

# Obligatorio 2 Taller de Programación

---

## Integrantes:

- Esteban Moroy - Nro. 338885
- Bruno Riera
- Ignacio Martínez

## Grupo: N1B

## Problema Elegido: Nro. 28

[Link al repositorio](#)

## Tabla de Contenido

- [Obligatorio 2 Taller de Programación](#)
  - [Tabla de Contenido](#)
  - [Presentación](#)
    - [Presentación del problema](#)
    - [¿Qué es GitHub?](#)
    - [¿Qué es GitHub Issues?](#)
    - [¿Para qué sirven los GitHub Issues?](#)
    - [Buenas Prácticas](#)
  - [Generar un repositorio de git local y subir a GitHub](#)
    - [Crear un repositorio en GitHub](#)
    - [Opcional - Configurar clave SSH](#)
  - [Crear un template de GitHub Issue en formato Markdown \(.md\)](#)
  - [Crear un template de GitHub Issue en formato Yaml \(.yaml\)](#)
  - [Fuentes](#)

## Presentación

### Presentación del problema

28. Problema: La empresa necesita de un mejor seguimiento acerca del reporte de problemas y mejoras en la documentación y el código, así como los bugs en su código.

Solución: En un repositorio en GitHub, implementar "templates" de GitHub Issues para al menos los siguientes diferentes tipos de reportes:

- Documentación
- Mejoras
- Pedido de nueva funcionalidad (Feature request)
- Bugs

Debe investigar buenas prácticas y elementos a incluir para cada tipo de reporte, así como ejemplificar con al menos 2 reportes cada categoría. Para el código, puede servirse de su entrega del Obligatorio 1, a la cual

puede realizarle modificaciones.

## ¿Qué es GitHub?

GitHub es una plataforma de desarrollo colaborativo para alojar proyectos utilizando el sistema de control de versiones Git. GitHub permite a los desarrolladores trabajar en proyectos en equipo, compartir código, rastrear cambios y colaborar en el desarrollo de software.

## ¿Qué es GitHub Issues?

GitHub Issues es una herramienta dentro de un repositorio de GitHub, consiste en un registro para discutir cualquier aspecto del proyecto. Cada issue tiene su propio hilo de discusión, donde los colaboradores pueden comentar, adjuntar archivos y linkear código relacionado. Los issues pueden incluir:

- Bugs: Reportes de errores o problemas en el código.
- Tareas: Tareas que necesitan completarse.
- Mejoras: Solicitudes de nuevas características o mejoras a las existentes.
- Preguntas: Preguntas o discusiones sobre el proyecto.

## ¿Para qué sirven los GitHub Issues?

**Seguimiento de Errores (Bugs):** Permiten a los usuarios y desarrolladores reportar errores que encuentran en el software, describirlos en detalle y proporcionar pasos para reproducirlos. Esto ayuda al equipo de desarrollo a identificar y solucionar problemas.

**Planificación de Características:** Ayudan a organizar y priorizar nuevas características o mejoras. Los colaboradores pueden discutir cómo implementar una nueva característica, sugerir cambios y asignar tareas específicas a los miembros del equipo.

**Gestión de Proyectos:** Facilitan la gestión de proyectos al permitir que las tareas se dividan en issues individuales, que luego pueden ser asignados a diferentes colaboradores. Esto proporciona una visión clara del progreso y lo que aún queda por hacer.

**Discusión y Colaboración:** Fomentan la discusión y colaboración entre los miembros del equipo. Los issues permiten que todos los involucrados en el proyecto contribuyan con ideas, revisen el trabajo de otros y propongan soluciones.

**Documentación de Decisiones:** Actúan como un registro de las decisiones tomadas durante el desarrollo del proyecto. Esto es útil para referirse en el futuro y entender por qué se tomaron ciertas decisiones.

**Automatización:** Pueden integrarse con otras herramientas y servicios de GitHub, como GitHub Actions, para automatizar flujos de trabajo. Por ejemplo, un issue puede cerrarse automáticamente cuando se fusiona un pull request que soluciona el problema descrito en el issue.

## Buenas Prácticas

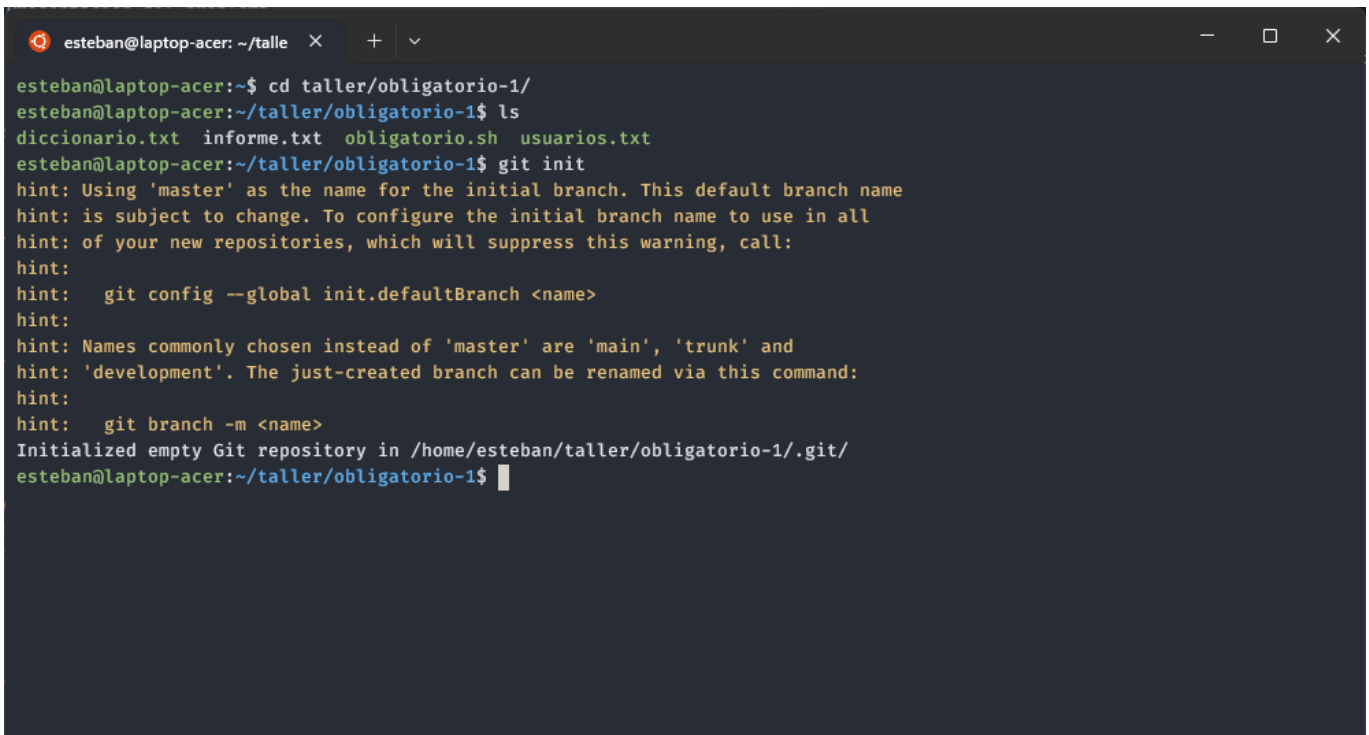
- **Estandarización de títulos:** Utilizar títulos claros y consistentes facilita la identificación rápida del problema o la mejora solicitada.
- **Pasos reproducibles:** Proporcionar pasos detallados para reproducir el problema permite a los desarrolladores entender exactamente cómo se puede manifestar el error o la necesidad de mejora.

- **Uso de etiquetas (Labels):** Las etiquetas categorizan los issues según su tipo (bug, mejora, documentación), estado (abierto, en progreso, cerrado) o prioridad (alta, media, baja), lo que ayuda a priorizar y gestionar eficazmente el trabajo.
- **Priorización e impacto:** Especificar la prioridad y el impacto del problema o la mejora ayuda a los equipos a asignar recursos adecuadamente y abordar primero los elementos más críticos.
- **Descripciones detalladas:** Proporcionar descripciones completas y detalladas del problema o la mejora solicitada ayuda a los desarrolladores a entender el contexto y la naturaleza del cambio necesario.
- **Alternativas consideradas:** Incluir alternativas o enfoques sugeridos muestra que se ha pensado de manera crítica sobre la solución y puede llevar a discusiones más informadas sobre la mejor manera de abordar el problema.
- **Contexto y justificación:** Explicar el contexto en el que se produce el problema o la razón por la cual se necesita la mejora proporciona una comprensión más profunda a los desarrolladores y a otros interesados.
- **Documentación completa:** Mantener la documentación actualizada y accesible asegura que todos los miembros del equipo estén informados y puedan colaborar eficazmente en la resolución de problemas y mejoras.

## Generar un repositorio de git local y subir a GitHub

1. Crear un repositorio de git local en la carpeta del proyecto

```
git init
```

A screenshot of a terminal window with a dark background. The window title is 'esteban@laptop-acer: ~/taller'. The terminal shows the following commands and output: 

```
esteban@laptop-acer:~$ cd taller/obligatorio-1/
esteban@laptop-acer:~/taller/obligatorio-1$ ls
diccionario.txt  informe.txt  obligatorio.sh  usuarios.txt
esteban@laptop-acer:~/taller/obligatorio-1$ git init
hint: Using 'master' as the name for the initial branch. This default branch name
hint: is subject to change. To configure the initial branch name to use in all
hint: of your new repositories, which will suppress this warning, call:
hint:
hint:   git config --global init.defaultBranch <name>
hint:
hint: Names commonly chosen instead of 'master' are 'main', 'trunk' and
hint: 'development'. The just-created branch can be renamed via this command:
hint:
hint:   git branch -m <name>
Initialized empty Git repository in /home/esteban/taller/obligatorio-1/.git/
esteban@laptop-acer:~/taller/obligatorio-1$
```

2. Agregar los archivos al repositorio

```
git add .
```

### 3. Agregar .gitignore para los archivos que se generan durante la ejecución

```
touch .gitignore
echo "usuarios.txt" >> .gitignore
echo "informe.txt" >> .gitignore
```

```
Initialized empty Git repository in /home/esteban/taller/obligatorio-1/.git/
esteban@laptop-acer:~/taller/obligatorio-1$ git add .
esteban@laptop-acer:~/taller/obligatorio-1$ touch .gitignore
esteban@laptop-acer:~/taller/obligatorio-1$ echo "usuarios.txt" >> .gitignore
esteban@laptop-acer:~/taller/obligatorio-1$ echo "informe.txt" >> .gitignore
esteban@laptop-acer:~/taller/obligatorio-1$ cat .gitignore
usuarios.txt
informe.txt
esteban@laptop-acer:~/taller/obligatorio-1$
```

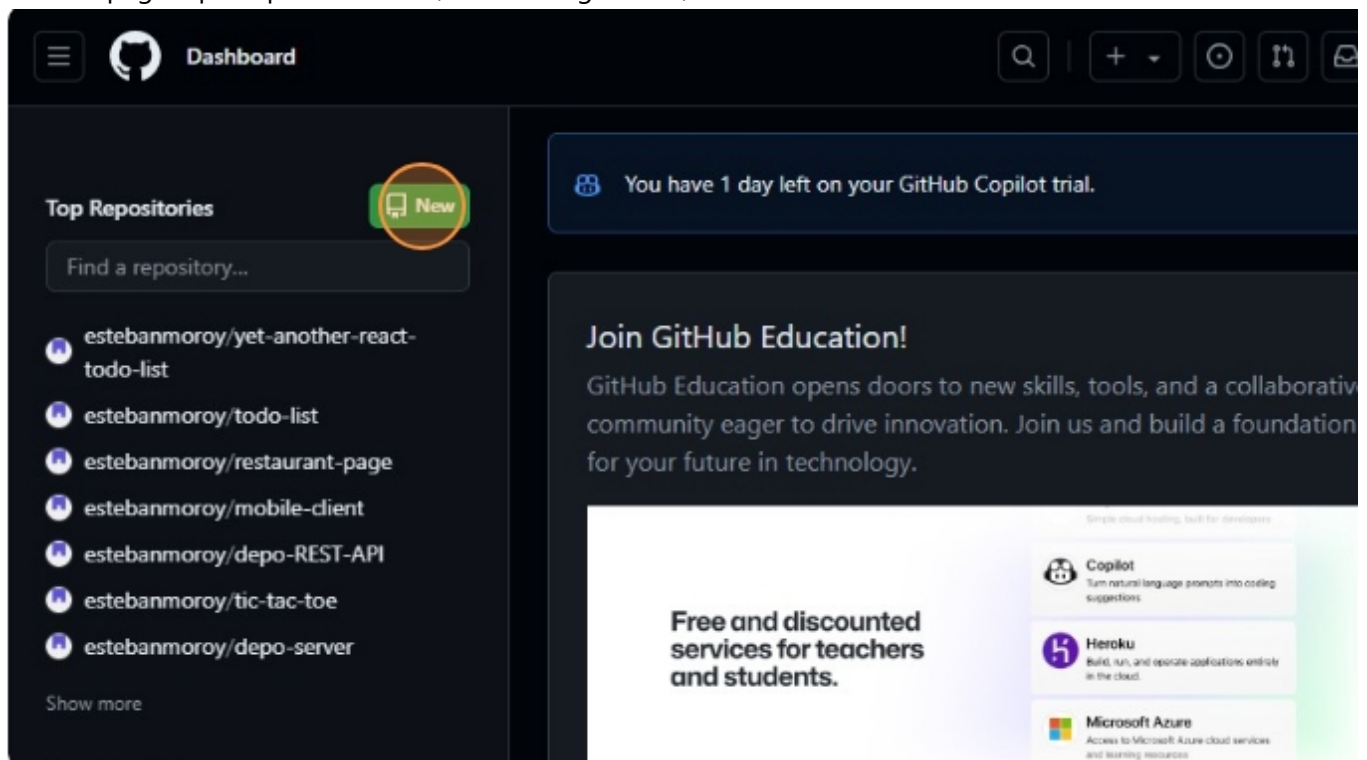
### 4. Crear un commit

```
git commit -m "Primer Commit"
```

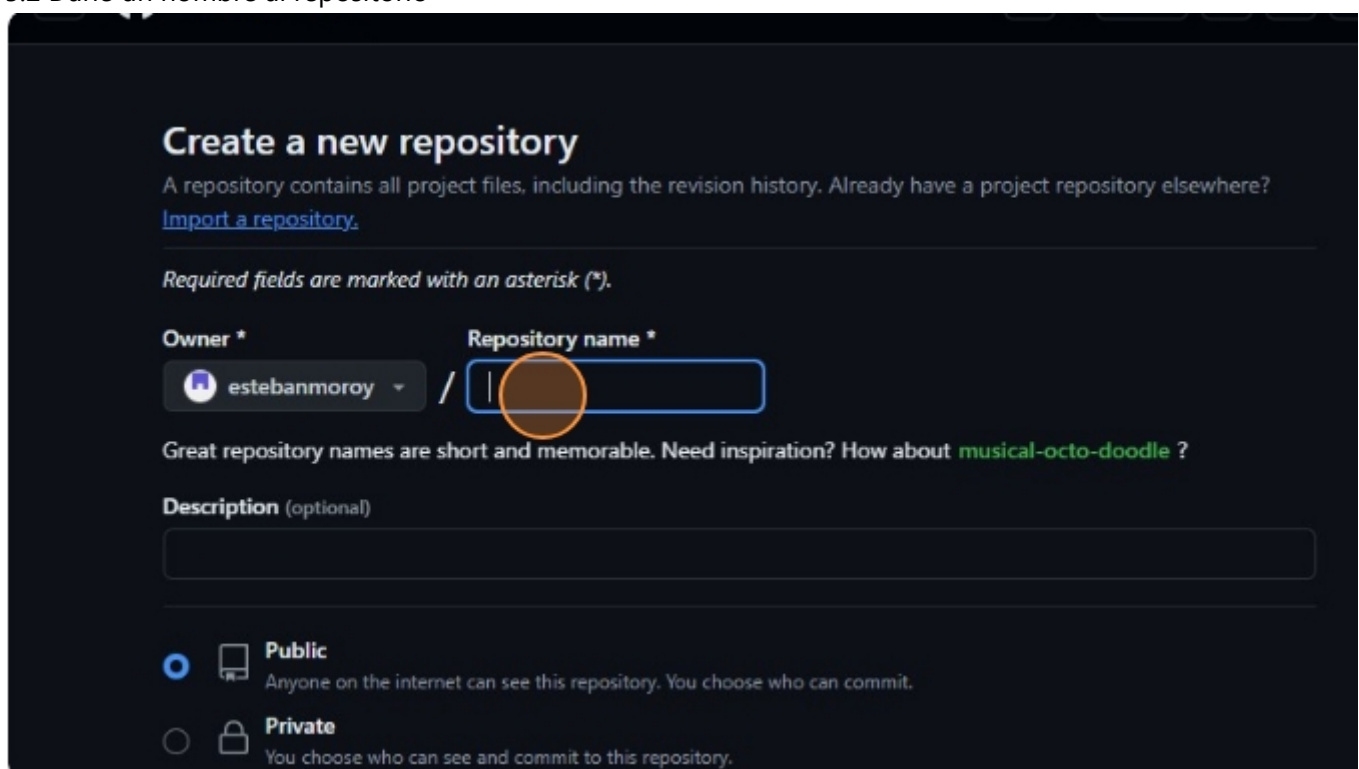
```
esteban@laptop-acer: ~/talle X + v
esteban@laptop-acer:~/taller/obligatorio-1$ git commit -m "Primer Commit"
[master (root-commit) 636dc29] Primer Commit
4 files changed, 175036 insertions(+)
create mode 100755 diccionario.txt
create mode 100644 informe.txt
create mode 100755 obligatorio.sh
create mode 100755 usuarios.txt
esteban@laptop-acer:~/taller/obligatorio-1$
```

## Crear un repositorio en GitHub

5.1 En la página principal de GitHub, una vez logueados, clickear en "New"




5.2 Darle un nombre al repositorio



### 5.3 Darle una descripción al repositorio

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)


Required fields are marked with an asterisk (\*).


Owner \*  **estebanmoroy** / Repository name \*

✔ obligatorio-1-taller is available.

Great repository names are short and memorable. Need inspiration? How about **musical-octo-doodle** ?

Description (optional)

☒  **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.

Initialize this repository with:

☐ Add a README file

### 5.4 Clickear en "Crear Nuevo Repositorio"

Initialize this repository with:

☐ Add a README file  
This is where you can write a long description for your project. [Learn more about READMEs.](#)

Add .gitignore

.gitignore template: **None** ▾

Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

Choose a license

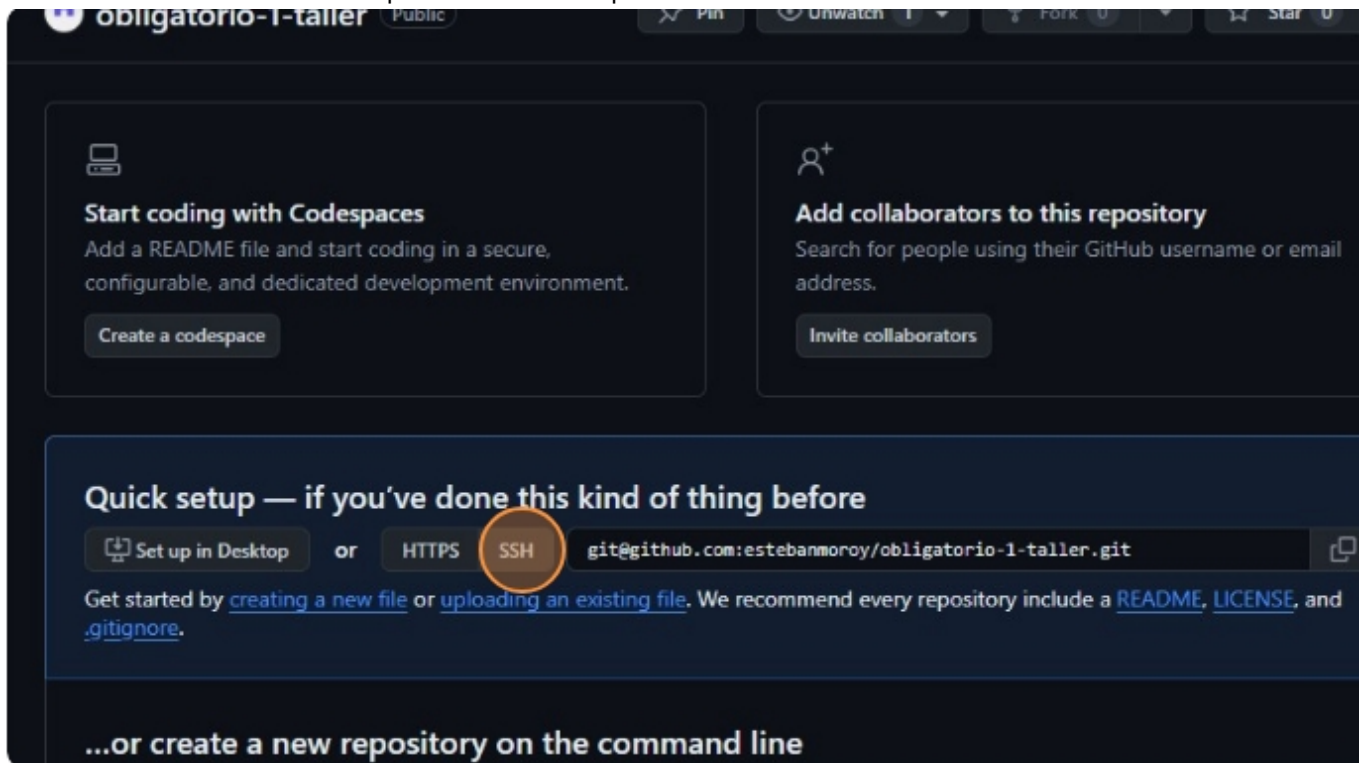
License: **None** ▾

A license tells others what they can and can't do with your code. [Learn more about licenses.](#)

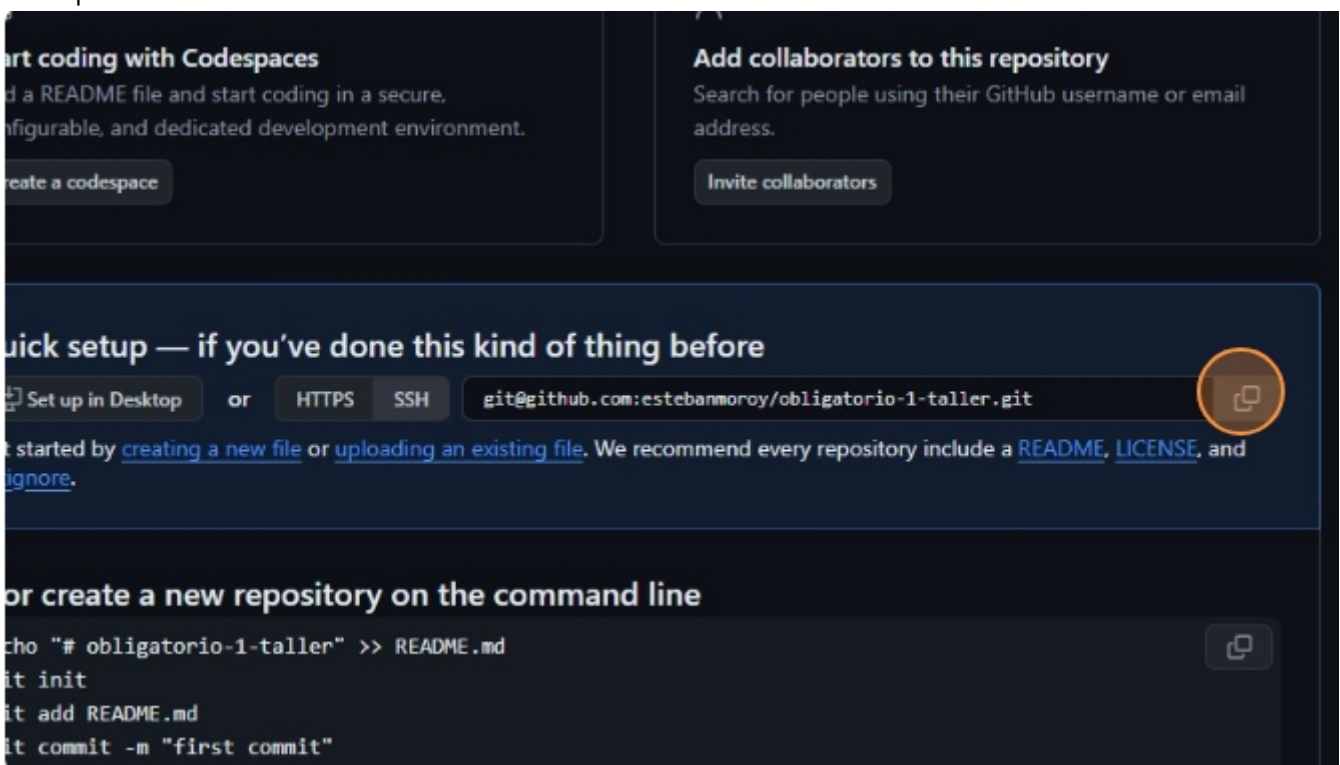
① You are creating a public repository in your personal account.

**Create repository**

5.5 En la sección de "Quick setup" seleccionar la opción SSH



5.6 Copiar la dirección SSH



5.7 Agregar la dirección SSH del repositorio remoto copiada

```
git remote add origin
```

Opcional - Configurar clave SSH



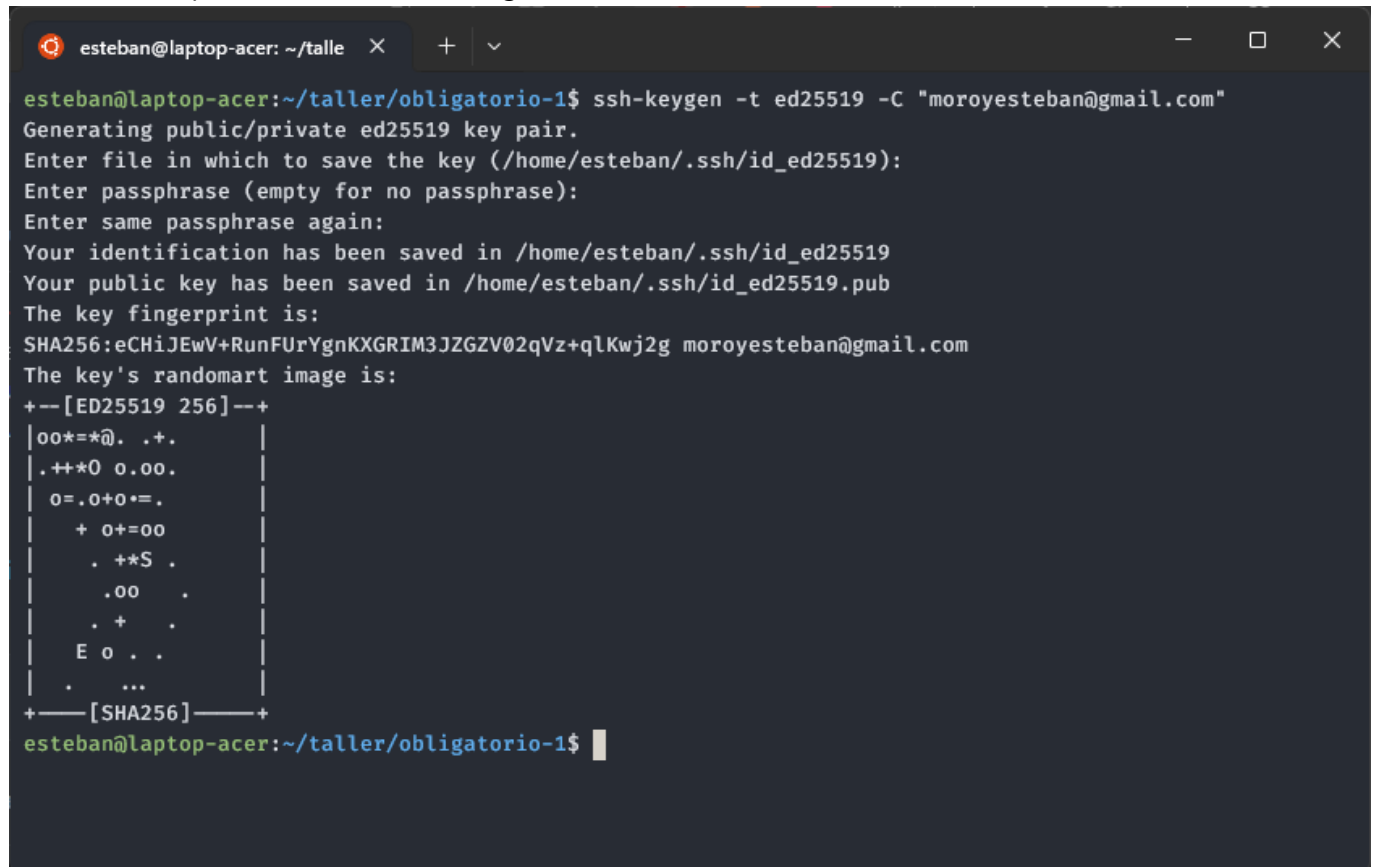
En caso de que no se tenga configurada una clave de autenticación remota SSH, se muestran los pasos para configurar una. Se realizó siguiendo instructivo publicado en GitHub:

### Generación de una nueva clave SSH y adición al agente SSH

#### 6.1 Generar llave SSH con el correo electrónico registrado en GitHub

```
ssh-keygen -t ed25519 -C "your_email@example.com"
```

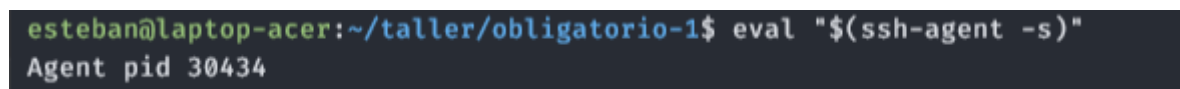
#### 6.2 Cuando se pida, escribir una frase segura como contraseña



```
esteban@laptop-acer: ~/taller
esteban@laptop-acer:~/taller/obligatorio-1$ ssh-keygen -t ed25519 -C "moroyesteban@gmail.com"
Generating public/private ed25519 key pair.
Enter file in which to save the key (/home/esteban/.ssh/id_ed25519):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/esteban/.ssh/id_ed25519
Your public key has been saved in /home/esteban/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:eCHiJEwV+RunFUrYgnKXGRIM3JZGZV02qVz+qlKwj2g moroyesteban@gmail.com
The key's randomart image is:
+--[ED25519 256]--+
|00*=*@. .+.
|.++*0 0.00.
| 0=.0+0+=.
|  + 0+=00
|  . ++S .
|  .00 .
|  . + .
|  E o . .
|  . ...
+-----[SHA256]-----+
esteban@laptop-acer:~/taller/obligatorio-1$
```

#### 6.3 Verificar si la clave se generó correctamente

```
$ eval "$(ssh-agent -s)"
```



```
esteban@laptop-acer:~/taller/obligatorio-1$ eval "$(ssh-agent -s)"
Agent pid 30434
```

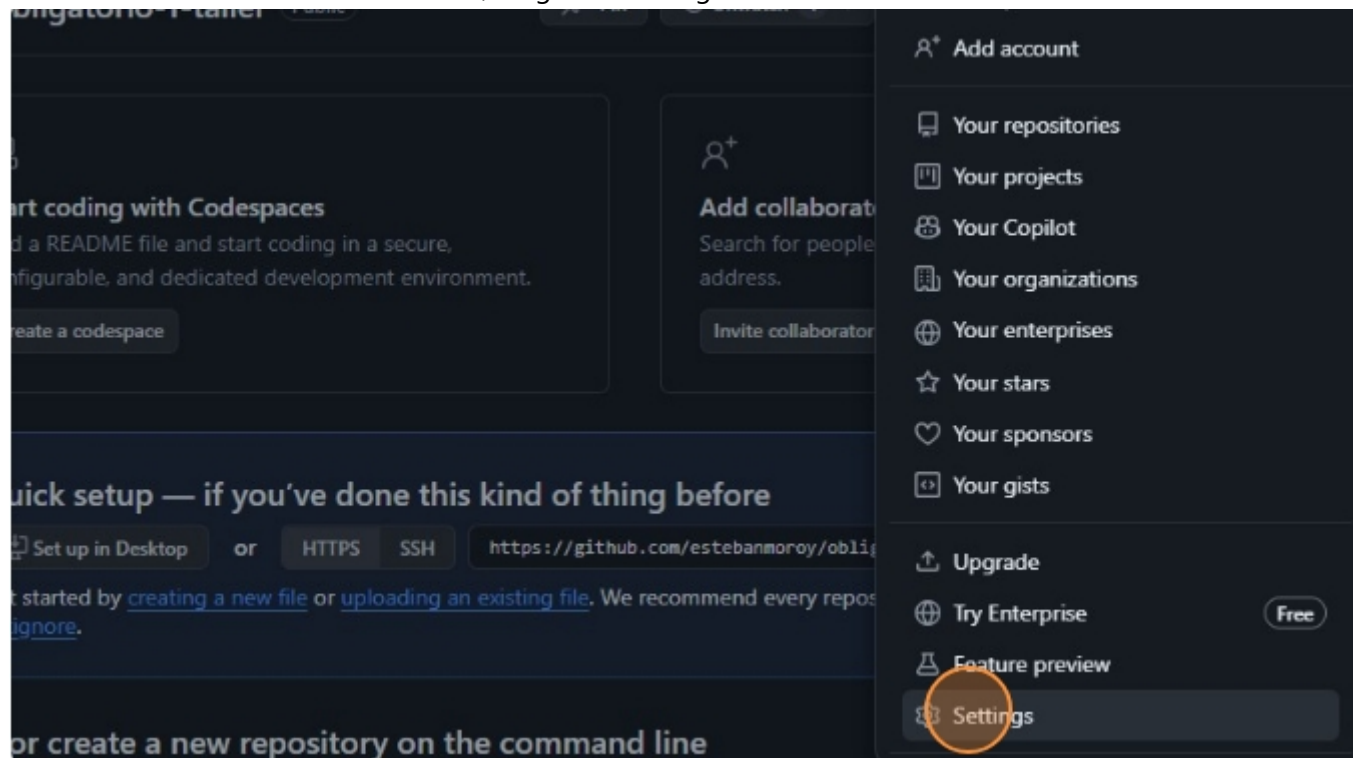
#### 6.4 Agregar la llave al ssh-agent

```
ssh-add ~/.ssh/id_ed25519
```

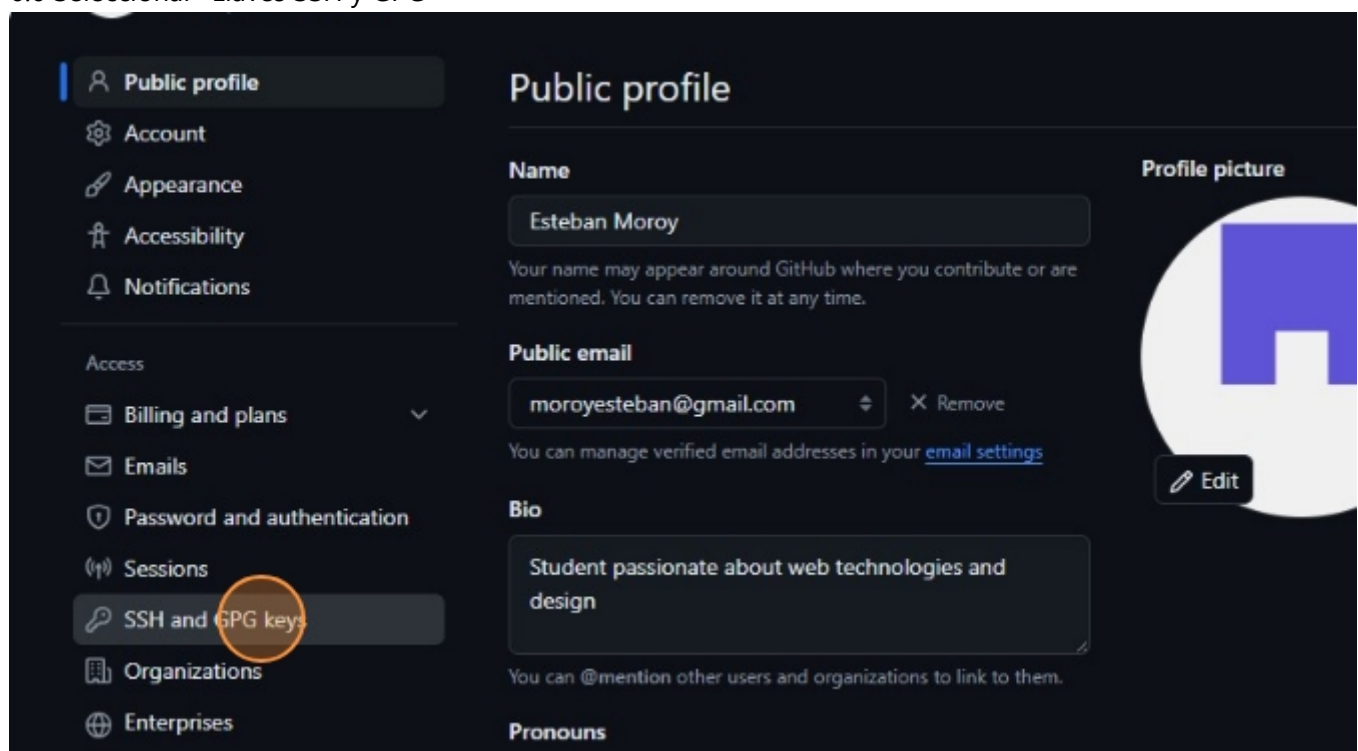


```
esteban@laptop-acer:~/taller/obligatorio-1$ ls -al ~/.ssh
total 20
drwx----- 2 esteban esteban 4096 Jun 16 22:20 .
drwxr-x--- 6 esteban esteban 4096 Jun 16 22:04 ..
-rw----- 1 esteban esteban  464 Jun 16 22:20 id_ed25519
-rw-r--r-- 1 esteban esteban  104 Jun 16 22:20 id_ed25519.pub
-rw-r--r-- 1 esteban esteban  142 Jun 16 22:04 known_hosts
esteban@laptop-acer:~/taller/obligatorio-1$ ssh-add ~/.ssh/id_ed25519
Enter passphrase for /home/esteban/.ssh/id_ed25519:
Identity added: /home/esteban/.ssh/id_ed25519 (moroyesteban@gmail.com)
esteban@laptop-acer:~/taller/obligatorio-1$
```

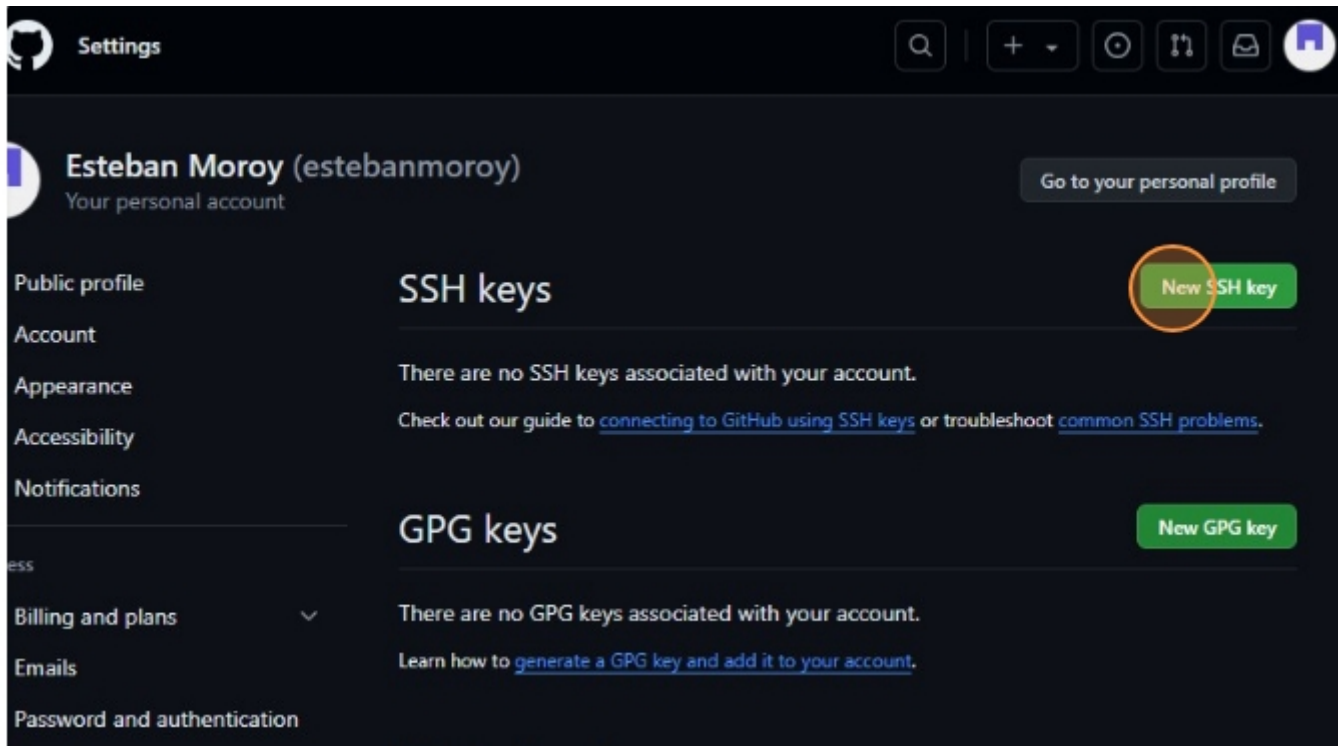
6.5 En la sección de usuario de GitHub, dirigirse a Settings



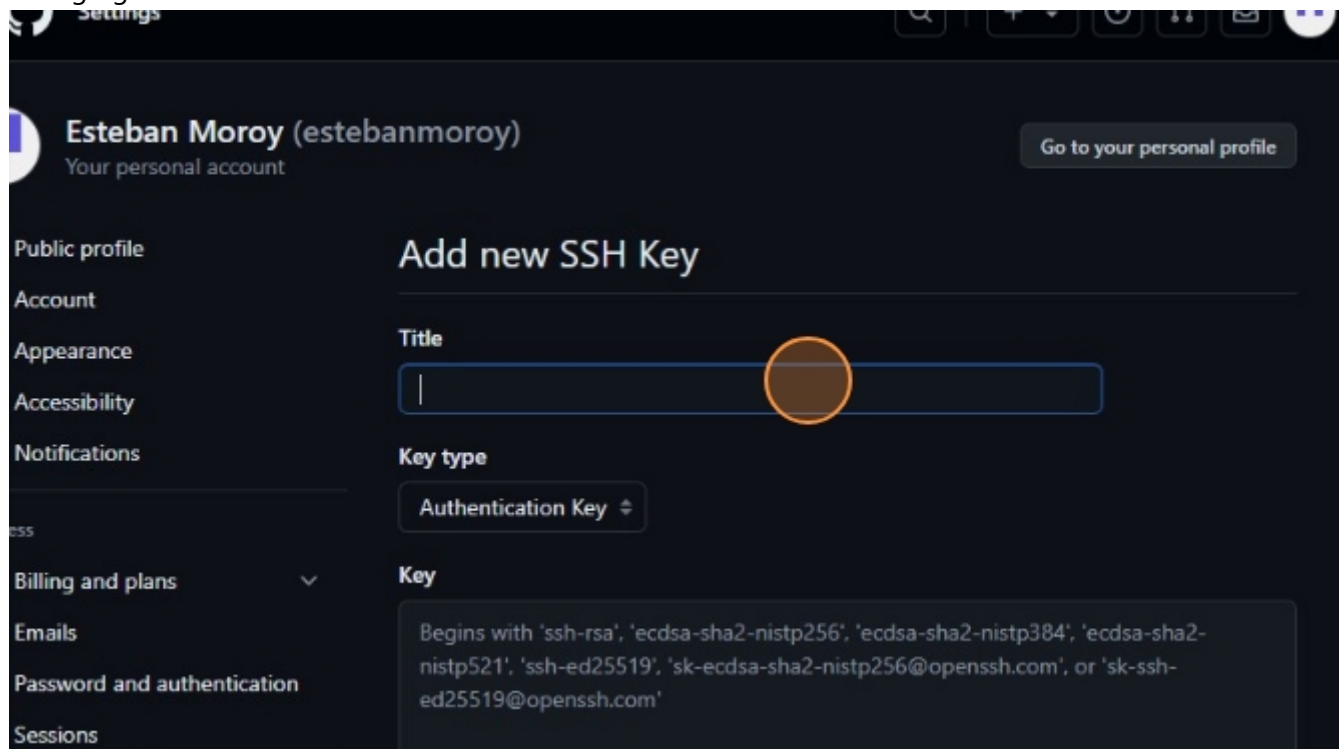
6.6 Seleccionar "Llaves SSH y GPG"



## 6.7 Seleccionar "Nueva llave SSH"



## 6.8 Agregar un nombre a la llave



## 6.9 En consola usar el comando debajo y pegar el resultado en el campo "Llave"

```
cat ~/.ssh/id_ed25519.pub
```

Public profile

Account

Appearance

Accessibility

Notifications

Billing and plans

Emails

Password and authentication

Sessions

**SSH and GPG keys**

Organizations

Enterprises

## Add new SSH Key

**Title**

laptop-acer

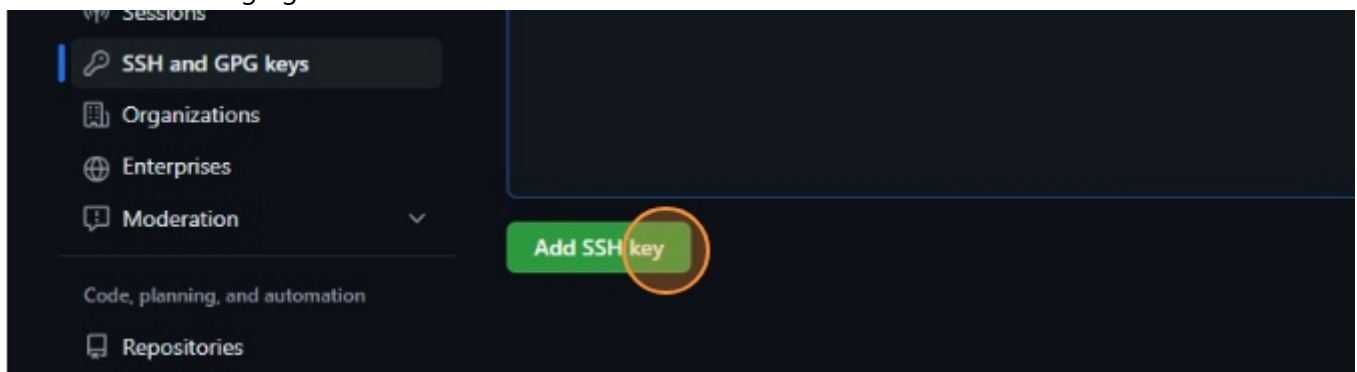
**Key type**

Authentication Key

**Key**

Begins with 'ssh-rsa', 'ecdsa-sha2-nistp256', 'ecdsa-sha2-nistp384', 'ecdsa-sha2-nistp521', 'ssh-ed25519', 'sk-ecdsa-sha2-nistp256@openssh.com', or 'sk-ssh-ed25519@openssh.com'

### 6.10 Seleccionar "Agregar llave SSH"

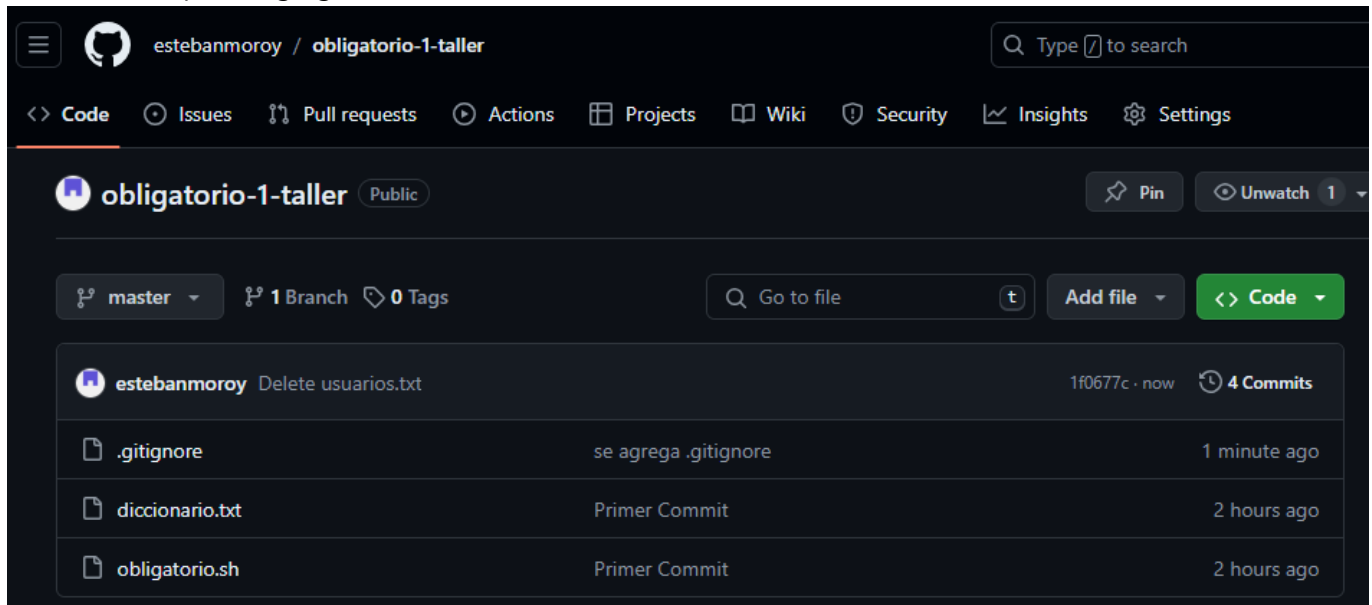


### 6.11 Subir los archivos al repositorio remoto

```
git push -u origin master
```

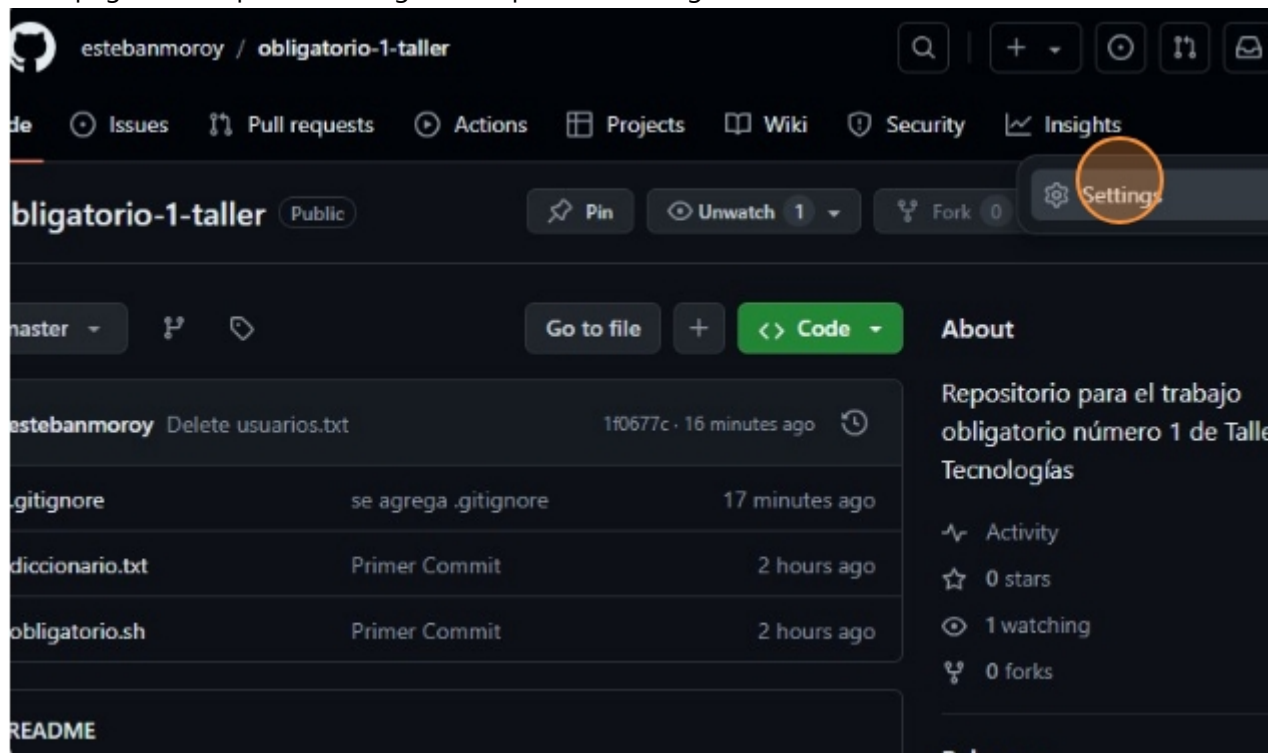
```
esteban@laptop-acer:~/taller/obligatorio-1$ git remote add origin git@github.com:estebanmoroy/obligatorio-1-taller.git
esteban@laptop-acer:~/taller/obligatorio-1$ git push -u origin master
Enumerating objects: 6, done.
Counting objects: 100% (6/6), done.
Delta compression using up to 4 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (6/6), 412.68 KiB | 1.79 MiB/s, done.
Total 6 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:estebanmoroy/obligatorio-1-taller.git
 * [new branch]      master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.
esteban@laptop-acer:~/taller/obligatorio-1$
```

## 6.12 Verificar que se agregaron los archivos correctamente

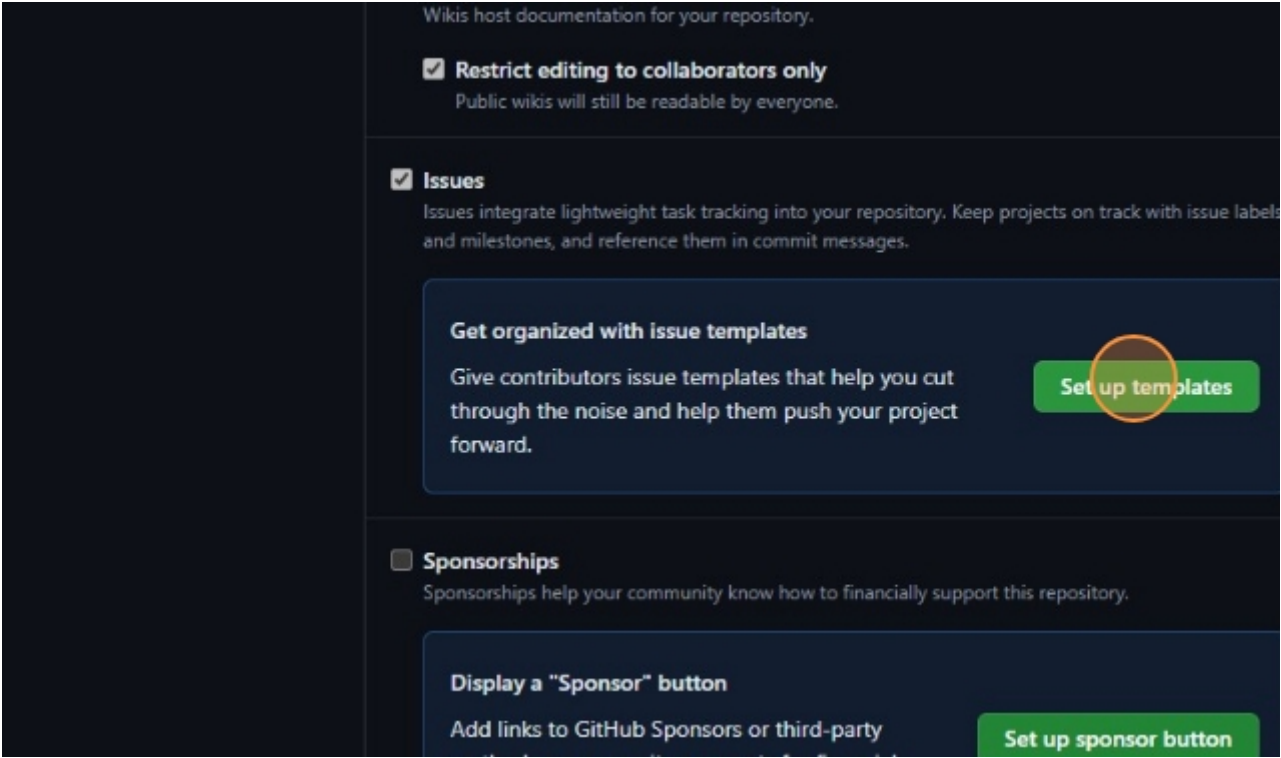


## Crear un template de GitHub Issue en formato Markdown (.md)

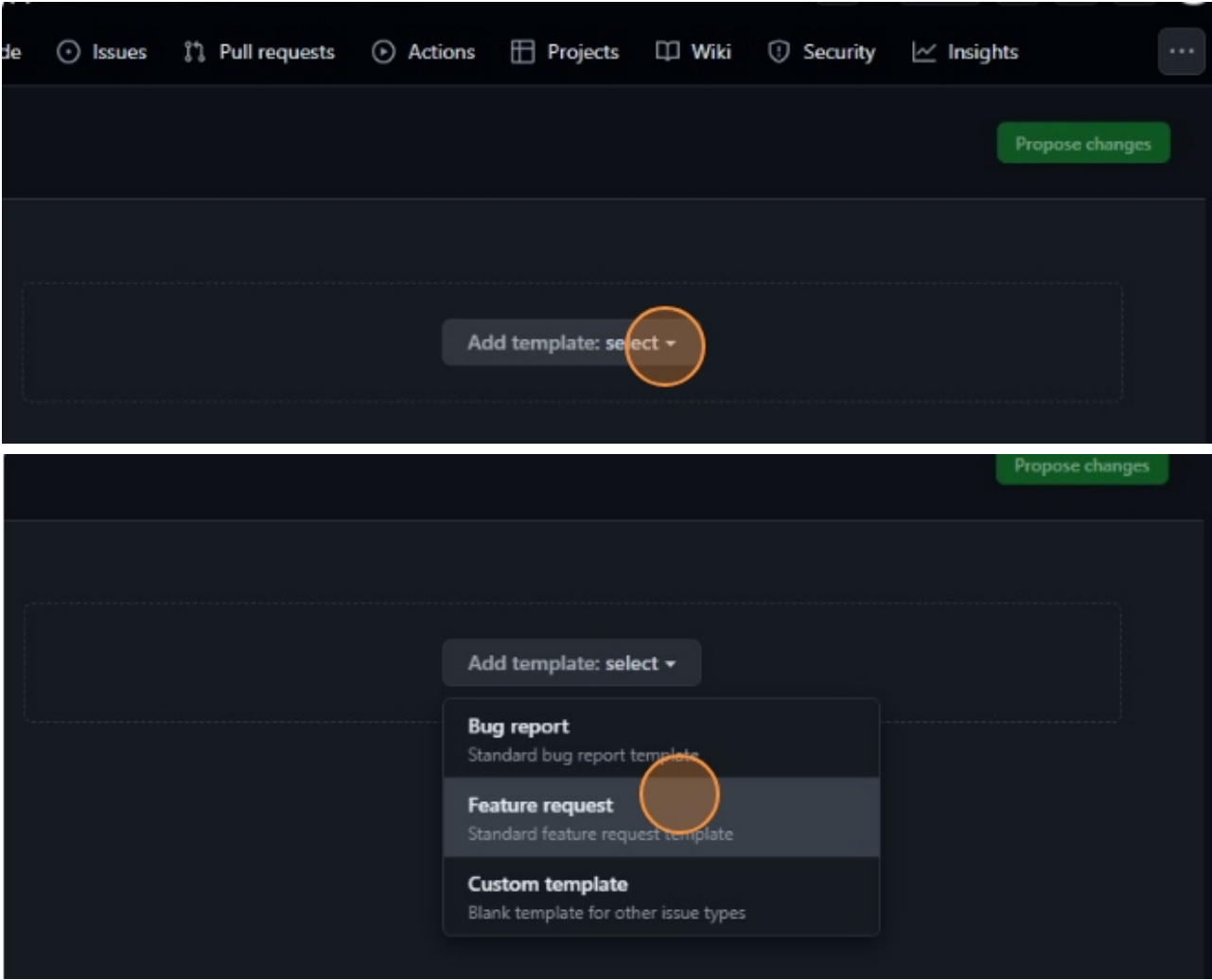
1. En la página del repositorio dirigirse a la pestaña Settings



2. En la sección de Issues, seleccionar Configurar Templates

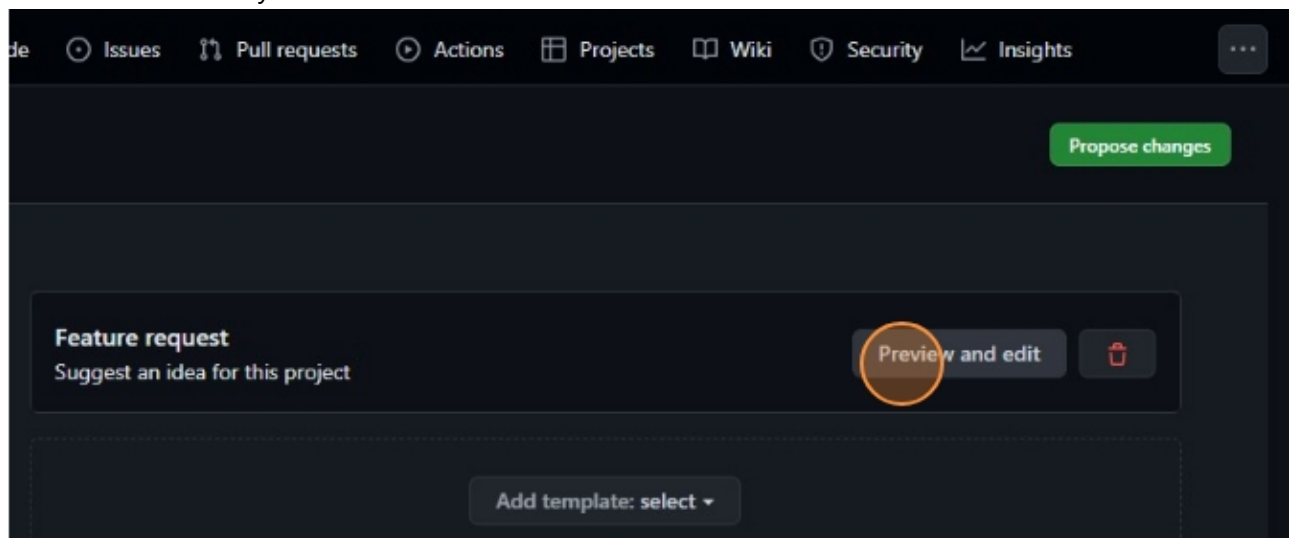


3. Seleccionar el tipo de template a configurar

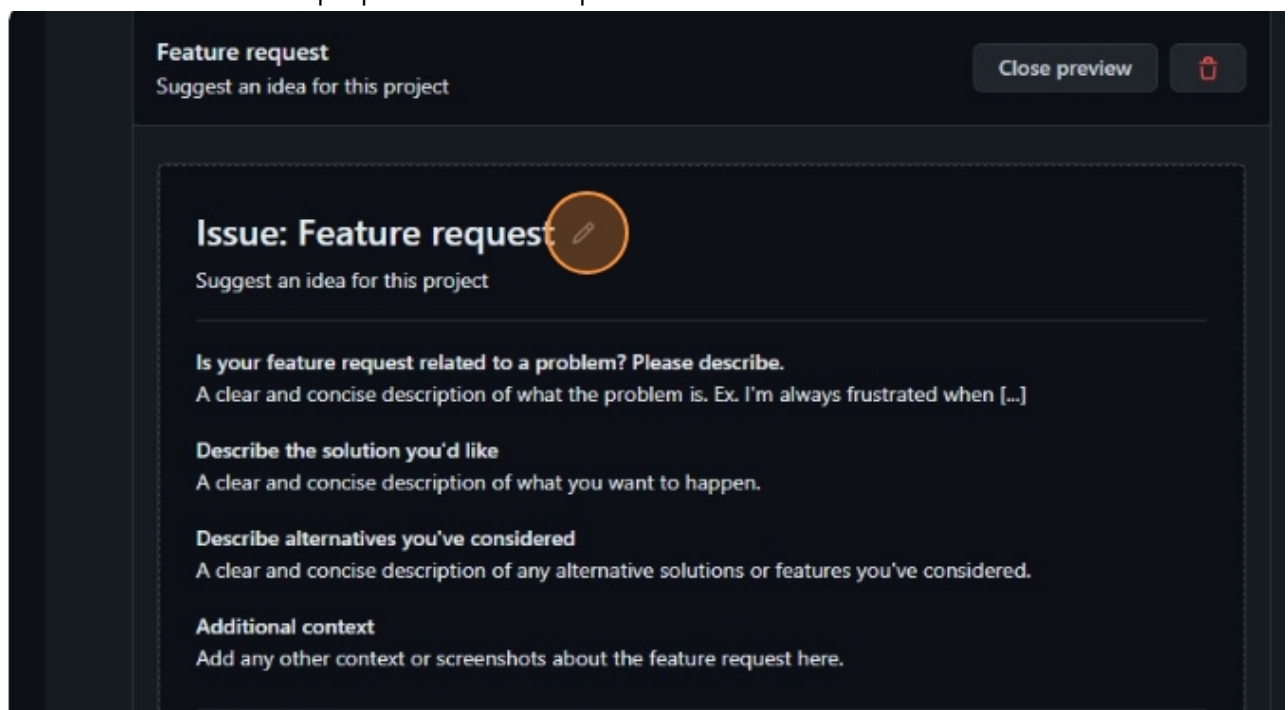




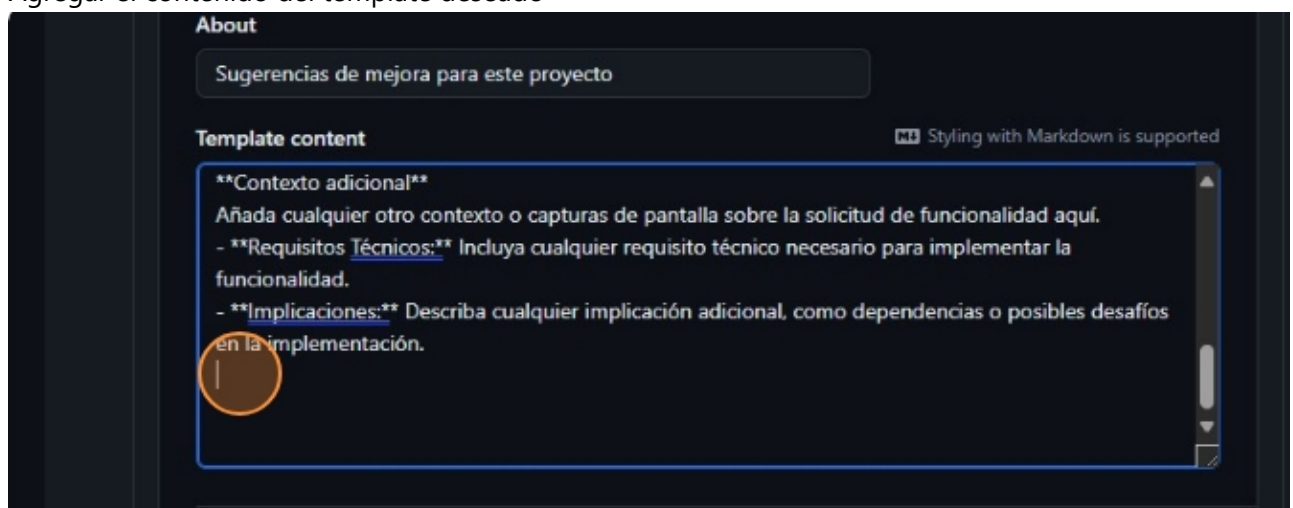
## 4. Seleccionar Preview y Editar



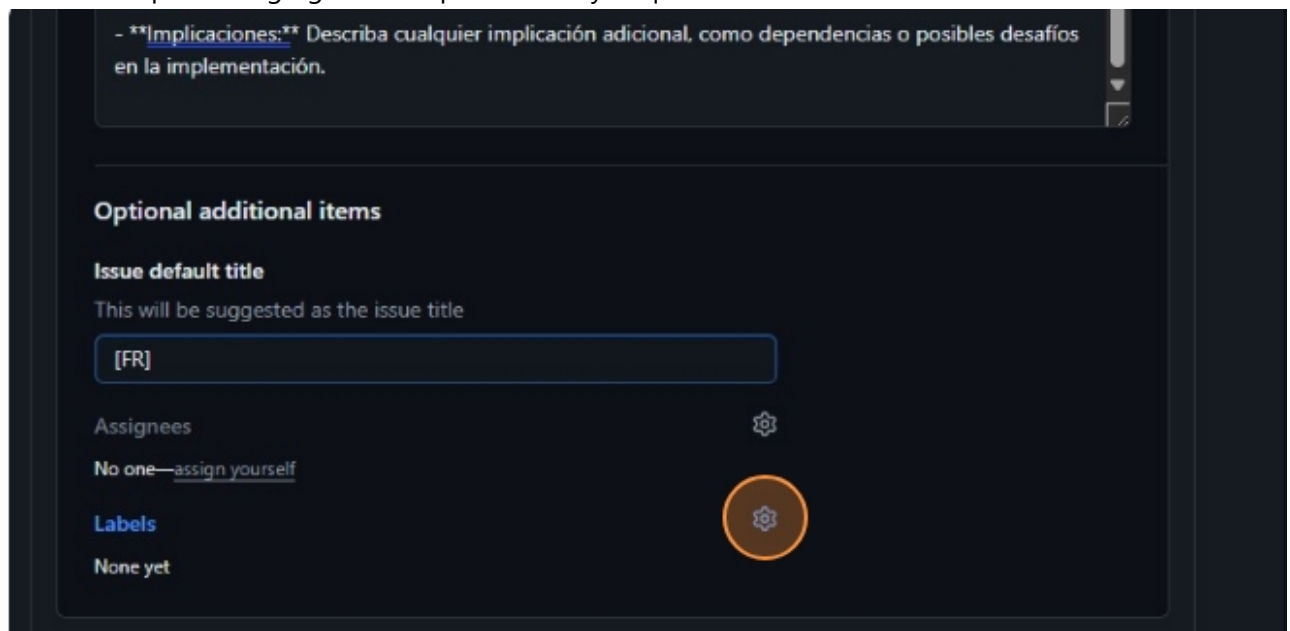
## 5. Clickear en el ícono del lápiz para editar el template



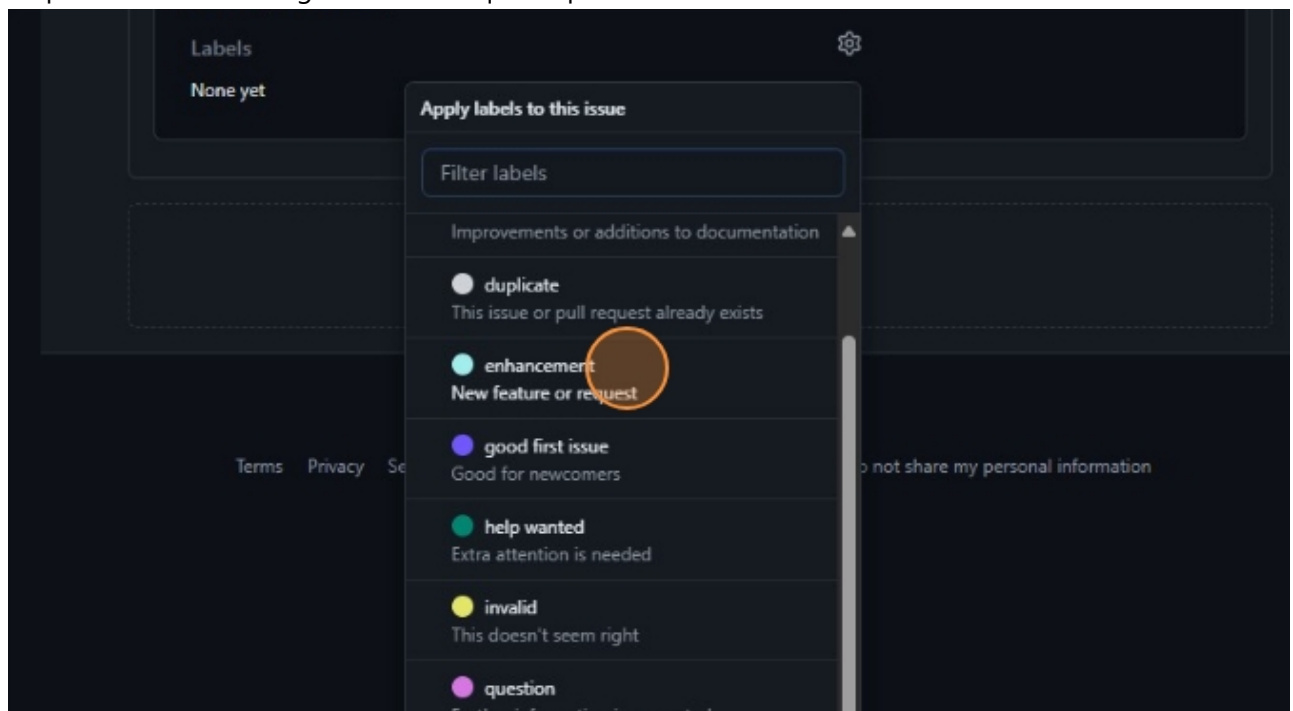
## 6. Agregar el contenido del template deseado



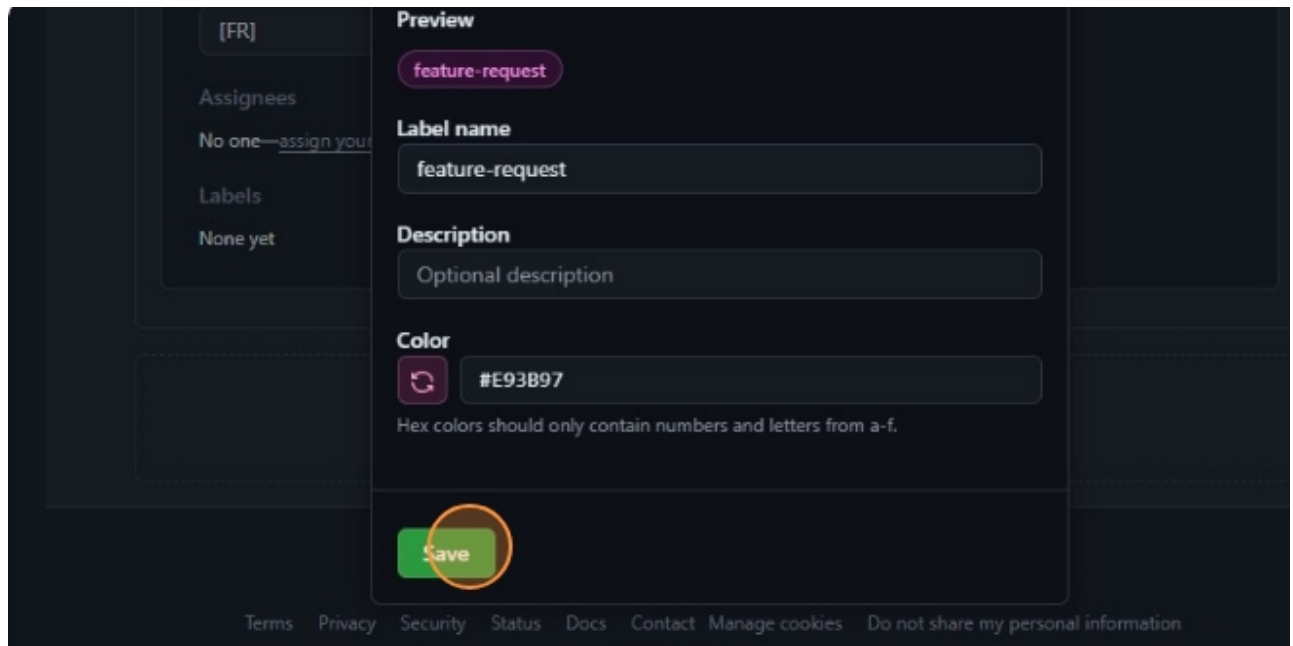
7. También se pueden agregar títulos por defecto y etiquetas



8. Se puede seleccionar alguna de las etiquetas predeterminadas







[FR]

Assignees

No one—[assign yourself](#)

Labels

None yet

**Preview**

feature-request


**Label name**

feature-request

**Description**

Optional description

**Color**

 #E93B97

Hex colors should only contain numbers and letters from a-f.

Save

[Terms](#) [Privacy](#) [Security](#) [Status](#) [Docs](#) [Contact](#) [Manage cookies](#) [Do not share my personal information](#)

9. O agregar una nueva customizada

assign yourself'. The 'Labels' section shows 'None yet'. A settings gear icon is circled in orange. A modal titled 'Apply labels to this issue' is open, showing 'feature-request' as a selected label with a checkmark and a pink dot, and an 'Edit labels' link at the bottom." data-bbox="118 369 932 676"/>

**Optional additional items**

**Issue default title**

This will be suggested as the issue title

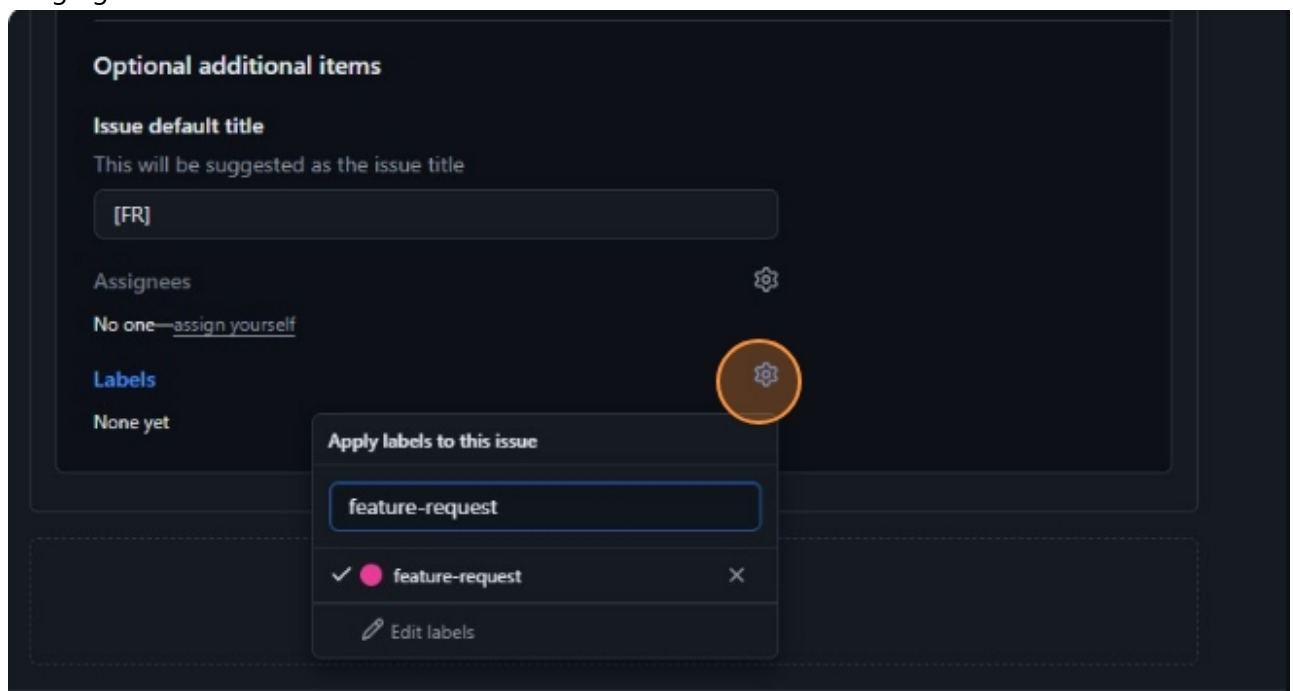
[FR]


Assignees

No one—[assign yourself](#)

**Labels**



None yet






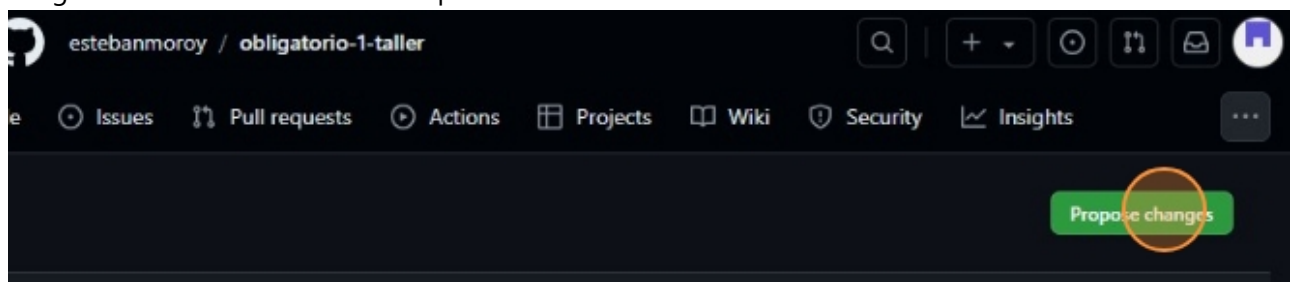
**Apply labels to this issue**

feature-request

✓  feature-request 

 Edit labels

10. Luego de finalizado clicar en Proponer cambios

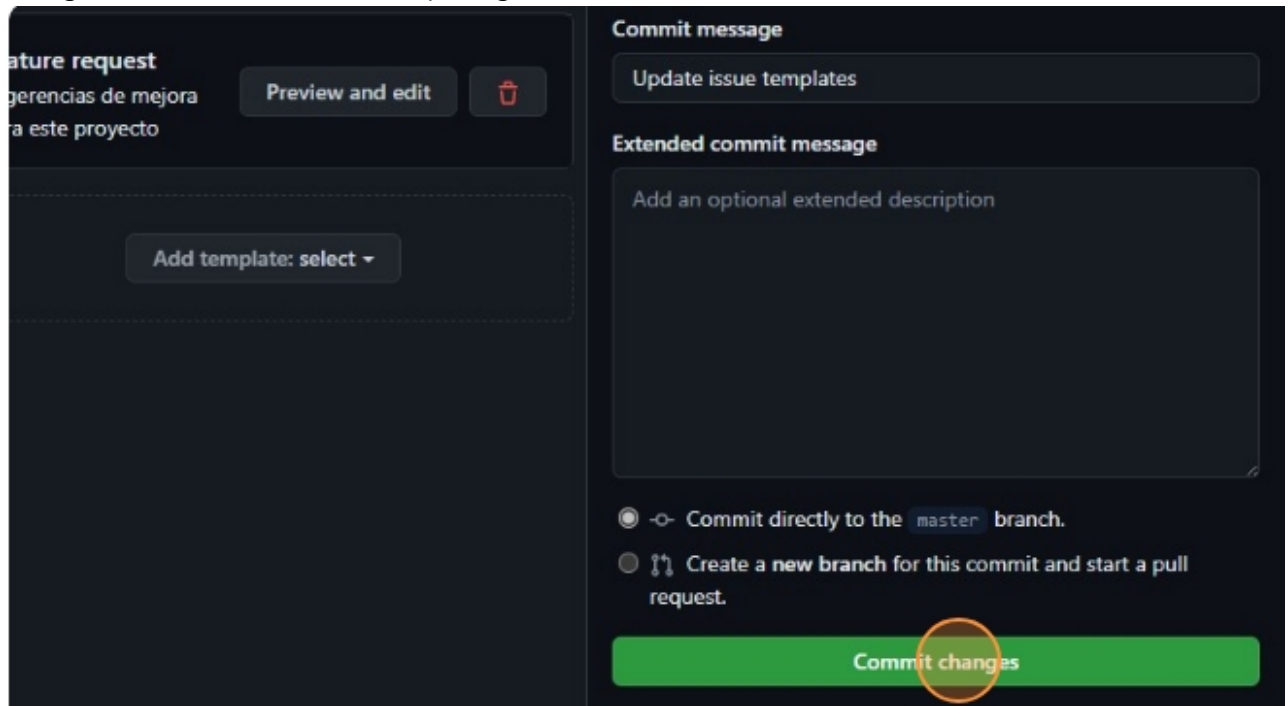


estebanmoroy / obligatorio-1-taller

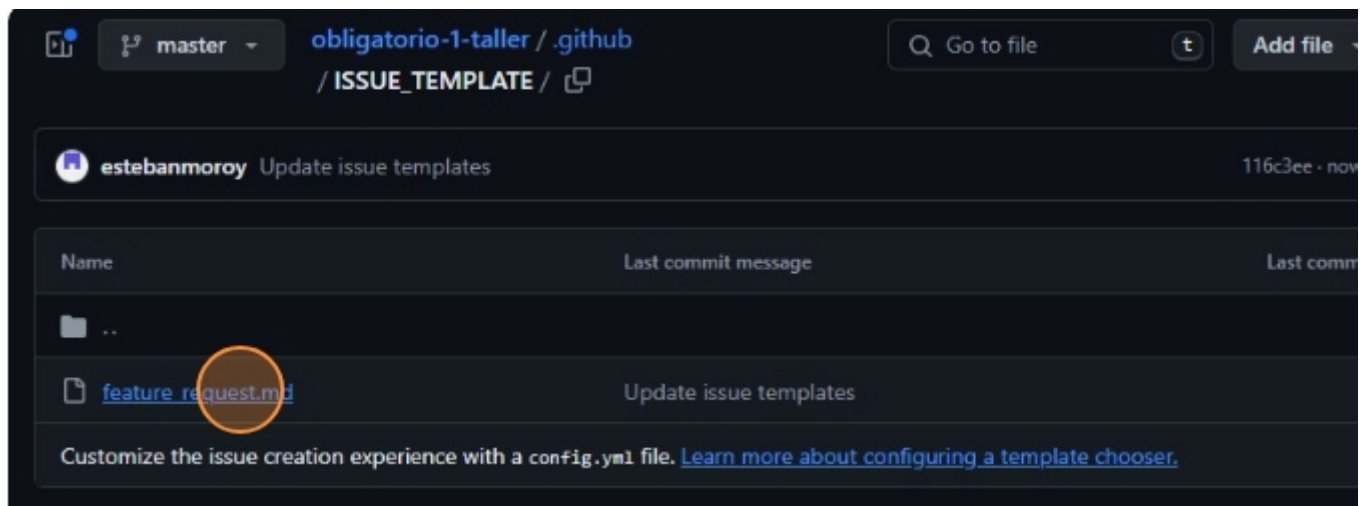
Issues Pull requests Actions Projects Wiki Security Insights

Propose changes

11. Esto genera un commit con el template generado en formato Markdown



The screenshot shows the GitHub commit interface for updating issue templates. On the left, there's a section for "feature request" with a "Preview and edit" button and a trash icon. Below it, a dashed box contains an "Add template: select" button. On the right, the "Commit message" section has a text input with "Update issue templates". Below that, the "Extended commit message" section has a text area with the placeholder "Add an optional extended description". At the bottom, there are two radio buttons: "Commit directly to the master branch." (selected) and "Create a new branch for this commit and start a pull request." Below the radio buttons is a green "Commit changes" button, which is circled in orange.



## Crear un template de GitHub Issue en formato Yaml (.yaml)

En caso de que se desee un mayor nivel de personalización, se pueden configurar los templates de Issues en formato Yaml, los cuales permiten generar campos específicos, listas desplegables, checkboxes, entre otros.

Pasos a seguir:

1. Actualizar el repositorio para traer los últimos cambios del servidor

```
git pull
```

2. Luego dirigirse a la carpeta `.github/ISSUE_TEMPLATE/`

```

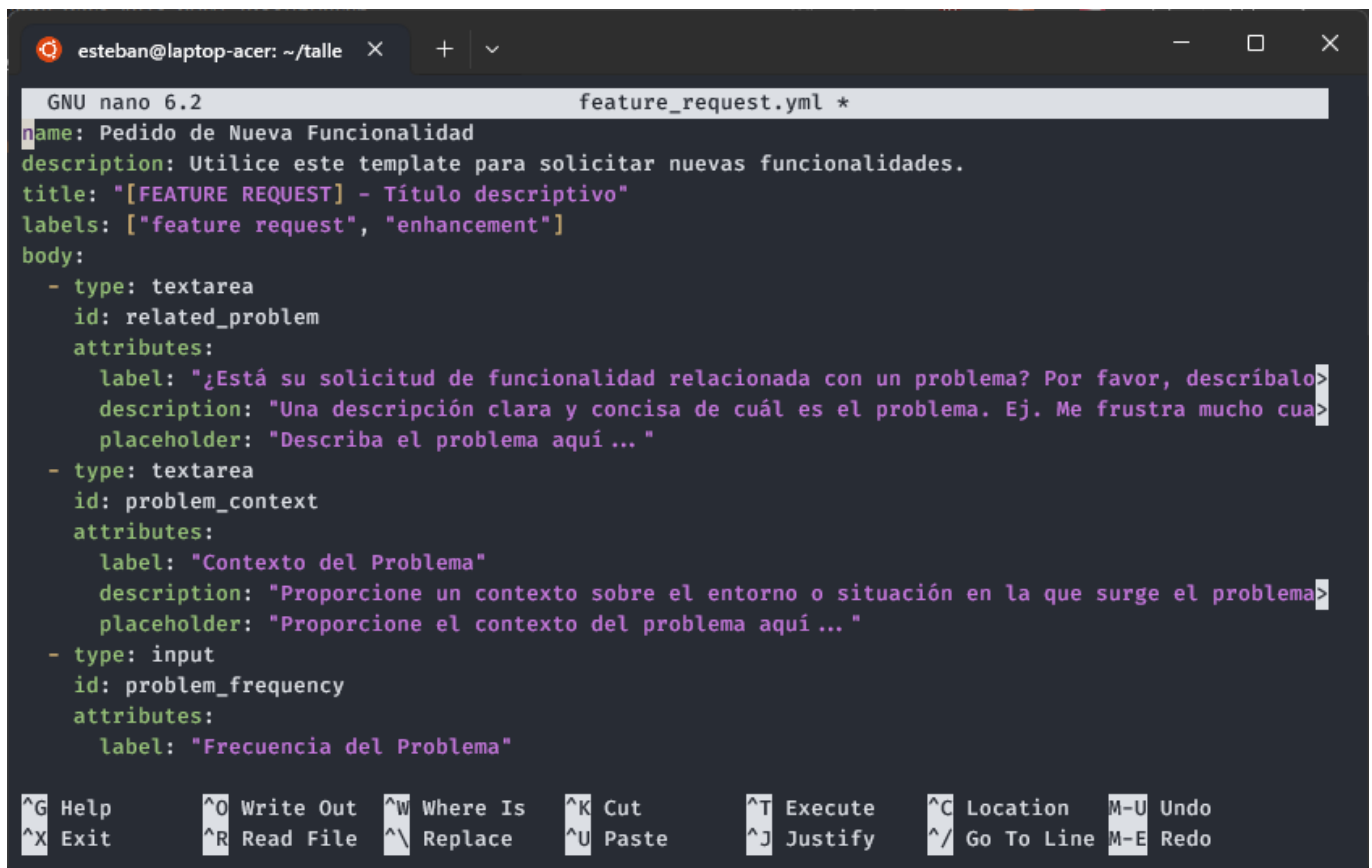
esteban@laptop-acer:~/taller/obligatorio-1$ git pull
remote: Enumerating objects: 6, done.
remote: Counting objects: 100% (6/6), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 5 (delta 1), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (5/5), 1.71 KiB | 291.00 KiB/s, done.
From github.com:estebanmoroy/obligatorio-1-taller
   1f0677c..116c3ee master    -> origin/master
Updating 1f0677c..116c3ee
Fast-forward
 .github/ISSUE_TEMPLATE/feature_request.md | 30 +++++
 1 file changed, 30 insertions(+)
 create mode 100644 .github/ISSUE_TEMPLATE/feature_request.md
esteban@laptop-acer:~/taller/obligatorio-1$ ls
diccionario.txt  obligatorio.sh  usuarios.txt
esteban@laptop-acer:~/taller/obligatorio-1$ cd .github/ISSUE_TEMPLATE/
esteban@laptop-acer:~/taller/obligatorio-1/.github/ISSUE_TEMPLATE$

```

3. Generar un archivo para el template, en este caso se llama "feature\_request.yml"

```
touch feature_request.yml
```

4. Escribir en el archivo siguiendo la convención para formularios de GitHub: [Syntax for GitHub's form schema](#)



```

GNU nano 6.2 feature_request.yml *
name: Pedido de Nueva Funcionalidad
description: Utilice este template para solicitar nuevas funcionalidades.
title: "[FEATURE REQUEST] - Título descriptivo"
labels: ["feature request", "enhancement"]
body:
  - type: textarea
    id: related_problem
    attributes:
      label: "¿Está su solicitud de funcionalidad relacionada con un problema? Por favor, descríbalos"
      description: "Una descripción clara y concisa de cuál es el problema. Ej. Me frustra mucho cua"
      placeholder: "Describe el problema aquí..."
  - type: textarea
    id: problem_context
    attributes:
      label: "Contexto del Problema"
      description: "Proporcione un contexto sobre el entorno o situación en la que surge el problema"
      placeholder: "Proporcione el contexto del problema aquí..."
  - type: input
    id: problem_frequency
    attributes:
      label: "Frecuencia del Problema"

^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute   ^C Location  M-U Undo
^X Exit      ^R Read File ^_ Replace   ^U Paste     ^J Justify   ^/_ Go To Line M-E Redo

```

5. Eliminar el archivo `.md` anteriormente generado

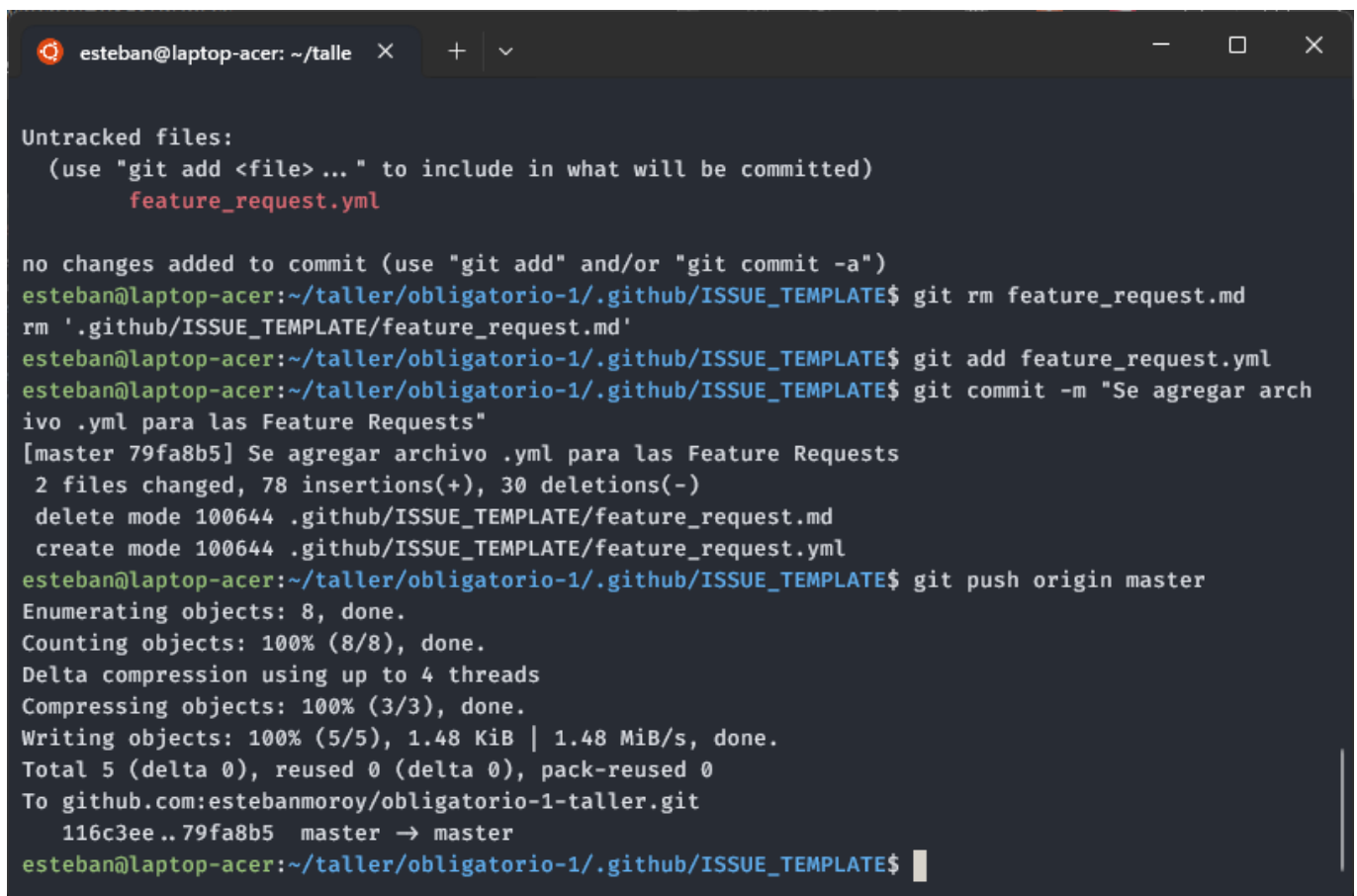
```
git rm feature_request.md
```

## 6. Y agregar el nuevo archivo .yaml generado

```
git add feature_request.yaml
```

## 7. Crear un commit para confirmar los cambios y pushearlos al repositorio

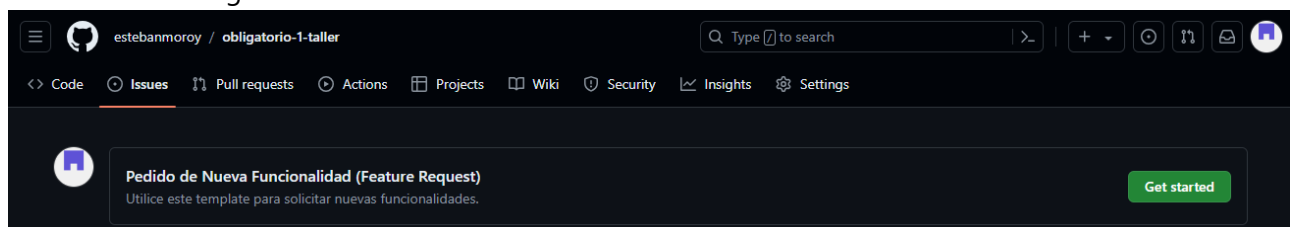
```
git commit -m "Se agrega archivo .yaml para las Feature Requests"
git push origin master
```



```
esteban@laptop-acer: ~/talle
Untracked files:
  (use "git add <file> ..." to include in what will be committed)
  feature_request.yaml

no changes added to commit (use "git add" and/or "git commit -a")
esteban@laptop-acer:~/taller/obligatorio-1/.github/ISSUE_TEMPLATE$ git rm feature_request.md
rm '.github/ISSUE_TEMPLATE/feature_request.md'
esteban@laptop-acer:~/taller/obligatorio-1/.github/ISSUE_TEMPLATE$ git add feature_request.yaml
esteban@laptop-acer:~/taller/obligatorio-1/.github/ISSUE_TEMPLATE$ git commit -m "Se agregar arch
ivo .yaml para las Feature Requests"
[master 79fa8b5] Se agregar archivo .yaml para las Feature Requests
 2 files changed, 78 insertions(+), 30 deletions(-)
 delete mode 100644 .github/ISSUE_TEMPLATE/feature_request.md
 create mode 100644 .github/ISSUE_TEMPLATE/feature_request.yaml
esteban@laptop-acer:~/taller/obligatorio-1/.github/ISSUE_TEMPLATE$ git push origin master
Enumerating objects: 8, done.
Counting objects: 100% (8/8), done.
Delta compression using up to 4 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (5/5), 1.48 KiB | 1.48 MiB/s, done.
Total 5 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:estebanmoroy/obligatorio-1-taller.git
 116c3ee..79fa8b5 master -> master
esteban@laptop-acer:~/taller/obligatorio-1/.github/ISSUE_TEMPLATE$
```


## 8. Verificar cambios generando un nuevo Issue



Ahora el formulario cuenta con varios campos de texto para cada sección solicitada

Issue: Pedido de Nueva Funcionalidad

Utilice este template para solicitar nuevas funcionalidades. If this doesn't look right, [choose a different type](#).



Add a title

[FEATURE REQUEST] - Título descriptivo

¿Está su solicitud de funcionalidad relacionada con un problema? Por favor, descríballo.

Una descripción clara y concisa de cuál es el problema. Ej. Me frustra mucho cuando [...]

Describe el problema aquí...

Contexto del Problema

Proporcione un contexto sobre el entorno o situación en la que surge el problema.

Proporcione el contexto del problema aquí...

Frecuencia del Problema

Indique la frecuencia con la que se encuentra este problema.

Indique la frecuencia del problema aquí...

Fuentes

- [YouTube - Github Issue Templates: A Look at the New Issue Templates](#)
- [Github - Configuring issue templates for your repository](#)
- [Github - Syntax for GitHub's form schema](#)