软件开发部DevOpt&TestOps

体系规范

日期：2021年8月

前言

为了规范软件开发部开发-运维体系及测试-运维体系正常合理运作，保证各项研发、测试、生产活动正常进行，降低研发、测试、运维成本，现对软件开发部研发、测试、运维等工作做出如下规范要求。

1. 分支管理规范
2. 项目立项之初由研发负责人在内网gitlab创建此项目的代码仓库，权限应设置为私有，项目参与人以“develop”身份加入到项目中。项目研发负责人（owner）对该仓库的分支、代码审核负直接责任。
3. 每个项目设立三套分支：
   1. 开发分支：开发人员进行代码协作的代码分支，使用专有分支`develop`，仅开发人员可操作
   2. 测试分支：测试人员进行回归验证的代码分支，使用专有分支`test`，仅测试人员可操作
   3. 生产分支：开发人员进行成果交付的代码分支，使用专有分支`master`，仅项目研发负责人可操作
4. 分支提交及合并遵循如下规范：
   1. 当有新需求就绪时，开发人员从最新的`master`拉取新的`feature-\*`分支，进行代码开发
   2. 当需求取得阶段性进展，需要内部验证时，开发人员把自己的`feature-\*`分支合并到`develop`分支并部署开发环境，然后基于开发环境进行内部验证
   3. 当需求完成，需要提测时，开发人员同步待提测的`feature-\*`分支给测试人员，由测试人员将此`feature-\*`分支合并到`test`分支并部署测试环境，然后测试人员基于测试环境进行回归验证
   4. 当回归测试通过，需要进行成果交付时，开发人员提请项目负责人审批，待审批通过后，把自己的`feature-\*`分支合并到`master`分支并部署生产环境，然后开发人员和测试人员共同基于生产环境进行成果验收
5. 有如下几点需要在此过程中注意：
   1. 自始至终，仅`feature-\*`分支可引入代码合并，`master`、`test`、`develop`间不允许互相合并
   2. `test`分支和测试环境归属于测试人员，开发人员不可进行操作和变更
6. Docker构建规范
7. 在后端项目创建时，应默认集成docker插件，以支持构建docker镜像。在需要构建镜像的模块添加maven插件，并添加Dockerfile。样例文件如下，注意pom文件中Dockerfile文件路径。



1. 前端项目创建时，应默认在项目根目录存放如下三个文件，pom和Dockerfile内容无需做任何修改，nginx.conf可按照接口代理情况进行配置，详情可咨询开发及运维同事。



1. 前端项目开发完毕需要打包时，应在相应分支根目录创建release文件夹，将打好包的内容放到此目录下并提交。
2. 配置文件管理规范
3. 后端项目应默认创建五个配置文件，分别为：
   1. application.yml 默认配置文件，存放一些通用配置
   2. application-dev.yml 开发环境配置文件，存放开发环境参数
   3. application-test.yml 测试环境配置文件，存放测试环境参数
   4. application-demo.yml 演示环境配置文件，存放演示环境参数
   5. application-prod.yml 生产环境配置文件，存放生产环境参数
4. CI脚本管理规范
5. CI部分由Gitlab-CI模块来完成功能，以后每个项目的不同分支下将会有一个.gitlab-ci.yml的文件来完成所需的镜像的打包构建和上传操作.该部分具体内容由开发人员联系运维共同商定后添加。

关于.gitlab-ci.yml配置的模板如下

1. 每个.gitlab-ci.yml文件，大致由image ,stages, variables, 和具体的stages步骤组成，stages多少与具体项目的模块数多少有关
2. 具体的stages内容需要由运维与对应的开发负责人共同商定，其中script脚本部分，指具体的maven打包构建命令，该部分内容由开发详细给出到运维。
3. only部分牵扯到代码构建触发规则的，则由运维与具体开发负责人或者项目负责人商定后制定。该部分一般商定后就不再更改，如有变更，开发负责人需及时联系运维更改
4. 提测标准规范