

ALIBAY使用指南（初稿）

第一步：申请内测资格，发送邮件到alibay@service.aliyun.com或者chengrong.zcr@alibaba-inc.com



插件化 需求管理
持续发布线 git 持续集成
Docker 图形化 代码托管
持续交付

Alibay

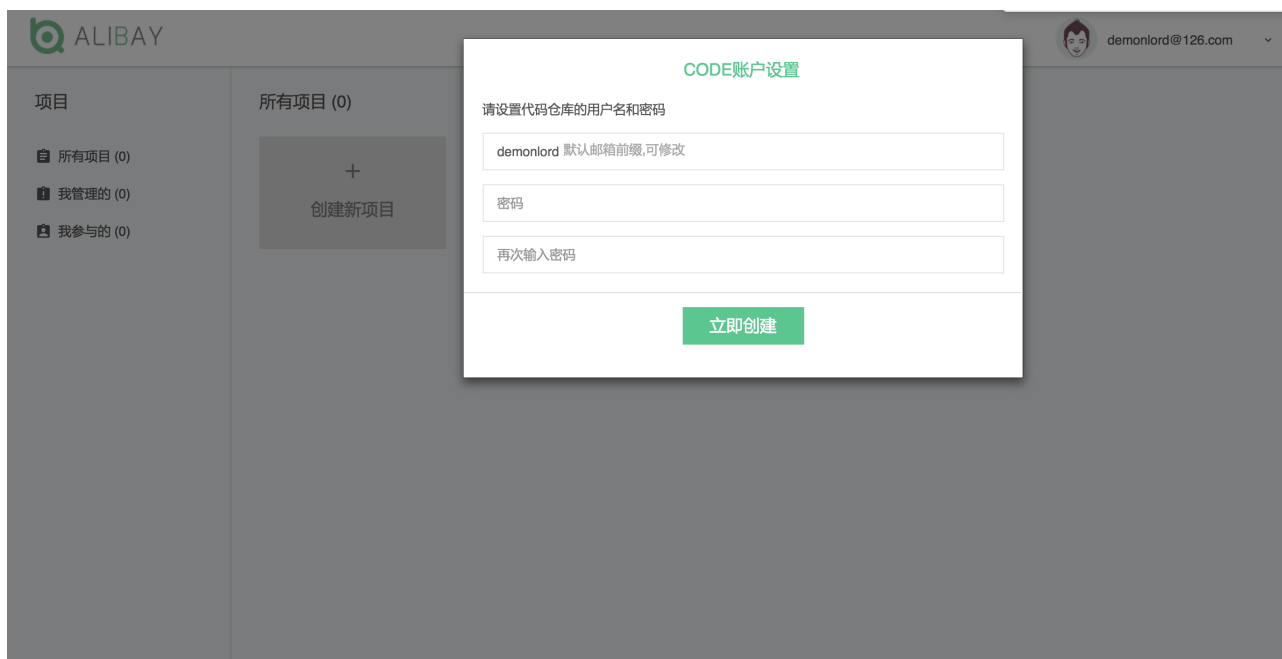
where your dreams come true.
实现梦想的港湾。

Alibay正在内测中...

请按照 [申请Alibay试用 + 公司名称 + 联系方式] 为主题发送邮件至

chengrong.zcr@alibaba-inc.com申请试用。

第二步：登陆之后填写代码仓库的用户名和密码，填写之后点击立即创建。



ALIBAY

项目

- 所有项目 (0)
- 我管理的 (0)
- 我参与的 (0)

所有项目 (0)

+ 创建新项目

CODE账户设置

请设置代码仓库的用户名和密码

demonlord 默认邮箱前缀,可修改

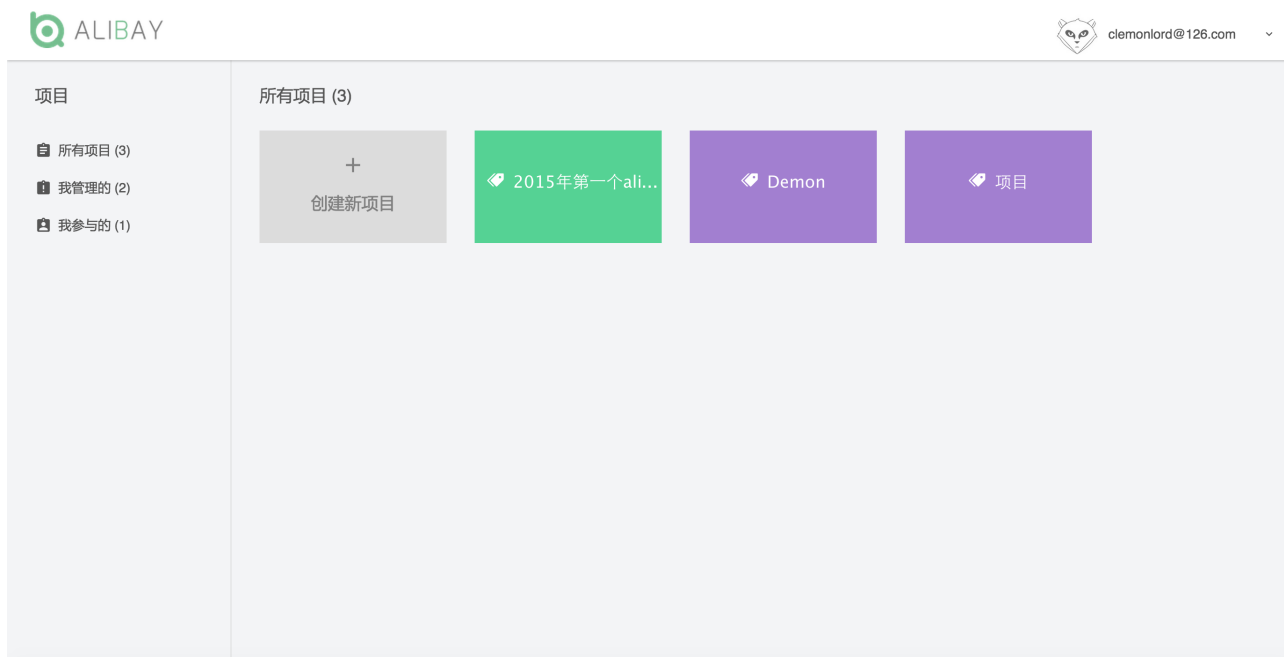
密码

再次输入密码

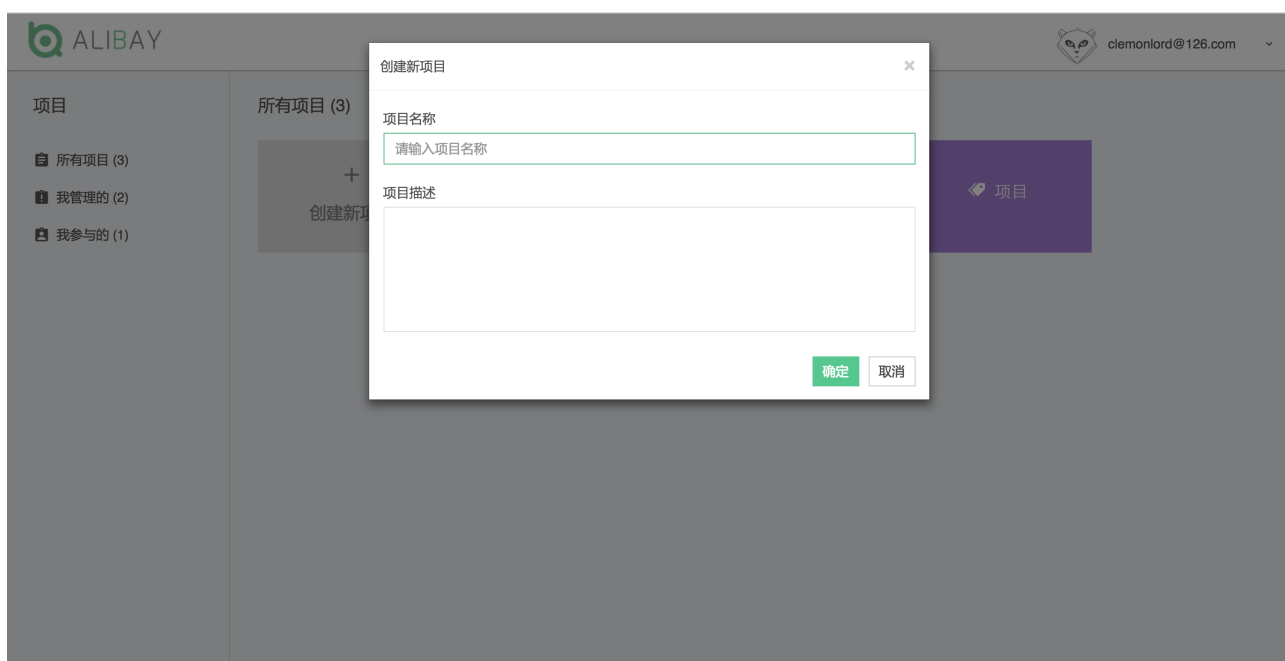
立即创建

demonlord@126.com

第三步：点击创建新项目，新建一个自己的项目。




输入项目名称和描述即可创建出你自己的项目，填写后点击确定。



第四步：创建项目后就进入了项目首页，如果需要回到项目列表，可以通过点击ALIBAY图标返回。



第五步：创建一个代码库，点击红色框中的图标或者右边栏中的  图标。



代码仓库中输入你所需要的代码库名字，只支持英文。

🕒

设置

↶↷

项目设置

👤

成员管理

</>

仓库管理

⚙️

部署公钥

代码仓库

创建或关联代码库

添加

代码仓库列表

当前没有已经关联的代码库，请创建或关联代码库

这里我们创建一个开源的Blog项目

代码仓库

创建或关联代码库

添加

代码仓库列表

当前没有已经关联的代码库，请创建或关联代码库

创建成功就可以看到代码下载地址

代码仓库

创建或关联代码库

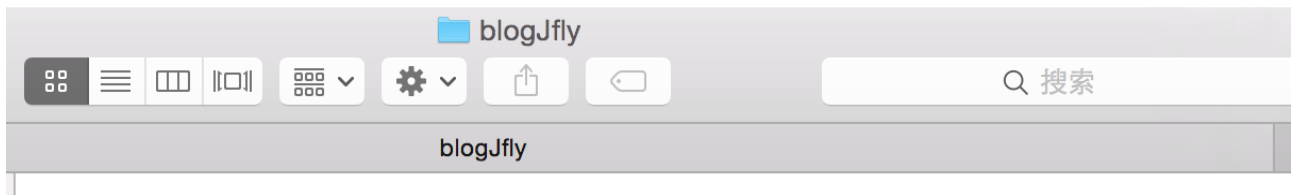
添加

代码仓库列表

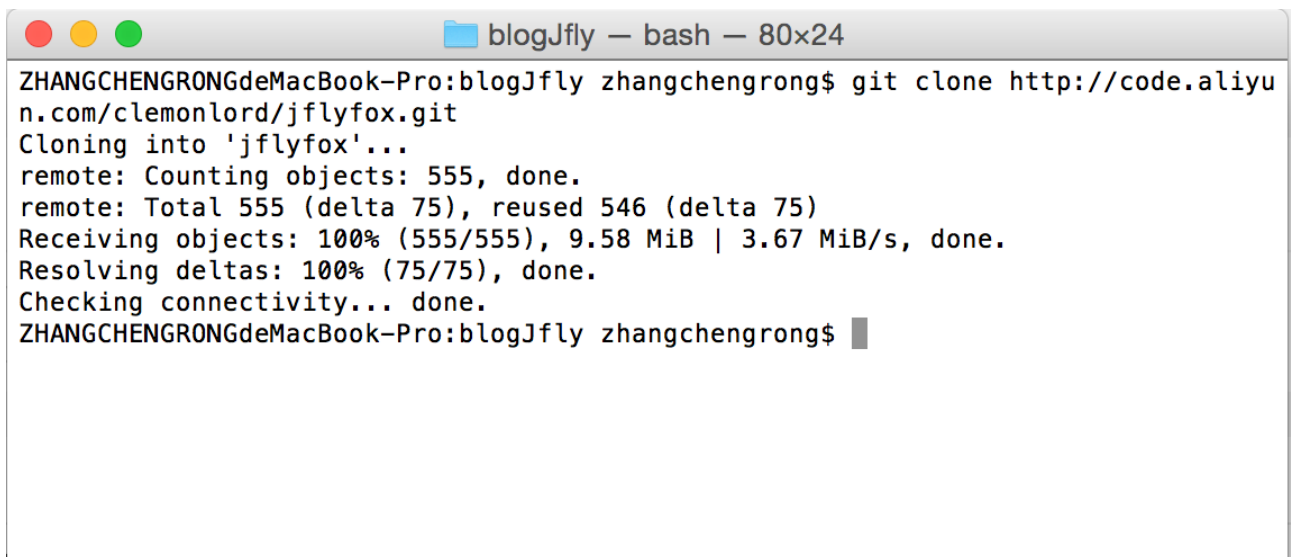
#	代码仓库	代码下载地址	操作
1	clemonlord/jflyfox	http://code.aliyun.com/clemonlord/jflyfox.git	<div>取消关联</div>

第六步：创建本地git仓库

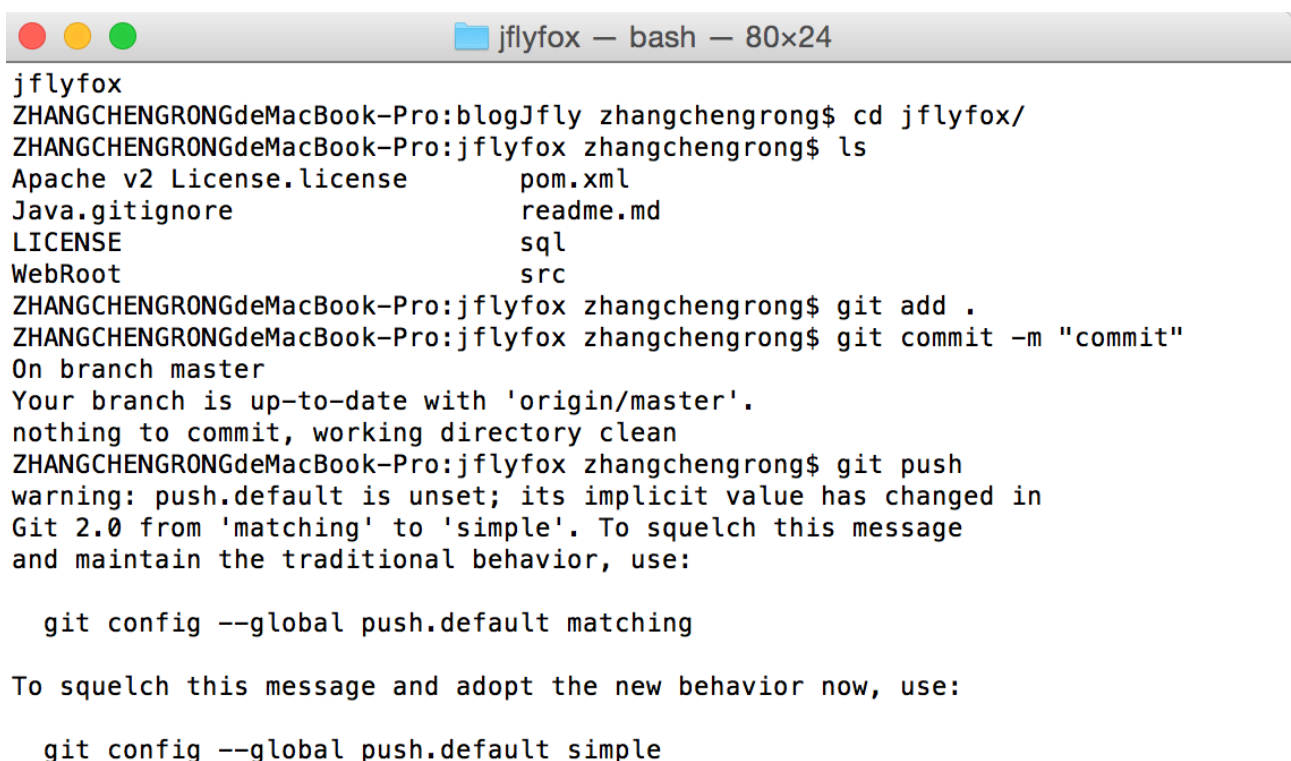
创建一个本地目录



打开命令终端进入该目录，执行git clone 代码下载地址，把刚才创建的仓库下载到本地。



第七步：提交代码到仓库中



第八步：开始配置持续发布线，如何配置火车模型发布线

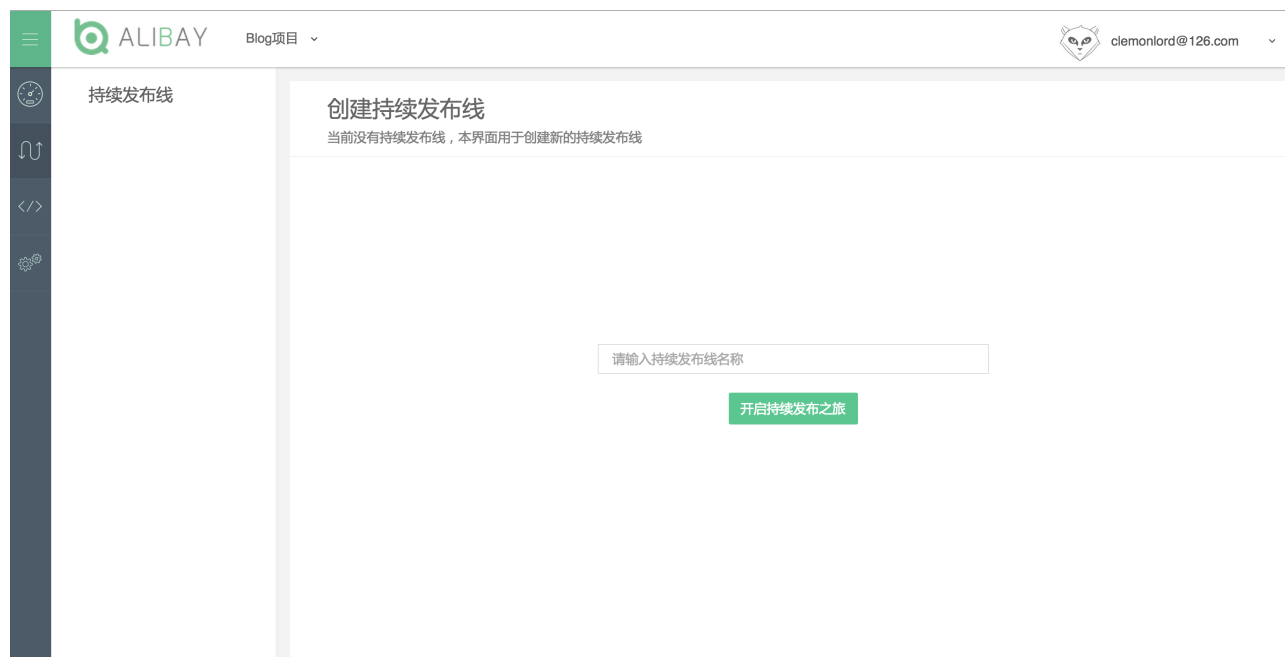
回到项目首页点击发布情况边上的按钮，或者点击右边栏



中图标，进入 workflow 创建页面。



进入持续发布线配置页面，输入持续发布线的名称，创建第一条持续发布线



进入持续发布线配置页面，后续需要增加发布线则点击持续发布线后面的+号按钮。

ALIBAY

Blog项目

clemonlord@126.com

持续发布线

blog发布

blog发布

在配置界面,点击或者拖拽不同区域可以对 workflow 配置

0 集成发布

工作流信息

名称

blog发布

详细信息

在配置界面,点击或者拖拽不同区域可

生效

点击红框中的起始节点

0 集成发布

可以对起始节点进行配置，所需要的就是通过下拉菜单选择代码仓库和侦听分支。

起始节点	触发器任务
请在右侧配置触发器	<div>集成发布</div> <div>选择插件Ali Git代码插件</div> <div>代码仓库e.g git@gitlab.company.com:project/project1.git</div> <div>侦听分支e.g master</div>

这里我们选择的是master分支

触发器任务	
集成发布	
选择插件	Ali Git代码插件
代码仓库	http://code.aliyun.com/clemonlord/jflyfox.git
侦听分支	master

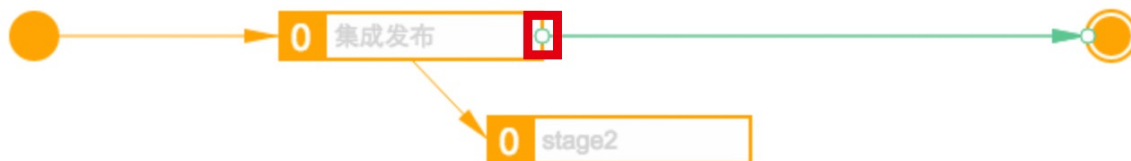
选中集成发布框，则框变成绿色，这时候可以按住红框中的绿色空心圆拖出新的节点。



变成如下图的图形



选中集成发布后面的连接线，按住红框中的绿色空心圆，移动到stage2上。



这时候图形变为如下情况



我们继续点击集成发布框，进行配置。

- 1.先修改活动名称为代码签入
- 2.点击任务列表，选择代码更新、编译、测试
- 3.勾选自动触发，自动完成

活动信息	任务列表 +
<p>活动名称 <input type="text" value="代码签入"/></p> <p><input type="checkbox"/> 自动触发</p> <p><input type="checkbox"/> 自动完成</p> <p><input type="checkbox"/> 异常通知</p>	<div>代码更新</div> <div>编译</div> <div>测试</div> <div>Tag</div> <div>部署</div>

代码更新不需要配置，检查下即可

活动信息	任务列表 +
<p>活动名称 <input type="text" value="代码签入"/></p> <p><input checked="" type="checkbox"/> 自动触发 <input type="text" value="立即执行"/></p> <p><input checked="" type="checkbox"/> 自动完成</p> <p><input type="checkbox"/> 异常通知</p>	<div>代码更新</div> <div>编译</div> <div>测试</div> <div>选择插件 <input type="text" value="Ali Git代码插件"/></div> <div>代码仓库 <input type="text" value="http://code.aliyun.com/clemonlord/flyfox.git"/></div> <div>侦听分支 <input type="text" value="master"/></div>

编译配置，选择maven版本，选择jdk版本，输入编译指令和你提交的代码的产出物路径。

活动信息	任务列表 +	
活动名称 <input type="text" value="代码签入"/>	代码更新	选择插件 <input type="text" value="Ali Maven编译插件"/>
<input checked="" type="checkbox"/> 自动触发 <input type="text" value="立即执行"/>	编译	Maven版本 <input type="text" value="Maven3"/>
<input checked="" type="checkbox"/> 自动完成	测试	JDK版本 <input type="text" value="jdk7"/>
<input type="checkbox"/> 异常通知		编译指令 <input type="text" value="mvn clean package"/>
		产出物路径 <input type="text" value="target/flyfox_mtgxxw.war"/>

单元测试配置，选择maven版本，选择jdk版本，输入测试指令。

活动信息	任务列表 +	
活动名称 <input type="text" value="代码签入"/>	代码更新	选择插件 <input type="text" value="Ali JUNIT单元测试插件"/>
<input checked="" type="checkbox"/> 自动触发 <input type="text" value="立即执行"/>	编译	Maven版本 <input type="text" value="Maven3"/>
<input checked="" type="checkbox"/> 自动完成	测试 ×	JDK版本 <input type="text" value="jdk7"/>
<input type="checkbox"/> 异常通知		测试指令 <input type="text" value="mvn test"/>

代码签入活动配置完成后，选中stage2活动节点。图形都是可以任意拖动的，摆成合适的形态。



这里我们修改stage2的活动名称为发布，如果需要请自行添加集成测试节点。同时新增部署任务，部署任务配置信息包含目标机器IP（可以多IP，使用“,”分割，注意是英文字符逗号）；部署路径，也就是指定包部署在您机器上的路径；部署脚本，启动您的应用需要自行配置一个部署脚本，包含停止应用，解压覆盖二进制包，和启动应用三大部分；最后填写登陆用户名。

活动信息	任务列表 +	
活动名称 <input type="text" value="发布"/>	部署	选择插件 <input type="text" value="Ali 部署插件"/>
<input type="checkbox"/> 自动触发		目标机器 <input type="text" value="120.26.40.103"/>
<input type="checkbox"/> 自动完成		部署路径 <input type="text" value="/usr/share/tomcat/webapps"/>
<input type="checkbox"/> 异常通知		部署脚本 <input type="text" value="/usr/share/tomcat/bin/restart.sh"/>
		登录用户 <input type="text" value="root"/> <input type="button" value="机器授权"/>

上图中的机器授权，需要您点击下，可以看到提示信息，点击蓝色的“点此”链接，把公钥复制下来。点击“点此”后并没有特别提示，拷贝的内容可以在本地文本编辑器中黏贴看到。

任务列表 +

部署

选择插件

Ali 部署插件

目标机器

120.26.148.121

部署路径

/usr/share/

部署脚本

/usr/share/

登录用户

root

机器授权

登陆到您的服务器中

Last login: Wed Sep 2 14:33:32 2015 from 42.120.74.94

Welcome to aliyun Elastic Compute Service!

```
[root@iZ23dnsn6cpZ ~]# vi .ssh/authorized_keys
[root@iZ23dnsn6cpZ ~]#
```

```
vi $HOME/.ssh/authorized_keys
```

添加刚才复制下来的公钥，并保存。

[illegible]

完成上述工作后回到持续发布线配置页面，点击生效按钮。完成 workflows 生效。



持续发布线就开始监听您的代码库，如果代码库有变化就会产生发布线实例。如果当前没代码提交，可以点击触发按钮，手工触发一条发布线实例。



点击后如下图所示

持续发布线 + #1 - blog发布

blog发布 进行中

在配置界面,点击或者拖拽不同区域可以对工作流配置

22 触发 编辑

控制台输出 历史信息 发布记录

运行编号	开始时间	当前活动节点	状态
#1	2015-09-02 14:45:39	代码签入	进行中

第一个节点运行成功则显示下图

The screenshot shows the Alibaba Cloud DevOps console interface. The top bar includes the ALIBAY logo, 'Blog项目', and a user profile 'clemonlord@126.com'. The left sidebar has icons for '持续发布线' (Continuous Release Line), 'blog发布' (Blog Release), and other tools. The main area displays a workflow configuration for '#1 - blog发布'. The workflow consists of two steps: '代码签名' (Code Signing) and '发布' (Release). The '代码签名' step is highlighted in green and shows a successful execution with the timestamp '2015-09-02 14:45:39' and duration '37.28秒'. The '发布' step is shown in grey. Below the workflow, there are tabs for '控制台输出' (Console Output), '历史信息' (History Information), and '发布记录' (Release Record). The '发布记录' tab is active, showing a table with columns: '运行编号' (Run ID), '开始时间' (Start Time), '当前活动节点' (Current Activity Node), and '状态' (Status). The table contains one entry for run '#1' with start time '2015-09-02 14:45:39', current node '发布', and status '待开始' (Waiting to start).

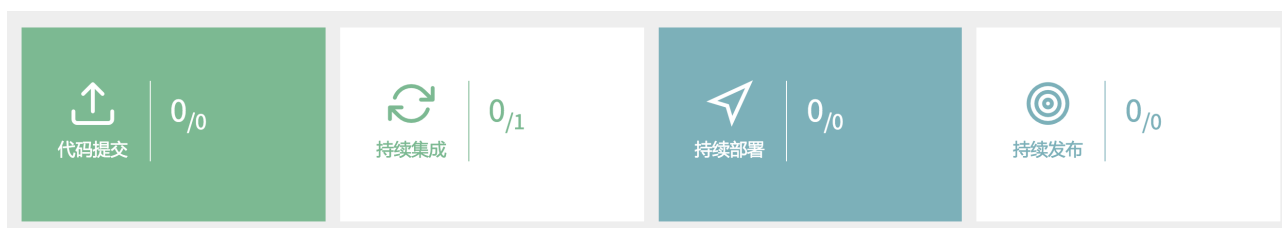
就此火车模型持续发布线完成配置。

第九步：点击左边栏



图标，回到项目首页。可以查看整个发布流程数据。包含git push了

多少次，持续集成了多少次，持续部署了多少次，持续发布了多少次等数据。



第十步：添加项目成员

在项目首页点击项目成员边上的



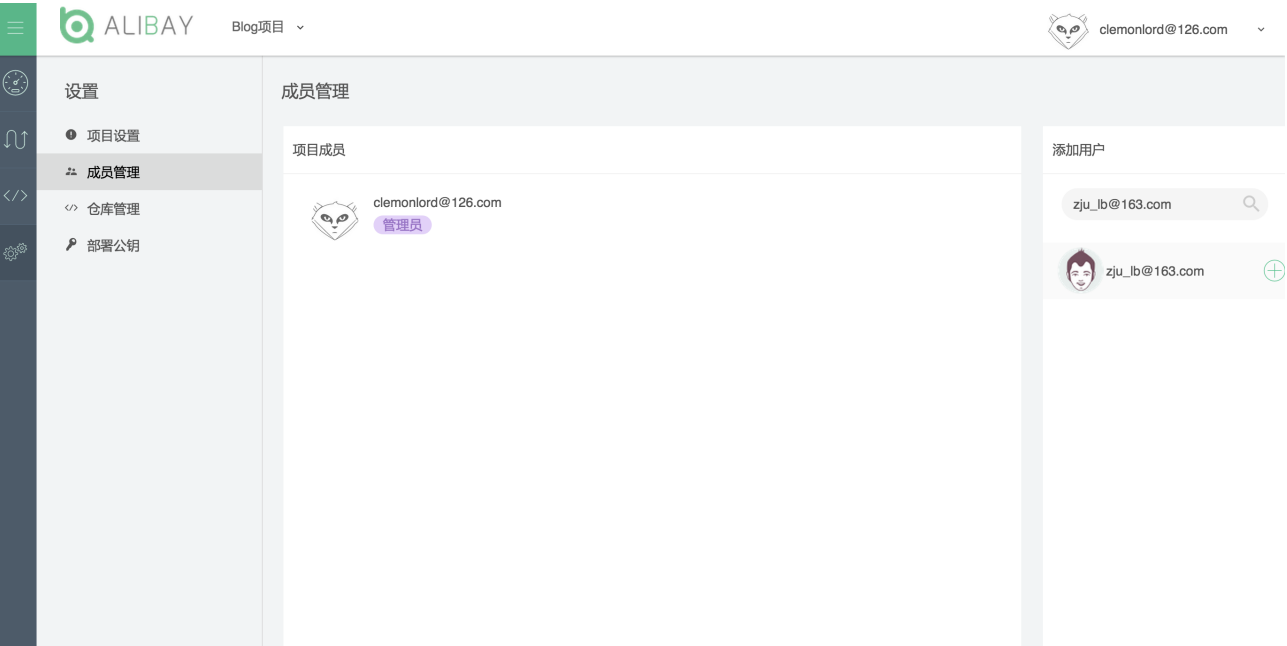
图标，或者



图标，选择成员管理。

The screenshot shows the '项目成员' (Project Members) page. The title '项目成员' is at the top left, and a right arrow icon is at the top right. Below the title, there is a large cat head icon and the text 'clemon...'. The page is mostly blank, suggesting a list of members is intended to be displayed here.

输入您需要添加的用户账号信息，点击绿色的+号图标则可以把该用户添加到您的项目中。



您可以设置该成员的角色，或者移出本项目

