

## PASO 0

Después de crear la aplicación en la app. **Descargamos el zip.** Te explico que se hace en cada botón.

**EDITAR** → editamos algún valor de la aplicación generada.

**ELIMINAR** → eliminamos la aplicación que hemos generado.

**DESCARGAR ZIP** → descargamos la demo de la app generada o solo la configuración (según lo que hayamos seleccionado).

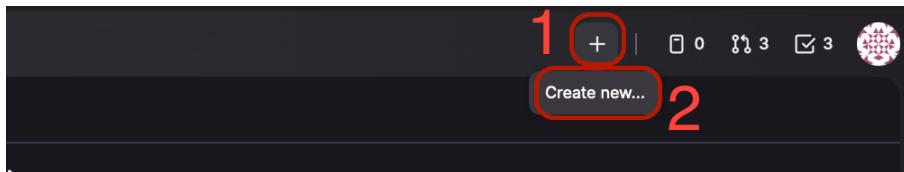
**VARIABLES (.TXT)** → este botón descarga un fichero .txt con algunas de las variables que hemos escrito cuando estábamos creando nuestra aplicación. (por ejemplo datos de base de datos, puerto de la aplicación...)

**ASISTENTE** → esto nos ayudara a generar las variables que necesitamos para que nuestra aplicación funcione correctamente.

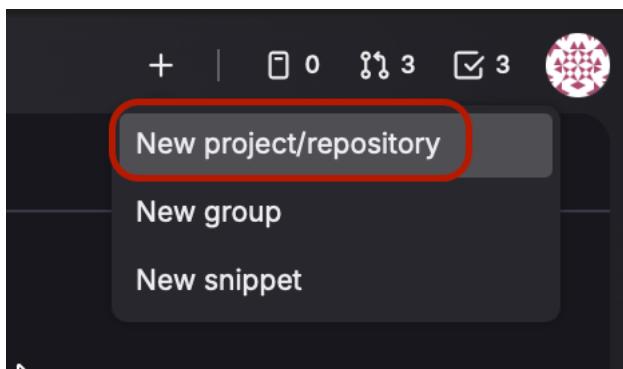
Nombre	Lenguaje	Base de datos	Proveedor CI/CD	Fecha creación	Pasos despliegue	Estado	Acciones
demo-mongo-py	PYTHON	POSTGRESQL	GITHUB	29/12/2025 20:21	0 / 6	--	<button>Editar</button> <button>Eliminar</button> <button>Descargar ZIP</button> <button>Variables (.txt)</button> <button>Asistente</button>

Una vez descargado el zip.

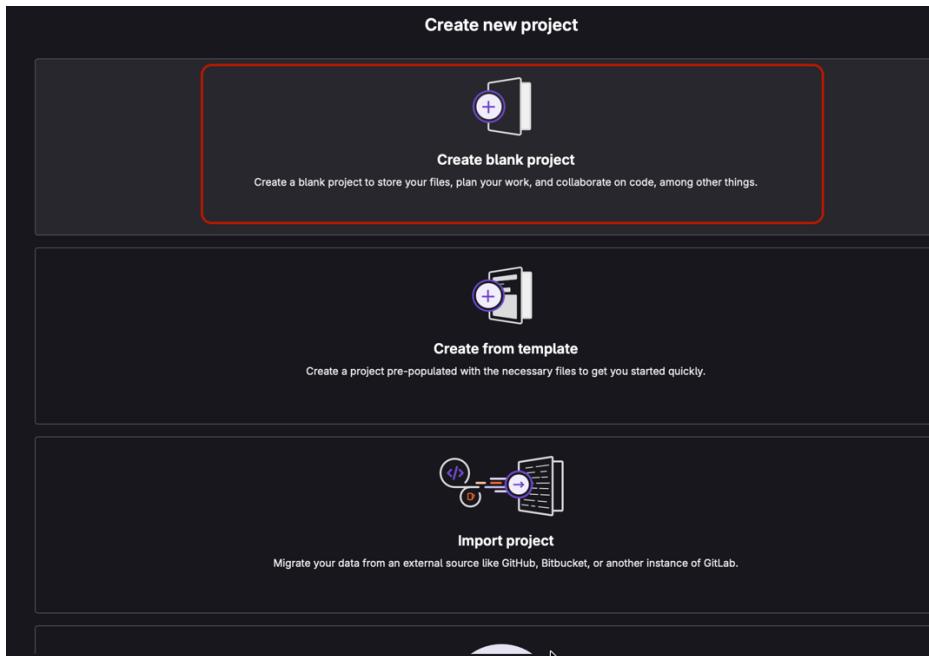
Lo primero que vamos a hacer es crear nuestro repositorio en gitlab. En la parte superior de nuestra cuenta de gitlab, damos al botón “+”



Pulsamos a “New Project/repository”.



Elegimos la opción: “create blank project”



1º Le damos un nombre a nuestro repositorio

2º Elegimos nuestro usuario en nuestro desplegable

3º Ponemos el Project público para que no falle con el sonar

4º damos a “create project”

**Create blank project**

Create a blank project to store your files, plan your work, and collaborate on code, among other things.

**Project name** 1 demo-prueba

Must start with a lowercase or uppercase letter, digit, emoji, or underscore. Can also contain dots, pluses, dashes, or spaces.

**Project URL** 2 https://gitlab.com/davidtome97

**Project slug** / demo-prueba

**Project deployment target (optional)** Select the deployment target

**Visibility Level** 3  Public  
The project can be accessed without any authentication.

**Project Configuration**

Initialize repository with a README  
Allows you to immediately clone this project's repository. Skip this if you plan to push up an existing repository.

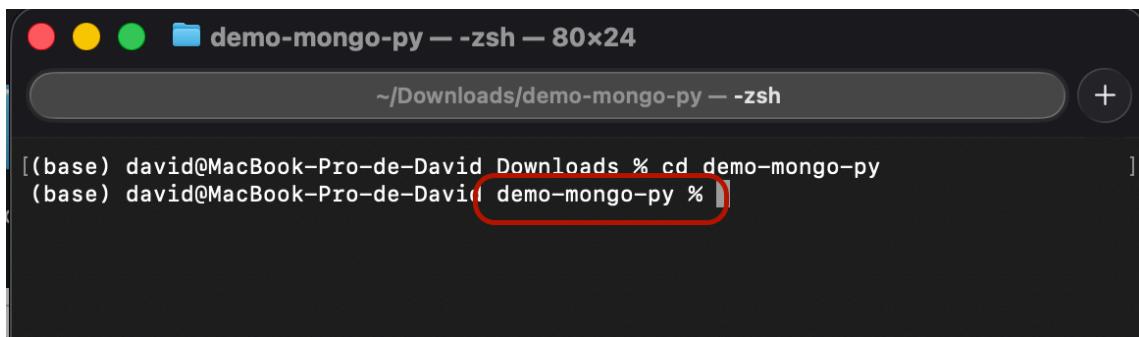
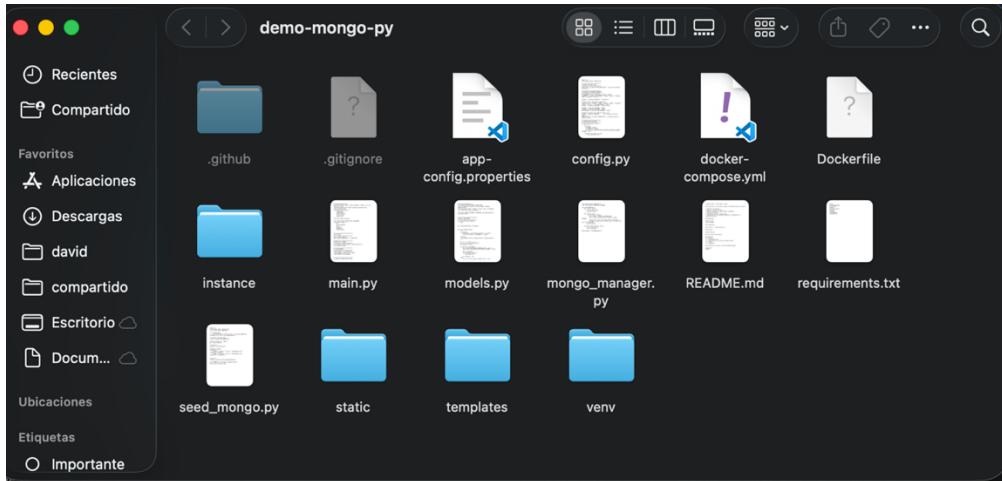
Enable Static Application Security Testing (SAST)  
Analyze your source code for known security vulnerabilities. Learn more.

Enable Secret Detection  
Scan your code for secrets and credentials to prevent unauthorized access. Learn more.

› Experimental settings

**Create project** 4 Cancel

Ahora descomprimimos el zip que hemos descargado antes y entramos dentro de el por terminal.



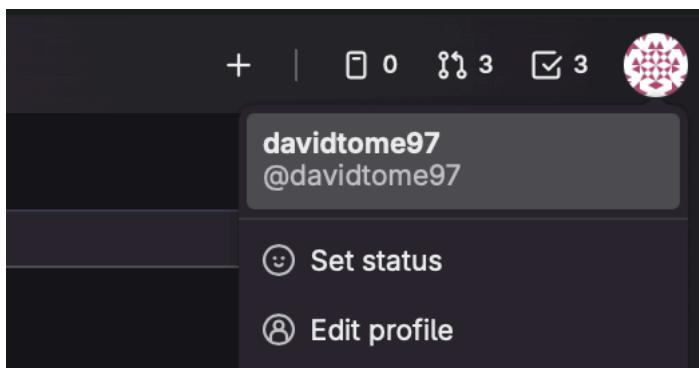
Ejecutamos esas líneas de comando, te las dejo escritas después de la imagen, si falla puede ser por las comillas, ponerlas a mano si eso.

```
git remote add origin https://gitlab.com/davidtome97/demo-prueba.git
```

```
git branch -M main
```

```
git push -uf origin main
```

si por un casual te da fallo por terminal y te pide usuario y contraseña, introduce tu usuario de github:



Y para la contraseña tendrás que crear un token, que solo vas a ver 1 vez:

Pulsamos a preferences → personal access tokens → add new token

The image consists of three screenshots illustrating the steps to create a new personal access token in GitLab.

- Screenshot 1: User Profile Menu**  
Shows the user profile dropdown menu with "Preferences" highlighted by a red box.
- Screenshot 2: User Settings**  
Shows the "User settings" sidebar with "Personal access tokens" highlighted by a red box.
- Screenshot 3: Personal Access Tokens Page**  
Shows the "Personal access tokens" page with:
  - "Active tokens": 2 (highlighted by a red box)
  - "Tokens expiring in 2 weeks": 0
  - "Revoked tokens": 3 (highlighted by a red box)
  - "Expired tokens": 0A red box highlights the "Add new token" button in the top right corner, which has a red number "1" above it.

Damos un nombre al token y marcamos las opciones de `read_repository` y `write_repository` y después creamos el token.

Personal access tokens

Add new token

Token name

tokenGitlab

Description (optional)

Expiration date

2026-10-15

Select scopes

Scopes set the permission levels granted to the token. Learn more

**read\_repository**  
Grants read-only access to repositories on private projects using Git-over-HTTP or the Repository Files API.

**write\_repository**  
Grants read-write access to repositories on private projects using Git-over-HTTP (not using the API).

**read\_user**  
Grants read-only access to your profile through the /user API endpoint, which includes username, public email, and full name. Also grants access to read-only API endpoints under /users.

**read\_registry**  
Grants read-only access to container registry images through the dependency proxy in private projects and virtual registries.

**read\_api**  
Grants read access to the API, including all groups and projects, the container registry, and the package registry.

**self\_rotate**  
Grants permission for token to rotate itself.

**write\_virtual\_registry**  
Grants read, write, and delete access to container images through the dependency proxy in private projects.

**write\_registry**  
Grants write access to container registry images on private projects. You need both read and write access to push images.

**api**  
Grants complete read/write access to the API, including all groups and projects, the container registry, the dependency proxy, and the package registry.

ahora vamos a subir nuestra carpeta a nuestro repositorio:

```
git add .
```

```
git commit -m "subir ficheros al repositorio"
```

```
git push origin main
```

Ya lo tenemos subido. Damos por finalizada esta parte.

The screenshot shows a GitHub repository named 'demo-prueba'. At the top, it displays 'Project information' with 4 commits, 1 branch, 0 tags, and 31KiB of project storage. Below this are options to add LICENSE, CHANGELOG, CONTRIBUTING, Auto DevOps, Kubernetes cluster, CI/CD, and Wiki, along with Configure Integrations. It was created on December 29, 2025. The main tab is selected, showing a merge commit from 'reset-demo-python' into 'main' by davidtome67 just now. The commit message is 'chore: initial demo-python project'. The file list shows the following structure and last commits:

Name	Last commit	Last update
.github	chore: initial demo-python project	8 minutes ago
static	chore: initial demo-python project	8 minutes ago
templates	chore: initial demo-python project	8 minutes ago
.DS_Store	chore: initial demo-python project	8 minutes ago
.gitignore	chore: initial demo-python project	8 minutes ago
Dockerfile	chore: initial demo-python project	8 minutes ago
README.md	Initial commit	34 minutes ago
app-config.properties	chore: initial demo-python project	8 minutes ago
config.py	chore: initial demo-python project	8 minutes ago