



链滴

# 基于 Gitlab 进行开发团队管理的尝试——0 0. 通过 issue 管理需求

作者: [crick77](#)

原文链接: <https://ld246.com/article/1574179406015>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

# What

产品经理、测试人员、开发人员统一在Gitlab中管理需求、bug。

# Why

为什么通过Gitlab issue管理，而不是Jira、Redmine等工具？

1. 开发团队最终交付物为项目代码，需求、bug最终都会转换为一行代码、一次MR。通过issue以让每一步都可以溯源。
2. Gitlab issue更轻量，markdown语法让issue更专注于内容本身

# How

1. 每组通过独立项目统一管理issue，在 Readme中描述使用方式及定义
2. 通过 milestone管理项目版本，对齐目标。节奏很重要
3. 通过 label管理优先级(P0|P-1|P-100)、类型(bug|feature)
4. 通过 board查看issue进度状态，配合To Do、Doing、Verify等label，定义issue生命周期
5. 通过模板定义 author需要提供的信息

# 项目Readme

# 新建 Issue 相关细节

- 模板：从以下 2 者中进行选择
  - feature
  - bug
- Assignee: 开发负责人
- Label:
  - 优先级
    - P-1: 全线 block 工作，直接在群里汇报，不需要走 gitlab，例如：
      - 网站无法使用
    - P0: block 个人工作
    - P1: 暂时不 block 工作，但是周尺度需要解决
    - P2: 暂时不 block 工作，周尺度外需要解决（配合 DDL-调整优先级）
  - label(可选)
    - BUG
    - Feature
  - 里程碑
    - 需求提出月份

# 验收-After 工程已完成测试

- 工程会在完成 BUG Feature 后指定相关人员做验收，默认谁提 feature BUG 谁做验收（可能会存特殊的指定）
- 开发完成后会放入verify看板，验收通过后由author关闭issue

# Gitlab issue模板

在项目 `.gitlab/issue_templates` 目录下的markdown文档, 可以在新建issue时被选择。利用这个性可以规范issue内容, 督促author提供有效信息。如:

## feature.md

### # 背景

> 回答为何要做: 不做会有怎样的问题; 做了会有怎样的收益;

### # 需求说明

> 回答要实现的目标是什么==需求说明

### # 方案

> 回答如何去做, 提供参考思路或模型 (可选-工程拍板)

### # 验证

> 回答何叫做到, 验证结果满足预期的标准有哪些, 是什么. (满足测试用例)

## bug.md

### # 步骤

> 本来想做的事情-描述

### # Bug行为

> 被 block 的点-描述

### # 期望行为

> 应该是什么样

### # 附件

> URL/相关信息-id 号/截图(整个界面的完整截图)

## 小技巧

1. **issue comment**移动board状态 -> `/board_move ~Doing`
2. 跨组关联 **issue** -> `group/project#issueNO`
3. **MR**中关闭关联的**issue** -> `close #9`
4. 添加子任务 -> `- [ ] subtask1`
5. 添加 **issue**耗时 -> `/spend 3d 5h 10m`