# Overview

Here is the Python web application deployment procedure using Jenkins,

1. Prepare server
   1. Install python

~# python --version

~# pip --version

* 1. Create virtual environment

1. Install jenkins
2. Install necessary plugin
   1. Git
3. Deploy a Python application
   1. Download a source code from git
   2. Provide python deployment code at the “shell command” section

#!/bin/bash

virtualenv -p python3 myenv

source myenv/bin/activate

pip3 install -r requirements.txt

cd reports

touch \*.xml

touch \*.report

cd ..

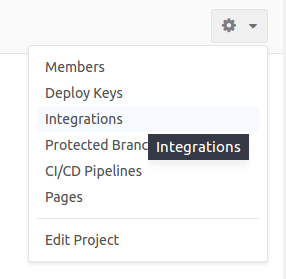
python3 manage.py jenkins – enable-coverage

* 1. Send an email confirmation (if needed)
  2. Add a scheduler/cron job (if needed)

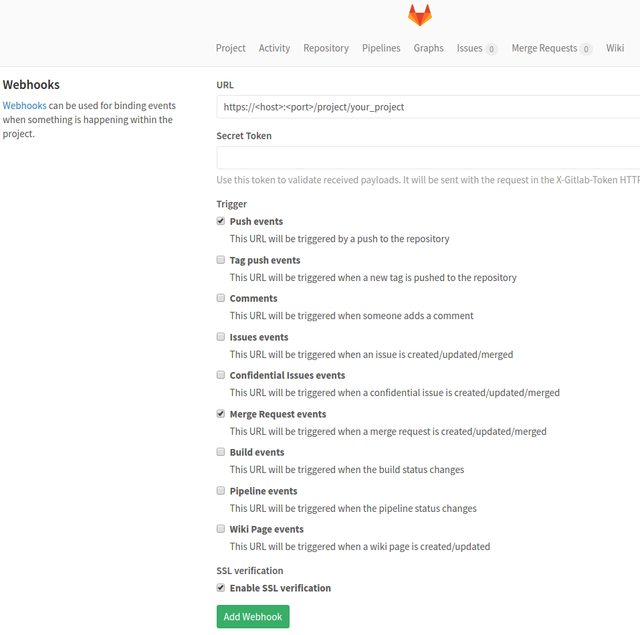
1. Testing
2. Monitoring

# build when a change is pushed to gitlab

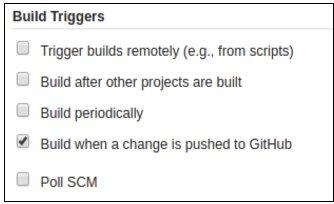
1. Install “Gitlab Plugin for Jenkins”
2. go to your Gitlab repository, go to the "Integrations" section of the repo settings

****

1. Then, set the webhook URL as : https://<jenkins-host>:<port>/project/<your job>



1. This way Gitlab will perform a POST request on your Jenkins Job each time the selected trigger will occur which will trigger your job task if you have configured the specified event to trigger the build.



# Reference

<https://www.stratoscale.com/blog/devops/using-jenkins-build-deploy-python-web-applications-part-2/>

<https://vahiwe.medium.com/build-a-django-ci-cd-pipeline-using-jenkins-e90cbe098970>