

華東理工大學

信息科学与工程学院

《软件工程》 实验报告八

系 别 计算机系

专 业 计算机科学与技术

年 级 2020 级

姓 名 刘子言

指导教师 阮 彤

2022-2023 学年 第 1 学期

实验八 持续集成实验（Jenkins）

一、实验目的

- 1、理解什么是持续集成，理解 Jenkins 的基本用途；
- 2、掌握 Jenkins 的基本用法，掌握基于 Jenkins 自动集成与构建 git 项目。

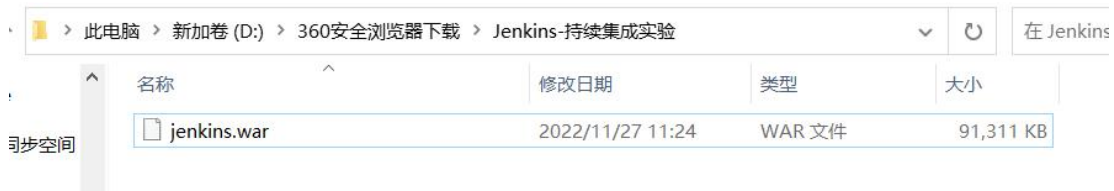
二、实验装置

个人 PC 机（win10），Jenkins, git, jdk-17.0.5。

三、实验内容

1、配置 Jenkins 及环境

（1）下载 Jenkins.war



（2）在下载目录终端运行指令

```
C:\ 管理员: C:\Windows\System32\cmd.exe - java -jar jenkins.war --httpPort=8090

Microsoft Windows [版本 10.0.19044.2251]
(c) Microsoft Corporation。保留所有权利。

D:\360安全浏览器下载\Jenkins-持续集成实验>java -jar jenkins.war --httpPort=8090
Running from: D:\360安全浏览器下载\Jenkins-持续集成实验\jenkins.war
webroot: $user.home/.jenkins
2022-11-27 03:25:51.882+0000 [id=45] INFO jenkins.install.SetupWizard#init:

*****
*****
*****

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

433d78b44c7e4565a484d50df612a834

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

*****
*****
*****

2022-11-27 03:26:13.264+0000 [id=47] INFO jenkins.InitReactorRunner$1#onAttained: Completed initialization
2022-11-27 03:26:13.293+0000 [id=27] INFO hudson.lifecycle.Lifecycle#onReady: Jenkins is fully up and running
2022-11-27 03:26:14.195+0000 [id=61] INFO h.m.DownloadService$Downloadable#load: Obtained the updated data fi
le for hudson.tasks.Maven.MavenInstaller
2022-11-27 03:26:14.196+0000 [id=61] INFO hudson.util.Retrier#start: Performed the action check updates serve
r successfully at the attempt #1
2022-11-27 03:26:14.203+0000 [id=61] INFO hudson.model.AsyncPeriodicWork#lambda$doRun$1: Finished Download me
tadata. 22,848 ms
```

如上图，Jenkins 启动成功。

(3) 打开 <http://localhost:8090>，按照浏览器链接内容初始化 Jenkins

入门

解锁 Jenkins

为了确保管理员安全地安装 Jenkins，密码已写入到日志中（[不知道在哪里?](#)）该文件在服务器上：

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

请从本地复制密码并粘贴到下面。

管理员密码

新手入门

自定义 Jenkins

插件通过附加特性来扩展 Jenkins 以满足不同的需求。

安装推荐的插件

安装 Jenkins 社区推荐的插件。

选择插件来安装

选择并安装最适合的插件。

新手入门

新手入门

<input type="radio"/> Folders	<input type="radio"/> OWASP Markup Formatter	<input type="radio"/> Build Timeout	<input type="radio"/> Credentials Binding	<div><div>++ Git Client ** Pipeline: Input Step ** Pipeline: Declarative Pipeline ** Java JSON Web Token (JJWT) ** OkHttp ** Github API Git ** Github Github Branch Source Pipeline: Github Groovy Libraries ** Pipeline Graph Analysis ** Pipeline: REST API ** JavaScript GUI Lib: Moment.js bundle Pipeline: Stage View Git SSH Build Agents Matrix Authorization Strategy PAM Authentication LDAP Email Extension Mailer ** Localization Support Localization: Chinese (Simplified) ** - 需要依赖</div></div>
<input type="radio"/> Timestamper	<input type="radio"/> Workspace Cleanup	<input type="radio"/> Ant	<input type="radio"/> Gradle	
<input type="radio"/> Pipeline	<input type="radio"/> Github Branch Source	<input type="radio"/> Pipeline: Github Groovy Libraries	<input type="radio"/> Pipeline: Stage View	
<input type="radio"/> Git	<input type="radio"/> SSH Build Agents	<input type="radio"/> Matrix Authorization Strategy	<input type="radio"/> PAM Authentication	
<input type="radio"/> LDAP	<input type="radio"/> Email Extension	<input type="radio"/> Mailer	<input type="radio"/> Localization: Chinese (Simplified)	

创建第一个管理员用户

用户名:

密码:

确认密码:

全名:

电子邮件地址:

Jenkins 2.361.4

[使用admin账户继续](#)

[保存并完成](#)

实例配置

Jenkins URL:

Jenkins URL 用于给各种Jenkins资源提供绝对路径链接的根地址。这意味着对于很多Jenkins特色是需要正确设置的,例如: 邮件通知、PR状态更新以及提供给构建步骤的BUILD_URL环境变量。

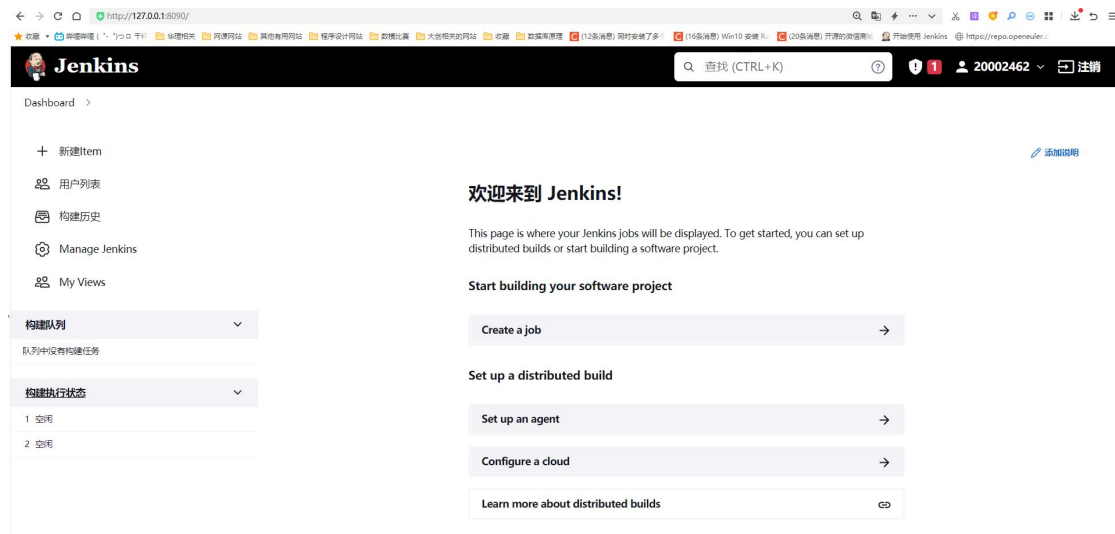
推荐的默认值显示在尚未保存, 如果可能的话这是根据当前请求生成的。最佳实践是要设置这个值, 用户可能会需要用到。这将会避免在分享或者查看链接时的困惑。

Jenkins已就绪!

Jenkins安装已完成。

[开始使用Jenkins](#)

(4) 点击开始使用 Jenkins



(5) 配置 JDK 和 GIT 的版本及路径:

JDK

JDK 安装

系统下JDK 安装列表

新增 JDK

≡ JDK

别名

jdk17

JAVA_HOME

C:\Program Files\Java\jdk-17.0.5

☐ Install automatically ?

新增 JDK

Git

Git installations

Git

Name

Default

Path to Git executable ?

D:\360安全浏览器下载\GIT\Git\bin\git.exe

☐ Install automatically ?

Add Git

2、登录 Gitee

(1) 打开 Gitee 官网，登录自己的账户



(2) 打开之前上传到仓库中的项目

这里可以使用之前在实验四（基于 Git 的版本管理）中，已经上传到 Gitee 上的项目“test4_project”来进行实验。



3、基于 Jenkins 自动集成与构建 Git 项目


(1) 使用 Jenkins 新建一个 job


取名“project20002462”：


输入一个任务名称

project20002462

» 必填项

**Freestyle project**
This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any build system, and this can be even used for something other than software build.

**构建一个maven项目**
构建一个maven项目.Jenkins利用你的POM文件,这样可以大大减轻构建配置.

**Pipeline**
Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.

（2）配置 Git 项目的 URL

项目的 URL 设置为 Gitee 项目 “test4_project” 的 URL

Dashboard > project20002462 >

Configuration

- General
- 源码管理
- 构建触发器
- 构建环境
- Build Steps
- 构建后操作

General

Enabled 

描述

[Plain text] [预览](#)

☐ Discard old builds ?

☒ GitHub 项目

项目 URL ?

https://gitee.com/xiaoyu-conghua/test4_project.git

[高级...](#)

Gitee 链接 ?

student20002462

（3）配置 Git 项目的源码管理

源码管理

☐ 无

☒ Git ?

Repositories ?

Repository URL ?

https://gitee.com/xiaoyu-conghua/test4_project.git

Credentials ?

1846206177@qq.com/***** (student20002462)

[+ 添加](#)

[高级...](#)

[Add Repository](#)

(4) 构建任务触发器

设置为每两分钟构建一次。

构建触发器

- ☐ 触发远程构建 (例如,使用脚本) ?
- ☐ Build after other projects are built ?
- ☒ Build periodically ?
- 日程表 ?
- H/2 * * * *
- Would last have run at 2022年11月27日星期日 中国标准时间 下午3:09:31; would next run at 2022年11月27日星期日 中国标准时间 下午3:09:31.
- ☐ Gitee webhook 触发构建, 需要在 Gitee webhook 中填写 URL: <http://127.0.0.1:8090/gitee-project/project20002462>
- ☐ GitHub hook trigger for GITScm polling ?
- ☐ Poll SCM ?

(5) 查看任务 project20002462 的状态

包括所有的构建情况及其记录。

**Jenkins**

Dashboard > project20002462 >

↑ 返回面板

📄 状态

</> 修改记录

📁 工作空间

▶ Build Now

⚙️ 配置

🗑️ 删除 Project

🔄 GitHub

✎ 重命名

Project project20002462

相关链接

- [Last build\(#1\), 1 分 42 秒之前](#)
- [Last stable build\(#1\), 1 分 42 秒之前](#)
- [Last successful build\(#1\), 1 分 42 秒之前](#)
- [Last completed build\(#1\), 1 分 42 秒之前](#)

 **Build History** 构建历史 ▼

✅ #2

2022年11月27日 下午3:17

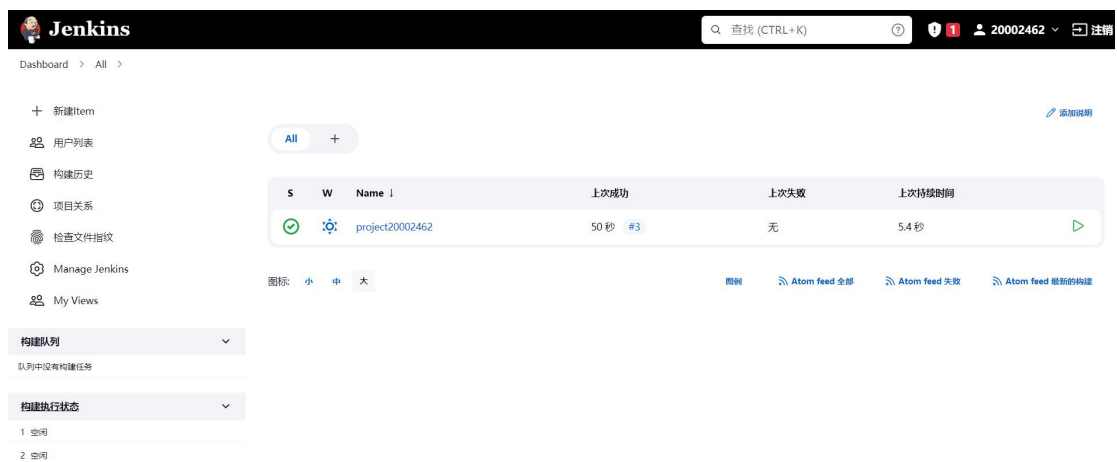
✅ #1

2022年11月27日 下午3:15

📡 Atom feed 全部

📡 Atom feed 失败

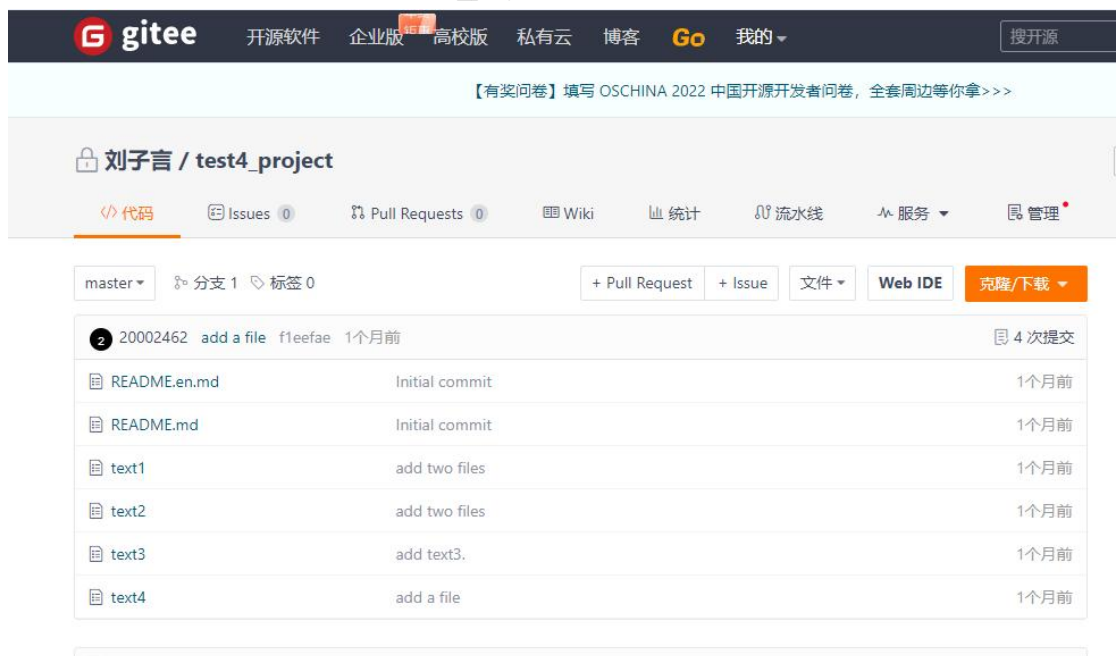
(6) 查看所有的 job 信息



The screenshot shows the Jenkins Dashboard. On the left is a sidebar with navigation links: 新建Item, 用户列表, 构建历史, 项目关系, 检查文件指纹, Manage Jenkins, and My Views. The main area displays a table of jobs. The table has columns: S, W, Name, 上次成功, 上次失败, and 上次持续时间. One job is listed: project20002462, which is successful (green checkmark), took 50 seconds, and has no failures. Below the table are buttons for '图标', '小', '中', '大', and '图例'. On the right, there are links for 'Atom feed 全部', 'Atom feed 失败', and 'Atom feed 最新的构建'.

S	W	Name	上次成功	上次失败	上次持续时间
✓	⚙	project20002462	50 秒 #3	无	5.4 秒

(7) 打开 Gitee, 查看已有项目 test4_project



The screenshot shows the Gitee project page for 刘子言 / test4_project. The top navigation bar includes links for 开源软件, 企业版, 高校版, 私有云, 博客, Go, and 我的. Below the navigation bar is a banner for a survey. The project page shows the repository name, tabs for 代码, Issues, Pull Requests, Wiki, 统计, 流水线, 服务, and 管理. The commit history is displayed with columns for commit message, author, and time. The commits listed are: 20002462 (add a file, f1eefae, 1个月前), README.en.md (Initial commit, 1个月前), README.md (Initial commit, 1个月前), text1 (add two files, 1个月前), text2 (add two files, 1个月前), text3 (add text3, 1个月前), and text4 (add a file, 1个月前).

(8) 修改 test4_project 项目 text1 文件中的代码 在 Git 上 commit 代码, 显示更新信息 update text1。



The screenshot shows the Gitee project page for 刘子言 / test4_project. The top navigation bar includes links for 开源软件, 企业版, 高校版, 私有云, 博客, Go, and 我的. Below the navigation bar is a banner for a survey. The project page shows the repository name, tabs for 代码, Issues, Pull Requests, Wiki, 统计, 流水线, 服务, and 管理. The commit history is displayed with columns for commit message, author, and time. The commits listed are: 20002462 (add a file, f1eefae, 1个月前), README.en.md (Initial commit, 1个月前), README.md (Initial commit, 1个月前), text1 (add two files, 1个月前), text2 (add two files, 1个月前), text3 (add text3, 1个月前), and text4 (add a file, 1个月前).

(9) 返回 Jenkins，查看新的 build#8 的相关记录

**Jenkins**

Dashboard > All > project20002462 >

↑ 返回面板

📄 状态

</> 修改记录

📁 工作空间

▶ Build Now

⚙️ 配置

🗑️ 删除 Project


🐙 GitHub

✏️ 重命名

修改记录

#8 (2022年11月27日 下午3:28:05)

1. update text1. (commit: eb815b4) — [noreply](#) / [detail](#)

**Jenkins**

Dashboard > All > project20002462 > #8

↑ 返回到工程

📄 状态集

</> 变更记录

🖨️ 控制台输出

📄 编辑编译信息

🗑️ 删除构建 '#8'

🔍 Git Build Data

← 上一次构建

→ 下一次构建

变更记录


Summary

1. update text1. (commit: eb815b4) ([details](#))

Commit eb815b42240d668e6a33488d4c148ac9e4306f08 by noreply
update text1.

Signed-off-by: 刘子言 <10426608+xiaoyu-conghua@user.noreply.gitee.com>
(commit: eb815b4)

📄 text1

**Jenkins**

Dashboard > All > project20002462 > #8

↑ 返回到工程

📄 状态集

</> 变更记录

🖨️ 控制台输出

📄 文本方式查看

📄 编辑编译信息

🗑️ 删除构建 '#8'


🔍 Git Build Data

← 上一次构建




→ 下一次构建

控制台输出

```
Started by user 20002462
Running as SYSTEM
Building in workspace C:\Users\Administrator\.jenkins\workspace\project20002462
The recommended git tool is: NONE
using credential 000001
> D:\360安全浏览器下载\GIT\git\bin\git.exe rev-parse --resolve-git-dir C:\Users\Administrator\.jenkins\workspace\project20002462\git # timeout=10
Fetching changes from the remote Git repository
> D:\360安全浏览器下载\GIT\git\bin\git.exe config remote.origin.url https://gitee.com/xiaoyu-conghua/test4_project.git # timeout=10
Fetching upstream changes from https://gitee.com/xiaoyu-conghua/test4_project.git
> D:\360安全浏览器下载\GIT\git\bin\git.exe --version # timeout=10
> git --version # 'git version 2.38.1.windows.1'
using GIT_ASKPASS to set credentials student20002462
> D:\360安全浏览器下载\GIT\git\bin\git.exe fetch --tags --force --progress -- https://gitee.com/xiaoyu-conghua/test4_project.git
refs/heads/*:refs/remotes/origin/* # timeout=10
> D:\360安全浏览器下载\GIT\git\bin\git.exe rev-parse "refs/remotes/origin/master^{commit}" # timeout=10
Checking out Revision eb815b42240d668e6a33488d4c148ac9e4306f08 (refs/remotes/origin/master)
> D:\360安全浏览器下载\GIT\git\bin\git.exe config core.sparsecheckout # timeout=10
> D:\360安全浏览器下载\GIT\git\bin\git.exe checkout -f eb815b42240d668e6a33488d4c148ac9e4306f08 # timeout=10
Commit message: "update text1."
> D:\360安全浏览器下载\GIT\git\bin\git.exe rev-list --no-walk f1ee4ae05e7033eb21e40642a89a990f5f57c9 # timeout=10
Finished: SUCCESS
```


 Jenkins


Q 查找 (CTRL+K)


   20002462


Dashboard > All > project20002462 > #8 > Git Build Data


↑ 返回到工程


 状态集

 变更记录

 控制台输出

 编辑编译信息

 删除构建 '#8'

 **Git Build Data**

← 上一次构建

→ 下一次构建

Git Build Data

Revision: eb815b42240d668e6a33488d4c148ac9e4306f08
Repository: https://gitee.com/xiaoyu-conghua/test4_project.git

- refs/remotes/origin/master

Built Branches

- refs/remotes/origin/master: Build #8 of Revision eb815b42240d668e6a33488d4c148ac9e4306f08 (refs/remotes/origin/master)

四、实验心得

通过本次实验，我了解了基于 Jenkins 持续集成的基本知识，掌握了 Jenkins 的配置方式及其基本用法。同时还利用自己在 Gitee 上的既有项目，实现了基于 Jenkins 自动集成与构建 Git 项目，收获颇丰。

Jenkins 作为一个基于 Java 开发的持续集成工具，对于监控持续重复的工作、持续集成软件项目起到了十分重要的作用，为项目的集成、构建带来了便利。这次实验的实践操作，也为我在之后的学习与项目开发中使用该工具打下了坚实的基础。