

湖南省零檬信息技术有限公司

创新教育·极致服务

CI/CD

产品设计-> 开发人员开发代码 -> 测试人员测试功能 -> 运维人员发布上线

- 持续集成 (Continuous integration, 简称CI)

开发人员提交了新代码之后, 立刻自动化进行构建、执行单元测试。目的是为了快速发现集成错误。

- 持续交付 (Continuous Delivery)

在持续集成的基础上, 自动将集成后的代码部署到更贴近真实运行环境中。

- 持续部署 (Continuous Deployment)

在持续交付的基础上, 自动将应用部署到生产环境中。

Jenkins

基于Java开发的持续集成工具, 开源免费, 官网下载: <https://jenkins.io/download/>

- 持续、自动地构建/测试软件项目。
- 更好地实现整个CI/CD流程。

Jenkins环境搭建

下载所需的软件包

1. Tomcat <https://archive.apache.org/dist/tomcat/tomcat-8/>
2. Jenkins <https://www.jenkins.io/download/>
3. Allure <https://github.com/allure-framework/allure2/releases>

部署步骤

- 下载war包: jenkins.war
- 部署到tomcat (jenkins.war包放到tomcat的webapps目录)
- 启动Tomcat (进入到tomcat的bin目录下, 找到startup.bat脚本执行启动即可)
- 验证:
 - <http://localhost:8080>验证Tomcat
 - <http://localhost:8080/jenkins>验证Jenkins

密码问题

获取密码

在第一次使用jenkins的时候默认会生成一个密码文件
%JENKINS_HOME%\secrets\initialAdminPassword
打开文件，复制其中的初始登录的密码，粘贴然后登录

修改密码

系统管理--》管理用户

启动后Jenkins提示离线

- 打开jenkins->系统管理->管理插件->高级
- 或打开链接：
<http://ip:端口/jenkins/pluginManager/advanced>
- 将升级站点
<http://updates.jenkins-ci.org/update-center.json>
换成<http://mirror.esuni.jp/jenkins/updates/update-center.json>
- 重启Tomcat

Jenkins全局设置

系统管理->全局工具配置

jdk 配置

如果机器上已经装了jdk，直接指定JAVA_HOME目录即可
jenkins版本较高，建议使用1.8版本的jdk。

maven配置

如果机器上有下载maven，指定一个MAVEN_HOME目录即可。

Jenkins创建任务

新建一个maven任务（如果没有这个选项，你需要在Jenkins里面安装Maven插件）

在管理插件中下载maven插件：maven intergration

新建->构建一个maven项目

源码管理

subversion -> 填写项目repository url+credentials验证

Build

- root pom指定
- goals and options（需要执行的maven命令：clean test）

构建触发器

构建触发器是Jenkins启动一个Job构建的条件，当条件满足时，Job启动。可以设置为测试人员commit后自动编译（比如5分钟update一次svn，如果有提交则执行构建），或定时构建（比如每天晚上0点定时构建）。

build periodically

隔一段时间build一次，不管版本库代码是否发生变化。

poll scm

隔一段时间比较一次源代码如果发生变更，那么就build。否则，不进行build。

时间表达式

时间表达式是由五部分组成，每一部分中间由空格隔开，如下：

MINUTE HOUR DOM MONTH DOW

1. 每天的晚上八点钟：

0 20 * * *

2. 每月的第10号晚上零点：

0 0 10 * *

3. 每天晚上的八点10分到15分

10-15 20 * * *

4. 每间隔10分钟

* /10 * * * *

5. 每间隔1小时

* * /1 * * *

6. 前三十分钟内每间隔5分钟

0-30/5 * * * *

7. 每天的20, 21, 23点执行

0 20,21,23 * * *

- MINUTE 分钟值 (0-59)
- HOUR 时钟值(0-23)
- DOM 日期 (1-31)
- MONTH 月份 (1-12)
- DOW 一周中的某一天(0-7) 0和7都代表周天

每一部分还可以设置为以下的值

- *: 指所有的值
- M-N: 指一个范围区间内的值
- M-N/X 或 */X: 指（在一个范围内/所有）满足间隔X的值
- A,B,...,Z: 列举了这一部分所有的值

Jenkins配置邮件

系统管理->系统设置:

Jenkins Location

Jenkins URL

`http://127.0.0.1:8080/jenkins/`

系统管理员邮件地址

`1425301899@qq.com`

设置Extended E-mail Notification:

进入“系统管理”-“系统设置”，对Extended E-mail Notification进行配置，具体参考下图：

Extended E-mail Notification

SMTP server

`smtp.qq.com`

邮件服务器，此处是qq邮箱

Default user E-mail suffix

`@qq.com`

邮箱后缀

☒ Use SMTP Authentication

User Name

`j81923`

发件人用户名

Password

.....

这个密码不是qq密码，而是在QQ邮箱申请
开通IMAP/SMTP服务后收到的密码

Use SSL

☒

SMTP port

`465`

QQ邮箱服务器端口号

Charset

`UTF-8`

Default Content Type

`HTML (text/html)`

邮件内容格式

☐ Use List-ID Email Header☐ Add 'Precedence: bulk' Email Header

Default Recipients

`40411@qq.com`

如何获取smtp服务密码？（QQ邮箱举例）

- 进入QQ邮箱主页，点击设置
找到“账户”页签
- 在开启服务下面，找到IMAP/SMTP服务对应的“开启”按钮点击
- 申请密保
按照弹窗里的提示给腾讯发送短信，你就会收到一个授权码，这个就是密码



设置“邮件通知”:

还是在系统管理-系统设置里面设置，找到“邮件通知”，内容和Extended E-mail Notification差不多，具体如下图：

邮件通知	
SMTP服务器	smtp.qq.com 此处填的QQ邮箱服务器
用户默认邮件后缀	@qq.com
<input checked="" type="checkbox"/> 使用SMTP认证	
用户名	81923
密码	*****
<input checked="" type="checkbox"/> 使用SSL协议	
SMTP端口	465
Reply-To Address	
字符集	UTF-8

测试:

输入默认接收人邮件，点击测试报错

解决方法:系统管理员邮件地址设置与发件人相同

Jenkins Location

<http://localhost:8080/jenkins/>

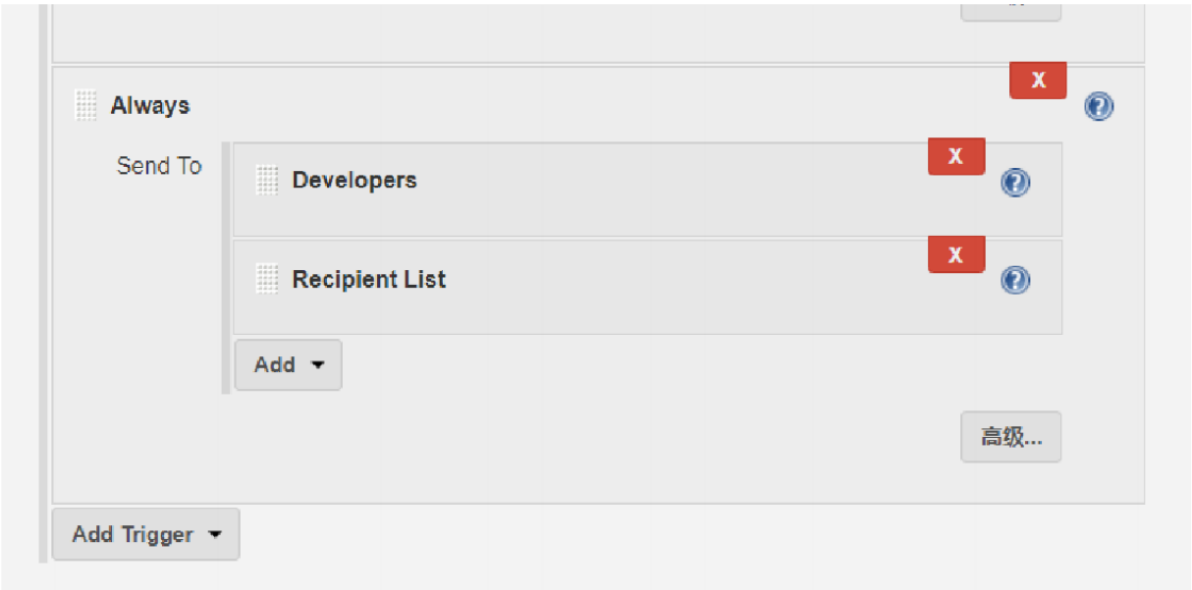
⚠ Please set a valid host name, instead of localhost

@qq.com

Editable Email Notification	
Disable Extended Email Publisher	<input type="checkbox"/> Allows the user to disable the publisher, while maintaining the settings
Project Recipient List	<div>4041103@qq.com 3041103@qq.com</div> <div>邮件接收人列表，多个接收人换行写</div>
Project Reply-To List	<div>\$DEFAULT_REPLYTO</div> <div>Comma-separated list of email address that should receive notifications for this project.</div>
Content Type	<div>HTML (text/html)</div> <div>邮件内容选择 "html"</div>
Default Subject	<div>自动化测试邮件</div> <div>邮件主题</div>
Default Content	<div><!DOCTYPE html> <html></div> <div>具体的邮件内容</div>

```
<hr>(请勿回复!)<br><hr>
项目名称: $PROJECT_NAME<br><hr>
构建编号: $BUILD_NUMBER<br><hr>
构建状态: $BUILD_STATUS<br><hr>
触发原因: ${CAUSE}<br><hr>
测试报告: <a href="$${BUILD_URL}allure/">${BUILD_URL}allure/</a>
<br><hr>
构建日志地址: <a href="$${BUILD_URL}console">${BUILD_URL}console/</a><br><hr>
```

触发条件，配置如下：



Jenkins Allure报表

安装Allure插件



配置全局工具

Allure Commandline

Allure Commandline 安装

新增 Allure Commandline

Allure Commandline

别名

allure2

安装目录

D:\Programs\allure-2.12.1

☐ Install automatically

删除 Allure Commandline

新增 Allure Commandline

系统下Allure Commandline 安装列表

保存

应用

添加构建后操作

配置job任务，构建后操作里面（add post-build action）

选择allure report

设置下allure-results目录的路径（相对路径，基于项目）

构建后操作

Allure Report

Disabled ☐

Results:

Path

target/allure-results

Aggregate downstream test results

Allure Report

Archive the artifacts

Build other projects

Deploy artifacts to Maven repository

Record fingerprints of files to track usage

Git Publisher

Editable Email Notification

Delete workspace when build is done

增加构建后操作步骤

报表展示

在任务页面的左边菜单里会出现allure report选项，点击即可访问到allure报告

将此url地址设置到邮件内容中发送给相关人

方式一：环境变量配置

系统管理->系统设置

Name: JAVA_TOOL_OPTIONS

Value: -Dfile.encoding=UTF-8

方式二：Jenkins全局设置

系统管理->系统设置

添加“全局属性”

设置key: LANG value: zh_CN.UTF-8