



2020

Profesores:

Martín Domínguez
Barragán, Pablo.

Diseño e Implementación de Estructuras de Datos

Guía 03 - Trabajar con GitHub

Contenido

Guía 3: Trabajar con GitHub	2
Paso 1: Descargar Git.....	2
Paso 2: Crear una cuenta en GitHub.....	2
Paso 3: Crear un repositorio en GitHub y subir nuestro proyecto	3

Guía 3: Trabajar con GitHub

Paso 1: Descargar Git

De la siguiente URL descargar el instalador para su sistema operativo:

<https://git-scm.com/>

En el ejemplo descargamos la versión del git para Windows.

La instalación muy simple consiste en darle Next consecutivamente, hasta que se instale.

Una vez instalada, probar en Símbolo del Sistema (cmd):

git --version

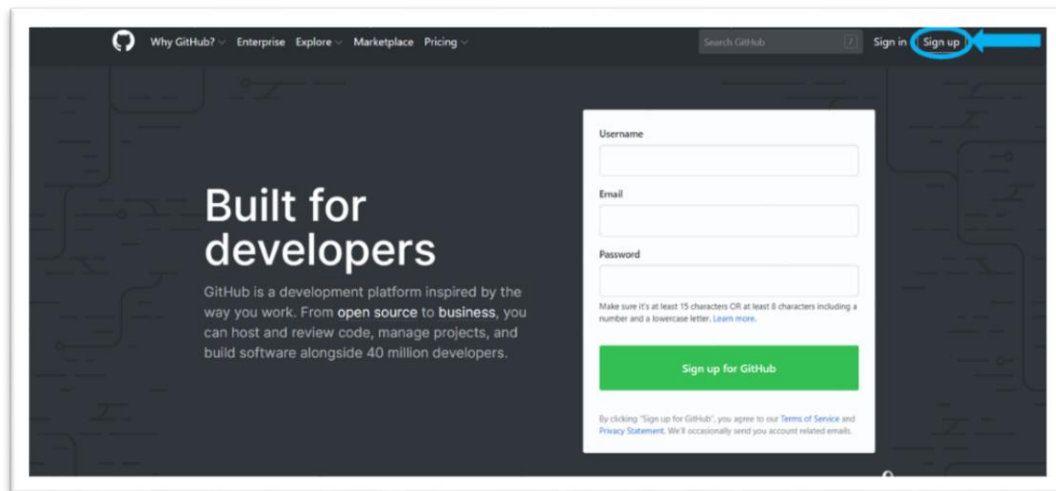


```
Símbolo del sistema
C:\Users\[redacted]>git --version
git version 2.25.1.windows.1
C:\Users\[redacted]>
```

Paso 2: Crear una cuenta en GitHub

En la siguiente URL puede entrar a la página oficial de GitHub:

<https://github.com/>



Join GitHub

Create your account

Username *

Email address *

Password *

Make sure it's at least 15 characters OR at least 8 characters including a number and a lowercase letter. [Learn more.](#)

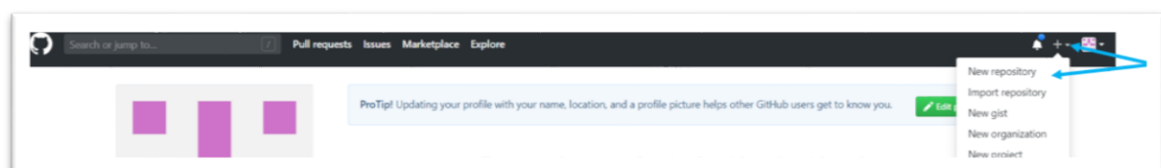
Email preferences

☐ Send me occasional product updates, announcements, and offers.

Verify your account

Paso 3: Crear un repositorio en GitHub y subir nuestro proyecto

Una vez creada la cuenta de GitHub, vamos a crear un repositorio para hacer referencia a nuestro proyecto de eclipse:



Create a new repository

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

Owner: [User Avatar] / **Repository name *** 1

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

Description (optional)

☒ **Public**
Anyone can see this repository. You choose who can commit.

☐ **Private**
You choose who can see and commit to this repository.

Skip this step if you're importing an existing repository.

☐ **Initialize this repository with a README**
This will let you immediately clone the repository to your computer.

Add .gitignore: **None** | Add a license: **None** ⓘ

Create repository 2

1. Pondremos el nombre de nuestro repositorio, en este caso died-lab01
2. Crear

Una vez creada, tendremos la siguiente ventana:

[User Avatar] / **died-lab01** Watch 0 Star 0 Fork 0

Code Issues 0 Pull requests 0 Actions Projects 0 Wiki Security Insights Settings

Quick setup — if you've done this kind of thing before

☒ Set up in Desktop or ☐ HTTPS ☐ SSH [https://github.com/\[User Avatar\]/died-lab01.git](https://github.com/[User Avatar]/died-lab01.git)

Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

...or create a new repository on the command line

```
echo "# died-lab01" >> README.md
git init
git add README.md
git commit -m "first commit"
git remote add origin https://github.com/[User Avatar]/died-lab01.git
git push -u origin master
```

...or push an existing repository from the command line

Tomaremos el ejemplo del proyecto que creamos en la guía 01, entonces abrimos el Símbolo del Sistema (cmd) y escribimos:

```
cd C:\Users\MiUsuario\eclipse - workspace\holamundo
git init
git add .
git commit -m "first commit"
git remote add origin https://github.com/<< MI_USUSARIO_GIT >>/died - lab01.git
git push -u origin master
```



```
Símbolo del sistema

C:\Users\[redacted]\eclipse-workspace\holamundo>git init
initialized empty Git repository in C:/Users/[redacted]/eclipse-workspace/holamundo/.git/

C:\Users\[redacted]\eclipse-workspace\holamundo>git add .

C:\Users\[redacted]\eclipse-workspace\holamundo>git commit -m "first commit"
master (root-commit) bce2848] first commit
5 files changed, 52 insertions(+)
create mode 100644 .classpath
create mode 100644 .gitignore
create mode 100644 .project
create mode 100644 .settings/org.eclipse.jdt.core.prefs
create mode 100644 src/ar/edu/utn/frsf/died/prueba/ImprimirHolaMundo.java

C:\Users\[redacted]\eclipse-workspace\holamundo>git remote add origin https://github.com/[redacted]/died-lab01.git

C:\Users\[redacted]\eclipse-workspace\holamundo>git push -u origin master
Enumerating objects: 15, done.
Counting objects: 100% (15/15), done.
Delta compression using up to 4 threads
Compressing objects: 100% (8/8), done.
Writing objects: 100% (15/15), 1.49 KiB | 507.00 KiB/s, done.
Total 15 (delta 0), reused 0 (delta 0)
To https://github.com/[redacted]/died-lab01.git
 * [new branch]      master -> master
branch 'master' set up to track remote branch 'master' from 'origin'.

C:\Users\[redacted]\eclipse-workspace\holamundo>
```

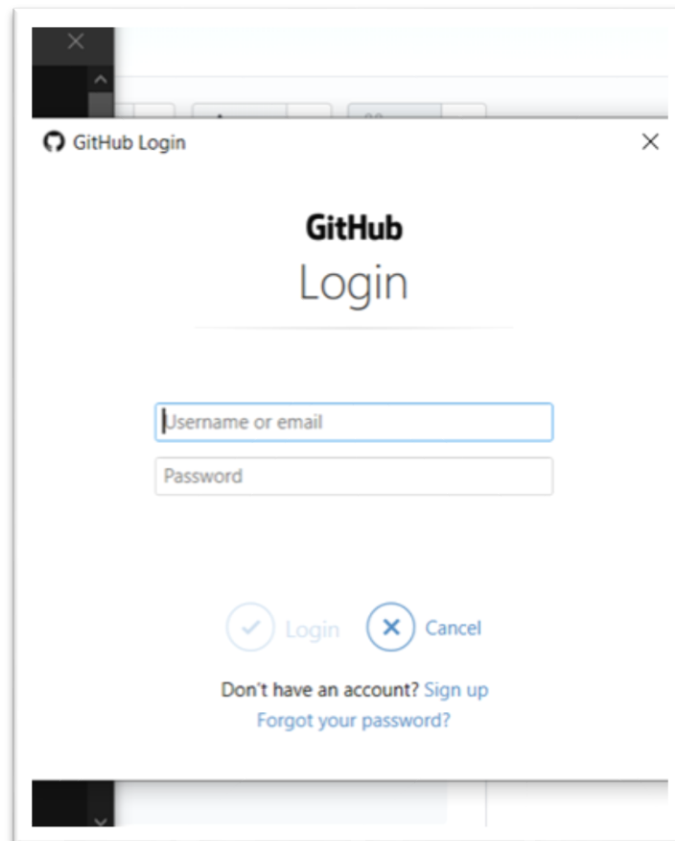
[Nota adicional]

Si les aparece el mensaje en el cmd:

```
git config --global user.email "you@example.com"
git config --global user.name "Your Name"
```

Escribir, igual como está registrado en el github.

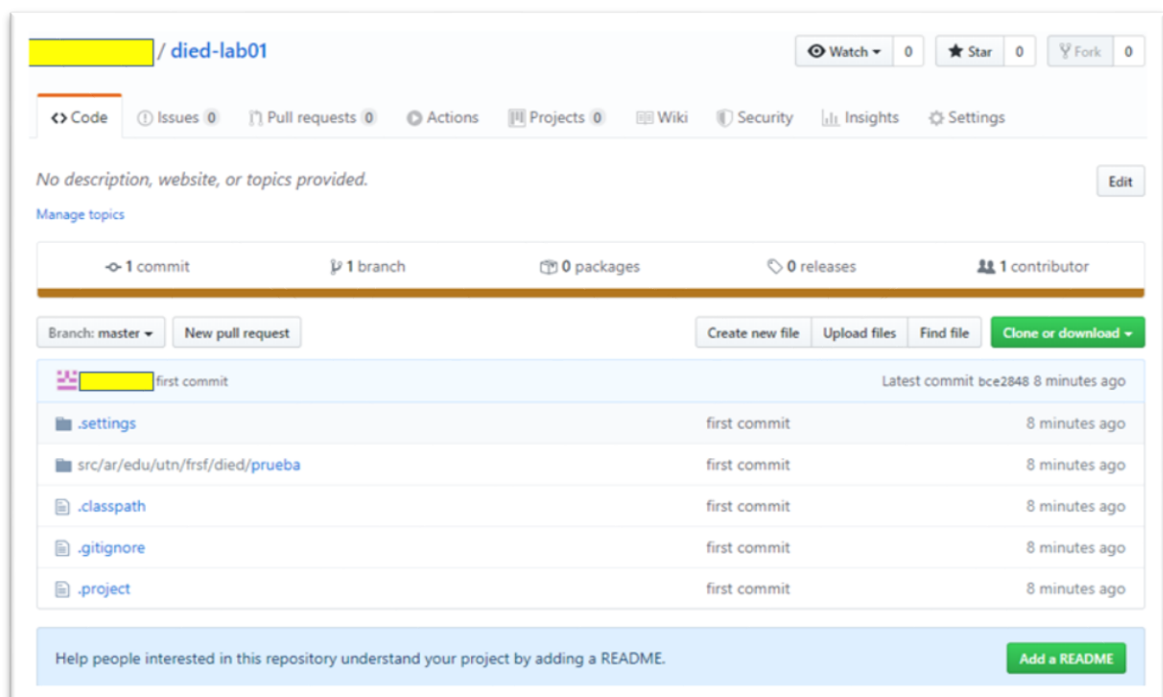
Si al momento de ejecutar git push -u origin master les aparece:



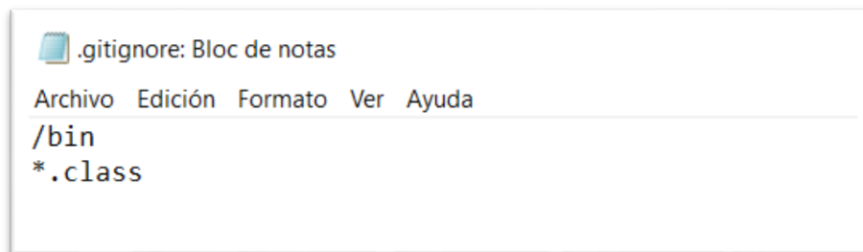
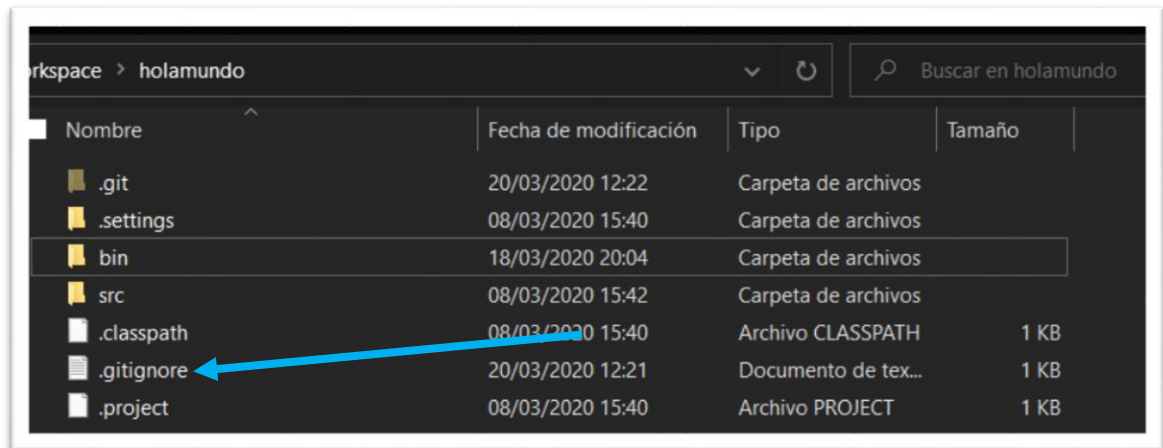
Significa que todo funciona correctamente, solo falta iniciar la sesión y comenzará a subir.

Desde nuestro repositorio donde accedemos:

https://github.com/MI_USUARIO_GIT/died-lab01



Podemos configurar también en nuestro archivo “.gitignore”, que permite evitar que se suban archivos innecesarios.



Como se muestra en la imagen, tenemos que evitar que se suban al repositorio de GitHub, los archivos de clase compilado “*.class” y la carpeta de “bin”.