



Infraestructura II

Paso a paso: Crear nuestra cuenta en GitLab

Vamos a crear nuestra cuenta en GitLab, que va a ser la herramienta utilizada para llevar a la práctica nuestros Pipelines.

Objetivo final de la práctica

- Completar la registración en la instancia de GitLab y dejar preparado nuestro usuario para poder subir código a los repositorios.
- Subir un código de prueba.

Formas de consumir GitLab

Primero debemos saber que GitLab puede consumirse de varias maneras:

- **Usando el SAAS público:** accediendo a <https://gitlab.com/> , registrándonos allí podemos comenzar en pocos segundos a utilizar todos los servicios, con el único inconveniente que para poder ejecutar los Pipelines debemos validar nuestra cuenta con una Tarjeta de Crédito, a pesar de no tener costo alguno
- **Instalándolo en nuestros servidores:** Siguiendo una serie de pasos dedicados para las distintas distribuciones de Linux, podríamos tener nuestro propio GitLab instalado, hasta inclusive cabe la posibilidad de ejecutarlo en contenedores Docker.

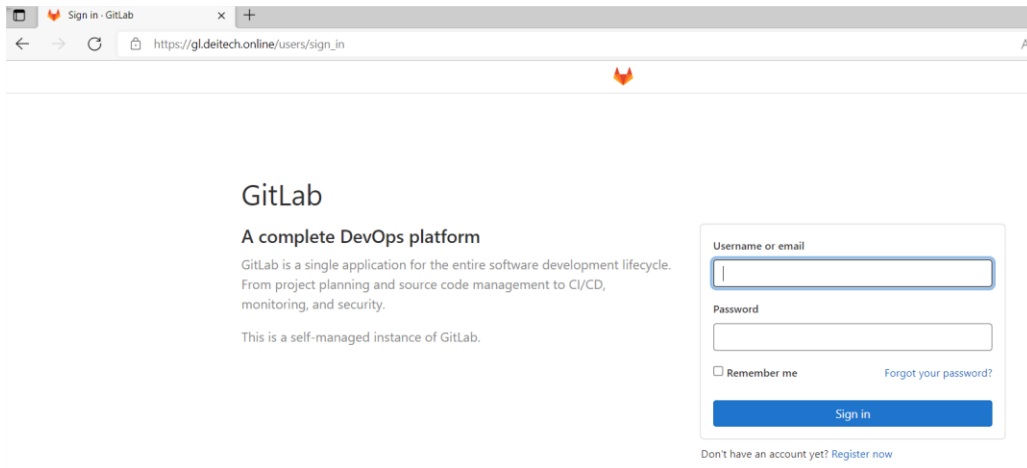
Esta última opción es la más recomendada y utilizada en el ámbito laboral, ahora, como salvedad, requerimos de un Hardware potente para hacerlo, ya que el mínimo hardware para que funcione óptimamente es 4 CPU y 8Gb RAM.

Ante esta situación, vamos a proveerle una instancia de GitLab para las practicas de Infraestructura II, la misma esta alojada en:

<https://gl.deitech.online/>

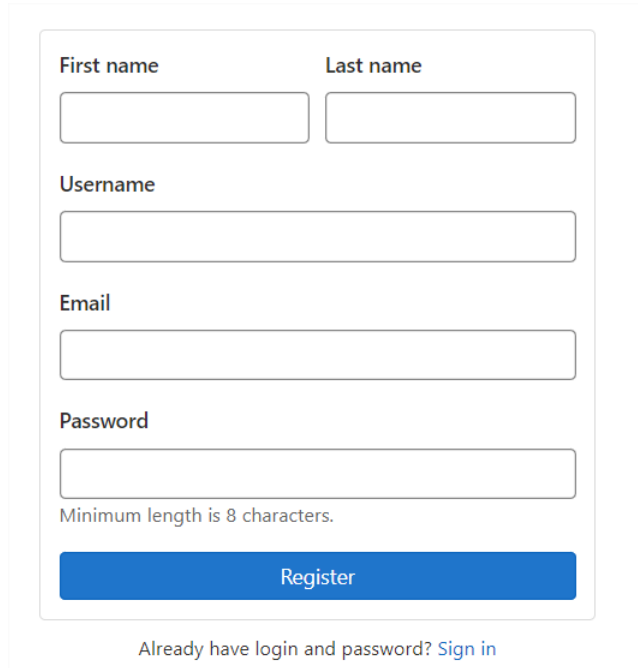
Registración y preparación del entorno

1. En la pantalla inicial, nos dirigimos a “Register now”



The screenshot shows a web browser window with the URL https://gl.deitech.online/users/sign_in. The page features the GitLab logo and the text "A complete DevOps platform". Below this, it states: "GitLab is a single application for the entire software development lifecycle. From project planning and source code management to CI/CD, monitoring, and security. This is a self-managed instance of GitLab." On the right side, there is a sign-in form with fields for "Username or email" and "Password". Below these fields are checkboxes for "Remember me" and a link for "Forgot your password?". A blue "Sign in" button is at the bottom of the form. At the very bottom of the page, there is a link that says "Don't have an account yet? Register now".

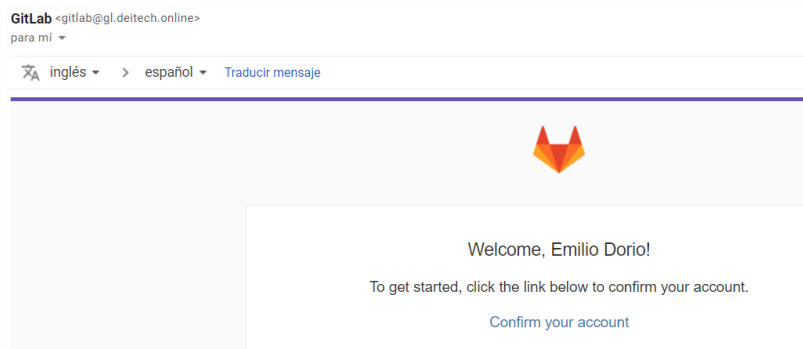
2. Los datos que debemos completar son los siguientes, automáticamente nos informara si el nombre de usuario o email seleccionado está en uso.




Registration form fields:


- First name
- Last name
- Username
- Email
- Password (Minimum length is 8 characters.)
- Register button
- Link: Already have login and password? [Sign in](#)

3. Recibiremos una confirmación y se nos informa que debemos ir a chequear nuestro email; revisar correctamente tanto Inbox como carpetas de Spam, la dirección desde donde se envía el email es gitlab@gl.deitech.online



4. Confirmamos nuestra cuenta, con lo cual ya estamos en condiciones de loguearnos, al hacerlo nos preguntara por que rol desempeñamos, seleccionamos "Devops Enginner" y como motivo "I Want to learn the basics of Git". Esto personalizara nuestra experiencia con la plataforma.
5. En la pantalla inicial de bienvenida seleccionamos "Create Project", luego "Create blank Project" y finalmente, completamos el dato de "Project Name"; dejamos la visibilidad de este en "Private".

1)  **Create a project**
Projects are where you store your code, access issues, wiki and other features of GitLab.

2)  **Create blank project**
Create a blank project to house your files, plan your work, and collaborate on code, among other things.

New project > Create blank project

Project name

Project URL **Project slug**

Want to house several dependent projects under the same namespace? [Create a group.](#)

Project description (optional)

Project deployment target (optional)

Visibility Level ⓘ
☒ **Private**
 Project access must be granted explicitly to each user. If this project is part of a group, access will be granted to members of the group.
☐ **Internal**
 The project can be accessed by any logged in user except external users.
☐ **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.

3)

6. Por defecto se nos crea un Repositorio Git en el cual podemos subir nuestros archivos, en el mismo se incluye un README.md con instrucciones básicas para dicho fin. Si bien como en todo repositorio Git podemos subir nuestros archivos a través de la interfaz Web, no es su uso habitual, así que ahora prepararemos nuestro entorno para poder subir código desde nuestro equipo; nos llevara unos pasos en los cuales debemos utilizar nuestra línea de comandos / consola

6.1. Abrimos en NUESTRO EQUIPO un Shell como GitBash o la Terminal de Linux con el propósito de generar una SSH Key, para ello ejecutamos:

```
ssh-keygen -t rsa
```

Nos preguntara por un nombre y un passphrase (opcional), dejamos todo por defecto

Leemos el contenido del mismo con

```
cat ~/.ssh/id_rsa.pub
```



se nos mostrara un contenido tal como

```
$ cat gitlab.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQGC/MaOt4T8HkpH2xFSsxAny5uNwXdZmnnx4r7hmYB
aHEHNwIzZKmxV9ZUrgN9cexso7Wx7QVT4uuDIOJZ23+Ndk0+gehgWw4Z9qxt7I4rimFgH17hsibAxGL
7dq1lEuqxUJXYmjwVKXwfi3byANepPaBMukmE/MjcYYenz5K03TIAJKUrbcuVam7EcMJzdfGRIiEf
NOLyoQ1tWALuzHvpjTnjC4YTkvxgMZ6jRw1cXav0yyCuzzJTtA5a2ynDHLGGSwraVlVui5g3wfVZM6P
s5NGkXv2912ppWjZzFWi/AFB417jw1kTC31c1z5FXArQYN+28ZAbznSPquS7q5o2c71M0r9FSkAZW5v
mhqanDi8VZ/xzAAVcg1rMpiNQ4mLHgtXaJjRx5JE8MR/wlgizJuRz0j07s53op3cZdZguKFSolHzWl
XQfEcNhn9Zp6VHD0yTbpETDCFQa/NzH9TMEoIWfAXe15/BXRUAS5qama9SBHvhre8Qw5cIE= AR-CAR
ERA-09+edorio@AR-CARRERA-09
```

No cerramos nuestro Shell.

6.2. Nos dirigimos al Gitlab, en la sección “SSH Keys” ([SSH Keys • User Settings • GitLab](#) ”

6.3. Copiamos todo el contenido que se mostro en el punto 6.1 con el comando cat (a partir de que comienza con ssh-rsa inclusive) y lo pegamos en la sección “Key”, le damos un nombre en “Title” y opcionalmente una fecha de expiración, luego de ello en “Add Key”.

User Settings > SSH Keys

SSH Keys

SSH keys allow you to establish a secure connection between your computer and GitLab.

Add an SSH key

Add an SSH key for secure access to GitLab. [Learn more.](#)

Key

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

Title

Expiration date

e.g. My MacBook key

dd/mm/yyyy

Give your individual key a title. This will be publicly visible.

Key can still be used after expiration.

Add key

6.4. Una vez guardada la misma en GitLab, debemos verificar que podemos conectarnos desde nuestro entorno, para ello volvemos a nuestro Shell y ejecutamos

```
ssh -T git@gl.deitech.online
```

Si es satisfactorio, debemos recibir el mensaje “welcome to GitLab, @MINOMBRE DE USUARIO!”

Con lo cual podremos subir nuestro código al repositorio de GitLab.

Subimos nuestro primer código

Como ejemplo, subiremos un código fuente hecho en Java, con toda la estructura por defecto que arma Maven para el proyecto, el mismo se puede descargar desde aquí <https://vps.deitech.online/dhinfra.rar>

1. Una vez descargado y descomprimido, nos dirigimos hacia la carpeta "dhinfra" en nuestro Shell. Una vez allí dentro, inicializamos el repositorio con el comando

```
git init
```

2. Agregamos el origen remoto con el comando, teniendo en cuenta que en lugar de "testdh" ira su nombre de usuario y en lugar de "infra2", el nombre de su proyecto

```
git remote add origin https://gl.deitech.online/testdh/infra2.git
```

3. Agregamos los archivos del proyecto con

```
git add .
```

4. Realizamos el commit con

```
git commit -m "1erCommit"
```

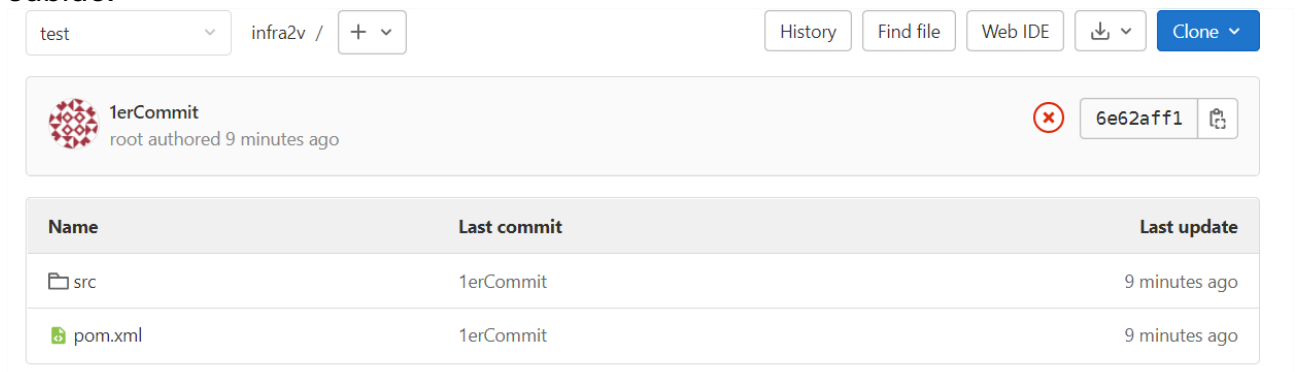
5. Cambiamos al Branch de test (IMPORTANTE: por defecto GitLab protege al Branch main, es por ello que usamos un Branch de test)

```
git branch -M test
```

6. Finalmente hacemos push de nuestro código

```
git push -uf origin test
```

7. Observamos en GitLab, en el Branch correspondiente que nuestro código ya está subido.



Name	Last commit	Last update
src	1erCommit	9 minutes ago
pom.xml	1erCommit	9 minutes ago