

# June 24, 2024

## 邮箱验证功能

### 配置

授权密码: FTGAVZFOTINYROSS

```
# 在 application.yml 里的 mail 配置
mail:
  protocol: smtps
  # 配置 smtp 服务器地址
  host: smtp.163.com
  # 发送者邮箱
  username: 15160078109@163.com
  # 配置授权密码
  password: FTGAVZFOTINYROSS
  # 端口号
  port: 587
  # 默认的邮件编码为 UTF-8
  default-encoding: UTF-8
```

## 将项目更新至 GitHub

将本地代码上传到 GitHub 仓库的指令流程如下:

### 1. 初始化 Git 仓库（如果还没有初始化）

如果你还没有在本地项目中初始化 Git 仓库，可以使用以下命令：

```
git init
```

### 2. 添加远程仓库

如果你还没有将远程仓库添加到本地仓库，可以使用以下命令：

```
git remote add origin https://github.com/MRYUHUI/Machinery-Mall.git
```

### 3. 添加所有更改

将所有更改添加到 Git 索引中：

```
git add .
```

## 4. 提交更改

提交更改并附带一条提交信息：

```
git commit -m "Your commit message"
```

## 5. 推送到远程仓库

将本地仓库中的更改推送到远程仓库。通常，第一次推送需要指定上游分支：

```
git push -u origin main
```

以后推送代码时，只需使用以下命令：

```
git push
```

## 常见问题和解决办法

### 1. 远程仓库已经存在

如果你尝试添加远程仓库时收到 `error: remote origin already exists.` 错误，说明已经配置了远程仓库。你可以使用以下命令查看远程仓库：

```
git remote -v
```

### 1. 更改默认分支名称

如果你的默认分支不是 `main` 而是 `master` 或其他名称，可以使用以下命令更改分支名称：

```
git branch -M main
```

### 1. 合并远程更改

在推送代码之前，建议先拉取远程仓库中的更改，以避免冲突：

```
git pull origin main
```

## 完整流程示例

假设你已经在本地项目中初始化了 Git 仓库，并且需要将更改推送到 GitHub 仓库，以下是完整的流程示例：

```
# 初始化 Git 仓库（如果还没有初始化）
git init

# 添加远程仓库（如果还没有添加）
git remote add origin https://github.com/MRYUHUI/Machinery-Mall.git

# 查看远程仓库
git remote -v

# 添加所有更改
```

```
git add .

# 提交更改
git commit -m "Your commit message"

# 合并远程更改 (可选)
git pull origin main --rebase

# 推送到远程仓库
git push -u origin main
```

## 将 GitHub 仓库更新到本地

打开终端（命令行）。

导航到你的本地仓库目录。

```
cd D:\A_Project\互聯網應用綜合實踐\Machinery-Mall
```

从远程仓库获取并合并更新：

使用 `git pull` 命令从远程仓库拉取更新并合并到本地分支：

```
git pull origin main
```