CICD with Jenkins
Lab Notes Part 1

Introduction

This manual accompanies the course *CICD with Jenkins”.*

Most of these labs have bee previously demonstrated during class. These lab notes are references to what was done in class.

Lab One: Setting up Jenkins and GitLab

This lab walks through the basic steps of setting up Jenkins for use in subsequent labs. If you already have Jenkins set up, you can skip the setup instrutions

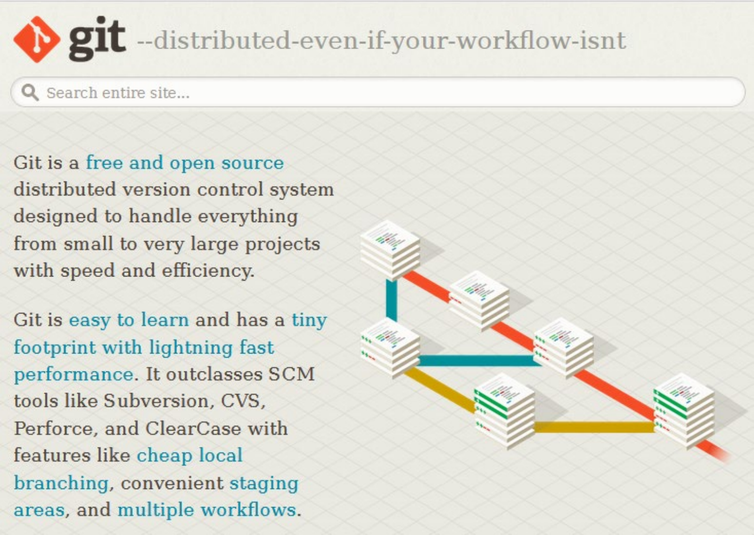
Verifying Your git Installation

You should already have git installed on your machine. To check, run the following command and verify that you get a version number as output – your number may be different.

```
C:\CICD>git --version
git version 2.33.1.windows.1
```

If you do not have git installed, download it and install it from the git website at:

https://git-scm.com/

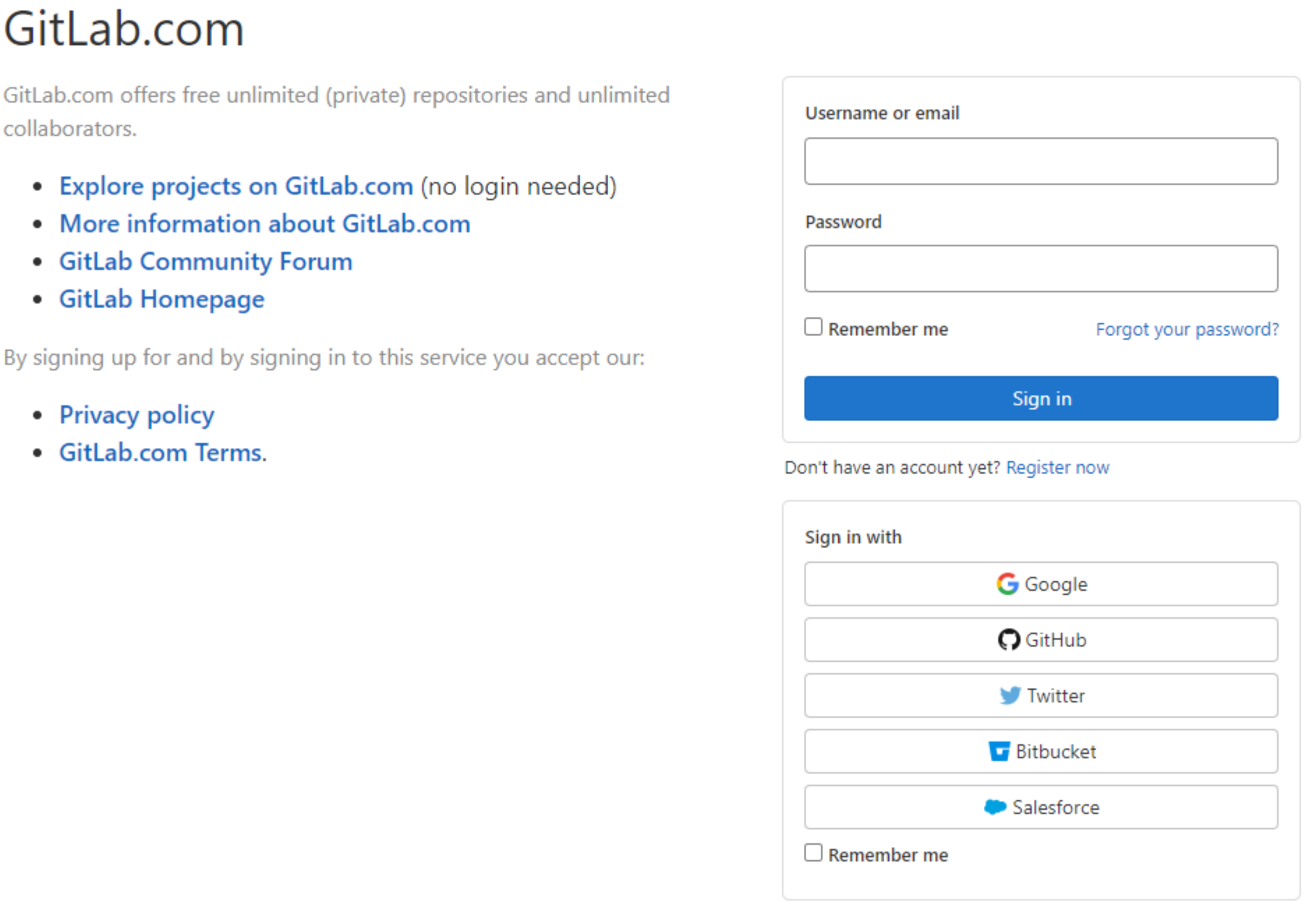


Git is a free and open source distributed version control system designed to handle everything from small to very large projects with speed and efficiency.

Git is easy to learn and has a tiny footprint with lightning fast performance. It outclasses SCM tools like Subversion, CVS, Perforce, and ClearCase with features like cheap local branching, convenient staging areas, and multiple workflows.

Setting up GitLab

If you already have a GitLab account, you can skip this step. If you don't have one, then go to GitLab.com and sign up for a free account. Since you will need to verify your email, use an account you will have access to in class. You do not need a credit



GitLab.com

GitLab.com offers free unlimited (private) repositories and unlimited collaborators.

- Explore projects on GitLab.com (no login needed)
- More information about GitLab.com
- GitLab Community Forum
- GitLab Homepage

By signing up for and by signing in to this service you accept our:

- Privacy policy
- GitLab.com Terms.

The process is standard and straight forward so it will not be documented here. It is a bit confusing since you must select the “login” option to get to the sign up page. A screenshot of the signup page is shown on the next page of the manual.

Setting up Credentials

First, you need to add your identification to your git global configuration. First check and see if they are already set. In the example below, no user information is set/

```
C:\CICD>git config --global –list

C:\CICD>
```

Set up a user name that will be used to identify all commits you make, as well and an email address by executing the commands shown on the next page.

```
C:\CICD>git config --global user.name "Rod Davison"

C:\CICD>git config --global user.email gitlab@exgnosis.ca

C:\CICD>git config --global --list
user.name=Rod Davison
user.email=gitlab@exgnosis.ca
```

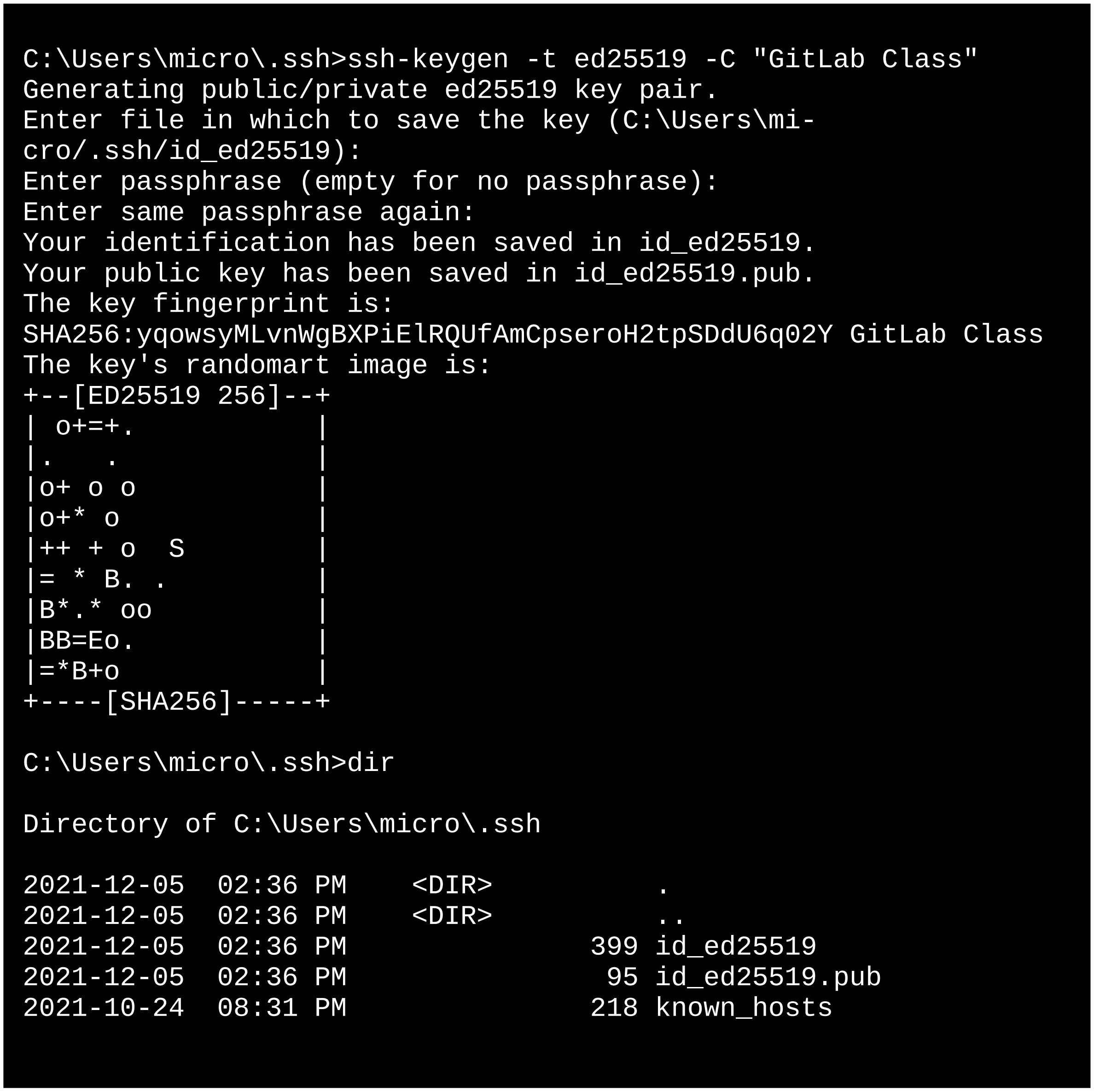
Creating the SSH Keys

Installing git ensures that there is an ssh installation on your machine. Navigate to the hidden .ssh directory on your machine, just under your user home directory. Depending on what you may have done with ssh before, there may be other files in this directory as well.

```
C:\Users\micro>cd .ssh

C:\Users\micro\.ssh>dir
 Volume in drive C has no label.
 Volume Serial Number is 2451-F407

 Directory of C:\Users\micro\.ssh

2021-12-03  05:03 PM    <DIR>          .
2021-12-03  05:03 PM    <DIR>          ..
2021-10-24  08:31 PM               218 known_hosts
```

Use the ssh-keygen command to generate a key pair. GitLab recommends using ed25519 type encryption. Add a comment, in the example on the next page, the comment is “GitLab Class”. Use the default filename suggested. This doesn’t work in Windows if you use your own filename

To be secure, we should provide a passphrase but since this key is only for demo purposes, no passphrase has been provided. This is NOT something you should do in your work environments. Ever.



```
C:\Users\micro\.ssh>ssh-keygen -t ed25519 -C "GitLab Class"
Generating public/private ed25519 key pair.
Enter file in which to save the key (C:\Users\mi-
cro/.ssh/id_ed25519):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in id_ed25519.
Your public key has been saved in id_ed25519.pub.
The key fingerprint is:
SHA256:yqowsyMLvnWgBXPiElRQUfAmCpseroH2tpSDdU6q02Y GitLab Class
The key's randomart image is:
+--[ED25519 256]--+
| o+=+.           |
|.   .            |
|o+ o o           |
|o+* o            |
|++ + o  S        |
|= * B. .         |
|B*.* oo          |
|BB=Eo.           |
|=*B+o            |
+----[SHA256]-----+

C:\Users\micro\.ssh>dir

Directory of C:\Users\micro\.ssh

2021-12-05  02:36 PM    <DIR>          .
2021-12-05  02:36 PM    <DIR>          ..
2021-12-05  02:36 PM               399 id_ed25519
2021-12-05  02:36 PM                95 id_ed25519.pub
2021-10-24  08:31 PM               218 known_hosts
```

Java

Jenkins needs JRE-8 or JRE-11. If you have a later version installed, you can download Java 8 or 11 and set your path variable to use that version temporarily. After class, you can reset your path variable and delete the version of Java you downloaded.

Installing Jenkins

You can follow the instructions on the Jenkins website based on the installer you are using. If you are using the WAR file, then the website states the following:

The Jenkins Web application ARchive (WAR) file can be started from the command line like this:

1. Download the latest stable Jenkins WAR file to an appropriate directory on your machine.
2. Open up a terminal/command prompt window to the download directory.
3. Run the command **java -jar jenkins.war**.
4. Browse to http://localhost:8080 and wait until the Unlock Jenkins page appears.

Setting up Jenkins

If you already have a Jenkins installed for the class, you can skip this step.

Run the command “java -jar Jenkins.war” and you will see out put that starts like that shown.

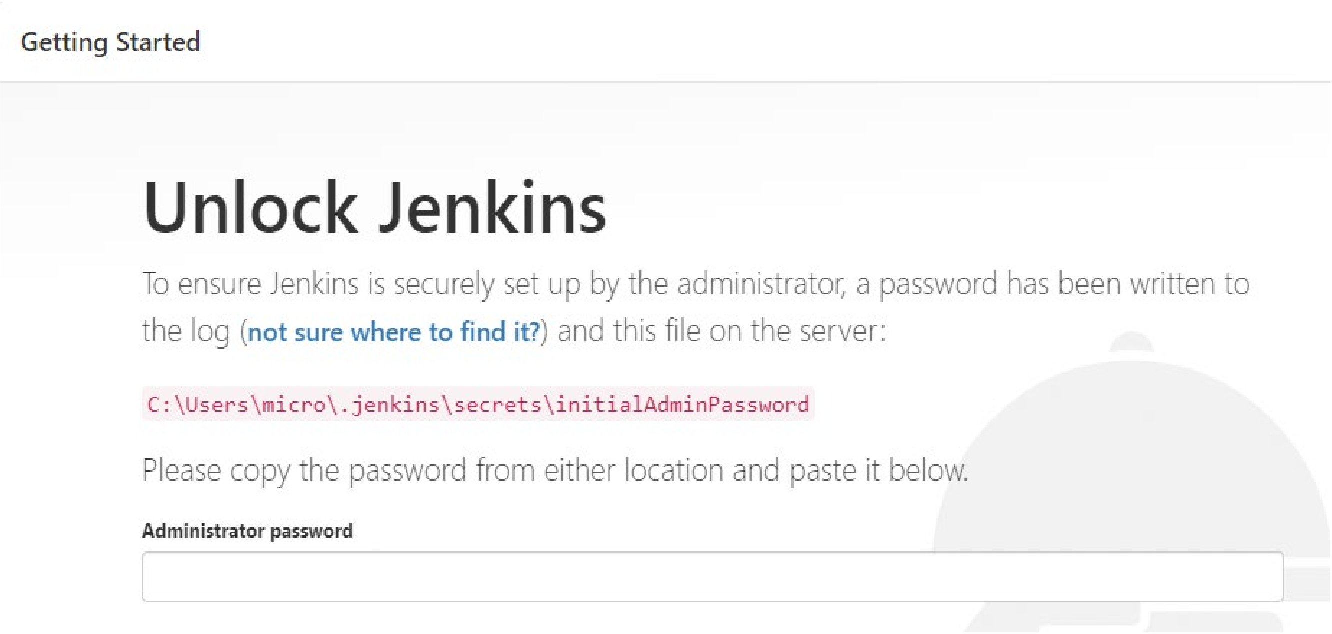
```
C:\tools\jenkins>java -version
java version "1.8.0_202"
Java(TM) SE Runtime Environment (build 1.8.0_202-b08)
Java HotSpot(TM) 64-Bit Server VM (build 25.202-b08, mixed mode)

C:\tools\jenkins>java -jar jenkins.war
Running from: C:\tools\jenkins\jenkins.war
webroot: $user.home/.jenkins
2021-12-12    21:19:24.267+0000    [id=1]                    INFO
org.eclipse.jetty.util.log.Log#initialized:  Logging  initialized
@641ms to org.eclipse.jetty.util.log.JavaUtilLog
2021-12-12 21:19:24.359+0000 [id=1]       INFO     winstone.Log-
ger#logInternal: Beginning extraction from war file
```

At the end of the output a generated password is displayed. Copy this password because you will need it in the next step.

```
*************************************************************
*************************************************************
*************************************************************

Jenkins initial setup is required. An admin user has been created
and a password generated.
Please use the following password to proceed to installation:

b9458783a578437fb9bd01f8d640a8d0

This may also be found at: C:\Users\micro\.jenkins\secrets\initialAdminPassword

*************************************************************
*************************************************************
*************************************************************
```

Going to “localhost:8080” brings up the initial login screen for the default admin account.

Enter the password from the previous step.



Getting Started

Unlock Jenkins

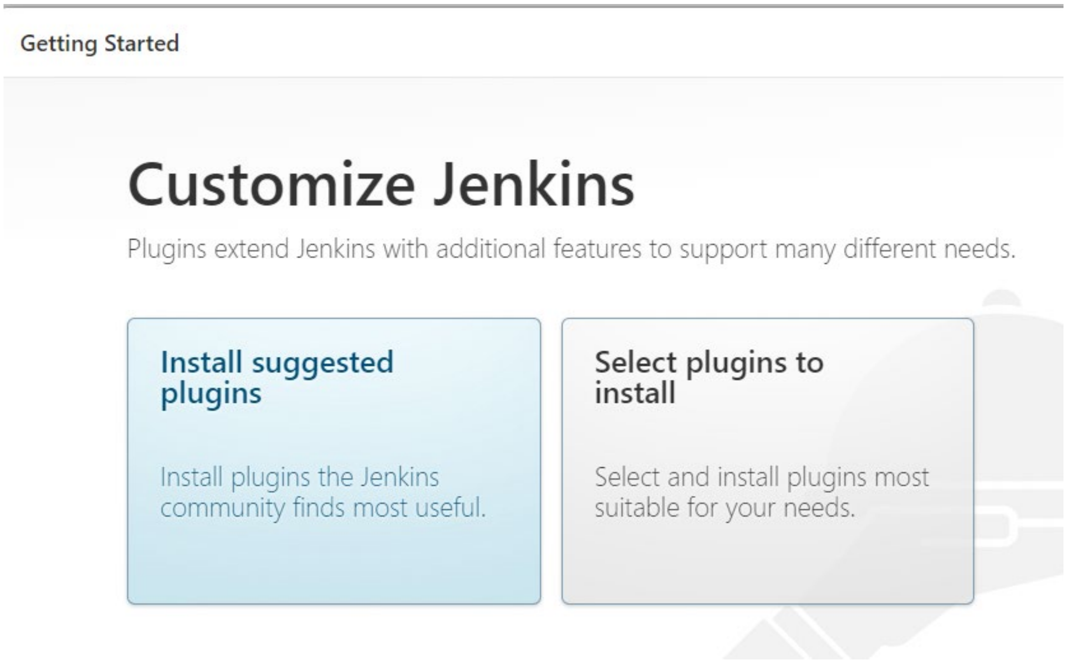
To ensure Jenkins is securely set up by the administrator, a password has been written to the log (not sure where to find it?) and this file on the server:

`C:\Users\micro\.jenkins\secrets\initialAdminPassword`

Please copy the password from either location and paste it below.

Administrator password

Then select the option to install the recommended plugins.



Customize Jenkins

Plugins extend Jenkins with additional features to support many different needs.

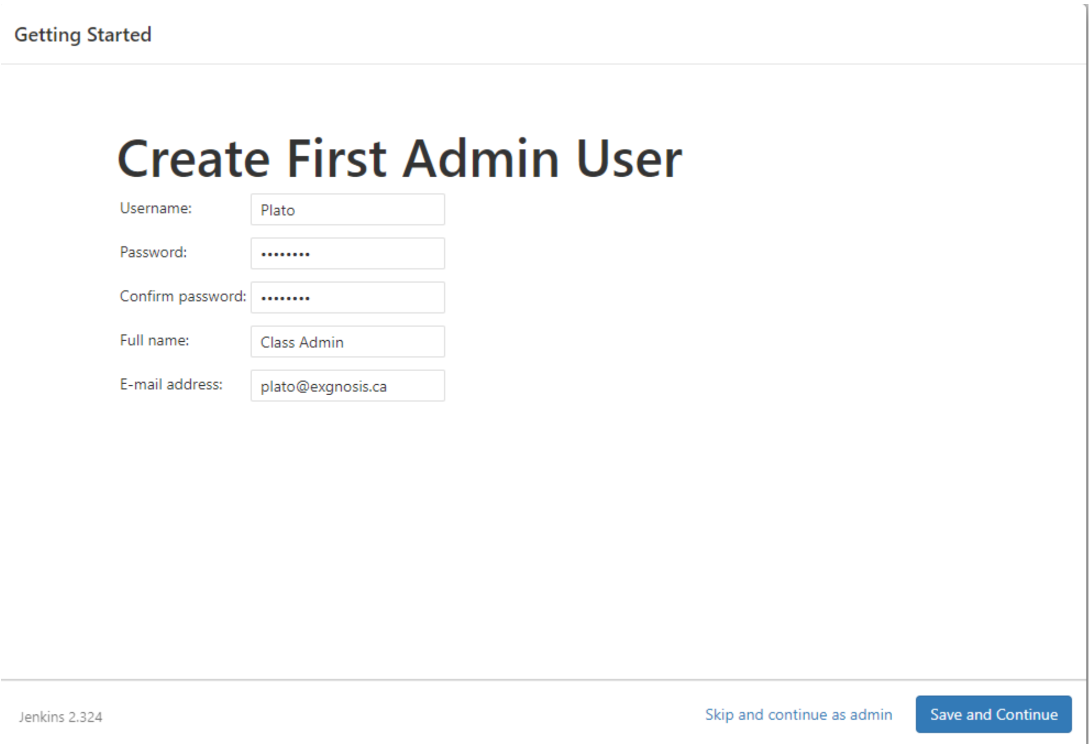
Install suggested plugins

Install plugins the Jenkins community finds most useful.

Select plugins to install

Select and install plugins most suitable for your needs.

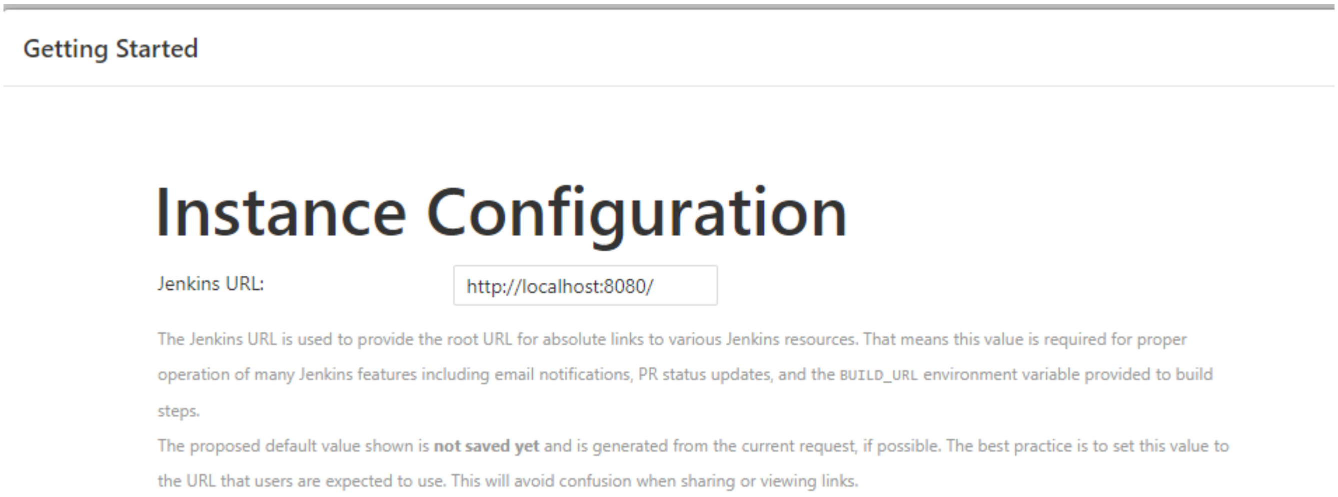
Once the plugins are installed, you will be asked to create and administrator account and password. You should not use the default admin account since it represents a security vulnerability (specifically, attackers would know the ID of the administrator.)



Create First Admin User

Username: Plato
Password: ••••••••
Confirm password: ••••••••
Full name: Class Admin
E-mail address: plato@exgnosis.ca

The next page up shows the URL that will be used to access Jenkins. Leave this as the default.



Getting Started

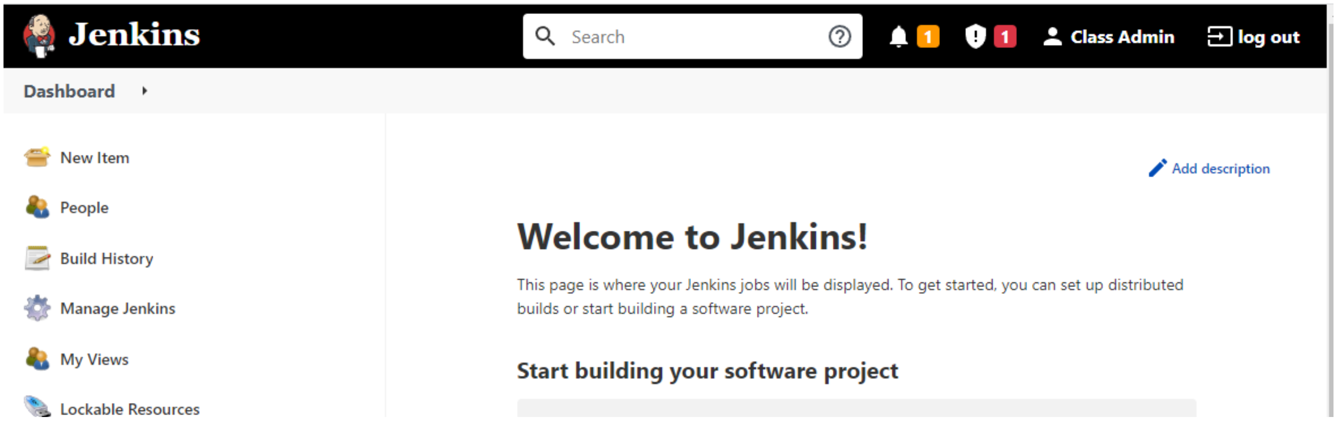
Instance Configuration

Jenkins URL: http://localhost:8080/

The Jenkins URL is used to provide the root URL for absolute links to various Jenkins resources. That means this value is required for proper operation of many Jenkins features including email notifications, PR status updates, and the `BUILD_URL` environment variable provided to build steps.

The proposed default value shown is **not saved yet** and is generated from the current request, if possible. The best practice is to set this value to the URL that users are expected to use. This will avoid confusion when sharing or viewing links.

This will bring you to the main Jenkins page logged in as the user you just created.



Welcome to Jenkins!

This page is where your Jenkins jobs will be displayed. To get started, you can set up distributed builds or start building a software project.

Start building your software project

Lab Two Create a Project

Create a new GitLab project with an autogenerated readme file from the create bank project option.

Open an issue to update the README file

Lab Three Create Feature Branch

Clone the lab using the SSH clone into a local directory.

```
C:\CICD>git clone git@gitlab.com:Exgnosis/cicd-lab-1.git
Cloning into 'cicd-lab-1'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (3/3), done.
```

Switch to the project directory and create the feature branch

```
C:\CICD\cicd-lab-1>git checkout -b readme-update
Switched to a new branch 'readme-update'
```

Edit the file and commit the changes

```
C:\CICD\cicd-lab-1>git add .

C:\CICD\cicd-lab-1>git status
On branch readme-update
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   README.md


C:\CICD\cicd-lab-1>git commit -m "Updates README File"
[readme-update 1e4ca61] Updates README File
 1 file changed, 5 insertions(+), 93 deletions(-)
 rewrite README.md (99%)
```

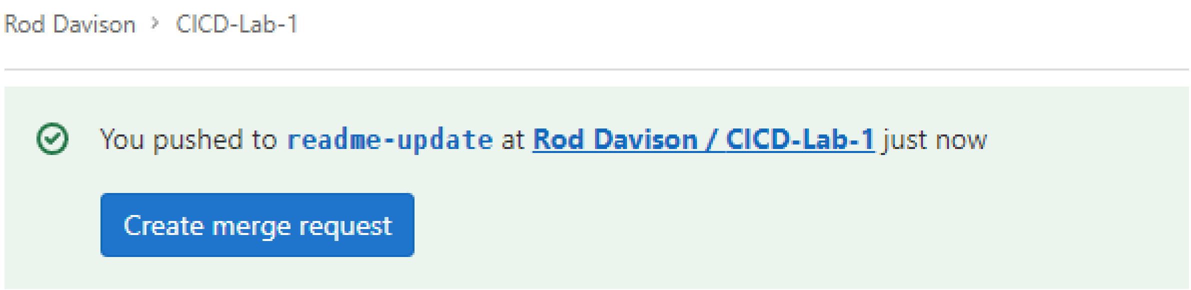
Push the changes to GitLab

```
C:\CICD\cicd-lab-1>git push -u origin readme-update
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 12 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 305 bytes | 305.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: To create a merge request for readme-update, visit:
remote:   https://gitlab.com/Exgnosis/cicd-lab-1/-/merge_requests/new?merge_request%5Bsource_branch%5D=readme-update
remote:
To gitlab.com:Exgnosis/cicd-lab-1.git
 * [new branch]      readme-update -> readme-update
Branch 'readme-update' set up to track remote branch 'readme-update' from 'origin'.
```

Go to GitLab, refresh the project page and ensure that the changes have been pushed correctly.

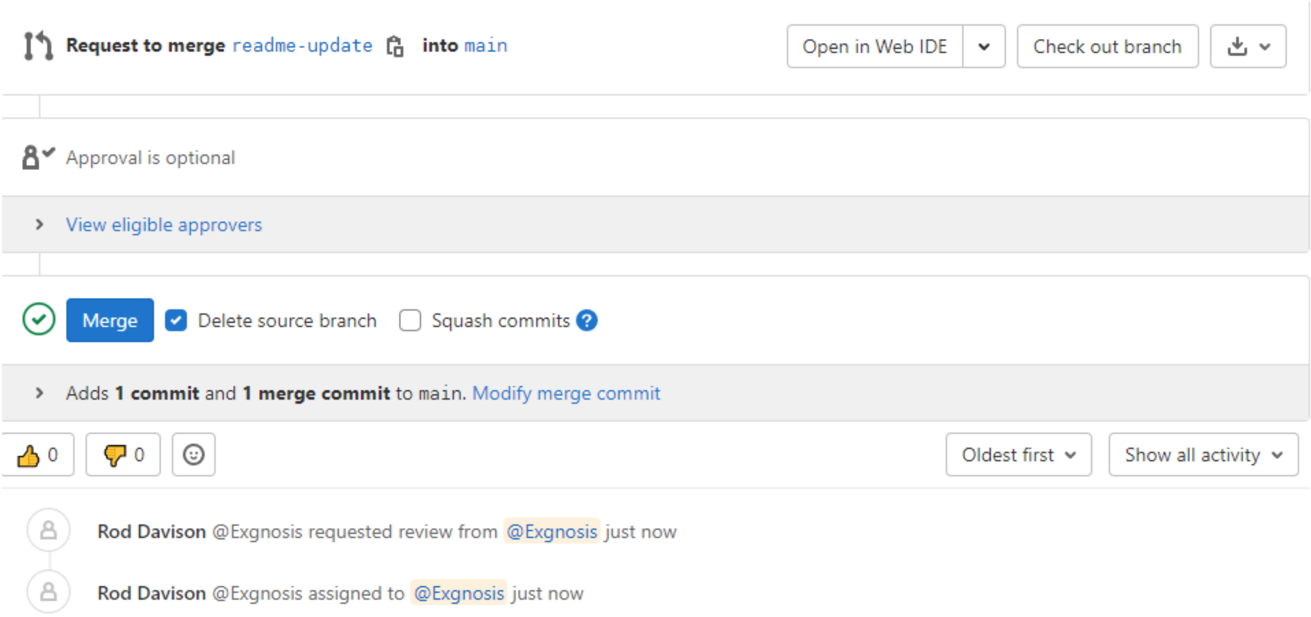
Lab Four Merge the Feature Branch

On the project page, there should be a notification to suggest creating a merge request



You pushed to `readme-update` at Rod Davison / CICD-Lab-1 just now

Create a merge request and assign it to yourself. Then perform the merge.



Request to merge `readme-update` **into** `main`

Approval is optional

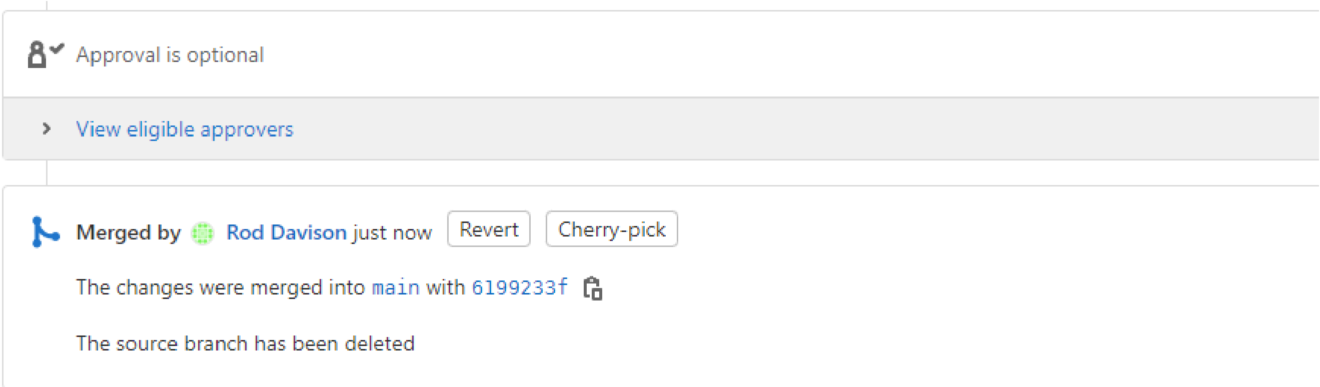
☑ Delete source branch □ Squash commits

Adds **1 commit** and **1 merge commit** to `main`. Modify merge commit

Rod Davison @Exgnosis requested review from @Exgnosis just now

Rod Davison @Exgnosis assigned to @Exgnosis just now

Confirm the merge is successful and that the feature branch has been removed



Approval is optional

Merged by Rod Davison just now

The changes were merged into `main` with `6199233f`

The source branch has been deleted

Finally, close the issue.