

Please visit website: <http://cxyroad.com>

偷偷分享下我们公司的研发规范~

=====

大家好，我是程序员鱼皮。前几天我分享了自己[创业一周年的复盘总结](<http://cxyroad.com/>”<https://mp.weixin.qq.com/s/PLITO3xq6eheDJKUu9j4IA>”)，其中提到了一点：随着团队的扩大，我们会更注重研发规范和技术沉淀。

有程序员朋友就问了：啥是研发规范？

还有朋友表示：鱼皮别拿咱当外人，把你们公司的研发规范发来看看？

可以，必须安排！

这篇文章就给大家简单分享下我们公司的研发规范，不过在开始前必须要明确2点：

0. 每个团队都应该根据情况定制自己的研发规范，别人的规范仅供参考，未必最适合你们团队。

1. 篇幅有限，本文仅分享一些我认为很重要的规范，并且移除了我们自己的敏感信息。

一、项目整体研发流程

1) 团队共同确认目标和规划

开会讨论，产出目标和规划文档

2) 产品调研和需求分析

产出调研报告和需求分析文档

3) 需求评审

开需求评审会，明确要做的需求和工作，评估工作量并明确工作时间节点。

4) 方案设计

产出方案设计文档，比如数据库表设计、页面设计、接口设计等。

5) 研发

包括各自开发、单元测试、前后端联调等

6) 测试和验收

包括研发自测、产品验收、组内验收等

7) 代码提交

提交可上线的代码，需要由负责人审查，通过后可合并

8) 部署上线

将代码发布到服务器上，组内进行上线通知并更新上线文档，上线后需要自行验证

9) 产品迭代

持续收集用户对新功能的反馈、并进行数据分析，从而验证改动效果，便于下一轮的更新迭代。

二、开发规范

开发前注意事项

- 1) 确保自己充分理解了业务和需求，需要先进行整体的方案设计；尤其是对于重要需求和核心业务，必须先跟组内同学核对方案并通过后，才能下手开发，避免重复工作。
- 2) 先熟悉项目再开发，建议阅读项目文档、项目代码、接口文档、前端组件文档等。
- 3) 慎重引入新的依赖或类库、或者升级版本，重大依赖变更需要和组内其他成员确认。
- 4) 熟悉团队已实现的功能和代码，尽量复用，避免重复开发。
- 5) 熟悉团队内部的研发规范，并在 IDE 中进行相应的配置，比如前端配置 ESLint、Prettier 等代码规范插件。

开发中注意事项

- 1) 开发新功能时，确保从项目仓库拉取 ****最新主分支**** 的代码。
- 2) 每个功能都要新建自己的分支进行开发，****千万不要直接修改主分支的代码****！注意分支名称要使用英文、足够语义化，不要和其他人的混淆。
- 3) 开发时，尽量复用现有的功能、模块、类、方法、对象代码。有现成的代码，就不要再重复编写。如无法复用，可以适当通过注释说明。
- 4) 开发时，遵循团队内部的研发规范，尽量参考现有项目代码的写法，尤其是不要使用和原项目不一致的格式、命名、写法，避免特立独行。
- 5) 开发过程中，有任何不明确的地方，不要凭空猜测，及时去联系项目的其他成员或负责人确认。

- 6) 开发过程中，每隔一段时间（比如 1 – 3 天）可以使用 `git pull` 同步一下最新的主分支代码，防止合并代码冲突。
- 7) 开发过程中，注意整体时间进度的把控，先完成再完美，有风险时及时反馈。
- 8) 开发时，需要格外注意对异常情况的捕获和处理。
- 9) 每个分支尽量保证纯净，尽量减少每次开发和提交时改动的代码量。建议每次开分支只改一个功能、Bug 或模块，不要把多个不相关的功能写在一起，并且非必要不修改。
- 10) 完成部分功能开发后，一定要自测！自测时，可以 Mock 假数据。**注意一定不要在线上测试、一定不要影响线上数据！**

三、代码提交规范

- 1) 只有通过测试和产品验收的代码，才能够发起合并到主分支的 PR 请求。在这之前可以提交到自己的分支。
- 2) 发起合并到主分支的 PR 前，**一定要完整阅读 3 遍自己的代码**，避免不规范的写法和无意义的改动。
- 3) 每次合并尽量只专注于一个功能或改动，避免多个功能耦合在一起合并，提高审查效率并降低改动风险。
- 4) 每次提交时，需要在 commit 信息中提供代码改动说明，还可以通过关联需求文档、测试用例、方案文档、效果截图等方式进行补充说明。

commit 信息可参考[《约定式提交》文档](<http://cxyroad.com/>"<https://www.conventionalcommits.org/zh-hans/v1.0.0/>"), 但不做强制要求。

- 5) 除非特殊情况，否则所有的代码必须经过至少一位项目负责人 Code

Review 审核通过后，才能合并；并且只有合并到主分支的代码才允许发布上线。

上线规范

上线前注意事项

1) 上线前，除了严格验证功能特性能否正常运行、并符合需求外，还要格外程序的：

- * 健壮性。比如给用户友好的错误提示、输入校验。
- * 安全性。防止越权操作、输入校验。
- * 稳定性。尽量保证调用 100% 成功，如果有几率失败，要考虑重试或容错策略。

2) 除非特殊情况，只有经过产品验证的功能、通过代码审核的主分支代码才允许发布上线。

3) 除非特殊情况，尽量在工作日上线（建议周二 ~ 周四），保证上线后出了问题时能够及时修复。

上线后注意事项

1) 上线后，一定要再次进行完整流程的测试，尤其要重点权限相关的功能测试。

2) 上线后，一定要在群内及时同步上线信息，周知相关的成员，如果遇到问题第一时间反馈。

3) 首次上线后，需要即时配置监控告警。

4) 上线验证通过、并经过内部群成员确认后，可以在外部用户群发布版本更新公告。

5) 上线后，即时更新项目的更新记录文档。

6) 注意，上线不是终点。上线后的一段时间（至少一周内），一定要持续观察自己负责的功能是否正常运行、持续接受用户反馈、通过数据分析来观察新功能的效果，期间有任何问题都需要即时修复处理，并且准备好下一期的改进迭代。

原文链接: <https://juejin.cn/post/7360486735798927396>