



zenika
<animés par la passion>

Gitlab Configuration Details

Version 1.18-SNAPSHOT

2018-08-23

Table of Contents

1. Introduction	2
2. Server configuration	2
3. Project creation	2
3.1. Merge Requests workflow	2
3.2. Pipeline creation	3
4. Appendix	5
4.1. Revision marks	5

Table 1. History

Date	Author	Detail
2018-08-23	bcouetil	Initial commit

1. Introduction

Gitlab in his [free online plan](#) offers a lot more than Github :

- 2,000 CI pipeline minutes per group per month on our shared runners
- Unlimited private projects and collaborators
 - Built-in CI/CD
 - Cycle Analytics
 - Issue Boards
 - Time tracking
 - Preview your changes with Review Apps
 - Publish static websites for free with GitLab Pages
 - Git LFS 2.0 support

2. Server configuration



TODO

3. Project creation

- Create a project in web interface.
 - check **Create README.md** or else **master** branch is not created
- Import

3.1. Merge Requests workflow

We assume that we will be the only one to push on that branch, so we do not check-out the branch, and just push there. This helps do fewer interactions in Eclipse.

Initialize the change

- Go to **Issues** → **New issue**
- Set information and create
- Click **Create merge request**
- In Eclipse, right click on your repo → **Pull**
 - The branch should appear under **Branches** → **Remote Tracking**
- Update the files...
- Commit
- Right click on local/master → Push Branch...

- Toggle
 - ☐ Configure Pull
 - ☐ Force
- Next → Finish

On each new dev iteration

- Commit with amend, ignore the warning
- Right click on local/master → Push Branch...
- Toggle
 - ☐ Configure Pull
 - ☒ Force
- Next → Finish

Merge in Gitlab

Browse the Merge Request in Gitlab

- ☒ Remove source branch

Remove the Remote branch in your git repo → Branches → Remote Tracking

Right click on your repo → **Pull**

- If you get an error, you have obviously configure Pull for the dead branch, so use **Pull...** instead and select **master** branch

3.2. Pipeline creation

```
image: maven:latest

stages:
  - build
  - test
  - deploy

variables:
  #MAVEN_CLI_OPTS: "-s cg-settings.xml --batch-mode"
  MAVEN_CLI_OPTS: "--batch-mode"
  MAVEN_OPTS: "-Dmaven.repo.local=.m2/repository"

cache:
  paths:
    - .m2/repository/

build:
  stage: build
  script:
    - apt-get update -qq && apt-get install -qq --assume-yes libc6-i386
    - mvn clean dependency:purge-local-repository
    - mvn install -DskipTests -Dassembly.skipAssembly=true

# the pipeline step must be named 'pages' for gitlab to deploy locally (in addition to github)
pages:
  stage: build
  script:
    #- ./src/scripts/asciidocOnlyModified.sh
    - ./src/scripts/asciidocHistory.sh .
    - apt-get update -qq && apt-get install -qq --assume-yes graphviz
    - git config --global user.email "gitlab@noreply.com"
    - git config --global user.name "GitLab"
    - mvn generate-resources -Dadoc.skip=false --non-recursive
    - cp src/docs/asciidoc/*.adoc target/generated-docs/
    - cp -R src/docs/asciidoc/subdocs target/generated-docs/
    - mvn scm-publish:publish-scm
    - mkdir public
    - mv target/generated-docs/* public/
  artifacts:
    paths:
      - public
  #only:
  # - master

unit-test:
  stage: test
  script:
    - mvn clean
    - mvn test -Dcheckstyle.skip=true -pl cg-utils

integration-test:
  stage: test
  script:
    - mvn clean
    - mvn verify failsafe:verify -Dcg.ut.skip=true -Dcheckstyle.skip=true -pl cg-utils

assembly:
  stage: deploy
  script:
    - mvn clean
    - mvn install -DskipTests -Dcheckstyle.skip=true -pl cg-utils
  #only:
  # - master
```

4. Appendix

4.1. Revision marks

Differences since last tag

