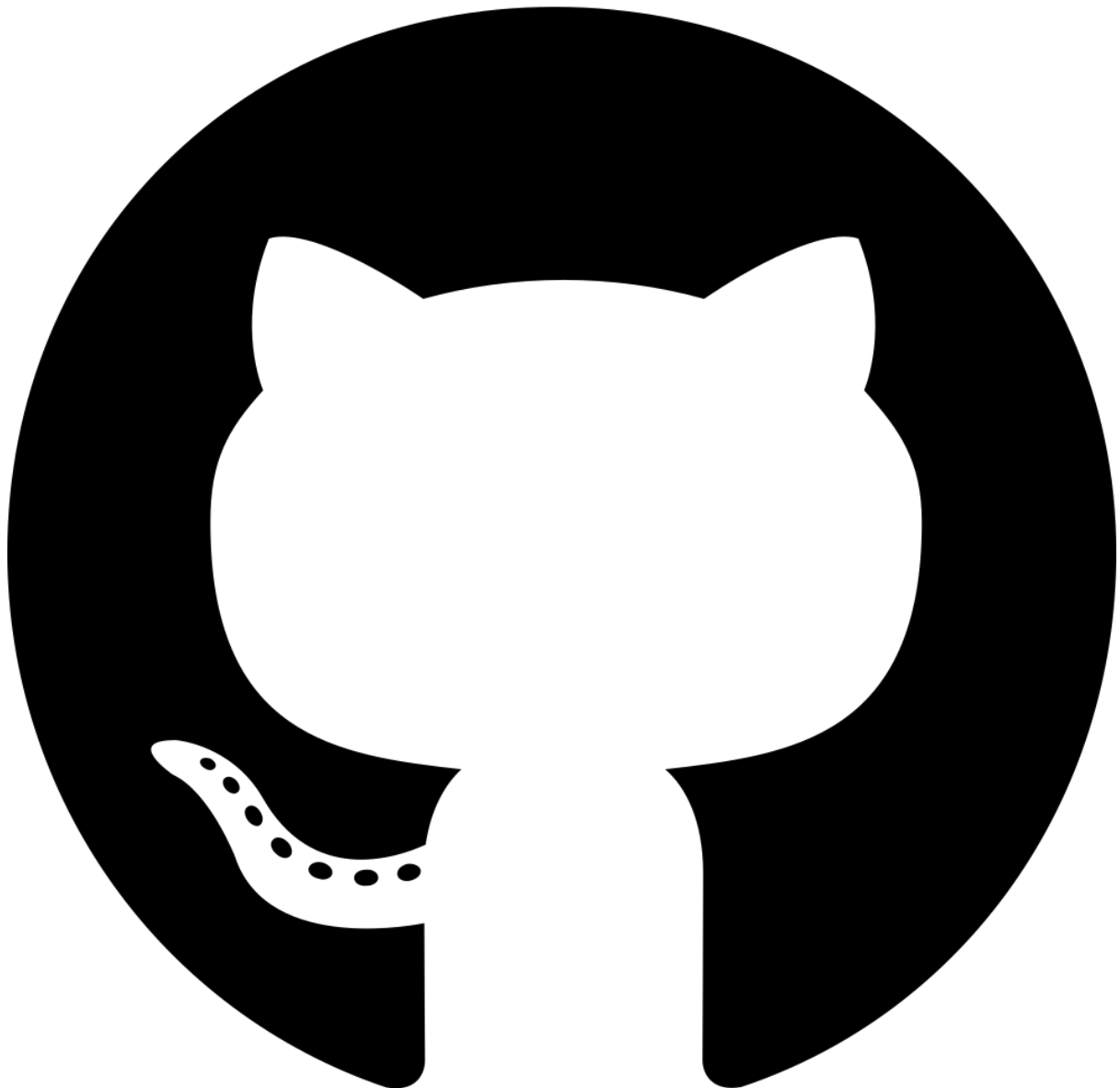


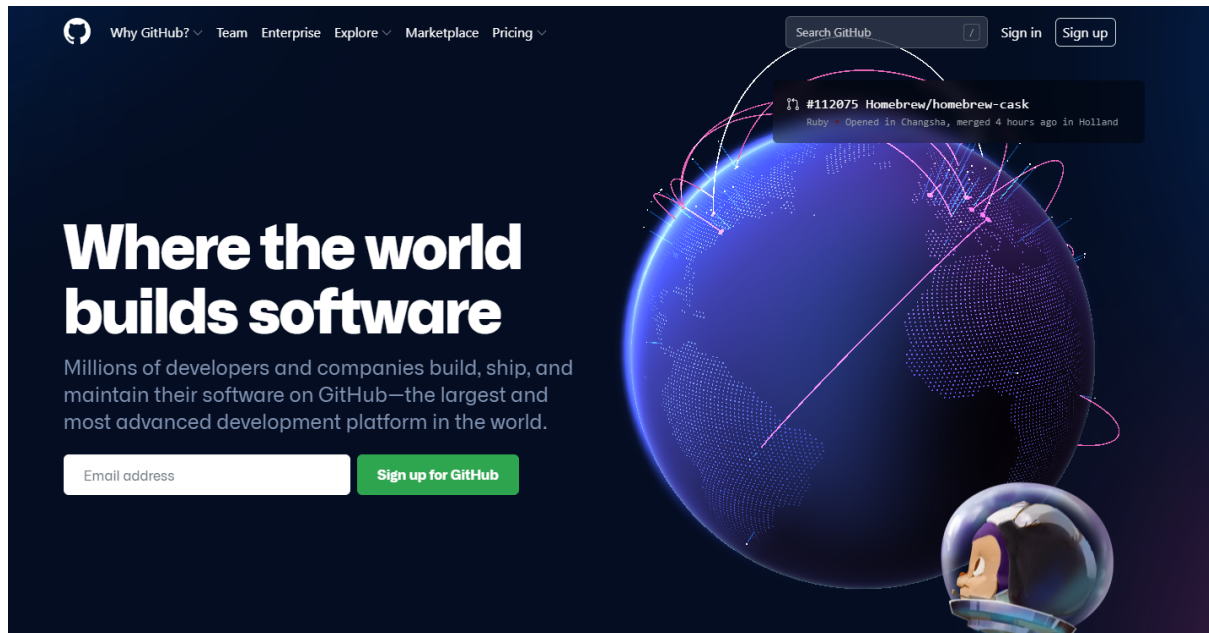
# CREACIÓN DE REPOSITORIO GITHUB



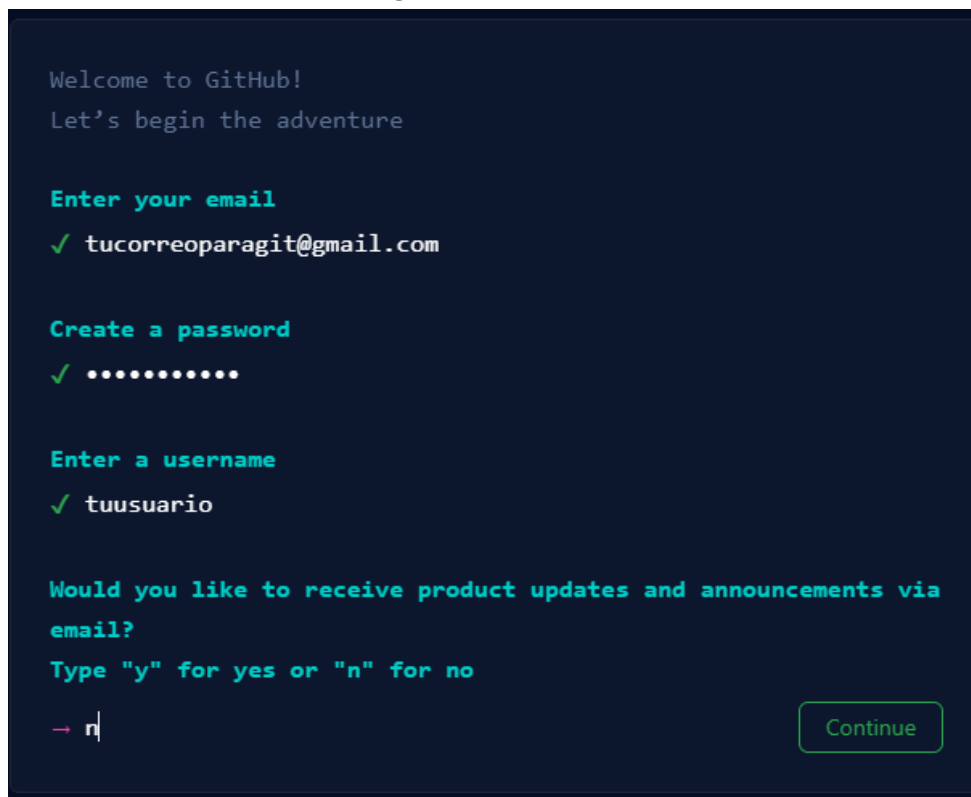
Para comenzar a trabajar con un nuevo repositorio, y usar GitHub primero debemos instalarlo en nuestro ordenador, además de crearnos una cuenta en la página web.

### Guía para acceder a GitHub y crear nuestro primer repositorio

Primero tenemos que crear nuestra cuenta de github para ellos accederemos a su página web



Una vez aquí le daremos a **Sign up** para crear nuestra cuenta



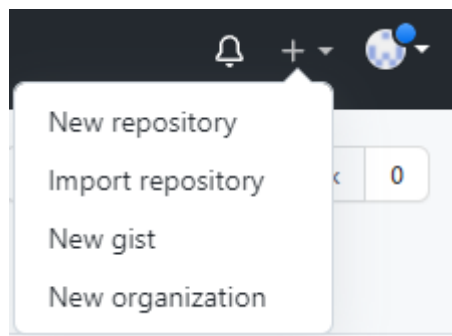
Nos saldrá esta pantalla donde tendremos que rellenar los datos correspondientes y nos enviarán un correo para verificar nuestra cuenta y algunas preguntas más para completar la información de tu perfil de usuario. Una vez realizado esto ya podremos acceder a github.

Ahora nos hace falta descargar la herramienta de comandos de git para ello entraremos en el siguiente enlace :<https://git-scm.com/>



Una vez dentro, le daremos a Download for Windows en la parte inferior derecha, para descargar la última versión de la herramienta.

A continuación, con la cuenta creada en Github, y la herramienta descargada e instalada ya podemos trabajar y crear nuestro primer repositorio.



Para crear nuestro repositorio una vez accedido a la página web, tendremos que clicar en el "+" y nos aparecerán varias opciones, una de ellas la de crear un nuevo repositorio.

Una vez dentro nos aparecerá “Create a new repository” donde tendremos que rellenar los datos que nos piden y ya estará creado el repositorio.

## Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere?

[Import a repository.](#)

Owner \*

Repository name \*



HillertJimenezlucas ▾

/

Great repository names are short and memorable. Need inspiration? How about [laughing-goggles?](#)

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:

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

☐ Add a README file

This is where you can write a long description for your project. [Learn more.](#)

☐ Add .gitignore

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

☐ Choose a license

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

Create repository

En segundo lugar, debemos crear la carpeta en nuestro ordenador en la cual trabajaremos con este nuevo repositorio, que luego subiremos a github para transformarlo en un repositorio remoto, para realizar esto debemos crear una carpeta en nuestro ordenador, y abrir nuestra pantalla de comandos (ej:Windows PowerShell) utilizando el comando “cd” para cambiar el directorio y poner el de la carpeta que hemos creado.

```
PS C:\Users\Lucas\Documents> cd C:\Users\Lucas\Documents\PROYECTO_01
```

Ahora podemos ya crear el repositorio remoto en GitHub escribiendo los siguientes comandos(en este ejemplo la carpeta la hemos llamado PROYECTO\_01).

```
echo "# PROYECTO_01" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin
https://github.com/HillertJimenezlucas/PROYECTO_01.git
git push -u origin main
```

Con la serie de comandos anterior ya se crea el repositorio remoto el cual podemos visualizar en la página web de GitHub además de crearse el archivo README.md.

The screenshot shows a GitHub repository page. At the top, there are buttons for 'Go to file', 'Add file', and 'Code'. Below these, the repository name 'HillertJimenezLucas Update texto info.txt' is displayed, along with the commit hash 'd3a316e', the time '1 hour ago', and '7 commits'. A table lists the files in the repository:

File	Commit	Time
doc	finalizado txt	1 hour ago
src	Update texto info.txt	1 hour ago
README.md	quinto commit	2 hours ago

Below the table, the README.md file is displayed. It contains the following text:

Lucas Hillert Jiménez 1ºASIR A Base de Datos

## PROJECTO\_01

En este proyecto vamos a hacer un primer repositorio en github con dos carpetas:

- "src" donde escribiremos información en un txt
- "doc" donde añadiremos un pdf con las explicaciones de instalacion y uso de github

On the right side of the repository page, there are sections for 'About', 'Releases', and 'Packages'. The 'About' section has a placeholder text: 'No description, website, or topics provided.' The 'Releases' section has a placeholder text: 'No releases published' and a link 'Create a new release'. The 'Packages' section has a placeholder text: 'No packages published' and a link 'Publish your first package'.

Esta imagen muestra también otros cambios realizados en el repositorio.

Para realizar cambios en el repositorio, podemos hacerlo directamente en el repositorio remoto que visualizamos en GitHub, o en el repositorio local en nuestra carpeta del proyecto. Si realizamos los cambios directamente en GitHub, tendremos que tener cuidado de luego intentar seguir trabajando en nuestro repositorio local, ya que tendremos distintas versiones, por lo tanto tendríamos que utilizar el comando `git clone`, y a continuación poner nuestro enlace del repositorio remoto que en mi caso sería

The screenshot shows the 'Code' dropdown menu in a GitHub repository. The menu is open, showing three options: 'Clone', 'Open with GitHub Desktop', and 'Download ZIP'. The 'Clone' option is selected, and it shows the following information:

- Buttons: 'Go to file', 'Add file', 'Code'.
- Clone options: 'HTTPS', 'SSH', 'GitHub CLI'.
- URL: `https://github.com/HillertJimenezLucas/PROJECTO_01.git` (with a copy icon).
- Text: 'Use Git or checkout with SVN using the web URL'.
- Buttons: 'Open with GitHub Desktop', 'Download ZIP'.

A large black arrow points from the URL in the 'Clone' section to the terminal command in the next block.

```
PS C:\Users\Lucas\Documents\PROJECTO_01> git clone https://github.com/HillertJimenezLucas/PROJECTO_01.git
```

Ahora sí podríamos seguir trabajando en nuestro repositorio local. Sin embargo si realizamos cambios primeramente en nuestro repositorio local, para actualizar esa información en el remoto, debemos usar la siguiente serie de comandos.

```
PS C:\Users\Lucas\Documents\PROYECTO_01> git add .
PS C:\Users\Lucas\Documents\PROYECTO_01> git commit -m "finalizado txt"
[main faddde2] finalizado txt
2 files changed, 23 insertions(+)
 rename src/texto.txt => "doc/informaci\303\263n de instalaci\303\263n y uso git hub.txt" (100%)
 create mode 100644 src/texto info.txt
PS C:\Users\Lucas\Documents\PROYECTO_01> git push origin main
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 4 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (5/5), 1.09 KiB | 1.09 MiB/s, done.
Total 5 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/HillertJimenezlucas/PROYECTO_01.git
 3c2c0d3..faddde2  main -> main
```

Dentro de las comillas después del comando `git commit -m "....."` ponemos un nombre al cambio realizado, y nos sirve de esta manera para guiarnos en el repositorio remoto e ir viendo las diferentes versiones que hemos ido actualizando.

Con esto ya habríamos finalizado nuestra creación de Repositorio en GitHub además de los comandos básicos para actualizar su información.