

Continuous Integration / Continuous Testing

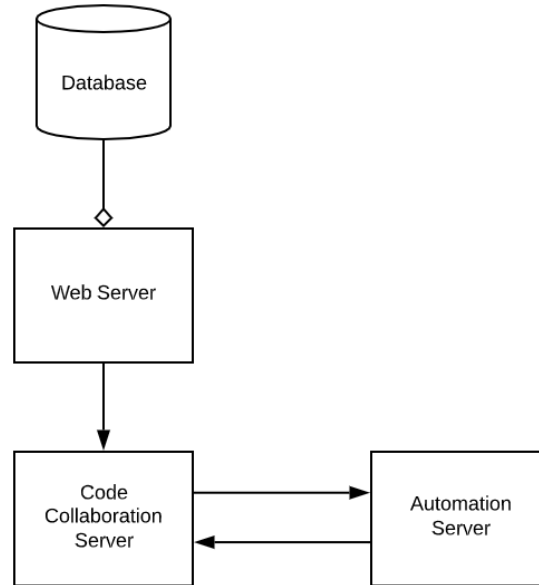
Seminary Part



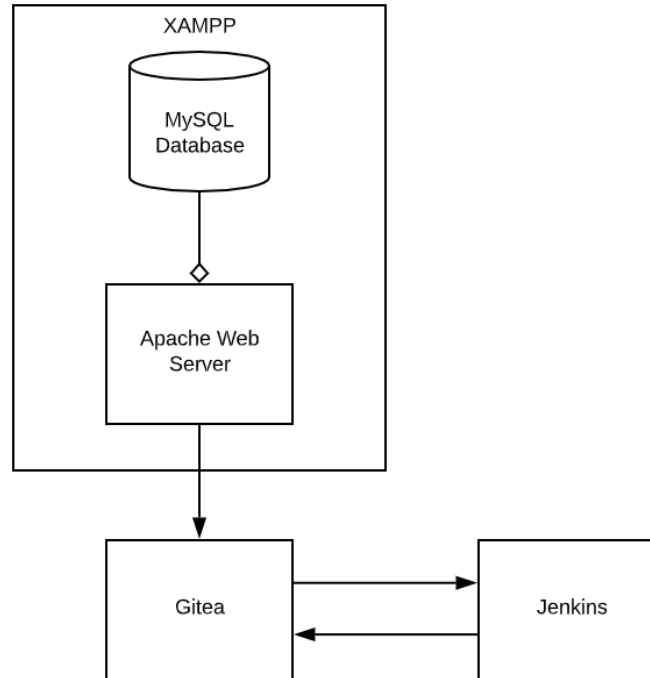
Concept

- › You are going to act as a true Software Engineer in a DevOps environment (“DevOps Engineer” – in marketing terms)
- › Based on a given architecture you are going to juggle between all the DevOps main functional areas (System Administration, Infrastructure, Platform and Site Reliability Engineering) and set-up from scratch the necessary tools and a Continuous Delivery System on top.

Arch – Infrastructure



Arch – Infra – Toolchain



1st Part: System Administration Ops (SysOps)

- › **Purpose:** install and set-up the necessary tools
- › **Points:** 1 / 5

Resources

Python 3.x installed

- › <https://www.python.org/downloads/>

Git installed

- › <https://git-scm.com/downloads>

Gitea: <https://github.com/go-gitea/gitea/>

- › Windows: <https://github.com/go-gitea/gitea/releases/download/v1.9.4/gitea-1.9.4-windows-4.0-386.exe>
- › Linux: <https://github.com/go-gitea/gitea/releases/download/v1.9.4/gitea-1.9.4-linux-386>

XAMPP: <https://www.apachefriends.org/>

- › Windows: <https://www.apachefriends.org/xampp-files/7.3.10/xampp-windows-x64-7.3.10-0-VC15-installer.exe>
- › Linux: <https://www.apachefriends.org/xampp-files/7.3.10/xampp-linux-x64-7.3.10-0-installer.run>

Jenkins: <https://jenkins.io/>

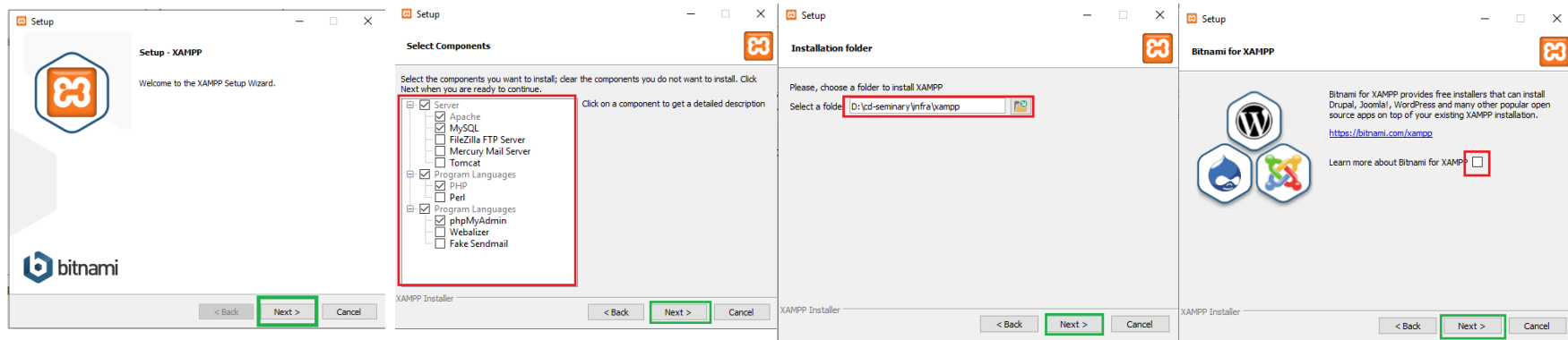
- › Windows: <https://jenkins.io/download/thank-you-downloading-windows-installer-stable>
- › Linux: <https://pkg.jenkins.io/debian-stable>

File System Layout

- › Create the following directory structure either in your main Drive on Windows or in your user home space on Linux
 - › D:\cd-seminary
 - › |——code
 - › |——infra
 - › |——gitea
 - › |——jenkins
 - › |——xampp

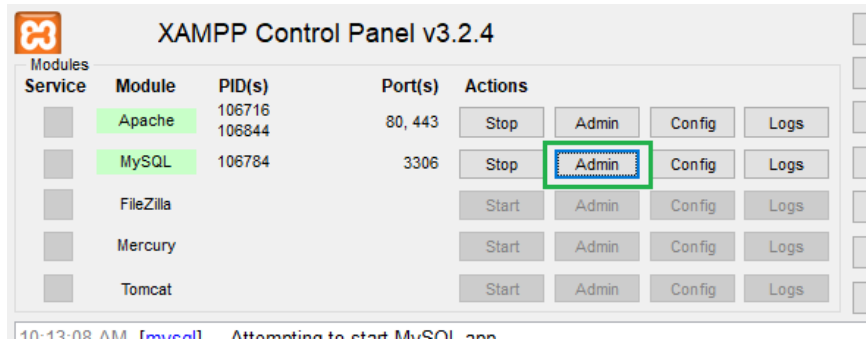
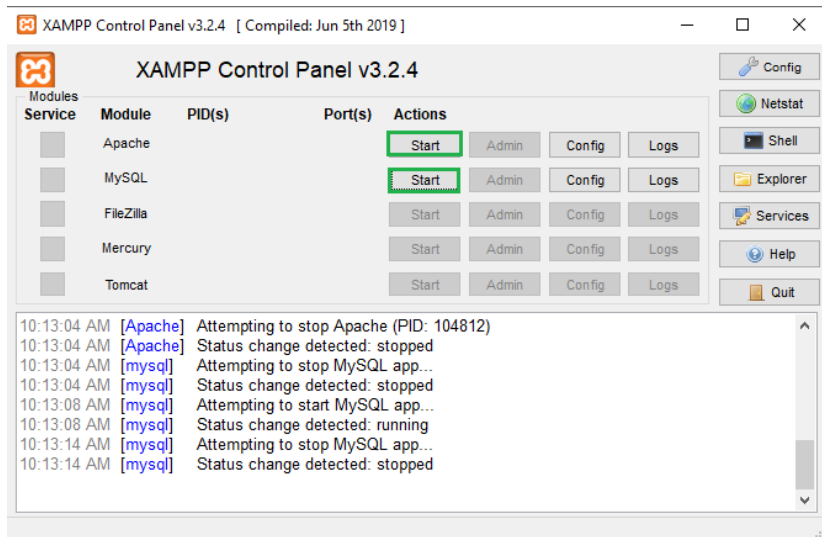
XAMPP Installation

Please follow the same steps to avoid issues!
Install only selected applications like in the print screen!



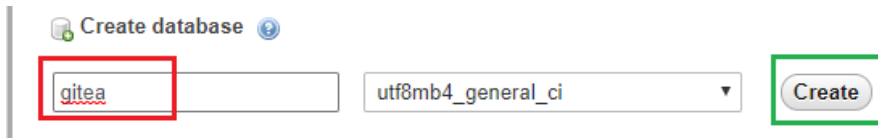
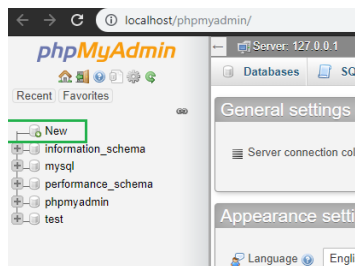
XAMPP Bring-up

- › Start the Apache Web Server & MySQL Database
- › Now access MySQL via the phpMyAdmin interface
- › A new tab into your default browser should be open on <http://localhost/phpmyadmin/>

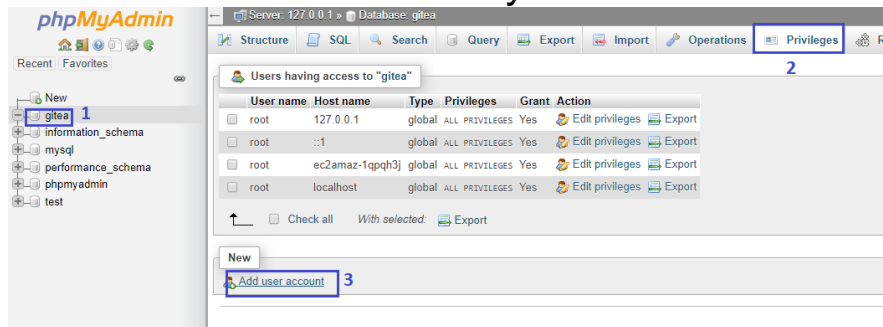


MySQL set-up

- › A new database has to be created in order to be populated with tables by the Gitea server.



- › A separate user has to be created to be used by the Gitea server



› username: gitea | password: gitea

Login Information

User name: Use text field:

Host name: Any host ⓘ

Password: Use text field: Strength: Extremely weak

Re-type:

Authentication Plugin: Native MySQL authentication ▼

Generate password:

Database for user account

☐ Create database with same name and grant all privileges.

☐ Grant all privileges on wildcard name (username_%).

☒ Grant all privileges on database gitea. ←

Global privileges ☐ Check all

Note: MySQL privilege names are expressed in English.

Data

- ☐ SELECT
- ☐ INSERT
- ☐ UPDATE
- ☐ DELETE
- ☐ FILE

Structure

- ☐ CREATE
- ☐ ALTER
- ☐ INDEX
- ☐ DROP
- ☐ CREATE TEMPORARY TABLES
- ☐ SHOW VIEW
- ☐ CREATE ROUTINE
- ☐ ALTER ROUTINE
- ☐ EXECUTE
- ☐ CREATE VIEW
- ☐ EVENT
- ☐ TRIGGER

Administration

- ☐ GRANT
- ☐ SUPER
- ☐ PROCESS
- ☐ RELOAD
- ☐ SHUTDOWN
- ☐ SHOW DATABASES
- ☐ LOCK TABLES
- ☐ REFERENCES
- ☐ REPLICATION CLIENT
- ☐ REPLICATION SLAVE
- ☐ CREATE USER

Resource limits

Note: Setting these options to 0 (zero) removes the limit.

MAX QUERIES PER HOUR

MAX UPDATES PER HOUR

MAX CONNECTIONS PER HOUR

MAX USER_CONNECTIONS

SSL

☒ REQUIRE NONE

☐ REQUIRE SSL

☐ REQUIRE X509

☐ SPECIFIED

REQUIRE CIPHER

REQUIRE ISSUER

REQUIRE SUBJECT

Gitea Bring-up

- › Copy the Gitea binary (.exe) into cd-seminary/infra/gitea and run it.
- › After the initialization, Gitea should be up on localhost:3000, open the page in a browser.
- › Click on either one of the Register or Sign In buttons for the Initial Configuration



- › Proceed by filling the fields as shown in the next pages

Part 1 / 3

Initial Configuration

If you run Gitea inside Docker, please read the [documentation](#) before changing any settings.

Database Settings

Gitea requires MySQL, PostgreSQL, MSSQL or SQLite3.

Database Type *	MySQL
Host *	127.0.0.1:3306
Username *	gitea
Password *	gitea
Database Name *	gitea
Charset *	utf8

Note to MySQL users: please use the InnoDB storage engine and if you use "utf8mb4", your InnoDB version must be greater than 5.6 .

Part 2 / 3

General Settings

Site Title *	Gitea: Git with a cup of tea	You can enter your company name here.
Repository Root Path *	D:\cd-seminary\infra\gitea-repositories	Remote Git repositories will be saved to this directory.
Git LFS Root Path	D:\cd-seminary\infra\gitea\data\lfs	Files tracked by Git LFS will be stored in this directory. Leave empty to disable.
Run As Username *	default	Enter the operating system username that Gitea runs as. Note that this user must have access to the repository root path.
SSH Server Domain *	localhost	Domain or host address for SSH clone URLs.
SSH Server Port	22	Port number your SSH server listens on. Leave empty to disable.
Gitea HTTP Listen Port *	3000	Port number the Gitea web server will listen on.
Gitea Base URL *	http://localhost:3000/	Base address for HTTP(S) clone URLs and email notifications.
Log Path *	D:\cd-seminary\infra\gitea\log	Log files will be written to this directory.

Part 3 / 3

Optional Settings

- ▶ Email Settings
- ▶ Server and Third-Party Service Settings
- ▼ Administrator Account Settings

Creating an administrator account is optional. The first registered user will automatically become an administrator.

Administrator Username

student

Password

..... student

Confirm Password

..... student

Email Address

student@issa.conti

Install Gitea



Dashboard

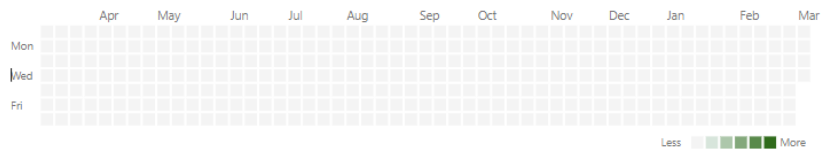
Issues

Pull Requests

Explore



student ▾



Repository

Organization

Repositories

0



Find a repository...



All

0

Sources

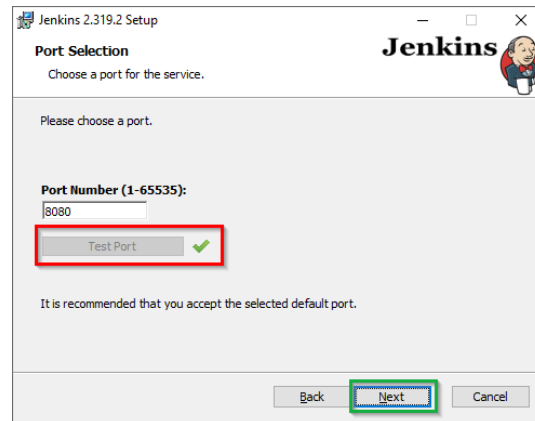
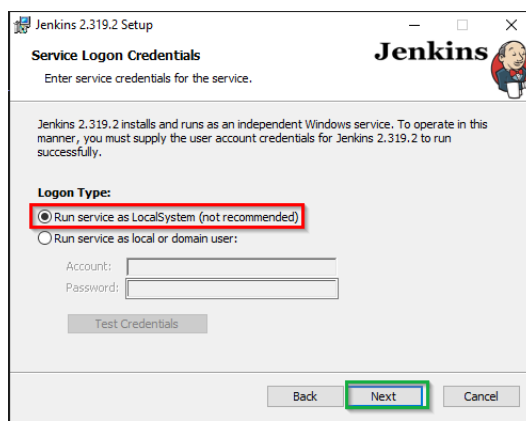
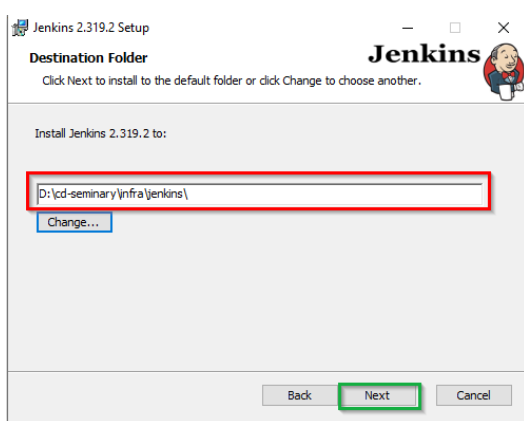
Forks

Mirrors

Collaborative

Jenkins Bring-up

- › Run the Jenkins binary and proceed to install it as shown in the next images.
- › **During Jenkins installation, you can test the port. If it is already used, please deactivate that application from services.msc**



- › After the installation ends. Jenkins will be available at **localhost:8080**

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:

`D:\cd-seminary\infra\jenkins\secrets\initialAdminPassword`

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

Administrator password



Continue

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.



If you encounter an error installing plugins, see links below:

- › <https://www.phpflow.com/misc/devops/how-to-manually-install-jenkins-plugin/>
- › <https://www.jenkins.io/doc/book/installing/offline/>

Create First Admin User

Username:

Password:

Confirm password:

Full name:

E-mail address:

- › Keep the defaults on the next step
- › Save & Finish
- › Jenkins should be up and you should be logged in with the 'jenkins' account

kins 2.190.1

[Continue as admin](#)

[Save and Continue](#)

2nd Part: Infrastructure Ops (InfraOps)

- › **Purpose:** Set-up the needed configurations that link the tools in between.
- › **Points:** 1 / 5

Install Gitea Jenkins plug-in

- › Access Jenkins pluginManager service and check for Gitea and install it.
<http://localhost:8080/pluginManager/available>

The screenshot shows the Jenkins Plugin Manager interface at `localhost:8080/pluginManager/available`. The page has a dark header with the Jenkins logo and a search bar. Below the header, there's a navigation bar with "Back to Dashboard" and "Manage Jenkins". The main content area has tabs for "Updates", "Available", "Installed", and "Advanced". The "Available" tab is selected, showing a table of available plugins. The table has columns for "Name" and "Version". The first row is for the "Gitea" plugin, which is checked in the "Install" column. The version is "1.1.2". Below the table, there are three buttons: "Install without restart" (highlighted with a red box and labeled 3), "Download now and install after restart", and "Check now". Above the table, there's a filter input field labeled "Filter: Gitea" (highlighted with a red box and labeled 1). The table header has a red box labeled 2 around the "Gitea" plugin entry.

1 Filter:

Install	Name	Version
<input checked="" type="checkbox"/>	Gitea SCM API implementation for Gitea	1.1.2

2

3

Update information obtained: 10 sec ago

Perform a Jenkins Safe Restart

- › Access <http://localhost:8080/safeRestart>
- › Click on **Yes**
- › Wait for the restart
- › Log in again (user: jenkins, password: jenkins)

Add Gitea server in Jenkins global scope

- › This is done in the system configuration page <http://localhost:8080/configure>
- › Scroll down to the **Gitea Server section** and **Add** a new one and then click on **Save**
- › If Jenkins can access the server, it should display the Gitea Version

The screenshot shows the Jenkins system configuration page. The 'Gitea Servers' section is highlighted with a red box. Below it, a new Gitea server is being added, with the 'Name' field set to 'issa-gitea' and the 'Server URL' set to 'http://localhost:3000'. The 'Gitea Version' is displayed as '1.9.3'. The 'Add' button is highlighted with a red box. Below the Gitea Servers section, there are sections for GitHub Servers, GitHub Enterprise Servers, Pipeline Model Definition, Global Pipeline Libraries, and Build-timeout Plugin > BuildStep Action. The 'Save' button is highlighted with a red box.

Gitea Servers

☐ Gitea Server

Name

Server URL

Gitea Version: 1.9.3

☐ Manage hooks

GitHub

GitHub Servers

GitHub Enterprise Servers

Pipeline Model Definition

Docker Label

Docker registry URL

Registry credentials

Global Pipeline Libraries

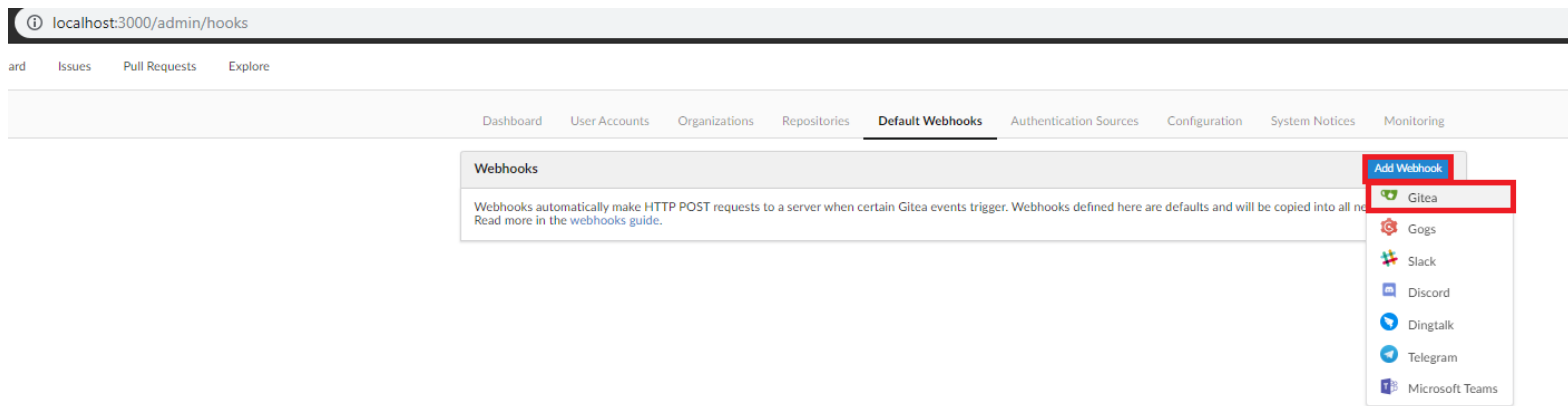
Sharable libraries available to any Pipeline jobs running on this system. These libraries will be trusted, meaning they run without "sandbox" restrictions and may use @Grab.

Build-timeout Plugin > BuildStep Action

☐ Enable BuildStep Action

Set-up the Gitea to Jenkins webhook

- › We are going to set-up a webhook where Gitea will perform HTTP POST requests each time a new event happens globally.
- › This is done on the Gitea side, in site admin's settings, in the Webhooks tab. Here is the direct link to it <http://localhost:3000/admin/hooks>



Add Default Webhook



Gitea will send POST requests with a specified content type to the target URL. Read more in the [webhooks guide](#).

Target URL *

http://localhost:8080/gitea-webhook/post

HTTP Method

POST

POST Content Type

application/json

Secret

Trigger On:

- ☐ Push Events
- ☒ All Events
- ☐ Custom Events...

☒ Active

Information about triggered events will be sent to this webhook URL.

Add Webhook

3rd Part: Platform Ops

- › **Purpose:** set-up the Continuous Delivery environment in Jenkins
- › **Points:** 1 / 5

Create the Gitea organization folder using the plugin

- › In the Jenkins home page, click on **New Item**
- › Then proceed to create a Gitea Organization folder

Enter an item name

cd-gitea-issa

» Required field

Freestyle project
This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any build system.

Pipeline
Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (for example, for building a project).

Multi-configuration project
Suitable for projects that need a large number of different configurations, such as testing on multiple environments.

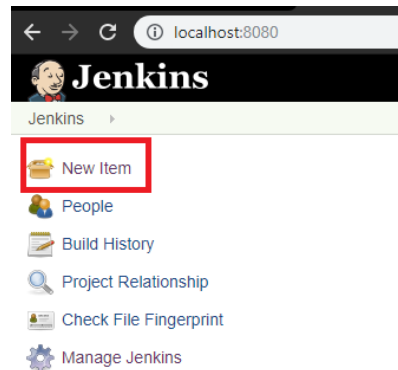
Folder
Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is as they are in different folders.

Multibranch Pipeline
Creates a set of Pipeline projects according to detected branches in one SCM repository.

Organization Folder
Creates a set of multibranch project subfolders by scanning for repositories.

OK

2 / 6



1 / 6

› You will be prompted to the Job configuration Page, don't save yet.

The screenshot shows the Jenkins Job Configuration Page for a job named "gitea!SSA Continuous Delivery". The "General" tab is selected, showing the job's name, description, and repository source. The description is "Continuous Delivery Project Code Collaboration Server: Gitea Organization: ISSA". The repository source is "Gitea Organization". The "Script Path" is "Jenkinsfile". The "Property strategy" section is partially visible at the bottom. The "Save" and "Apply" buttons are at the bottom left. A red box highlights the job name and description. A red box highlights the "Gitea Organization" option in the repository source dropdown. A red box highlights the "Gitea Organization" option in the repository source dropdown. A red box highlights the "Gitea Organization" option in the repository source dropdown.

General Projects Scan Organization Folder Triggers Orphaned Item Strategy Health metrics Properties Child Health metrics Child Orphaned Item Strategy Child Scan Triggers

Pipeline Libraries Automatic branch project triggering

Display Name

gitea!SSA Continuous Delivery

Description

Continuous Delivery Project
Code Collaboration Server: Gitea
Organization: ISSA

[Plain text] Preview

Disable

☐ (No new builds within this Organization Folder will be executed until it is re-enabled)

Projects

Repository Sources

Add

GitHub Organization

Gitea Organization

Single repository

Script Path ?

Jenkinsfile

Add

Property strategy

Save Apply

3 / 6

› You will be prompted to the Job configuration Page, don't save yet.

Projects

Gitea Organization

Server

Credentials

Owner

Behaviours

Discover branches

Strategy

Discover pull requests from origin

Strategy

Discover pull requests from forks

Strategy

Trust

Project Recognizers

Pipeline Jenkinsfile

Script Path

4 / 6

- › Also you have to add the credentials that have access to the specified organization (Owner)
- › These credentials are going to be available only for **cd-gitea-ISSA**

5 / 6

Server

issa-gitea (http://localhost:3000)

Credentials

- none -

cd-gitea-ISSA

Jenkins

Owner

ISSA

Server

issa-gitea (http://localhost:3000)

Credentials

- none -

student/*****

ISSA

Folder Credentials Provider: gitea:ISSA Continuous Delivery

Add Credentials

Domain

Global credentials (unrestricted)

Kind

Username with password

Username

student

☐ Treat username as secret

Password

student

ID

Description

Add Cancel

- › Go down into the job configuration page and update the following configurations and **Save**

Scan Gitea Organization Triggers

☒ Periodically if not otherwise run

Interval

Child Scan Triggers

These are the triggers that the children of this organization folder will use. Does not apply to grandchildren.

☒ Periodically if not otherwise run

Interval

Pipeline Libraries

Sharable libraries available to any Pipeline jobs inside this folder. These libraries will be untrusted, meaning their code runs in the Groovy sandbox.

Pipeline Model Definition

Docker Label

Docker registry URL

Registry credentials

Automatic branch project triggering

Branch names to build automatically

6 / 6

4th Part: Site Reliability Engineering Ops (SREOps)

- › **Purpose:** set-up the production environment, software development environment and the required configuration management particularities
- › **Points:** 1 / 5

Create a new Gitea organization

- › Access Gitea homepage <http://localhost:3000/> and proceed creating your ISSA organization

The screenshot shows the Gitea homepage at <http://localhost:3000/>. The user is logged in as 'student'. The navigation bar includes 'Dashboard', 'Issues', 'Pull Requests', and 'Explore'. The main content area shows a calendar and a list of organizations. The 'Organization' tab is selected, and the 'My Organizations' section shows a '+' button. The 'New Organization' form is displayed, with the 'Organization Name' field containing 'ISSA'. The 'Visibility' section has three options: 'Public' (selected), 'Limited (Visible to logged in users only)', and 'Private (Visible only to organization members)'. The 'Create Organization' button is highlighted.

localhost:3000

Dashboard Issues Pull Requests Explore

student

Nov Dec Jan Feb Mar Apr May Jun Jul Aug Sep Oct

Mon

Tue

Wed

Thu

Repository Organization 1

My Organizations 0 + 2

New Organization

Organization Name * ISSA

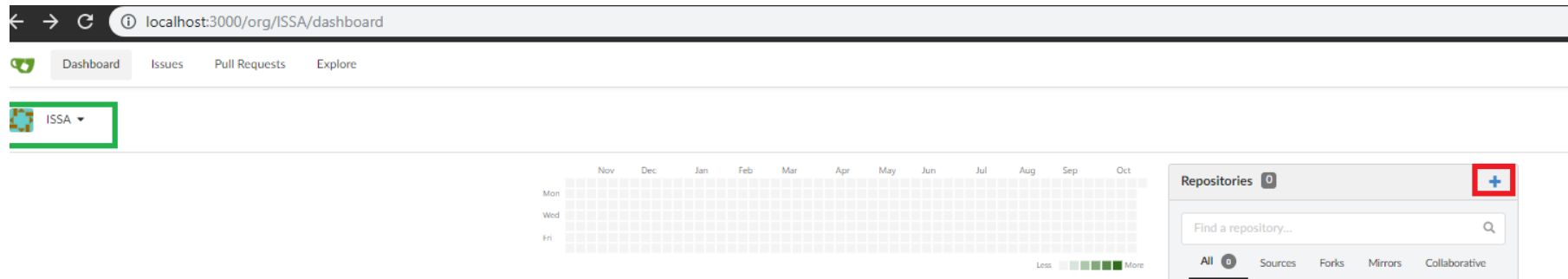
Organization names should be short and memorable.

Visibility * ☒ Public ☐ Limited (Visible to logged in users only) ☐ Private (Visible only to organization members)

Create Organization Cancel

Create your project Git repository

- › From your ISSA's organization dashboard page
<http://localhost:3000/org/ISSA/dashboard>, proceed creating your project's Git repository



New Repository

Owner*  ISSA ▼

Repository Name*

Good repository names use short, memorable and unique keywords.

Visibility ☐ Make Repository Private

Description

.gitignore

License

README

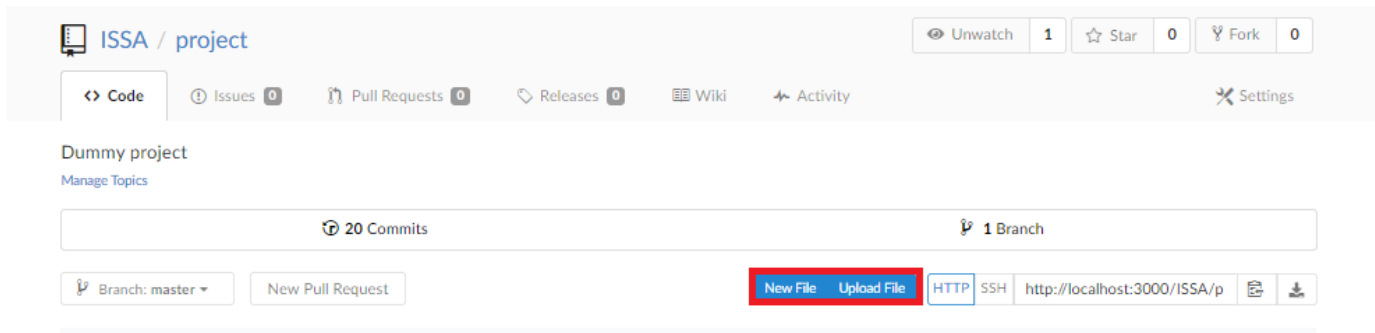
☒ Initialize Repository (Adds .gitignore, License and README)

Create Repository

Cancel

How to commit a new file(s)

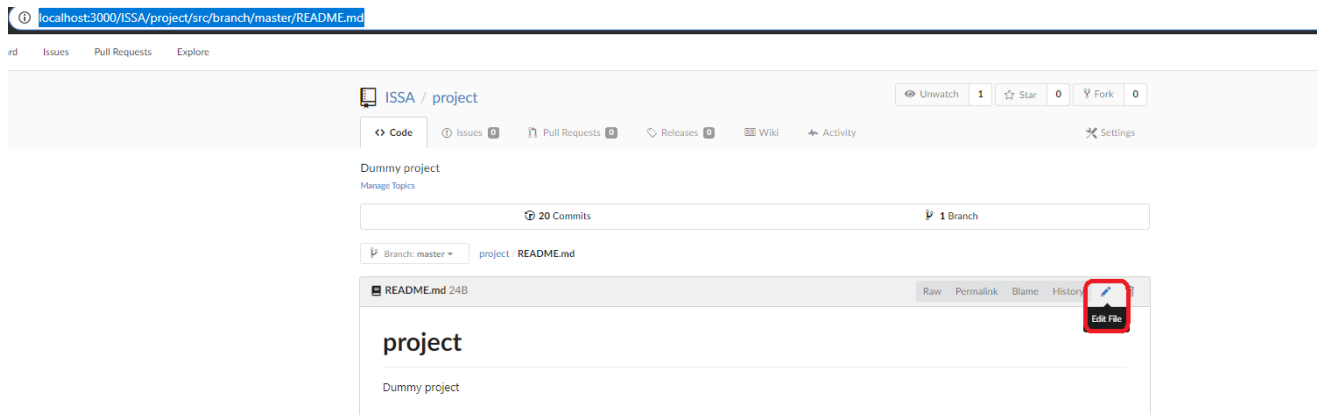
- › From your repository page <http://localhost:3000/ISSA/project>. Via the New File or Upload File



- › The New File will prompt your to a page where you can specify the file name (or relative path) and edit it on the spot.
- › The Upload File will prompt your to a page where you can drag and drop files to be committed

How to edit a committed file

- › From your repository page <http://localhost:3000/ISSA/project>. Navigate to the desired file to be edited. README.md for example <http://localhost:3000/ISSA/project/src/branch/master/README.md> and click on the edit button.



Initial Jenkinsfile

- › Commit a new file called **Jenkinsfile**
- › It should have only one line
 - › **sleep 10**
- › After that go to <http://localhost:8080/job/cd-issa-gitea/>, the Gitea Jenkins Project that you created at the 3rd Step.
- › From the menu of the left, click on **Scan Gitea Organization Now**.
- › Now the ISSA/project repository should have it's own **project** folder with it's own Jenkins job for the **master** branch that only sleeps 10 seconds at each new change / commit.

Add your project source code

- › Go to <https://github.com/whymex/issa-cd-fii> and port the following files, keeping the directory structure in your Gitea repository
 - › **dev-requirements.txt**
 - › **requirements.txt**
 - › **project/__init__.py**
 - › **project/project.py**
- › **Commit each file individually.** 4 Jenkins builds (that still do nothing) should get triggered.

Jenkinsfile code

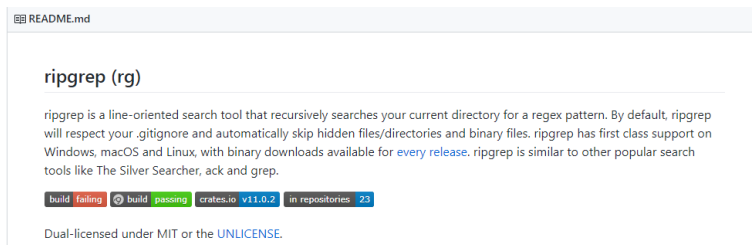
- › Edit the Jenkinsfile from your Gitea repository and copy paste the code from <https://github.com/whymex/issa-cd-fii/blob/master/Jenkinsfile>
- › Observe what's happening in Jenkins and check the Console Output
 - › Hint: In case of error, check system environment variable for python
- › Observe what happened in Gitea after the pipeline execution

Fix the code style and make the pipeline succeed

- › Fix the variables naming from **project/project.py** by replacing x and y with first_number and second_number
- › Commit the changes and analyze the Jenkins pipeline execution

Exercise: Add a build badge to README.md

- › **Points:** 1 / 5
- › The majority of the opensource project have inside their README.md a badge like icon that displays the build / pipeline execution status.
- › Example: <https://github.com/BurntSushi/ripgrep>



- › **Hints:**
 - › Use the following plug-in and check it's documentation how to set it up and add the necessary links to your README.md
<https://github.com/jenkinsci/embeddable-build-status-plugin>
 - › Allow anonymous users read access at Jenkins.