软件项目研发管理规范

(开发-试行)

**Date:2020.6**

**软件组**

目录

[一、 设计阶段](#_Toc17707)

[1.PRD文档输出及评审(外部输入)](#_Toc8906)

[2.UE设计输出及评审(外部输入)](#_Toc14604)

[3.UI设计输出及评审(外部输入)](#_Toc28184)

[-----------------------------------------------------------------](#_Toc6127)

[1.系统架构设计](#_Toc28198)

[2.系统详细设计](#_Toc9275)

[3.工作任务分解，版本研发计划制定](#_Toc31739)

[二、 开发阶段](#_Toc23090)

[1.编码开发相关要求](#_Toc21724)

[1.1.java开发规范/安卓/前端开发规范](#_Toc5087)

[1.2YApi接口文档](#_Toc15615)

[1.3数据库设计](#_Toc30106)

[1.4配置文件](#_Toc15494)

[1.5基线版本模板](#_Toc12153)

[1.6公共能力SDK](#_Toc29100)

[1.7项目开发基础规则](#_Toc17247)

[2.研发管理相关工作](#_Toc2319)

[2.1SVN工程](#_Toc15509)

[2.2git工程](#_Toc13680)

[2.3Sonar检查](#_Toc27914)

[2.4Jekins工程](#_Toc2666)

[2.5代码检视工作](#_Toc9645)

[2.6每日站会](#_Toc31215)

[2.7.项目开发周报](#_Toc19069)

[2.8.技术总结文档](#_Toc29316)

[2.9.开发过程中需求细节明确](#_Toc26698)

[三、 测试阶段](#_Toc19610)

[1.提测准入条件](#_Toc24884)

[1.1冒烟测试用例执行通过](#_Toc26487)

[1.2 Sonar检查问题修复](#_Toc11884)

[1.3 YApi接口文档提供](#_Toc6787)

[1.4测试环境部署说明](#_Toc26826)

[1.5列举对公共服务、基础组件、外部平台的依赖](#_Toc1536)

[2.版本测试阶段问题收敛](#_Toc5932)

[3.代码审核后合入](#_Toc26223)

[4.安全扫描和问题修复](#_Toc30911)

[5.UI设计还原问题修复](#_Toc26675)

[6.测试阶段的git管理](#_Toc29344)

[7.yapi上接口、数据库脚本、配置文件控制](#_Toc9517)

[8.Jira问题处理要求](#_Toc17272)

[9.生产环境申请](#_Toc1133)

[四、 上线阶段](#_Toc29951)

[1.封板测试阶段准备上线部署文档](#_Toc13351)

[2.上线过程中的问题定位](#_Toc647)

[3.上线操作时间点](#_Toc30859)

[五、 项目总结](#_Toc9477)

[1.项目总结](#_Toc7735)

[六、 线上支撑](#_Toc30765)

[1.线上问题反馈](#_Toc5465)

[2.功能增强需求](#_Toc2457)

[七、 遗留问题](#_Toc8892)

[八、 其他](#_Toc10781)

[1.需求变更场景](#_Toc15949)

整体流程图

图片包含 游戏机, 截图

描述已自动生成

# 设计阶段

## 1.PRD文档输出及评审(外部输入)

PRD需求说明文档，由产品经理输出。

对文档的要求：与客户沟通，客户反馈同意；与开发沟通，评审通过。

提供评审通过PRD后，开发进入项目开发初期阶段，包括：工作量评估、UE设计、架构设计等。

PRD文档模板由产品经理决定。

## 2.UE设计输出及评审(外部输入)

UE设计文档，由UE设计工程师输出。

对文档的要求：与产品经理、开发沟通，评审通过。

提供评审通过UE后，开发进入项目开发阶段，包括：详细设计、后台编码开发。

UE设计文档模板由资源统筹组决定。

## 3.UI设计输出及评审(外部输入)

UI设计文档，由UI设计工程师输出。

对文档的要求：与产品经理、开发沟通，评审通过。

提供评审通过UI后，开发进入项目开发阶段，包括：Web、APP、小程序等前端编码开发。

UE设计文档模板由资源统筹组决定。

## 4.系统概要设计

(包括基础组件、模块、整体架构)

根据PRD和UE设计文档，进行对应的系统架构设计，由开发负责人或项目开发人员输出。

对文档的要求：召集相关人员，进行系统概要设计评审。

## 5.系统详细设计

(包括流程梳理、逻辑梳理、数据库设计)

根据PRD、UE、系统架构设计文档，进行对应的系统详细设计，由项目开发人员输出。

对文档的要求：召集开发负责人、项目开发参与人员，进行系统详细设计评审。

**系统详细设计模板，见附件。**

系统架构设计与详细设计某些项目，可合并简化

## 6.工作任务分解，版本研发计划制定

(项目里程碑、工作量大小、难易程度、优先级、依赖关系、与测试沟通测试计划)

根据PRD、UE、系统架构设计文档，进行对应的工作量评估、工作任务分解和详细研发计划制定，由开发负责人输出。

其中，开发阶段由开发侧输出；测试阶段，与测试沟通后，由测试侧输出。

对文档的要求：整体时间计划，与项目质量管理室确认；召集项目开发参与人员，进行任务分解和计划评审。

**版本研发计划模板，见附件。**

# 开发阶段

## 1.编码开发相关要求

### 1.1.java开发规范/安卓/前端开发规范

责任人：项目组全体成员

代码开发过程前，学习开发规范，开发过程中按照对应的开发规范进行开发。

JAVA开发规范：**阿里巴巴Java开发手册，见附件**。

**安卓开发规范：待定**

**IOS开发规范：待定**

**前端开发规范：待定**

### 1.2接口文档

责任人：项目Java开发

后台微服务提供的接口都必须输出到YAPI（设计阶段） 上

要求：接口录入后必须进行评审，接口说明明确，输出输出值的格式、返回值文字说明明确，接口限制，返回码，参数限制(长度、特殊字符、枚举、整型范围等)明确

接口变更，注意版本向前兼容

### 1.3数据库设计

责任人：项目Java开发

在设计阶段，已经进行了初步的数据库设计，开发过程中涉及到数据库脚本变更的，需要进行评审，并归档到git

数据库脚本注意增量脚本、全量脚本，数据库变更版本向前兼容

### 1.4配置文件

责任人：项目Java开发

微服务项目创建时包含本地环境、开发环境、测试环境、生产环境配置文件，并在Apollo配置中心 进行管理

配置文件的注意事项包括

1.涉及文件上传的服务 server.tomcat.basedir  
2.所有ip、用户、密码相关的配置项是否已修改为生产环境  
3.微服务的名字、端口等配置是否已确认过  
4.禁用SpringConfigServer的配置是否增加  
5.使用的session的地方是否有增加namespace的配置  
6.数据库连接是否使用druid  
7.使用kafka的服务，kafka配置是否正确  
8.禁用Actuator端点的配置检查

9.密码相关信息是否已加密

## 2.研发管理相关工作

### 2.1SVN工程

svn://ip:port/研发小结/项目管理，责任人:项目负责人

项目各阶段的输出件，归档到SVN目录下

### 2.2git工程

[http://ip:port/](http://172.16.1.21:18080/)，责任人:项目负责人、项目开发人员

在项目开始前由项目负责人通知运维同事统一创建项目创建web、安卓、ios、小程序、后台等项目，不接收项目成员零星的项目创建申请。

分支管理：

按照思想进行分支划分，对master、dev、bugfix分支进行保护，开发者在dev或者bugfix分支切出自己的分支进行任务开发，完成后进行merge request操作，具体merge request操作可参考附件文档；

代码合入注释格式为

【功能问题】: 需求名称或jira单号

【提交说明】:说明此次提交修改点

【提交人】: xx

### ~~2.3Sonar检查（当前已不做要求）~~

[http://ip:port/sonar/](http://172.16.1.29:19000/sonar/)，责任人:项目负责人、项目开发人员

进入编码阶段的一周内，通知运维同事对git工程创建sonar检查

总体要求，所有在研项目都必须上git和sonar，新项目bug或漏洞清零，已封板项目在有新的版本时，将bug或漏洞清零

sonar的主要检查项包括findbug/pmd/checkstyle/圈复杂度检查

### 2.4Jekins工程

[http://ip:port/jenkins/](http://172.16.1.22:18081/jenkins/)，责任人:项目负责人

进入编码阶段的一周内，通知运维同事对git工程进行自动化编译打包

**编译打包对工程的要求，包括项目名和版本号等**

在明确项目对应的开发、测试环境后，进行自动化jar、web部署

版本开发过程中，jar包出现变动，必须及时通知运维同事，否则打包会有问题

**要求，必须提前通知运维同事进行sonar、jekins工程配置，例如临时通知要求1小时后必须出包这种工作，可不接收**

### 2.5代码检视工作

项目开发阶段组织代码检视，责任人:项目负责人

要求，梳理项目重点模块，至少对重点模板进行2次代码检视，非重点模块进行1次代码检视(代码检视次数待定)，代码检视情况和修改情况在SVN归档

代码检视工具暂定使用git

### 2.6每日站会

开发和测试阶段都需要进行每日站会，责任人:项目负责人、项目开发人员

要求，每人反馈昨天工作内容，今天计划，当前遇到的风险或问题，每人控制在30s-60s

记录开发成员反馈的风险或问题，由项目负责人进行解决，并通报解决进度，会上不解决问题，只抛出问题，待会后如有需求单独拉会讨论

### 2.7.项目开发周报

项目周报，责任人:项目负责人

通常情况下，一周反馈一次项目进展，项目紧急时，可调整为日报、双日报等，周报内容包含里程碑、本周整体进展、下周整体计划、内部/外部风险问题、详细工作

**周报模板，见附件**

### 2.8.技术总结文档

倡导及时进行技术总结，责任人:项目负责人、项目开发人员

项目开发过程，定位解决问题过程中，多进行总结

### 2.9.开发过程中需求细节明确

开发时，对需求细节的确认，责任人:项目负责人

需求细节，如唯一性限制，长度限制、逻辑关系等，在开发阶段有疑问时，项目负责人统一负责解答，如有必要，可向产品经理进行求证。

# 测试阶段

## 1.提测准入条件

责任人:项目负责人

1.1冒烟测试用例执行通过

在开发提测前一周提供冒烟测试用例，提测前3天提供全部测试用例并评审

~~1.2 Sonar检查问题修复（运维同事线下申请）~~

1.3 Yapi接口文档提供

1.4测试环境部署说明

1.5列举对公共服务、基础组件、外部平台的依赖

便于测试、运维同事对项目基础信息的了解，

包括依赖的服务、版本号

**待和测试沟通提测条件是否需要补充，以及确认提测模板**

## 2.版本测试阶段问题收敛

责任人:项目负责人

版本测试阶段，测试版本通常规划3-4轮，要求必须在3轮内，问题收敛，如果不收敛，及时分析原因，

要求不能出现版本打回情况，避免问题阻塞情况，版本被打回的第一责任人是项目负责人，需要项目负责人进行回溯。

## 3.代码审核后合入

责任人:项目负责人、项目开发人员

第三轮测试开始代码合入必须进行审核，审核定主审核人和备份审核人

代码必须审核才能合入，使用gitlab的merge功能

版本测试阶段的代码合入只能由问题单或需求触发，禁止开发发现问题自行修复不通知测试

## 4.安全扫描和问题修复

责任人:项目负责人、项目开发人员

最迟在第三轮测试中，进行安全扫描申请，问题修改

## 5.UI设计还原问题修复

责任人:项目负责人、项目开发人员

时间：一般第三轮测试开始进行申请；

向UI设计团队申请进行UI还原度测试，输出UI还原度测试报告，并修复

## 6.测试阶段的git管理

责任人:项目负责人

~~每轮提测和~~封版成功都需要在git上创建tag，并说明此次提测的内容时间等；

## 7.Jira问题处理要求

责任人:项目负责人、项目开发人员

要求在测试阶段，阻塞问题半天内解决，严重问题1天内解决；其他问题，2天内解决，无法及时解决的反馈给项目负责人；

Jira问题的处理，要求在备注中详细描述

格式为：

【代码提交】：提交记录

【问题原因】：包括代码原因/需求不对/原始设计不对等的描述

【修改办法】：包括代码修改方案/修改后的逻辑等描述

【影响范围】：分析此问题修改可能影响的其他功能，如果不涉及，填写不涉及

处理流程：负责修复的jira问题处理人在处理完问题单之后进行上述详细描述并将问题单分配给其他开发组内同事进行交叉验证，被分配同事在确定bug及修复情况的情况下在开发环境对问题单进行验证，验证完成后对问题单进行“已解决”操作。

问题单重开：理论上经过交叉验证的bug单不允许被测试团队重新打开，如重新打开需与测试人员进行沟通确认bug原因，如确实是理解问题，则重新分配给模块开发者。

## 8.生产环境申请

责任人:项目负责人/项目经理

生产环境申请中包含机器信息、内网网络要求、外网ip信息等

注意:生产环境网络变更只能在周二或周四，要开通外网映射必须先通过安全渗透测试

# 上线阶段

## 1.封板测试阶段准备上线部署文档

责任人:项目负责人

(机器申请、网络申请、版本包准备、配置文件、数据库脚本、基础组件情况)

如果基础组件较为复杂，还需提供部署指导文档

**上线部署模板，见附件**

## 2.上线过程中的问题定位

责任人:项目负责人、项目开发人员

包括部署过程中的问题、测试验证过程中的问题

## 3.上线操作时间点

责任人:项目负责人

进入封板测试阶段时，要求开始进行生产环境基础组件的部署，同时进行网络策略的开通。

# 项目总结

## 1.项目总结

责任人:项目负责人

项目进度情况、人力投入情况、代码情况、代码检视情况、问题单情况(重点关注阻塞、致命、严重问题)、开发过程中的项目问题、提供文档规范性、版本打回情况、后续改进、前期总结改进情况

根据项目各个阶段的问题进行回顾

项目中的技术总结情况回顾

优秀、失败案例，奖惩措施

# 线上支撑

## 1.线上问题反馈

第一责任人为项目负责人，

影响用户使用的严重问题，在用户要求的时间段内定位解决，最多1天内解决，并输出问题根因报告，

线上问题的修复，必须在封板tag基础上创建branch，并通过merge方式合入代码，并提供代码变动点，功能影响点。

线上问题，测试必须在线上版本复现后，更新修复版本，确保问题解决。

按周/月度汇总现网问题列表及解决情况

**问题根因报告模板，见附件**

## 2.功能增强需求

责任人:项目负责人

功能增强需求反馈给产品经理，由产品经理跟踪

# 遗留问题

1.测试环境-生产环境的云服务器信息的一致，基础组件的配置一致的保证

**待和测试、运维讨论**

2.将sonar检查集成到编译打包的一部分，验证实施方案

**待讨论**

3.后续项目进展跟踪使用JIRA或者禅道，待明确后，再进行对应的调整

4.基础组件、第三方组件的版本管理、安全漏洞管理

**此问题当前无有效的解决方案**

4.开发分支规范

master 分支只保留目前线上版本代码

dev 分支代码为开发环境代码

允许开发人员自建以自己姓名拼音为名称的分支，作为开发者本人的主开发分支。

开发人员主分支必需继承自dev分支，即新建开发人员主分支时使用dev分支作为夫分支

feature 分支：代码为新增特性代码，在需要为现有代码基础上新增功能时新建该分支。

# 其他

## 1.需求变更场景

整体策略

在PRD、UE、UI和系统设计阶段，发生需求变更，在不影响整体计划的基础上，以接受需求变更为主；

进入代码开发阶段后，发生需求变更，在不影响客户诉求的基础上，以将需求延期到下一轮版本为主。

实施方式

与普通的需求类似，需要进行PRD-->UE-->UI-->详细设计等阶段。