# 前期准备

## 在IDEA中拉取Github代码。

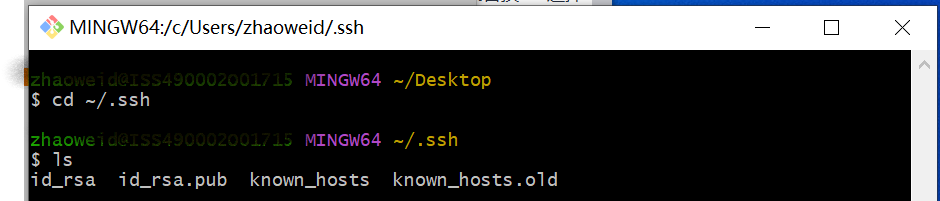
### 基于http协议：<https://github.com/iSoftStoneGroup/EazyBuilder.git>

### 基于SSH协议：[git@github.com:iSoftStoneGroup/EazyBuilder.git](mailto:git@github.com:iSoftStoneGroup/EazyBuilder.git)

#### 检查是否已有SSH Key

$ cd ~/.ssh

$ ls

（ 检查改目录下是否有id\_rsa.pub 或 id\_dsa.pub 文件，如果存在则直接添加到github ）

#### 若没有，创建SSH Key

# 这里填写自己github注册的邮箱

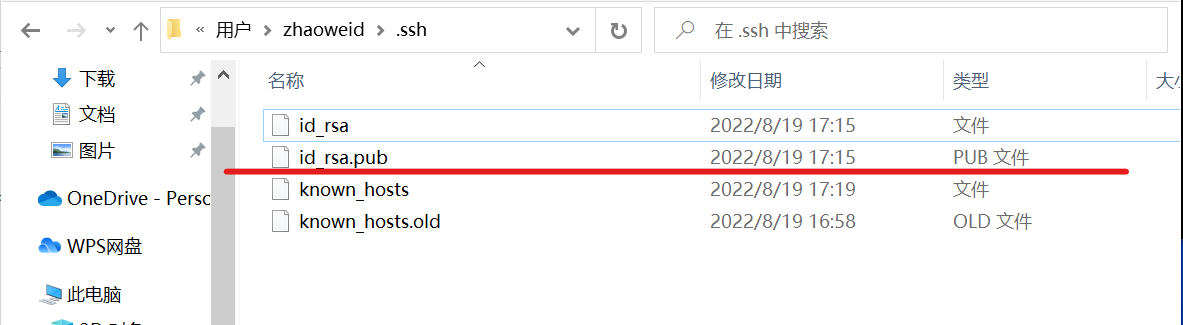
$ ssh-keygen -t rsa -C “@qq.com”



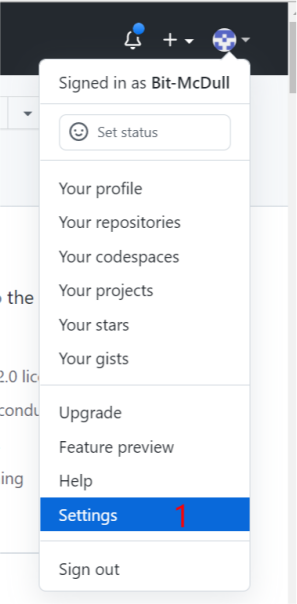
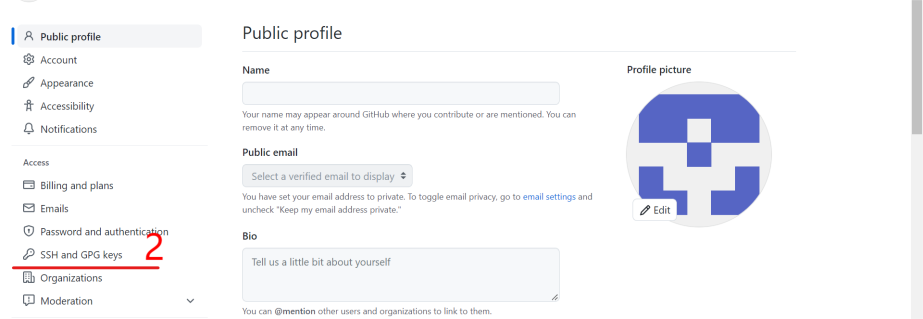
此处会弹出提示框，第一次是让你输入一个文件名，用于保存SSH Key，这里我们直接回车使用默认，接下来会提示输入两次密码，该密码是以后push文件所用的密码，这里我们直接回车默认不需要密码。

#### 添加SSH Key到Github

首先我们要找到生成的ssh key，windows的路径一般在C:\Users\Administrator.ssh下，复制 公钥id\_rsa.pub的内容。



#### 在Github中添加公钥id\_rsa.pub的内容,在GitHub上，查找SSH and CPG keys:

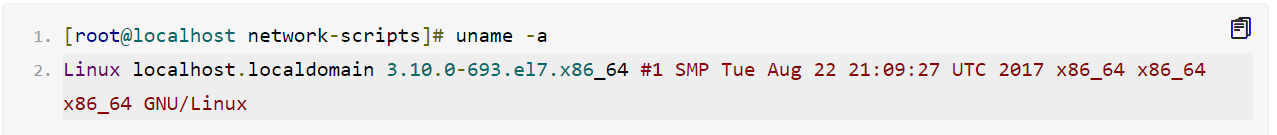




## **虚拟机Centos7中安装docker**

### root账户登录，查看内核版本如下：(系统的内核版本需要高于 3.10 ,验证是否支持 docker 。)

# uname -a



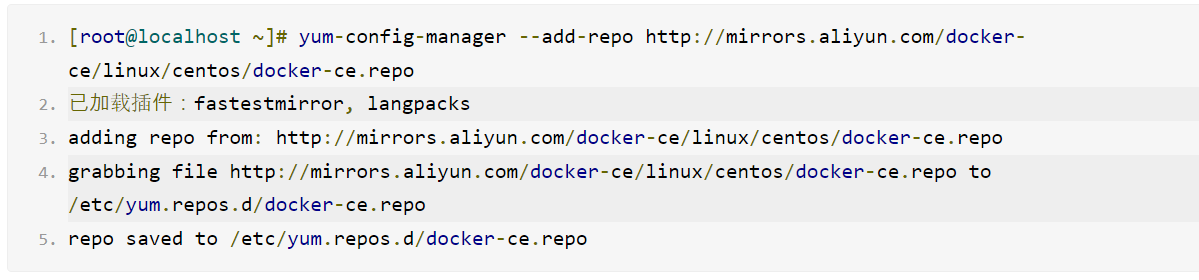
### 把yum包更新到最新（提示：新环境可随意操作，生产环境酌情更新）

# sudo yum -y update



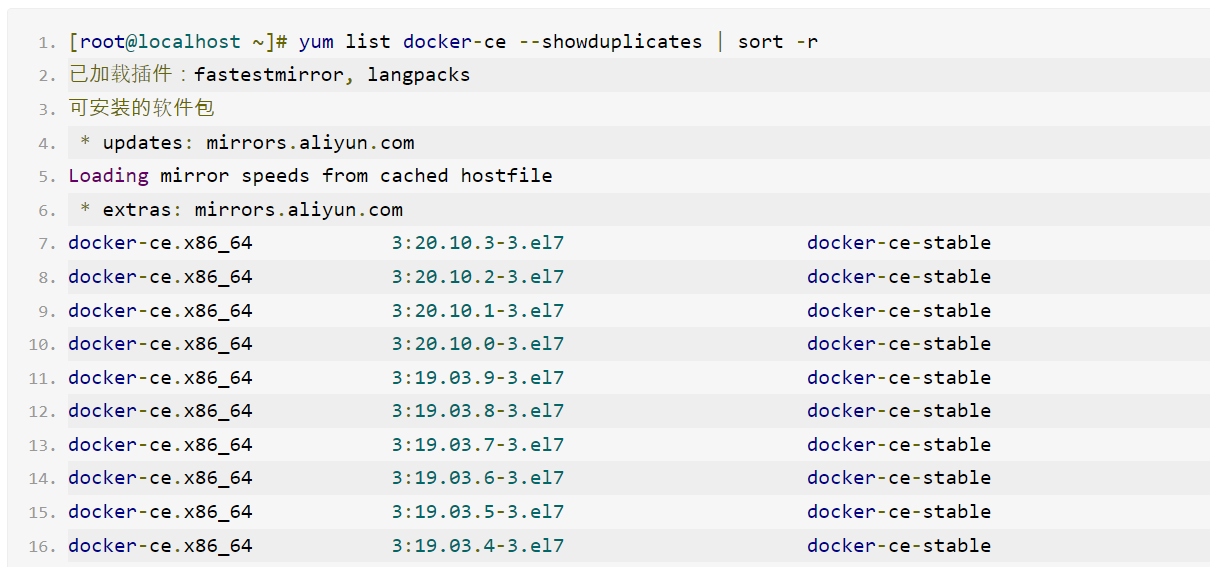
### 设置yum源(也可以设置别的yum源)

# yum-config-manager --add-repo <http://mirrors.aliyun.com/docker-ce/linux/centos/docker-ce.repo>



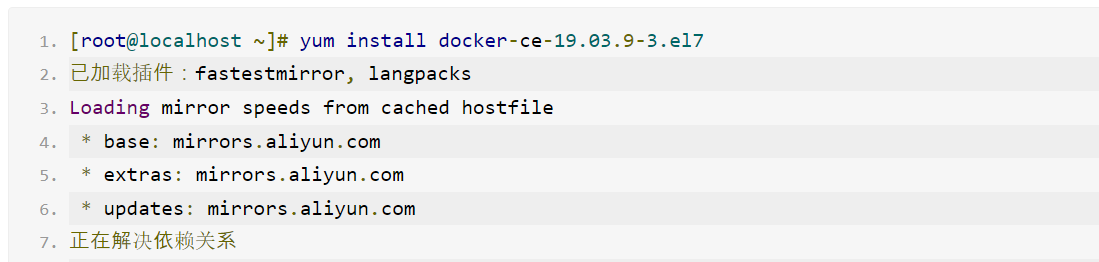
### 查看所有仓库中所有docker版本，并选择特定版本安装

# yum list docker-ce --showduplicates | sort -r



### 安装Docker

# sudo yum install docker-ce-19.03.9-3.el7



### 启动、加入开机自启

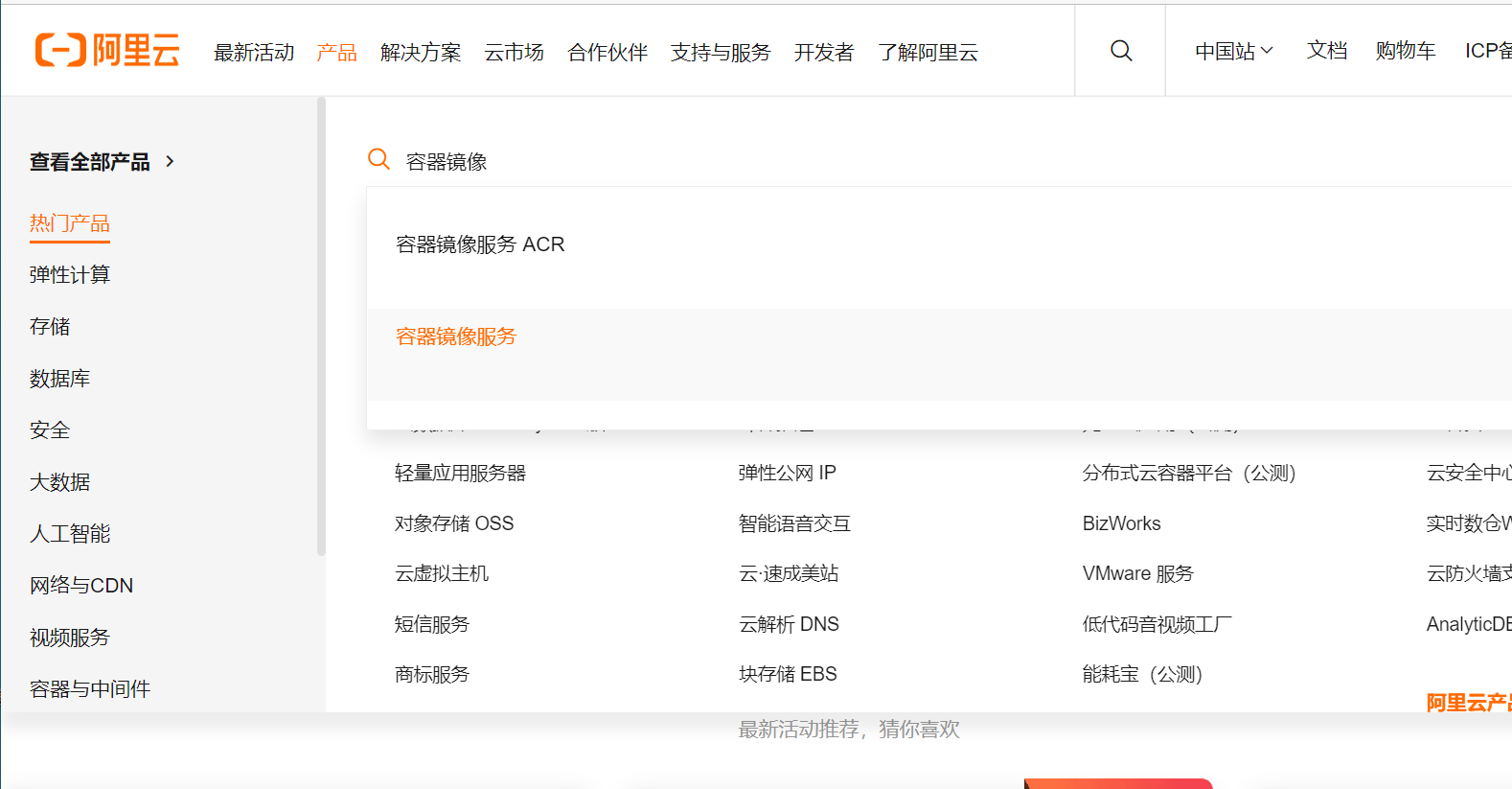
# sudo systemctl start docker

# sudo systemctl enable docker

### 测试是否成功

# docker version

### 配置阿里云加速（官网：<https://www.aliyun.com/> ），登陆后在产品菜单中搜索 容器镜像服务



### 9.点击 管理控制台



### 10.点击 镜像加速器，选择对应操作系统，按照步骤配置镜像加速器。



## **安装docker-compose**

### docker-compose（也可以选择别的版本）

curl -L [http://mirror.azure.cn/docker-toolbox/linux/compose/1.25.4/docker-compose-Linux-x86\_64](http://mirror.azure.cn/docker-toolbox/linux/compose/1.25.4/docker-compose-Linux-x86_64" \t "https://www.kuangstudy.com/bbs/_blank) -o /usr/local/bin/docker-compose

### 赋予执行权限

sudo chmod +x /usr/local/bin/docker-compose

### 测试是否成功

docker-compose -version

# Docker中安装启动项目

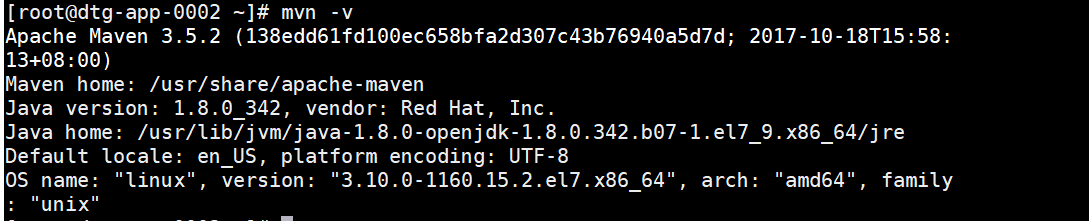
## 在Linux操作系统中安装配置Maven。

### maven官方下载地址如下：

https://maven.apache.org/download.cgi

### 查看Maven版本

mvn -v



## eazybuilder制作镜像，依耐maven docker plugin,需要在docker中开启远程api接口。

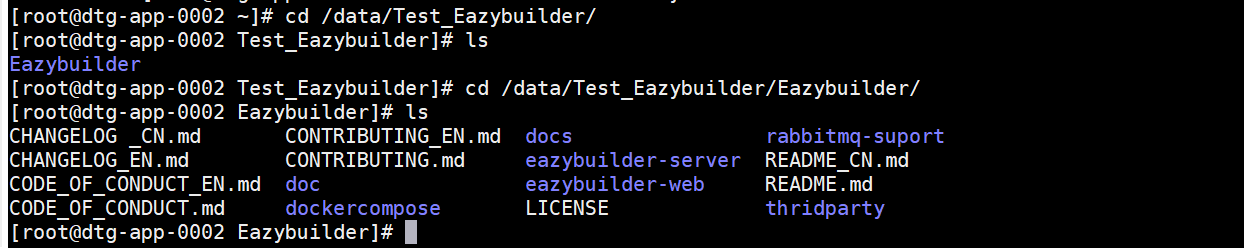
1. **修改文件：**

/usr/lib/systemd/system/docker.service

**更改配置为（将127.0.0.1改成自己的ip）：**ExecStart=/usr/bin/dockerd -H tcp://127.0.0.1:22375 -H fd:// --containerd=/run/containerd/containerd.sock

1. **查看是否生效**  
   docker -H 127.0.0.0:22375 ps

## 将Eazybuilder项目存放到Liunx系统（本文以CentOS7为例）的指定位置，进入项目所在目录。



### 进入dockercompose目录执行首先执行

# sh startMvn.sh

**再执行命令：**

# docker-compose -f eazybuilder.dependent.yml up -d

#### 若执行失败，删除当前文件中的mysql文件夹，然后先启动mysql，使用工具连接mysql，再创建nacos数据库和ci数据库，再nacos数据库中导入并执行nacos.sql文件。

1. **再在地址栏输入：<http://127.0.0.1:8848/nacos> (将127.0.0.1换成自己的ip)，登录nacos（账号：nacos 密码: nacos）,点击导入配置，将当前文件中nacos\_config.zip文件上传（将127.0.0.1换成自己的ip）。**



### 2.然后执行以下命令

# docker-compose -f eazybuilder.client.yml up -d

### 使用工具连接mysql，在ci数据库中，执行以下sql语句：

#### insert into `ci\_guard`(`id`,`guard\_type`,`level`,`name`,`threshold\_max`,`threshold\_min`) values

(1,5,NULL,'新增代码单元测试覆盖率',50,NULL),(2,4,NULL,'全量代码单元测试覆盖率',10,NULL),

(3,3,NULL,'单元测试通过率',50,NULL),

(4,0,NULL,'BUG(阻断)',10,NULL),

(5,1,NULL,'安全漏洞（阻断）',10,NULL),

(6,2,NULL,'编码规范（阻断）',10,NULL);

1. **更改base.url为自己eazybuilder访问地址，更改build.docker-host为自己docker远程地址**

Insert into `ci\_config`(`id`,`cfg\_key`,`cfg\_val`,`description`) values

(1,'jenkins.csrf.crumb','true',NULL),

(2,'registry.url','registry-devops.cn',NULL),

(3,'base.url','http://upms-web.eazybuilder/ci/',NULL),

(4,'build.docker-host','0.0.0.0:22375','maven docker plugin依耐的docker api');

## 操作界面

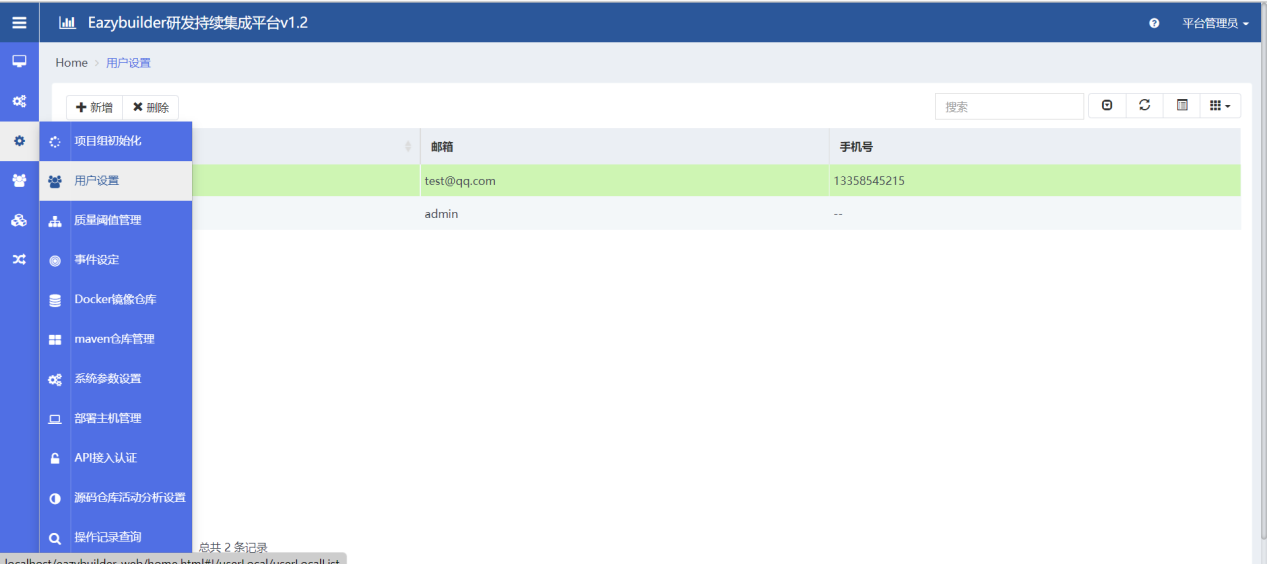
### **登陆地址（将127.0.0.1换成自己ip）：**[http://127.0.0.1:30091/eazybuilder-web/login.html](http://localhost/eazybuilder-web/login.html)

### 初始账号：admin 初始密码：admin

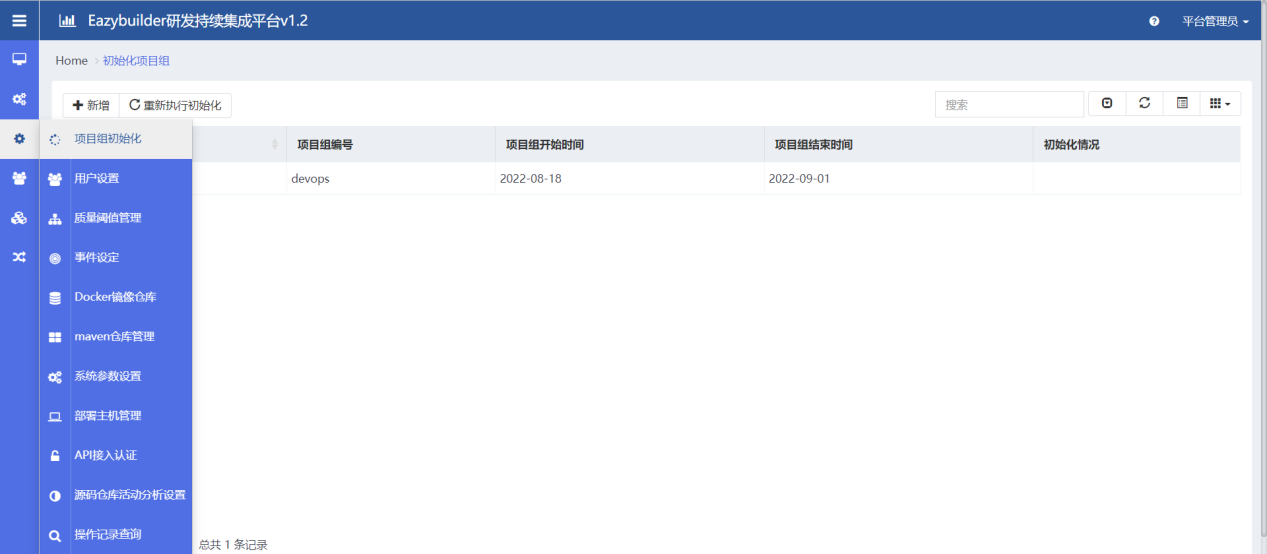




### 在系统管理 ---用户设置中添加用户



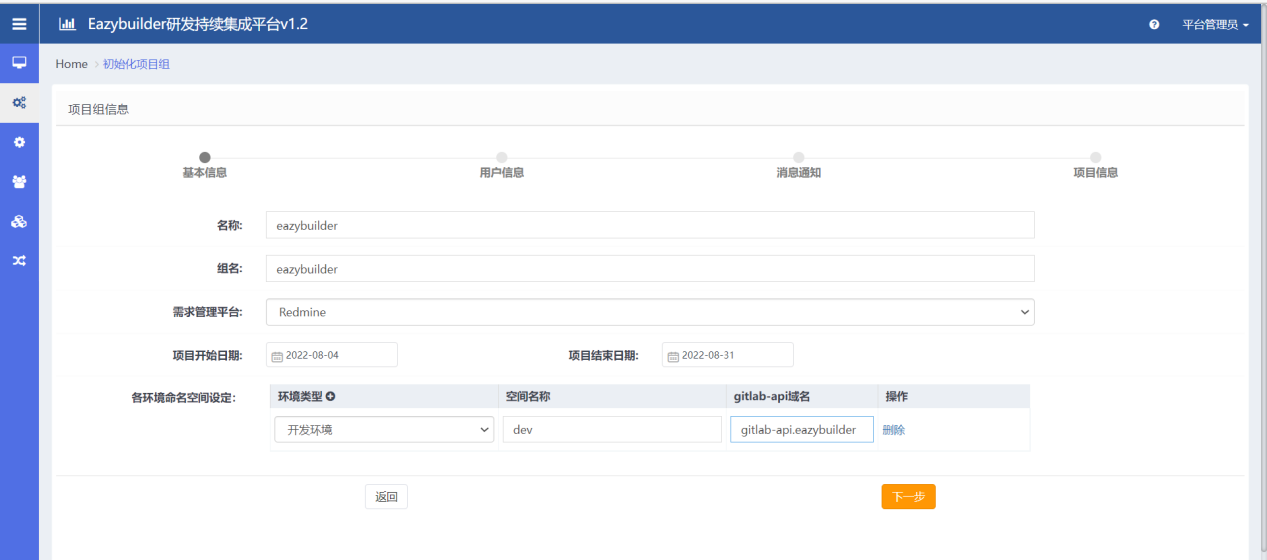
### 系统管理---项目组初始化添加项目组



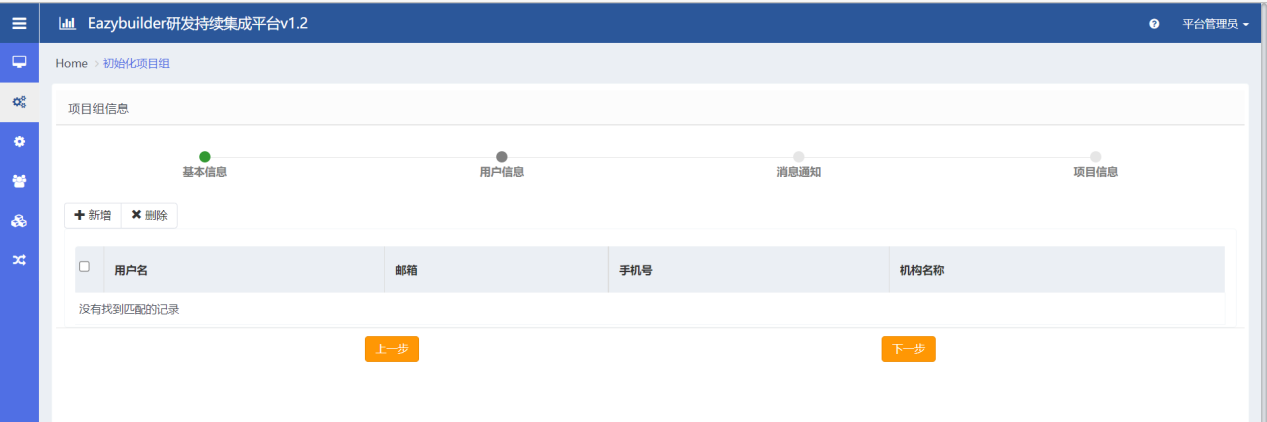
#### 进入新增页面，填写项目基本信息，

##### 其中组名要和gitlab、redmine（或禅道）关联，所以只支持小写英文。

##### 最下方是命名空间设定，主要用于docker镜像发布、nacos配置文件发布

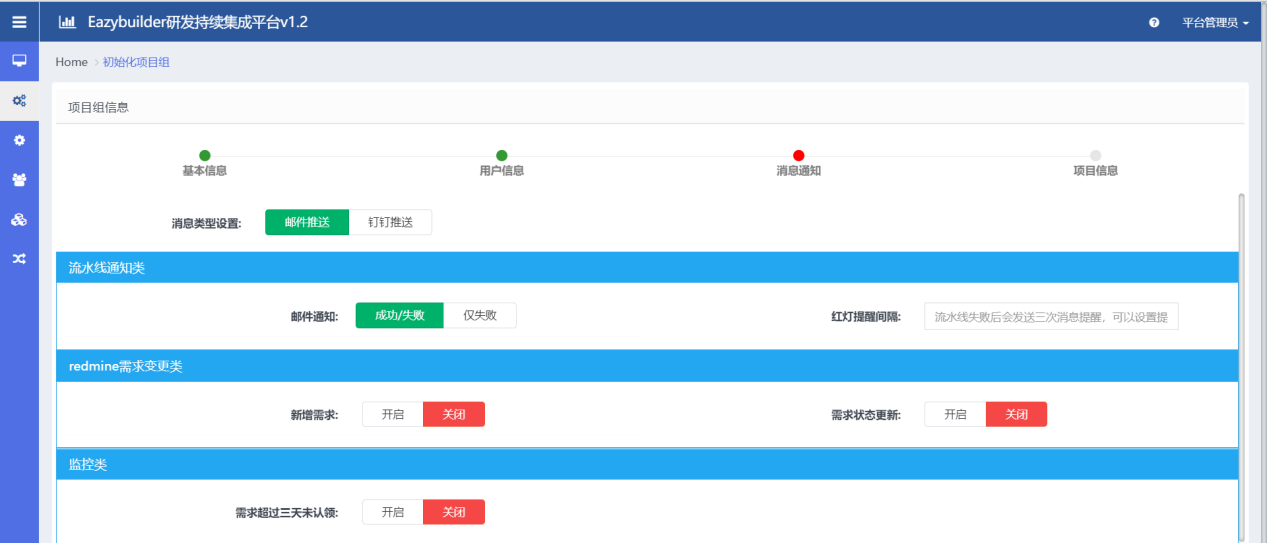


##### 填写完基本信息之后点击下一步进入用户信息设置页面





##### 通知类型有邮件和钉钉两种，若选择钉钉需要设置机器人webhook信息



##### 点击下一步进入项目信息设置页面（也可以直接点保存跳过此步骤，在项目设置菜单里进行设置）

