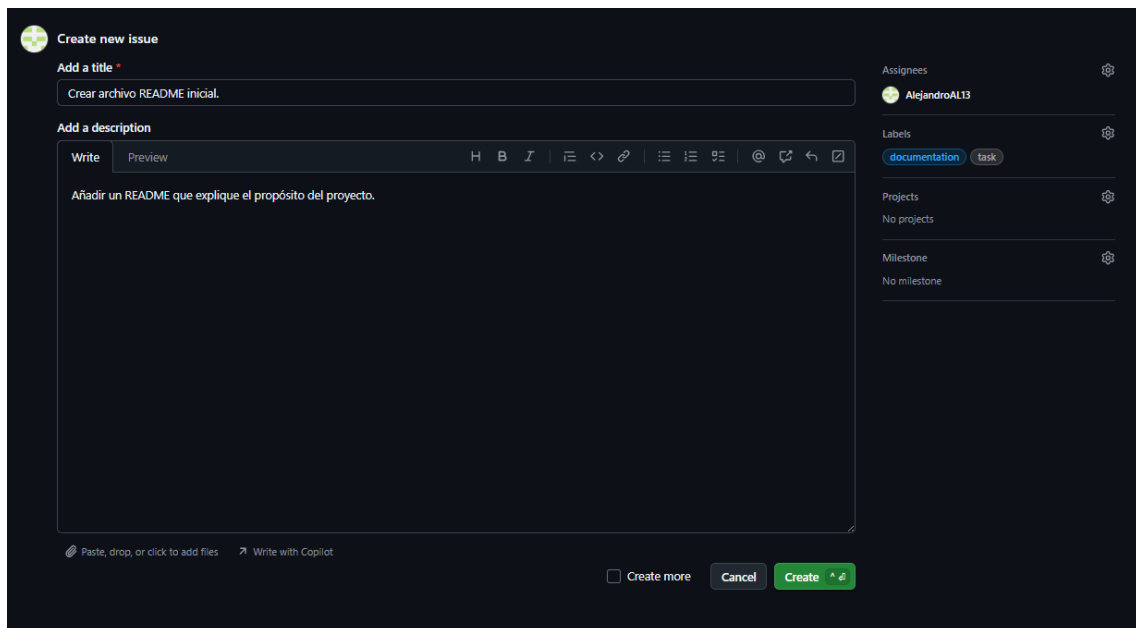


Creamos el repositorio desde github y lo clonamos para poder usarlo desde una terminal(GitBash en mi caso).

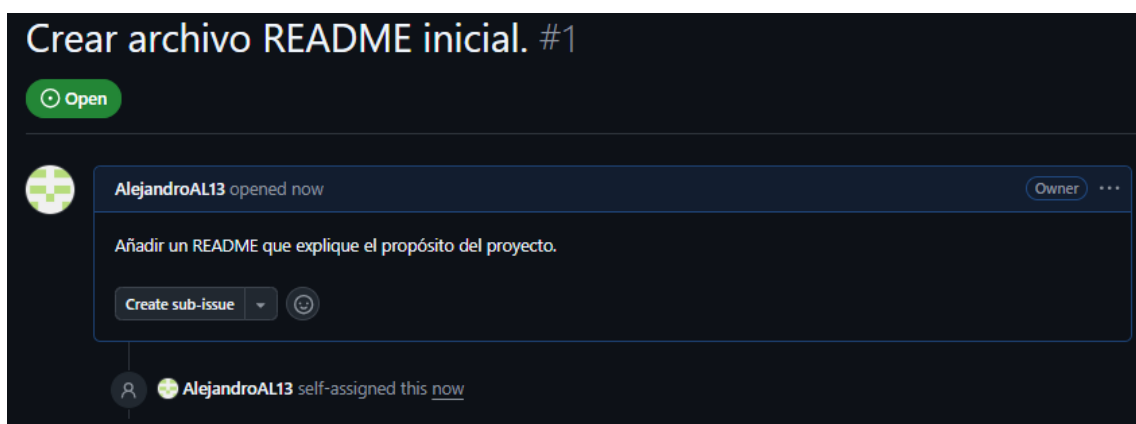
```
alex1@Alexportati1 MINGW64 ~  
$ git clone https://github.com/AlejandroAL13/tarea-kanban  
Cloning into 'tarea-kanban'...  
warning: You appear to have cloned an empty repository.
```

Empezamos añadiendo los “issues” que son como las tareas que nos podemos asignar con sus correspondientes etiquetas para saber en un futuro la parte que tenemos que hacer de nuestro proyecto. Con esto en caso de formar parte de un proyecto con más gente podemos asignar tareas en función de lo que debe hacer cada uno para que no sea un caos la organización.



The screenshot shows the 'Create new issue' interface on GitHub. The title field contains 'Crear archivo README inicial.' The description field contains 'Añadir un README que explique el propósito del proyecto.' The right sidebar shows the 'Assignees' section with 'AlejandroAL13' selected, and the 'Labels' section with 'documentation' and 'task' selected. The 'Projects' and 'Milestone' sections are empty. At the bottom, there are buttons for 'Create more', 'Cancel', and 'Create'.

El primer “issue” será crear un README inicial.



The screenshot shows the GitHub issue page for 'Crear archivo README inicial. #1'. The issue is opened by 'AlejandroAL13' and is self-assigned. The description field contains 'Añadir un README que explique el propósito del proyecto.' The 'Create sub-issue' button is visible. The 'Owner' dropdown menu is open, showing 'AlejandroAL13' as the owner.

Desde nuestra terminal creamos un carpeta `.github/workflow` para tenerlo todo organizado, dentro de workflow creamos un archivo llamado `verify.yml` el cual nos servirá para revisar que las características de nuestro proyecto funcionen correctamente al iniciarlo.

```
alex1@Alexportatil MINGW64 ~
$ cd tarea-kanban/

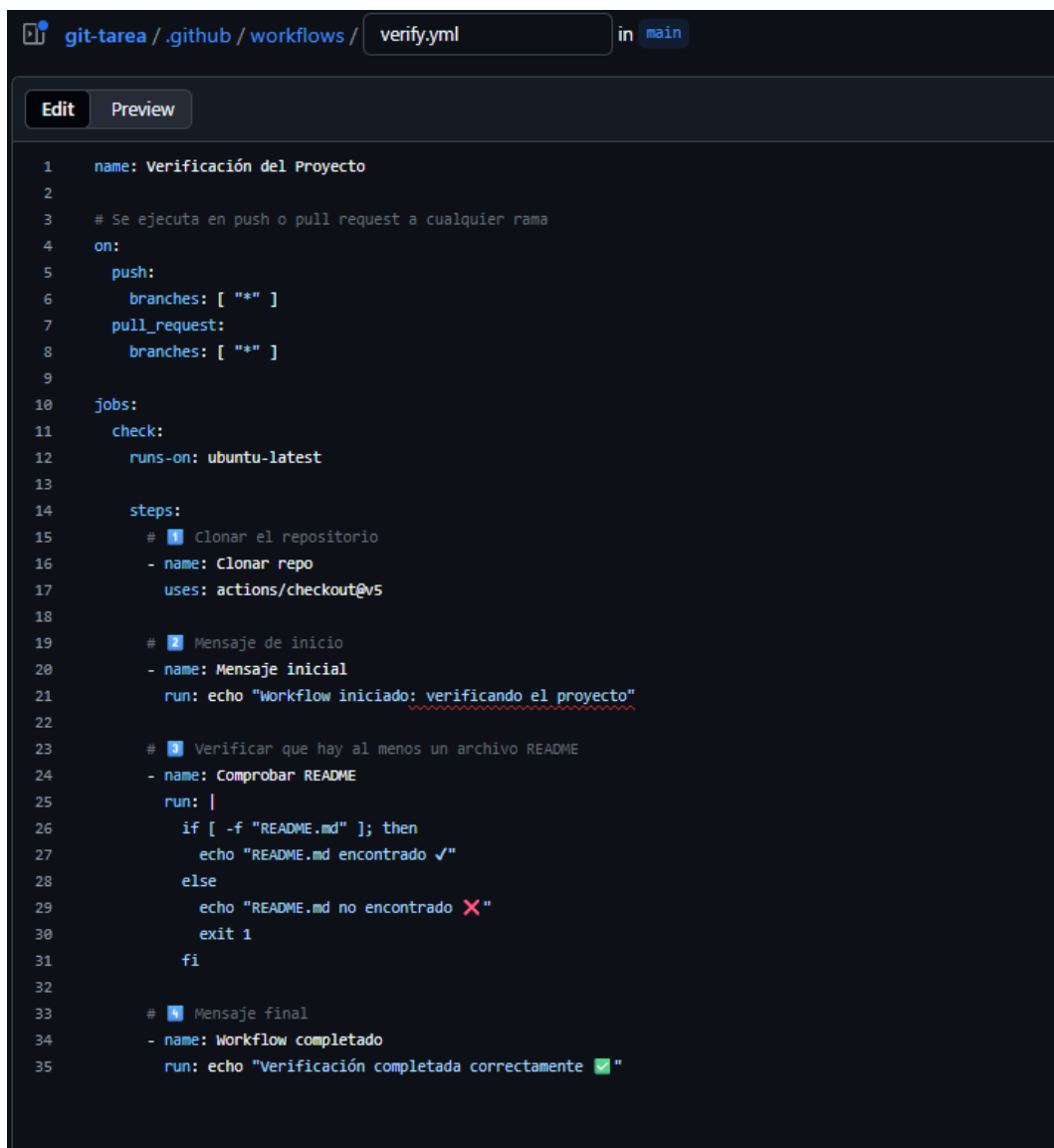
alex1@Alexportatil MINGW64 ~/tarea-kanban (main)
$ mkdir .github/workflow
mkdir: cannot create directory '.github/workflow': No such file or directory

alex1@Alexportatil MINGW64 ~/tarea-kanban (main)
$ mkdir -p .github/workflows/verify.yml

alex1@Alexportatil MINGW64 ~/tarea-kanban (main)
$ cd .github

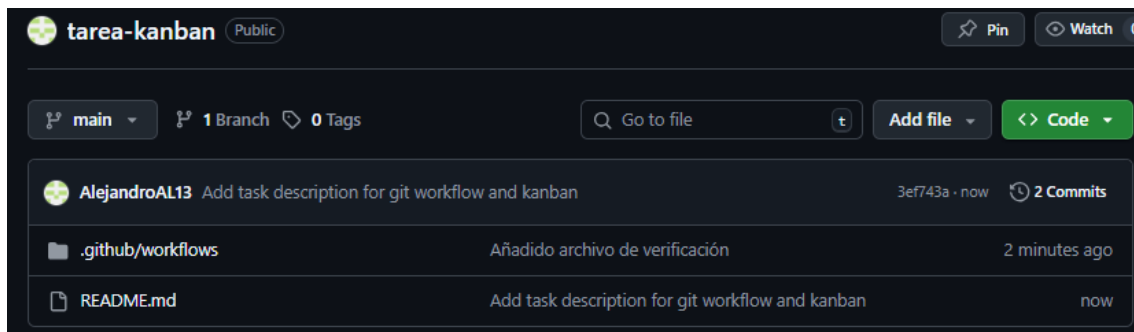
alex1@Alexportatil MINGW64 ~/tarea-kanban/.github (main)
$ cd workflows/

alex1@Alexportatil MINGW64 ~/tarea-kanban/.github/workflows (main)
$ git add verify.yml
```



```
1  name: Verificación del Proyecto
2
3  # Se ejecuta en push o pull request a cualquier rama
4  on:
5    push:
6      branches: [ "*" ]
7    pull_request:
8      branches: [ "*" ]
9
10 jobs:
11   check:
12     runs-on: ubuntu-latest
13
14     steps:
15       # 📄 Clonar el repositorio
16       - name: Clonar repo
17         uses: actions/checkout@v5
18
19       # 🗉 Mensaje de inicio
20       - name: Mensaje inicial
21         run: echo "Workflow iniciado: verificando el proyecto"
22
23       # 📖 Verificar que hay al menos un archivo README
24       - name: Comprobar README
25         run: |
26           if [ -f "README.md" ]; then
27             echo "README.md encontrado ✓"
28           else
29             echo "README.md no encontrado ✗"
30             exit 1
31           fi
32
33       # 📄 Mensaje final
34       - name: Workflow completado
35         run: echo "Verificación completada correctamente ✅"
```

Aquí podemos comprobar que los primeros pasos que se han realizado correctamente.



En el apartado proyectos de nuestro repositorio creamos uno nuevo, elegimos la opción de Kanban. Por lo que podemos observar este tipo de proyecto es uno que nos ayudará a mantener un orden pudiendo ver que tareas faltan por empezar, termina o la que ya estén acabadas, de igual forma que podremos ver a quien se le ha asignado cada tarea.

Arrastrando una tarea de una columna a otra podremos ver como avanzan, se pueden crear más columnas para aclarar si una tarea está en progreso o se acaba de empezar a completar.

