



# 语联灵犀项目 - GitHub 协作与版本控制指南

文档维护人：前端负责人 适用对象：全体研发成员

为了保证代码库的整洁和每次交接的顺利，本项目采用“**单线程接力模式**”。即：同一时间只有一个人在修改主仓库代码。

## 核心原则：交通灯法则

在开始写代码前，请务必在群里确认当前状态：

1.  **红灯 (有人在跑)**：队友正在开发中，**请勿 Push 代码**，防止冲突。
2.  **绿灯 (无人占用)**：上一位队友已 Push 完毕，你可以 Pull 最新代码并开始工作。

## 第一部分：如何下载代码

### 场景 A：我是第一次加入项目

如果你电脑上还没有项目文件夹：

```
1 # 1. 打开你想存放项目的文件夹终端
2 # 2. 克隆仓库（只需做一次）
3 git clone https://github.com/Azurezjz/yulian-lingxi.git
```

### 场景 B：轮到我干活了

在写任何一行代码之前，必须先同步上一位队友的进度。如果不更新直接写，提交时会报错。

```
1 # 1. 进入项目目录
2 cd yulian-lingxi
3
4 # 2. 拉取最新代码
5 git pull
```




如果提示 `Already up to date`，说明你是最新的，可以开始干活。

## 第二部分：如何上传代码 (交棒)

当你完成了你的模块（如：写完了后端接口，或画完了前端页面），并**本地测试通过**后：

### 1. 提交前进行代码处理

为了保持代码库的专业性，请在提交前执行“**三删原则**”：

-  **删除调试代码**：删掉所有的 `console.log`, `print()`, `alert()` 等测试语句。
-  **删除废弃代码**：不要保留大段被注释掉的旧代码（Git 有历史记录，不需要你保留尸体）。
-  **删除敏感信息**：**绝对不要**上传 API Key、密码或数据库连接串（请放在 `.env` 文件或本地配置文件中）。

## 2. 执行提交三部曲

打开终端，依次执行：

```
1 # 第一步：将修改加入暂存区
2 git add .
3
4 # 第二步：提交并填写备注（请遵循下方的备注规范）
5 git commit -m "feat: 完成了后端天气查询接口"
6
7 # 第三步：推送到 GitHub
8 git push
```



## 第三部分：如何获取特定版本的代码

如果你想查看之前的某个版本：

1. 点击代码列表上方的 **Releases** 或 **Tags**。
2. 找到对应的版本（如 `stage-1`），点击 **Source code (zip)** 下载。
3. 解压后，就是一个独立的、那个时期的项目包。



## 常见问题急救

Q1: `git push` 时报错 `fetch first` 或 `rejected`？

- **原因：**有人在你之前提交了代码，你现在的版本落后了。
- **解决：**
  1. 先执行 `git pull` 把队友的代码拉下来。
  2. 如果没有冲突，Git 会自动合并。
  3. 再次执行 `git push`。