

Git代码管理规约

1. commit注释规范

使用关键字开头，体现更改类型和涉及的范围：

feat: 新增功能

fix: 修复错误

docs: 文档变更

style: 格式或风格改进（不影响功能）

refactor: 重构代码（不影响现有功能）

test: 测试相关更改

chore: 构建脚本、依赖升级等维护性工作

ci: 持续集成配置更改

perf: 性能优化

2. 分支管理

- a. 一个需求建一个dev分支，比如dev/dealer，本地也可以建自己的对应分支比如dev/dealer-h,也可以直接在dev/dealer上开发，看自己习惯。
- b. 测试时多个需求合并到test-all，test-all代码与测试环境对应
- c. 上线会归档到release/xxx，release/xxx最高版本上线后与正式环境对应，上线前可视为uat。
- d. bugfix/xxx：bug修复分支，从最新版切出。
- e. master通常作为归档，就不轻易提交，与正式环境对应。
- f. 名称带"/"有利于分组，比如dev/开头的分支会自动归为一组。

3. 分支合并

- a. 共用的分支合并时用merge,保留提交记录，方便查阅及追溯。
- b. 自己独用的分支可以用rebase,也可以merge。