



Manual de Configuración y Usuario

Configuración y Manejo de GitLab

Dirección de Tecnologías e Información

Elaborado por:

Mgs. Norma Arcos
Mgs. Javier Chillogallo

Versión 1.0
Fecha de emisión: 8 de mayo de 2024
Quito – Ecuador

Contenido

1. INSTALACION DEL SERVIDOR DE VERSIONAMIENTO DE CODIGO FUENTE GITLAB	3
2. DESCARGA DE IMAGEN DE GITLAB.....	3
3. IMPLEMENTACION DE OVA(Formato estándar de maquinas virtuales) ...	3
4. CONFIGURACION DEL SERVIDOR	3
5. MANEJO DEL SERVIDOR DE CORREO ELECTRONICO GITLAB	5
Merge Request.....	11
6. BIBLIOGRAFIA.....	14
7. FIRMAS.....	14

- 1. INSTALACION DEL SERVIDOR DE VERSIONAMIENTO DE CODIGO FUENTE GITLAB**
- 2. DESCARGA DE IMAGEN DE GITLAB**

- Para el despliegue del servidor gitlab, se utiliza la imagen del sitio bitnami que **se dedica a crear instaladores y paquetes de aplicaciones web, así como máquinas virtuales con entornos de desarrollo completos.**

Link git: <https://bitnami.com/stack/gitlab/virtual-machine>

- 3. IMPLEMENTACION DE OVA(Formato estándar de maquinas virtuales)**

- La imagen del sitio se intalo en la tecnología se servidores vmware del Congope.
- Se asigno la ip: 172.24.5.35
- Se asigno el nombre git.congope.gob.ec en el dns interno.

- 4. CONFIGURACION DEL SERVIDOR**

CORREO ELECTRONICO:

Para la habilitación del correo electrónico en el servidor se realizaron las siguientes tareas:

- Se configuro una cuenta en el servicio mailjet.com para utilizarlo para el envío de correos desde el servidor gitlab.
 - Las credenciales del mismo fueron enviadas a los correos electrónicos de los funcionarios de la Dirección de Tecnologías de Información
- Con el correo electrónico se realizó la siguiente configuración en el archivo /etc/gitlab/gitlab.rb

```

gitlab_rails['smtp_enable'] = true
gitlab_rails['smtp_address'] = "in-v3.mailjet.com"
gitlab_rails['smtp_port'] = 587
gitlab_rails['smtp_user_name'] = "d367bxxxxxxxxxxxxxxxxxxxxxx 7ee"
gitlab_rails['smtp_password'] = "9a7cd8xxxxxxxxxxxxxxxxxxxxxx da"
gitlab_rails['smtp_domain'] = "in-v3.mailjet.com"
gitlab_rails['smtp_authentication'] = "plain"
gitlab_rails['smtp_enable_starttls_auto'] = true
gitlab_rails['smtp_tls'] = false
#gitlab_rails['smtp_ssl'] = false
# gitlab_rails['smtp_pool'] = false

###! **Can be: 'none', 'peer', 'client_once', 'fail_if_no_peer_cert'**
###! Docs: http://api.rubyonrails.org/classes/ActionMailer/Base.html
#gitlab_rails['smtp_openssl_verify_mode'] = 'peer'

# gitlab_rails['smtp_ca_path'] = "/etc/ssl/certs"
# gitlab_rails['smtp_ca_file'] = "/etc/ssl/certs/ca-certificates.crt"

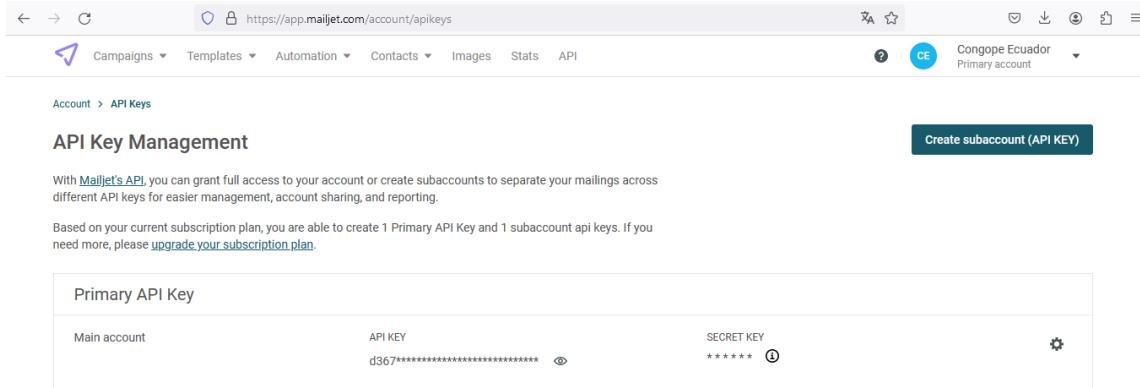
### Email Settings

# gitlab_rails['gitlab_email_enabled'] = true

### If your SMTP server does not like the default 'From: gitlab@gitlab.example.com'
### can change the 'From' with this setting.
gitlab_rails['gitlab_email_from'] = 'gitlab@congope.gob.ec'
#gitlab_rails['gitlab_email_display_name'] = 'Example'
gitlab_rails['gitlab_email_reply_to'] = 'noreply@congope.gob.ec'
# gitlab_rails['gitlab_email_subject_suffix'] = ''

```

El usuario y contraseña se configuran en el sitio mailjet.com, en el username se ubica el apikey y en el smtp_password el secretkey



The screenshot shows the Mailjet API Keys Management page. At the top, there's a navigation bar with links for Campaigns, Templates, Automation, Contacts, Images, Stats, API, and a user account dropdown for 'Congope Ecuador Primary account'. Below the navigation is a breadcrumb trail 'Account > API Keys' and a 'Create subaccount (API KEY)' button. The main section is titled 'API Key Management' and contains a table for the 'Primary API Key'. The table has one row for the 'Main account' with columns for 'API KEY' (containing 'd367*****') and 'SECRET KEY' (containing '*****'). There's also a gear icon for settings.

Nota: el servidor configurado tiene una limitante de envío de correos electrónicos de 6000 correos mensuales o 200 diarios, si el equipo de la Dirección de Tecnología e Información requiere realizar algún cambio en la parametrización del correo electrónico lo puede realizar en este apartado, también puede revisar la documentación en <https://docs.gitlab.com/omnibus/settings/smtp.html>

CONFIGURACIÓN DEL DNS INTERNO:

- Una vez asignado el dns con el que se ubicara al servidor en la red interna de igual forma se configuro el nombre por defecto para la ubicación en los repositorios
- Se edita el archivo /etc/gitlab/gitlab.rb

```
## GitLab URL
##! URL on which GitLab will be reachable.
##! For more details on configuring external_url see:
##! https://docs.gitlab.com/omnibus/settings/configuration.html#configuring-the-external-url-for-gitlab
##!
##! Note: During installation/upgrades, the value of the environment variable
##! EXTERNAL_URL will be used to populate/replace this value.
##! On AWS EC2 instances, we also attempt to fetch the public hostname/IP
##! address from AWS. For more details, see:
##! https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/instancedata-data-retrieval.html
external_url 'https://git.congope.gob.ec'

## Roles for multi-instance GitLab
##! The default is to have no roles enabled, which results in GitLab running as an all-in-one instance.
##! Options:
##!   redis_sentinel_role redis_master_role redis_replica_role geo_primary_role geo_secondary_role
##!   postgres_role consul_role application_role monitoring_role
##! For more details on each role, see:
##! https://docs.gitlab.com/omnibus/roles/index.html#roles
##!
# roles ['redis_sentinel_role', 'redis_master_role']
```

EJECUCION DE LA CONFIGURACION:

Para que los cambios sean ejecutados en el servidor es necesario realizar las siguientes acciones:

- Ejecutar los comandos
 - `sudo gitlab-ctl reconfigure`
 - `sudo gitlab-ctl restart`
- Una vez realizada esta configuración se debe realizar una verificación del envío de los correos realizando las siguiente acciones
 - Abrir la consola gitlab rails ejecutando el siguiente comando
 - `gitlab-rails console`
 - Una vez realizada esta tarea se puede comprobar el envío del correo electrónico ejecutando la siguiente instrucción
 - `Notify.test_email('destination_email@address.com', 'Message Subject', 'Message Body').deliver_now`
- Si todo es exitoso el resultado debe mostrarse se la siguiente forma.

```
irb(main):004:0> Notify.test_email('echillogallo@congope.gob.ec', 'Message Subject', 'Message Body').deliver_now
Delivered mail 66100a5770ad7_79192cec93071@debian.mail (3379.0ms)
=> #<Mail::Message:477340> Multipart: false, Headers: <Date: Fri, 05 Apr 2024 09:27:35 -0500>, <From: GitLab <congopetics@outlook.com>>,
<Reply-To: GitLab <noreply@git.congope.gob.ec>>, <To: echillogallo@congope.gob.ec>, <Message-ID: <66100a5770ad7_79192cec93071@debian.mail
>>, <Subject: Message Subject>, <Mime-Version: 1.0>, <Content-Type: text/html; charset=UTF-8>, <Content-Transfer-Encoding: 7bit>, <Auto-submitted: auto-generated>, <X-Auto-Response-Suppress: All>
irb(main):005:0>
```

5. MANEJO DEL SERVIDOR DE CORREO ELECTRONICO GITLAB

Para el login se creó un usuario admin denominado congopegit, las credenciales fueron enviadas al correo electrónico de los funcionarios de la Dirección.

Nota: Este es el usuario admin de GitLab, pero se pueden crear mas administradores de acuerdo con la necesidad institucional.

Observación: Cuando de instala la aplicación se crea un usuario root por defecto, si lo requieren pueden eliminar al mismo o cambiar la contraseña.



The screenshot shows the 'Projects' section of the GitLab web interface. On the left, there's a sidebar with links for 'Projects', 'Groups', 'Issues', 'Merge requests', 'To-Do List', 'Milestones', 'Snippets', and 'Activity'. The main area shows four projects under the 'Yours' tab:

- congopegit / CongopeBackSFGProv (Owner)
- congopegit / CongopeBddSFGProv (Owner)
- congopegit / CongopeFrontSFGProv (Owner)
- congopegit / CongopeSFGProv_V1 (Owner)

Each project entry includes a star icon, a green checkmark, a blue checkmark, a yellow checkmark, and a red checkmark, followed by the text 'Updated 2 months ago'.

Dentro de esta pantalla se crean los proyectos para el uso del GitLab, para el presente proyecto se han creado cuatro proyectos, pero se pueden crear la cantidad de proyectos que consideren necesario.

USUARIOS

En el área de administración existen las opciones que utiliza el Gitlab, para el caso puntual se agregan los usuarios que estimen conveniente.

The screenshot shows the 'Users' management page in the Admin Area. It lists the following users:

Name	Projects	Groups	Created on	Last activity	Action
Administrator (Admin) admin@example.com	0	0	Mar 27, 2024	Apr 04, 2024	Edit ::
congopegit (Admin) congopetics@outlook.com	4	1	Jan 27, 2024	Apr 04, 2024	Edit
Edwin Javier Chilogallo Solano echilogallo@congope.gob.ec	3	1	Apr 04, 2024	Jan 26, 2024	Edit ::
GitLab Alert Bot (Bot) alert@172.24.5.35	0	0	Jan 26, 2024	Never	
GitLab Support Bot (Bot) support@172.24.5.35	0	0	Jan 26, 2024	Never	

Para agregar nuevos usuarios se debe utilizar la opción New User y mostrara lo siguiente:

Admin Area / New User

New user

Account

Name

Username

Email

Password

 Reset link will be generated and sent to the user. User will be forced to set the password on first sign in.

Access

Default access

[Create user](#)

[Cancel](#)

En esta pantalla se crean los usuarios correspondientes, por recomendación crear usuarios con nombres completos, en username el usuario del correo institucional y en email el correo institucional, una vez creado el correo se enviara una invitación al correo registrado para que el colaborador ingrese al sitio.



Hi Edwin Javier Chillogallo Solano!

The Administrator created an account for you. Now you are a member of the company GitLab application.

login..... echillogallo@congope.gob.ec

[Click here to set your password](#)

This link is valid for 2 days. After it expires, you can [request a new one](#).

De igual forma puede Editar o Eliminar el usuario de acuerdo como lo consideren necesario.

Edwin Javier Chillogallo Solano

[Impersonate](#)[New identity](#)[Edit](#)

⋮

[Account](#) [Groups and projects](#) [SSH keys](#) [Identities](#) [Impersonation Tokens](#)

Edwin Javier Chillogallo Solano

Profile page: [echillogallo](#)

Profile

Member since Apr 4, 2024 8:28pm

Account:

Name: Edwin Javier Chillogallo Solano

Username: echillogallo

Email: echillogallo@congope.gob.ec Verified

ID: 11

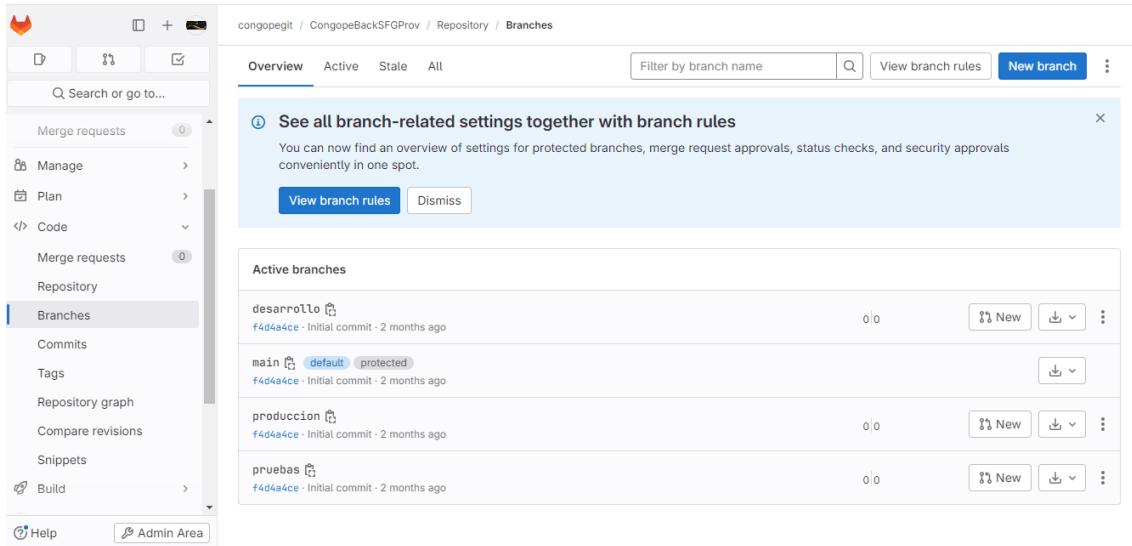
Namespace ID: 16

PROYECTOS

Cada uno de los proyectos tiene una configuración particular, sin embargo es necesario al menos realizar las siguientes actividades

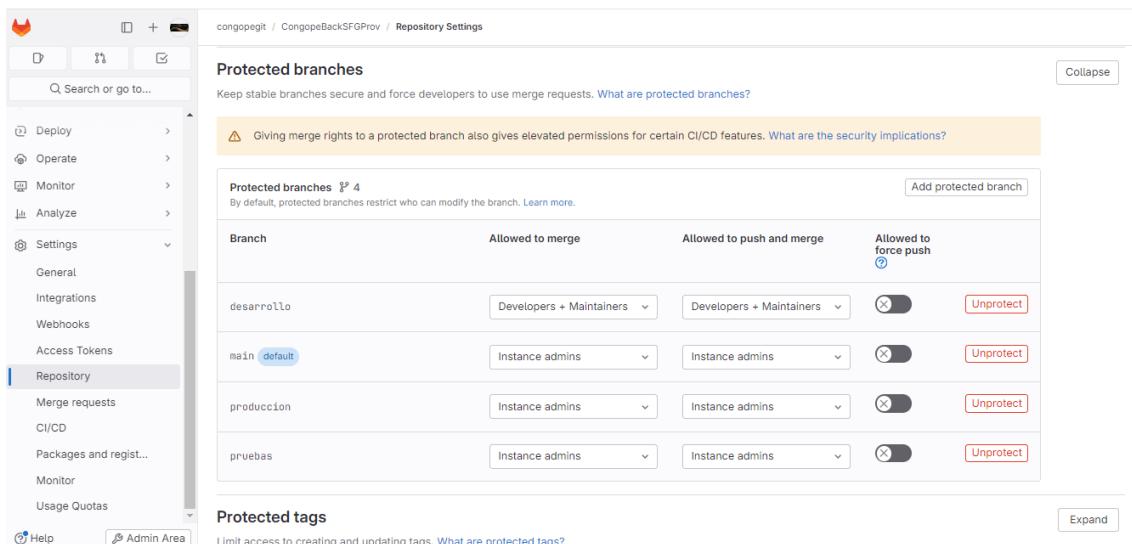
- CREAR RAMAS Y BRINDARLES SEGURIDAD
 - Esto se realiza con el objetivo de mantener seguras las ramas en donde se encuentre el código fuente.
 - Se recomienda al menos tener la siguiente estructura
 - Main: rama principal del proyecto, únicamente el usuario admin podría realizar merge en esta rama
 - Producción: rama de producción del proyecto, esta rama tendría el código fuente de producción, debería tener el mismo código del main, únicamente el usuario admin podría realizar merge en esta rama
 - Pruebas: Rama donde se encuentra el código que ya fue probado en desarrollo y necesita ser probado previo su despliegue en producción. únicamente el usuario admin podría realizar merge en esta rama, hasta que se designen usuarios de test y ambientes para este propósito
 - Desarrollo: En esta rama el equipo de desarrollo podría realizar los cambios necesarios.

- Esta información se parametriza ingresando a un proyecto en el apartado codes Branches.



The screenshot shows the 'Branches' page for the project 'congopegit / CongopeBackSFGProv'. It displays four active branches: 'desarrollo', 'main', 'produccion', and 'pruebas'. The 'main' branch is highlighted as the default and is marked as protected. A modal window titled 'See all branch-related settings together with branch rules' is open, providing information about protected branches and merge request approvals. Below the modal, there's a section for 'Active branches' with detailed information for each branch, including commit history and action buttons like 'New', 'Download', and 'More'.

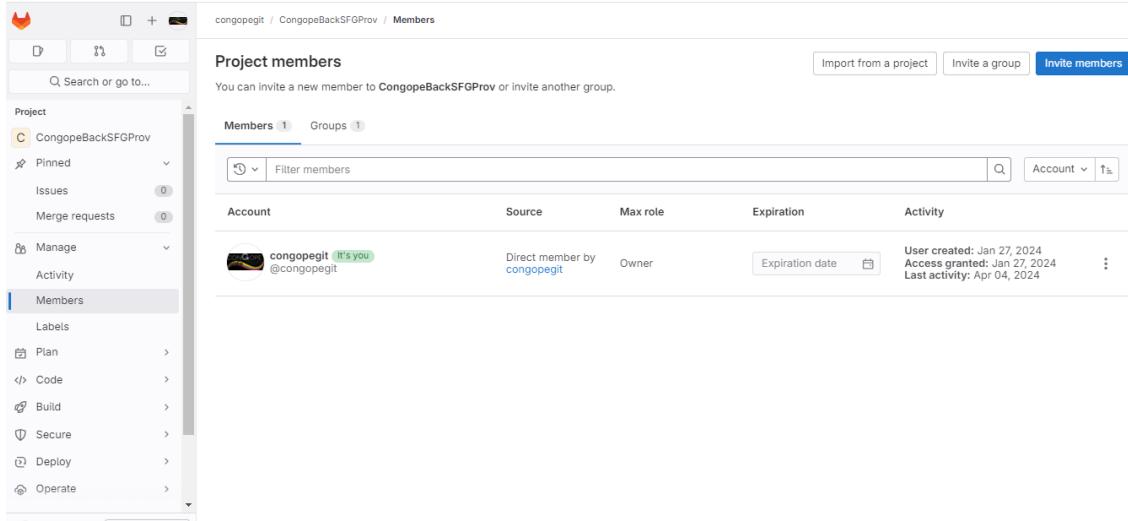
Haciendo clic en el ícono View Branch rules se puede cambiar la configuración mencionada



The screenshot shows the 'Repository Settings' page for the same project. Under the 'Protected branches' section, it lists the four branches with their current protection status and merge rights. Each row includes dropdown menus for selecting users or groups and a red 'Unprotect' button. The 'main' branch is currently set to 'default' and has 'Instance admins' selected for both merge and push permissions. The other branches have 'Developers + Maintainers' selected.

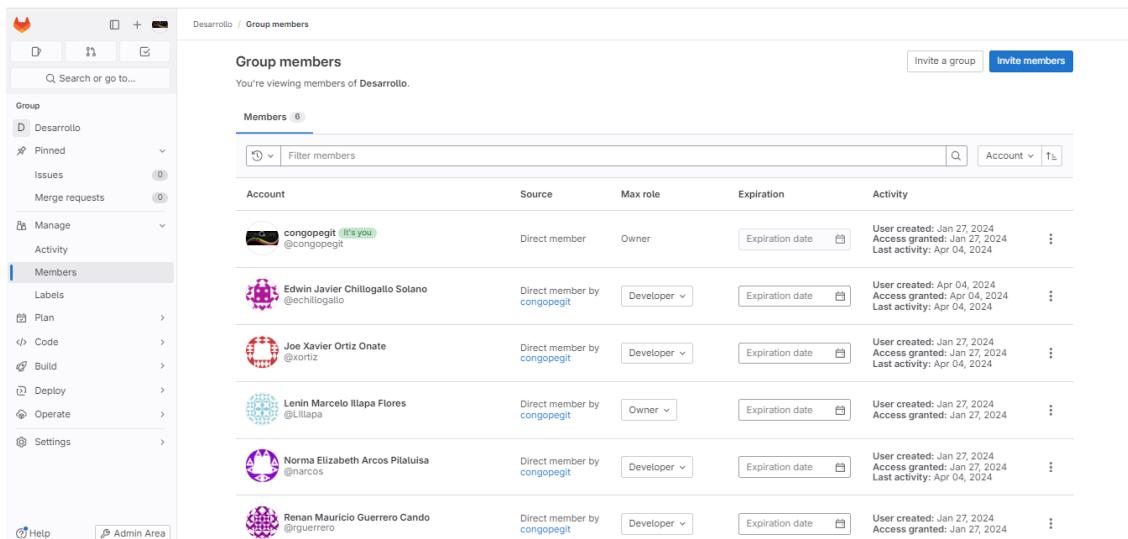
- **COLABORADORES**

Para cada proyecto se debe agregar los colaboradores nombrados con el propósito de permitirles ver únicamente el proyecto que deben visualizar



Para organizar a los colaboradores también se puede crear grupos los mismos que se les asigna roles de acceso a los proyectos.

Para este caso se ha creado un grupo denominado desarrollo



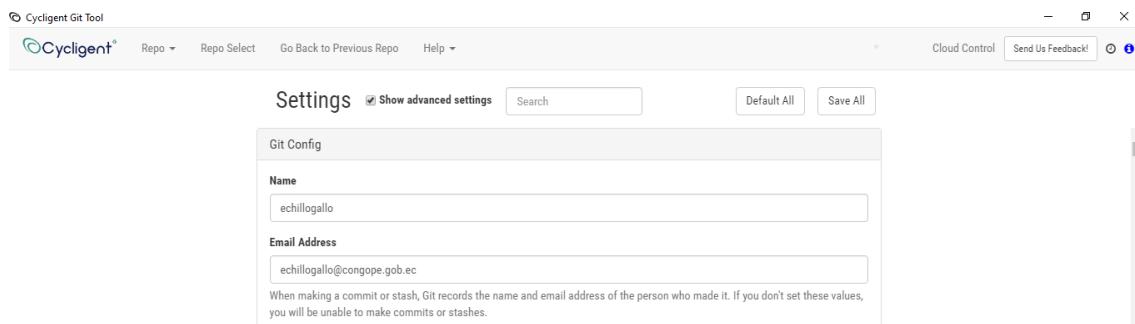
De acuerdo a lo indicado en la parte anterior se puede agregar nuevos miembros a los grupos, pero para realizar esta tarea es necesario primero crear los usuarios.

Los usuarios que se vayan creando deben tener los niveles mas bajos de privilegios.

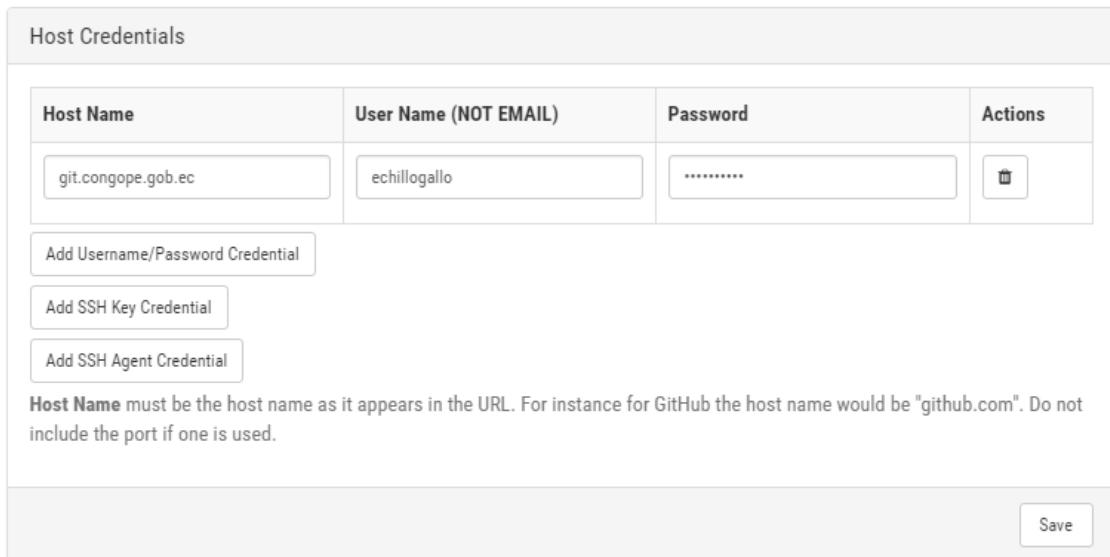
Merge Request.

Para utilizar la herramienta gitlab se puede realizarlo con cualquier cliente de git, a modo de ejercicio se va a utilizar la herramienta CycligentGitTool.

Para utilizar esta herramienta se va a configurar un sitio, clonando el repositorio



The screenshot shows the 'Settings' tab of the Cycligent Git Tool. Under the 'Git Config' section, the 'Name' field is set to 'echillogallo' and the 'Email Address' field is set to 'echillogallo@congope.gob.ec'. A note below states: 'When making a commit or stash, Git records the name and email address of the person who made it. If you don't set these values, you will not be able to make commits or stashes.'



The screenshot shows the 'Host Credentials' page. It displays a table with one row for the host 'git.congope.gob.ec'. The 'User Name (NOT EMAIL)' is 'echillogallo' and the 'Password' is masked as '.....'. There are buttons for 'Add Username/Password Credential', 'Add SSH Key Credential', and 'Add SSH Agent Credential'. A note at the bottom says: 'Host Name must be the host name as it appears in the URL. For instance for GitHub the host name would be "github.com". Do not include the port if one is used.' A 'Save' button is located at the bottom right.

Una vez realizada esta configuración, se realiza la configuración del repositorio

Credentials

Use Global Settings

Repository Information

Remote Repository URL

<https://git.congope.gob.ec/congopegit/congopeBackSFGProv.git>

 Test Connection

Successfully connected to the remote.

Branch to checkout (**OPTIONAL**)

Branch

If you do not specify a branch then the default branch, usually master, will be checked-out.

Folder to initialize as the cloned git repo (folder must be empty or must not exist)

Path



Repository display title (only used for display, not related to path)

Title

Image file (**OPTIONAL**)

Path



If an image file is not specified, an image will be created from the title. The image that will be used is shown below:



En esta configuración se determina en que cliente se va a clonar el sitio.

Luego de esta configuración ya tendremos una copia del sitio, listo para empezarlo a trabajar

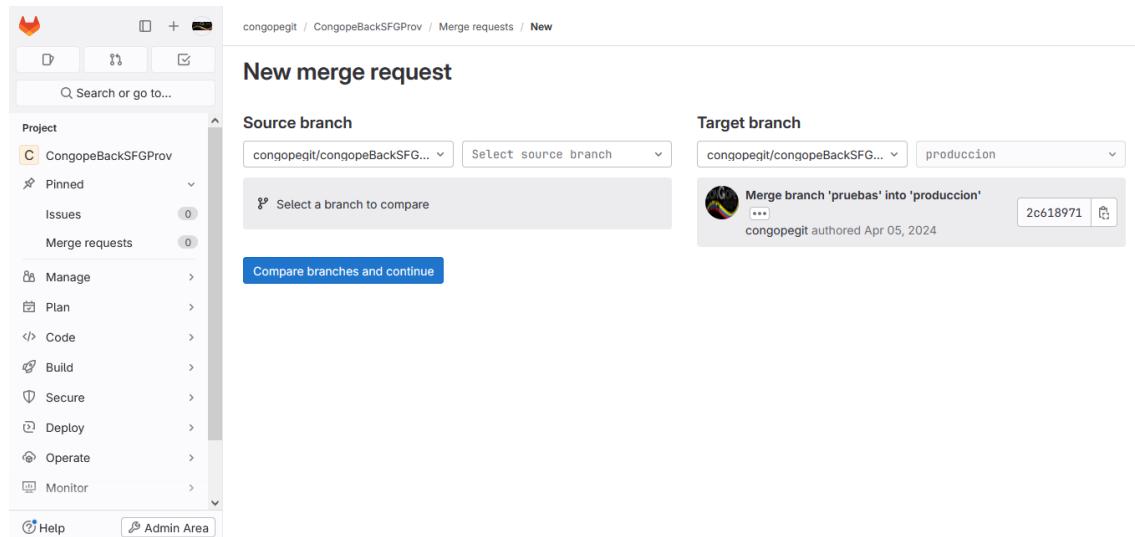
A partir de este punto podemos realizar las acciones en el git que correspondan al sitio, es decir crear una nueva rama y sobre esta realizar las acciones que se requieran de cambios.

La herramienta Cycligent detecta estos cambios para que sean subidos al servidor de versionamiento de código fuente.

Un vesh que se realiza el push al servidor gitlab se puede ya realizar las solicitudes de fusión o merge en las ramas.

Una vez realizado el push se debe ingresar al servidor y solicitar el merge respectivo.

El equipo de desarrollo puede realizar el merge a la rama de desarrollo sin embargo por políticas únicamente puede realizar la solicitud de merge a las ramas de pruebas, producción y main.



En este punto existen dos momentos

- La solicitud. - en donde el responsable solicita el merge de las ramas con esta acción se entiende que se desea ya probar el código fuente desarrollado.
- El merge o fusión. - Que ejecuta la fusión de las ramas para tener la última versión del cambio en el servidor git y con esta versión realizar el despliegue.

El orden de fusión debe ser el siguiente

EQUIPO DESARROLLO	DE	SERVIDOR GIT	SERVIDOR GIT
RAMA DESARROLLADA	RAMA DESARROLLADA		
	RAMA DESARROLLADA	RAMA DESARROLLO	
	RAMA DESARROLLO		RAMA PRUEBAS
	RAMA PRUEBAS		RAMA PRODUCCION
	RAMA PRODUCCION		RAMA MAIN

La organización de ramas es únicamente por recomendación sin embargo se puede organizar este proceso de forma que el equipo del congope pueda administrar de acuerdo con los roles internos que manejen.

6. BIBLIOGRAFIA

- <https://docs.gitlab.com/>
- <https://www.cycligent.com/git-tool>

7. FIRMAS

	CARGO	FIRMA	FECHA
ELABORADO POR	Mgs. Javier Chillogallo Profesional Contratado		08-05-2024
ELABORADO POR	Mgs. Norma Arcos Profesional Contratado		08-05-2024