Imangine图片分享网站项目管理书

1352845 郭雨卿

1352953 朱嘉业

项目综述

# 项目概要

本项目是一个PC平台的图片素材分享网站，用户通过找到符合自己品味的图片，从而找到与自己品味相投的好友，彼此之间建立社交圈，在圈内分享符合自己品味的图片，进一步形成了各种风格的专辑、小组。本网站主要包括三大模块：

1. 个人主页管理模块：用户可以在个人主页上传图片，并以专辑的形式归类图片，同时以小组长或者管理员的身份管理小组。
2. 社交模块：用户可以对图片、专辑和小组进行收藏、评论与分享；可以关注好友，浏览好友动态，并且针对感兴趣的话题发起讨论小组。
3. 浏览模块：用户可以浏览自己的主页信息和他人的主页信息，这些信息包括该用户拥有的图片、专辑及其参加的小组。可以通过图片的标签关键字，对图片进行搜索。
4. 账户管理模块：用户对账户的注册、登录功能，以及对个人信息的修改。

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 项目代码 | 项目名称 | | | | | 客户 | |
| 20160612 | Imangine图片分享网站 | | | | | Imangine公司 | |
| 项目经理 | **配置经理** | | **业务经理** | **项目副经理** | | 配置副经理 | |
| 朱嘉业 | 朱嘉业 | | 郭雨卿 | 郭雨卿 | | 郭雨卿 | |
| 项目类型 | **平台** | | | | | **项目阶段数** | |
| 新开发 | WinXP及其以上、OS X 10.7 Lion及其以上  CentOS 7(Tomcat + MySQL 5.5.37 + Java) | | | | |  | |
| 项目开始时间 | | | **项目结束时间** | | | **总预算** | |
| 2016/5/1 | | |  | | | 20万 | |
| 项目与客户联系人 | | | | | | | |
| 姓名 | | **手机号** | | | **QQ** | | **邮箱** |
| 郭雨卿 | | 13701764276 | | | 497434004 | | Charlottie77@163.com |
| 项目边界 | | | | | | | |
| 一、 项目需求  - 个人主页管理模块  该模块的功能均为用户登录后方可使用的功能。  用户可以在个人主页上传图片，并以专辑的形式归类图片，同时以小组长或者管理员的身份管理小组。   1. 用户从本地上传图片，并填写图片的标题与说明，为图片添加标签。图片的标题不可重复。 2. 用户可以新建一个专辑，并填写专辑的标题、说明和标签，同时必须上传一张图片作为专辑封面。上传封面部分同第一项。 3. 该用户所拥有的图片、专辑以及其建立的小组，均在用户的个人主页上予以展示，用户可以对其进行管理： 4. 用户可以对图片进行管理。用户可以对某一张图片的信息进行修改或者删除图片。 5. 用户可以对专辑进行管理。用户可以对某一专辑的信息进行修改，或者删除该专辑（专辑下的图片同样被删除）。用户也可以对专辑中的图片的信息进行修改或者将该图片删除（即移出专辑），该部分参见a项。 6. 用户可以对小组进行管理。当用户是小组建立者时，用户可以对某一小组的信息进行修改，或者解散该小组（该小组下的专辑和图片同样被删除），可以对该小组的用户进行管理（如添加、删除成员（删除成员后，该成员在小组中上传的图片、建立的专辑以及其相关的评论都会被删除））。用户也可以对小组中的专辑和图片进行管理，该部分参见a、b项。   - 社交模块  1. 用户可以通过点击图片、专辑对应的心形按钮，对图片、专辑进行收藏，系统将该图片或者专辑加入用户的“收藏”图片列表中，心形按钮变为实心。用户若点击小组对应的心形按钮，则表示申请加入该小组，心形按钮变为实心。用户再次点击实心的心形按钮，表明取消收藏图片或专辑、退出小组，心形按钮变为空心。  2. 用户可以在图片、专辑、小组的详细信息页面下方的评论栏中，输入合法的评论并提交。  3. 用户可以在图片、专辑、小组的详细信息页面点击分享按钮，将其分享到第三方平台。  4. 用户可以在他人主页，点击用户相应的心形按钮，来关注该用户。  5. 用户可以发起一个小组，填写小组的基本信息，并上传一张图片作为小组封面，建立属于自己的小组。  - 浏览模块  1. 用户可以浏览最新上传的图片列表、关注的好友最新上传的图片列表以及系统每日最热门的图片推荐列表。  2. 用户可以浏览最新建立的专辑列表、关注的好友最新建立的专辑列表以及系统每日最热门的专辑推荐列表。  3. 用户可以浏览最新建立的小组列表、关注的好友最新建立的小组列表以及系统每日最热门的小组推荐列表。  4. 用户可以浏览某一张图片的详细信息，包括：适宜分辨率的大图、图片名称、描述、上传日期、收藏数、收藏者头像列表、评论数、评论内容。  5. 用户可以浏览某一专辑的详细信息，包括：专辑名称、专辑描述、专辑所属标签、专辑创建时间、专辑收录的图片、图片名称缩略、专辑封面的适宜分辨率大图。用户可以点击其中图片进入图片详细信息页面，其内容参照第4项。  6.用户可以浏览某一小组的详细信息，包括：小组名称、小组所属标签、小组创建时间、小组的专辑的封面图片、专辑名称、小组人数、部分成员头像列表（点击进入可浏览全部成员列表）、小组互动数、小组互动内容。用户可以点击其中专辑进入专辑详细信息页面，其内容参照第5项。  - 账户管理模块  1. 用户可以输入已有的用户名和密码登录自己的账号。若用户连续输错密码三次，则在2小时内不能再次登录。  2. 用户可以输入未被使用过的用户名和密码，并同意网站协议来进行账号注册。  二、非功能性需求  非功能性需求？？安全性 软件质量-用户界面易用性  三、项目接口  - 用户接口  即PC端的网页前端的界面部分。  - 外部接口  第三方分享平台接口  - 内部接口  黑人？？？？  四、其他说明  2. 用户的主页包含以下部分：  1. 图片  a. 该用户上传的图片，显示为My Pictures列表。  b. 该用户收藏的图片，显示为Favorite Pictures列表。  2. 专辑  a. 该用户建立的专辑，显示为My Albums列表。  b. 该用户收藏的专辑，显示为Favorite Albums列表。  3. 小组  a. 该用户建立的小组（即用户为小组组长），显示为My Groups列表。  b. 该用户加入的小组，显示为Favorite Groups列表。 | | | | | | | |
| 项目增值服务 | | | | | | | |
| 微信H5宣传小应用  文档网站 | | | | | | | |
| 公司目标 | | | | | | | |
| 建立一个清新简洁风格的图片分享网站，以简洁、开放、丰富 、品味为主要特征的图片分享与社交相融合的平台。 | | | | | | | |

# 项目成果

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 里程碑日期 | 里程碑 | 交付物 |
| 1 | 2016-7-1 | 立项 |  |
| 2 | 2016-7-15 | 可行性研究完成 | 可行性报告 |
| 3 | 2016-8-1 | 需求分析完成 | 需求规约 |
| 4 | 2016-8-15 | 计划阶段完成 | 项目企划书（采购计划、开发计划、测试计划、人力计划） |
| 5 | 2016-9-1 | 培训完成 | 培训报告 |
| 6 | 2016-10-1 | 第一次迭代完成 | 该迭代概要设计书、详细设计书、测试计划书、测试报告、增量交付 |
| 7 | 2016-11-7 | 第二次迭代完成 | 该迭代概要设计书、详细设计书、测试计划书、测试报告、增量交付 |
| 8 | 2016-12-7 | 第三次迭代完成 | 该迭代概要设计书、详细设计书、测试计划书、测试报告、增量交付 |
| 9 | 2017-1-5 | 整理合并工作完成 | 会议报告、回归测试计划、回归测试报告 |
| 10 | 2017-1-20 | 项目后期工作完成 | 项目总结报告 |

# 项目假设

1. 项目中用户上传的图片，如有侵权行为，均为用户个人行为，与Imangine图片分享网站无关。

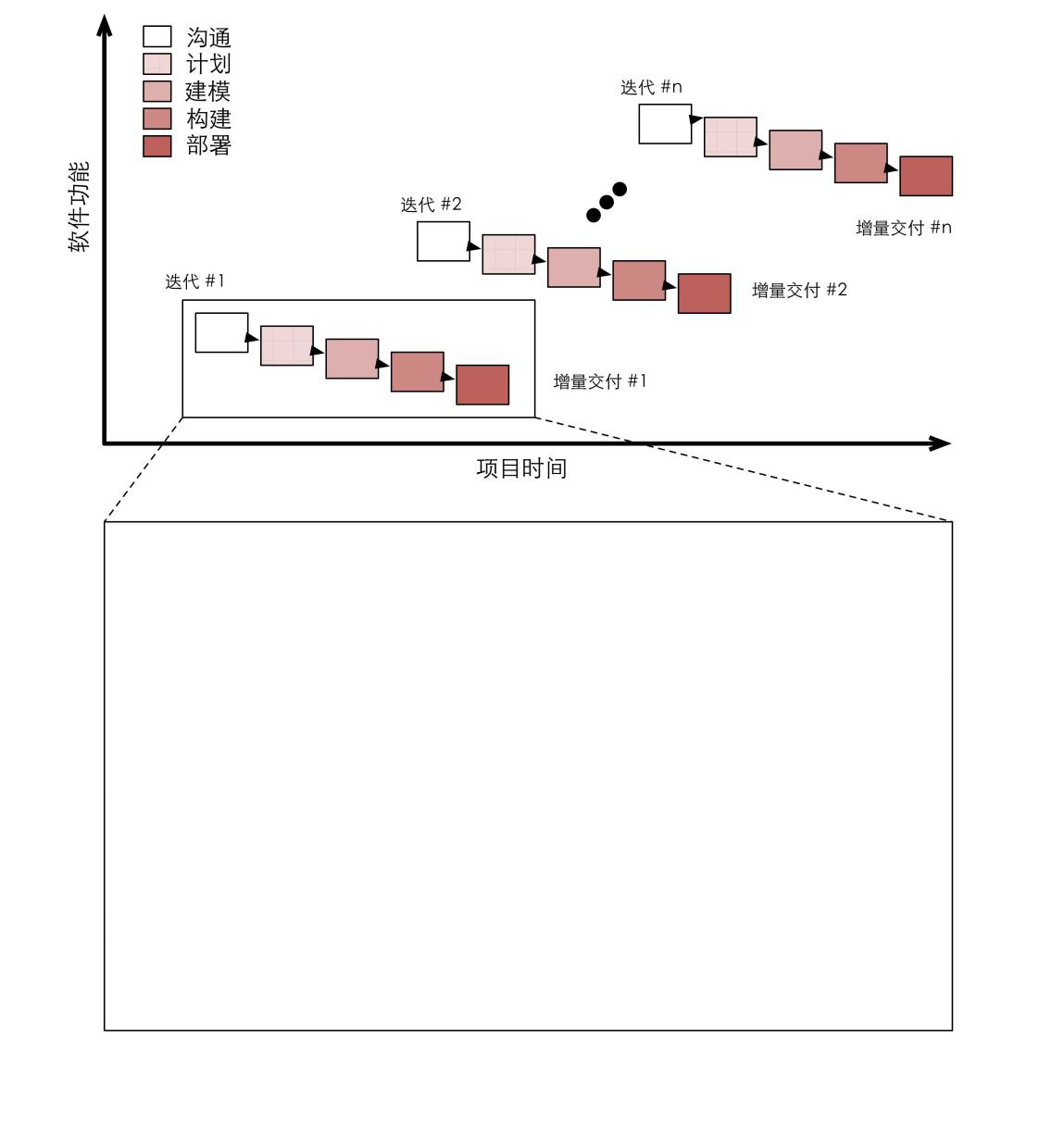
2. 由于技术上或是需求上造成的变更，或多或少会影响项目进度，这些影响将会在下一个增量中加入，并与甲方协商确认。

项目计划

# 项目过程

## 过程模型

本项目采用增量模型。



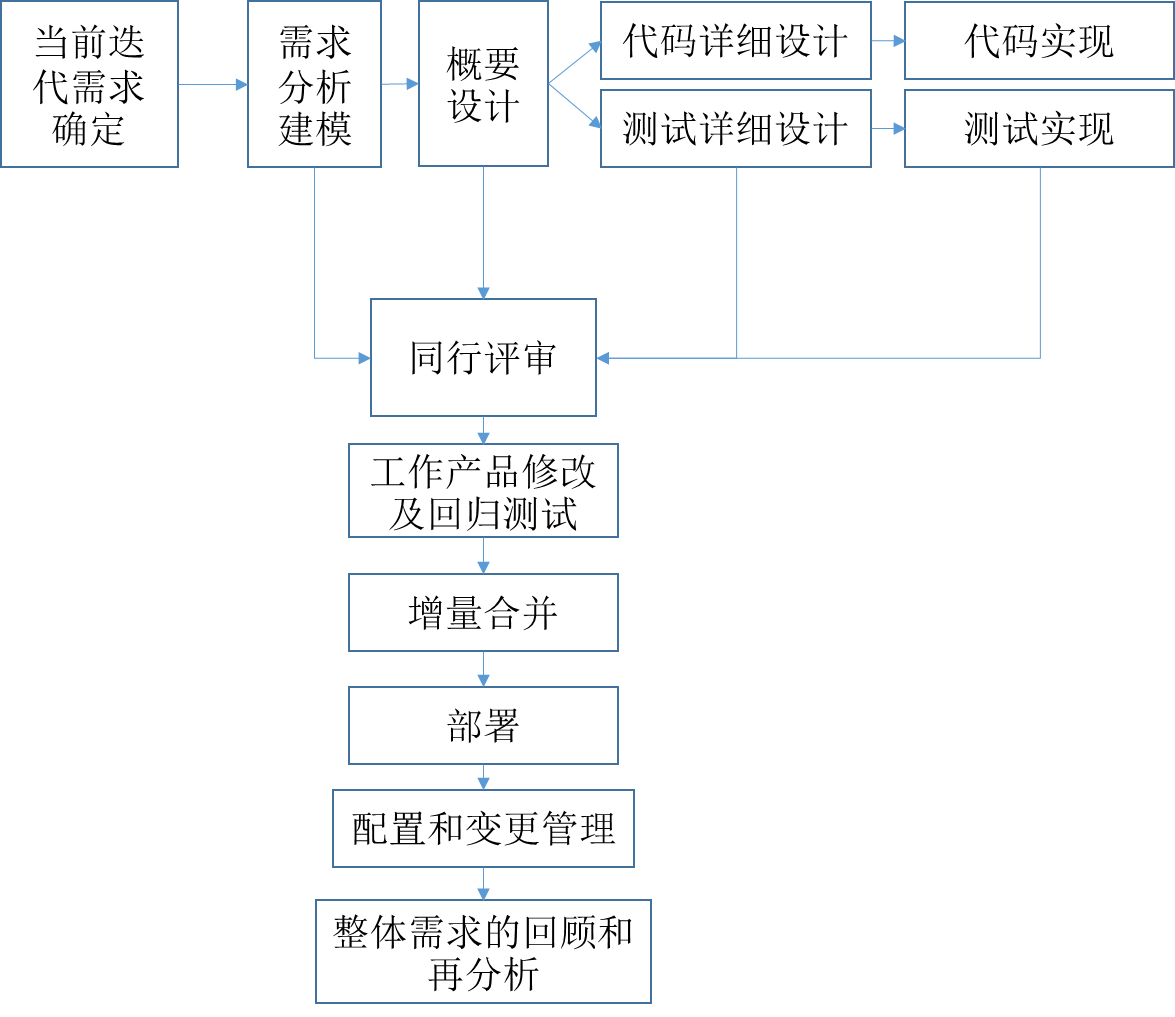
选择理由是：Imangine图片分享网站的需求明确并且有相似的项目以供参考，项目平台为稳定的PC浏览器端。因此，在稳定的框架和平台上，来开发和增加具体的业务功能，是一个较好的选择。此外，由于项目的需求具有各个模块相对独立、层次结构清晰的特点，可以方便合理的进行拆卸，从而设计增量包，对系统的可维护性是一个极大的提高。整个系统是由一个个构件集成在一起的，当需求变更时只变更部分部件，而不必影响整个系统。

将项目分为三个阶段：

1. 第一阶段是整体的需求分析与前期准备工作（包括培训）。

2. 第二阶段是增量阶段，包括一个个增量的迭代和合并。

每一个增量包的流程图如下：



3. 第三阶段是最后的交付阶段，包括对整个项目的总结的Closure阶段。

## 裁剪要点

|  |  |  |
| --- | --- | --- |
| 与标准模式的差异 | 增加\修改\删除 | 理由 |
| 将任务划分为更多独立的小任务 | 修改 | 由于开发团队成员的开发技术经验不一，所以需要降低每一个任务的难度 |
| 对每个开发人员最初提供的工作产品（文档和代码）进行同行评审 | 修改 | 由于开发团队成员的开发技术经验不一，多设置节点进行走查。 |
| 在安排进度时应考虑开发人员学习掌握技术的时间 | 增加 | 由于开发团队成员的开发技术经验不一，有些成员可能需要一定的时间对相关技术进行学习。 |
| 适当减少开发基线中需要配置的配置项 | 减少 | 由于项目规模不大并且其所在的平台较为简单，所以有些配置是不需要的，比如安装手册、可执行程序及其链接模块、用户手册等。 |

## 需求变更管理过程

|  |
| --- |
| 需求变更跟踪 |
| 1. 客户要求的需求变更必须被登记记录在ChangeRequest.xls中。登记条目有：变更编号、变更简述、变更影响、变更状态和关键日期。其中变更状态参照需求变更生命周期。 |
| 2. 被登记的需求变更都需要进行影响分析。主要考虑原则有：变更所涉及的工作量、对工作产品交付承诺的影响、对成本和交付质量的影响、对整体日程和工作量的潜在影响等。 |
| 3. 估算修改变更量所需的工作量 |
| 4. 重新估算交付日程表 |
| 5. 计算累积变更并分析影响 |
| 6. 检查是否超过阈值。若工作量、日程、累积变更等超过可容阈值，需上报高级经理 |
| 7. 获得客户批准 |
| 8. 进入返工 |

## 需求变更生命周期

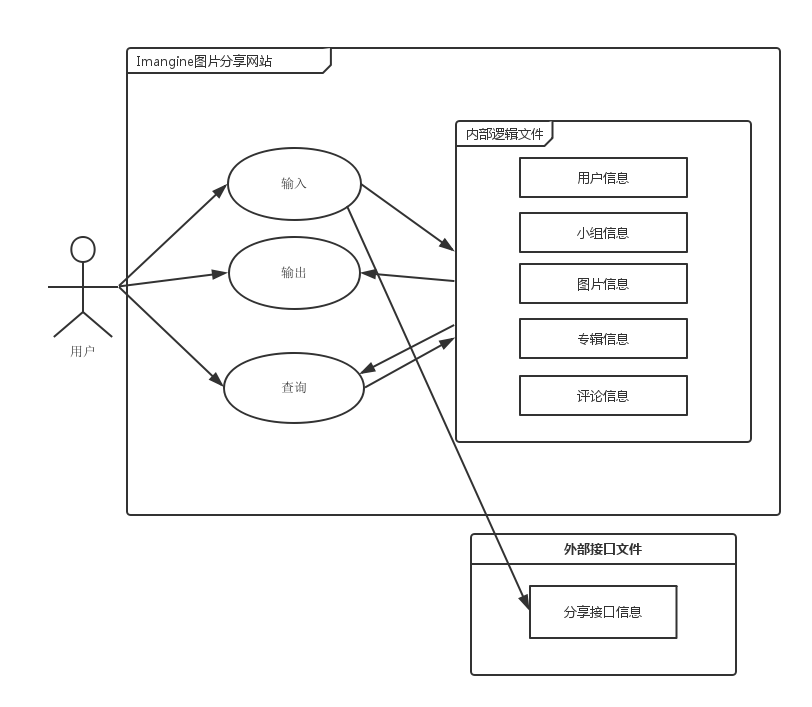
|  |  |
| --- | --- |
| 状态 | 说明 |
| 已登记 | 该需求变更已经登记在ChangeRequests.xls |
| 已批准 | 需求变更分析结果得到高级经理和客户的批准 |
| 返工中 | 得到批准后，按照项目经理制定的计划返工 |
| 验证中 | 返工完成后，需要验证各类规约是否修改一致 |
| 测试中 | 验证完成后，需要额外对代码进行测试 |
| 完结 | 测试通过 |
| 挂起 | 测试不通过，需进一步检查 |
| Note：状态之间可以流水跳转，也可以随机跳转。 | |

## 需求变更可跟踪性

利用需求管理工具RequisitePro进行跟踪

# 估算

## 系统抽象



## 功能点估算

相关数据如下：

|  |
| --- |
| 用户表 |
| User\_id  User\_name  Password  Avatar  Email  City  Birthday  Description |
| 好友表 |
| Friend\_id |
| Token表 |
| Token  expirattion |

|  |
| --- |
| 小组表 |
| Group\_id  User  Name  Member\_num  Set\_data  Cover\_path  Description |
| 小组专辑列表 |
| Album\_id |
| 小组成员列表 |
| User\_id |

|  |
| --- |
| 图片表 |
| Pic\_id  User  Path  Upload\_date  Title  Description |
| 图片收藏者表 |
| User\_id |
| 图片标签表 |
| tag |

|  |
| --- |
| 专辑表 |
| Album\_id  User  Pic\_num  Set\_date  Description  Theme  Cover\_path |
| 专辑收藏者列表 |
| User\_id |
| 专辑图片列表 |
| Pic\_id |

|  |
| --- |
| 评论表 |
| Comment\_id  User  Comment\_date  content |
| 图片评论列表 |
| Pictures\_id |
| 小组评论列表 |
| Group\_id |

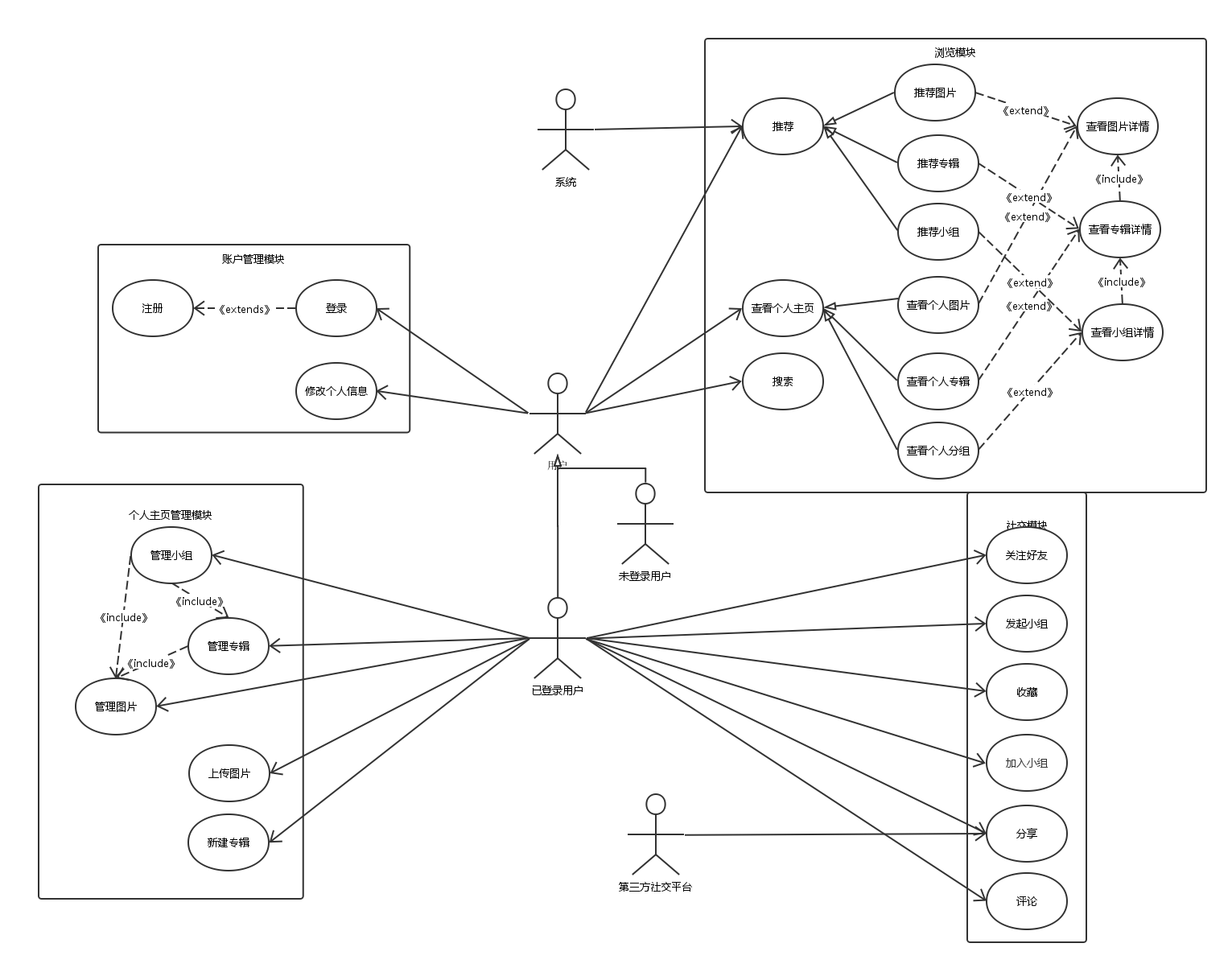
|  |
| --- |
| 分享接口信息 |
| API\_id  share\_intro (标题信息)  share\_url  share\_appkey  share\_pic  share\_site |

识别出的ILF和EIF功能点个数如下表所示：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ILF内部逻辑文件 | RET | DET | 复杂度 | 未调整FP个数 |
| 用户表 | 3 | 11 | 低 | 7 |
| 小组表 | 3 | 9 | 低 | 7 |
| 图片表 | 3 | 8 | 低 | 7 |
| 专辑表 | 3 | 9 | 低 | 7 |
| 评论表 | 3 | 6 | 低 | 7 |
| EIF外部接口文件 | **RET** | **DET** | **复杂度** | **未调整FP个数** |
| 分享接口信息表 | 1 | 6 | 低 | 5 |

因此，DFP为 DFP = ILF + EIF = 40个。

Imangine图片分享网站的用例如下：



识别出来的EI, EQ和EO个数如下表所示。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| EI | FTR | DET | 复杂度 | 未调整FP个数 |
| 注册-添加用户信息 | 1 | 8 | 低 | 3 |
| 登录 | 2 | 3 | 低 | 3 |
| 修改个人信息 | 1 | 8 | 低 | 3 |
| 上传图片 | 1 | 6 | 低 | 3 |
| 新建专辑 | 1 | 7 | 低 | 3 |
| 管理图片-修改图片信息 | 1 | 6 | 低 | 3 |
| 管理图片-删除图片 | 1 | 1 | 低 | 3 |
| 管理专辑-修改专辑信息 | 1 | 7 | 低 | 3 |
| 管理专辑-删除专辑 | 1 | 1 | 低 | 3 |
| 管理专辑-删除专辑图片 | 2 | 2 | 低 | 3 |
| 管理专辑-移动专辑图片 | 2 | 3 | 低 | 3 |
| 管理小组-修改小组信息 | 1 | 7 | 低 | 3 |
| 管理小组-解散小组 | 1 | 1 | 低 | 3 |
| 管理小组-添加用户 | 2 | 2 | 低 | 3 |
| 管理小组-删除用户 | 2 | 2 | 低 | 3 |
| 关注好友 | 2 | 2 | 低 | 3 |
| 发起小组 | 2 | 7 | 低 | 3 |
| 收藏图片 | 3 | 2 | 中 | 4 |
| 收藏专辑 | 3 | 2 | 中 | 4 |
| 加入小组 | 3 | 2 | 中 | 4 |
| 评论图片 | 3 | 6 | 中 | 4 |
| 评论专辑 | 3 | 6 | 中 | 4 |
| 小组发言 | 3 | 5 | 中 | 4 |
| 共计：75 | | | | |
| EQ | **FTR** | **DET** | **复杂度** | **未调整FP个数** |
| 查询用户详细信息 | 1 | 8 | 低 | 3 |
| 查询专辑详细信息 | 1 | 7 | 低 | 3 |
| 查询图片详细信息 | 1 | 6 | 低 | 3 |
| 查询小组详细信息 | 1 | 7 | 低 | 3 |
| 查询用户图片列表 | 2 | 2 | 低 | 3 |
| 查询用户专辑列表 | 2 | 2 | 低 | 3 |
| 查询用户小组列表 | 2 | 2 | 低 | 3 |
| 查询专辑图片列表 | 2 | 2 | 低 | 3 |
| 查询小组专辑列表 | 2 | 2 | 低 | 3 |
| 查询小组成员列表 | 2 | 2 | 低 | 3 |
| 查询网站图片列表 | 1 | 5 | 低 | 3 |
| 查询网站专辑列表 | 1 | 5 | 低 | 3 |
| 查询网站小组列表 | 1 | 5 | 低 | 3 |
| 共计：12 | | | | |
| EO | **FTR** | **DET** | **复杂度** | **未调整FP个数** |
| 分享图片 | 1 | 6 | 低 | 4 |
| 查询用户详细信息 | 1 | 8 | 低 | 4 |
| 查询专辑详细信息 | 1 | 7 | 低 | 4 |
| 查询图片详细信息 | 1 | 6 | 低 | 4 |
| 查询小组详细信息 | 1 | 7 | 低 | 4 |
| 查询用户图片列表 | 2 | 2 | 低 | 4 |
| 查询用户专辑列表 | 2 | 2 | 低 | 4 |
| 查询用户小组列表 | 2 | 2 | 低 | 4 |
| 查询专辑图片列表 | 2 | 2 | 低 | 4 |
| 查询小组专辑列表 | 2 | 2 | 低 | 4 |
| 查询小组成员列表 | 2 | 2 | 低 | 4 |
| 查询网站图片列表 | 1 | 5 | 低 | 4 |
| 查询网站专辑列表 | 1 | 5 | 低 | 4 |
| 查询网站小组列表 | 1 | 5 | 低 | 4 |
| 查询网站用户列表 | 1 | 3 | 低 | 4 |
| 共计：40 | | | | |

由此，事务功能点TFP = EI + EO + EQ = 75 + 12 + 40 = 127

所以，未调整功能点UFP = TFP + DFP = 127 + 40 = 167

进一步考虑，本系统的通用系统特性及其影响程度如下表所示：

|  |  |
| --- | --- |
| 系统特性 | 分数 |
| 数据通讯 | 3 |
| 分布式数据处理 | 2 |
| 性能 | 3 |
| 大业务量配置 | 2 |
| 事务处理率 | 3 |
| 在线数据输入 | 5 |
| 最终用户效率 | 4 |
| 在线更新 | 4 |
| 复杂处理 | 1 |
| 可复用性 | 1 |
| 易安装性 | 0 |
| 易操作性 | 1 |
| 多场地 | 0 |
| 支持变更 | 0 |
| 合计：29 | |

根据国际IFPUG组织的软件项目分类：新开发项目、二次开发项目、功能增强项目，确定Imangine图片分享网站为新开发项目且不需要与已有系统进行整合，故有估算公式如下：

VAF = ( sum of ( DI ) \* 0.01 ) + 0.65 = 29\*0.01 + 0.65 = 0.94

AFP = UFP \* VAF

由此，调整后功能点AFP = UFP \* 0.94 = 175 \* 0.94 = 157

根据日本IPA SEC组织提供的工作量估算公式：

假设每日工作8小时，Effort = e^0.542 \* FP^1.154 = 588人-天

## 用例估算

用例分类依据如下表：

|  |  |  |
| --- | --- | --- |
| 用例类型 | 描述 | 因子 |
| 简单 | 0-3个事务 | 5 |
| 中等 | 4-7个事务 | 10 |
| 复杂 | >7个事务 | 15 |

本项目中的用例列表如下：

|  |  |  |  |
| --- | --- | --- | --- |
| 用例编号 | 用例名称 | 包含事务个数 | 复杂度 |
| 1 | 登录 | 4 | 中等 |
| 2 | 注册 | 8 | 复杂 |
| 3 | 修改个人信息 | 3 | 简单 |
| 4 | 上传图片 | 4 | 中等 |
| 5 | 新建专辑 | 5 | 中等 |
| 6 | 管理图片 | 6 | 中等 |
| 7 | 管理专辑 | 22 | 复杂 |
| 8 | 管理小组 | 50 | 复杂 |
| 9 | 查看个人主页 | 9 | 复杂 |
|  | 查看个人图片 | 4 | 中等 |
|  | 查看个人专辑 | 14 | 复杂 |
|  | 查看个人分组 | 24 | 复杂 |
|  | 查看图片详情 | 4 | 中等 |
|  | 查看专辑详情 | 5 | 中等 |
|  | 查看小组详情 | 5 | 中等 |
| 10 | 搜索 | 5 | 中等 |
| 11 | 关注好友 | 2 | 简单 |
| 12 | 发起小组 | 5 | 中等 |
| 13 | 收藏 | 4 | 中等 |
| 14 | 加入小组 | 2 | 简单 |
| 15 | 分享 | 4 | 中等 |
| 16 | 评论/发言 | 9 | 复杂 |

|  |  |  |  |
| --- | --- | --- | --- |
| 估算工作量 | | | |
| 用例 | 工作量（基于以往项目） | 个数 | 总工作量（人天） |
| 简单用例 | 1 人-天 | 3 | 3 |
| 中等用例 | 5 人-天 | 12 | 60 |
| 复杂用例 | 8 人-天 | 7 | 56 |
| 共计 |  |  | 119 |

计算未调整用例点 UUCP = 3 \* 5 + 12 \* 10 + 7 \* 15 = 240

考虑技术复杂因子TCP如下表，得分为5表示对项目很重要，0表示非项目要求

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 技术复杂度因子 | 权重 | 得分 |
| 1 | Distributed system | 2 | 2 |
| 2 | Response or throughput performance objectives | 1 | 5 |
| 3 | End-user efficiency(online) | 1 | 5 |
| 4 | Complex internal processing | 1 | 2 |
| 5 | Code must be reusable | 1 | 4 |
| 6 | Easy to install | 0.5 | 0 |
| 7 | Easy to use | 0.5 | 4 |
| 8 | Portable | 2 | 5 |
| 9 | Easy to change | 1 | 3 |
| 10 | Concurrent | 1 | 5 |
| 11 | Includes special security features | 1 | 3 |
| 12 | Provides direct access for third parties | 1 | 0 |
| 13 | Special user training facilities required | 1 | 0 |

根据权重计算：

TFactor = 2\*2+5+5+2+4+0.5\*4+2\*5+3+5+3 = 43

TCP = 0.6 + (0.01 \* TFactor) = 1.03

进一步考虑环境调整因子EF如下表：

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 环境调整因子 | 权重 | 得分 |
| 1 | Familiar with internet process | 1.5 | 2 |
| 2 | Application experience | 0.5 | 2 |
| 3 | Object-oriented experience | 1 | 2 |
| 4 | Lead analyst capability | 0.5 | 0 |
| 5 | Motivation | 1 | 2 |
| 6 | Stable requirements | 2 | 3 |
| 7 | Part-time workers | -1 | 0 |
| 8 | Difficult programming language | -1 | 0 |

根据权重计算EFactor = 1.5\*2+0.5\*2+2+2+2\*3 = 14

EF = 1.4 + ( -0.03 \* EFactor ) = 0.97

最后，根据技术复杂度和环境因素：UCP = UUCP \* TCP \*EF = 115\*1.03\*0.97 = 239.8

根据标准工作量1个UCP对应20人-小时，得到工作量为4795人-小时。

假设每日工作时间为8小时，即需要599人-天。

## 综合估算结果

综合FP估算和Use Case两种估算方法得到的工作量估计，求其平均得到工作量为（588+599）/2 = 593.5人-天（假设每日工作时间为8小时 ）

然后由Process Database中相似项目信息得到：

|  |  |  |
| --- | --- | --- |
| Phase-wise Effort Estimate | | |
| Activity/Phase | 人-天 | 占总工作量比例 |
| Requirements | 59 | 10 |
| Design | 71 | 12 |
| Build | 173 | 29 |
| Integration testing | 42 | 7 |
| Regression testing | 12 | 2 |
| Acceptance testing | 36 | 6 |
| Project management | 89 | 15 |
| Configuration management | 18 | 3 |
| Training | 59 | 10 |
| Others | 35 | 6 |
| Estimated effort | 594 | 100% |

|  |  |  |
| --- | --- | --- |
| Effort Estimate By Iterations | Person-Days | %Of Total Effort |
| Project initiation | 30 | 5 |
| Inception phase | 30 | 5 |
| Elaboration phase: iteration 1 | 53 | 9 |
| Elabrotion phase: iteration 2 | 42 | 7 |
| Construction phase: iteration 1 | 30 | 5 |
| Construction phase: iteration 2 | 30 | 5 |
| Construction phase: iteration 3 | 24 | 4 |
| Transition phase | 131 | 22 |
| Project closure | 11 | 2 |
| Project management | 89 | 15 |
| Configuration management | 17 | 3 |
| Traning | 60 | 10 |
| Others | 47 | 8 |
| Total estimated effort | 594 | 100% |

# 时刻表

详见附件：Imangine Project.mpp

# 人员

|  |  |  |
| --- | --- | --- |
| 按角色 | | |
| 角色 | 要求人数 | 日期 |
| 项目经理 | 1 | 2016-7-1 至 2017-1-20 |
| 开发者及测试人员 | 4 | 2016-7-1 至 2017-1-20 |
| DBA | 1 | 2016-7-1 至 2017-1-20 |
| 技术经理 | 1 | 2016-7-1 至 2017-1-20 |
| 总计 | 7 | 2016-7-1 至 2017-1-20 |

|  |  |  |  |
| --- | --- | --- | --- |
| 按技能和经验 | | | |
| 领域 | 总数 | 经验（1-12月）人数 | 经验（>12月）人数 |
| 前端开发 | 1 | 1 | 0 |
| Java开发 | 3 | 2 | 1 |
| 数据库管理及优化 | 1 | 1 | 0 |
| 配置管理与项目管理 | 1 | 1 | 0 |
| 技术经理 | 1 | 0 | 1 |
| 人员需求计划 | | | |
| 月份 | Offshore | Onsite | 总数 |
| 2016-7-1 至 2017-1-20 | 0 | 4 | 4 |

# 开发环境

|  |  |
| --- | --- |
| 硬件 | 软件 |
| Macbook/PC | Mac OS X , Win 8.1, IntelliJ IDEA |
| 高配置服务器 | Apache， MySQL |

# 硬件与软件资源需求

|  |  |
| --- | --- |
| 描述 | 所需数量 |
| Macbook with 256 RAM/PC | 7 |
| 16GB space on server | 1 |
| IntelliJ IDEA | 4 |
| MySQL | 4 |
| Requisite Pro | 1 |
| Microsoft Project 2010 | 1 |
| Bugzilla | 4 |

# 工具

UML建模工具：ProcessOn

配置管理工具：Github

需求管理工具：Requisite Pro

缺陷跟踪工具：Bugzilla

项目管理工具：Microsoft Project 2013

集成测试服务工具：Xcode Server， GitHub

测试结果分析工具：XcodeCoverage

测试工具：XCTest测试套件

# 培训

|  |  |  |
| --- | --- | --- |
| 培训领域 | 时长 | 免去条件 |
| 技术领域 | | |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
| 过程相关 | | |
|  |  |  |
|  |  |  |
|  |  |  |

# 质量计划

## 质量目标

|  |  |  |
| --- | --- | --- |
| 项目质量目标 | | |
| 目标 | 值 | 目标设置基础 |
| 产生的缺陷总数 | 180 | 0.038缺陷每人时。 |
| 质量（接受缺陷密度） | 9 | 预估检测到缺陷数的5%或者更少 |
| 生产力 | 15功能点人/月 | 设置平均的缺陷移除率为93% |
| 时刻表 | 按时交付 | 85%的项目可以按时交付，所有项目都应以按时交付为目标。 |
| 质量成本 | 35% | 根据类似项目有所提高 |

## 检测缺陷评估

|  |  |  |  |
| --- | --- | --- | --- |
| 审核/测试阶段 | 被检测的缺陷预估 | 百分比 | 预估基础 |
| 需求和设计审核 | 36 | 20% | 类似项目Synergy和PCB |
| 代码走查 | 36 | 20% | 类似项目Synergy和PCB |
| 单元测试 | 72 | 40% | 类似项目Synergy和PCB |
| 集成测试和回归测试 | 41 | 17% | 类似项目Synergy和PCB |
| 验收测试 | 5 | 3% | 类似项目Synergy和PCB |
| 总计 | 132 | 100% | 类似项目Synergy和PCB |

## 达到质量目标的策略

|  |  |
| --- | --- |
| 策略 | 期望 |
| 按时间进行同行评审，而不是等到某一工作产品完成之后才进行同行评审。 | 由于开发人员整体经验不足，所以采取此策略，尽早预防缺陷。 |
| 结合Github做代码提交申请和代码审核，直至审核通过才合并代码。 | 有规范的代码提交记录，保证每一次代码合并都产生完整的可执行项目。 |
| 按照增量模型开发的思路在每一个增量结束后进行总结和审核。 | 高效执行每一次迭代，有效计划下一次迭代。 |
| 每一个迭代结束后，要对整体需求进行回顾和再分析 | 保证整体需求不偏差，每一次迭代的产品都要尽量符合最初需求制定；如果有偏差，就应该在下一次迭代中进行调整和修复。 |

## 审核

|  |  |  |
| --- | --- | --- |
| 审核点 | 审核对象 | 审核类型 |
| 整体需求分析阶段结束时 | 项目需求分析计划  项目时刻表 | 团队审核  SQA审核 |
| 每个增量的需求确定时 | 项目需求规约  用例条目  界面框架 | 团队审核 |
| 每个增量的需求分析建模完成时 | 项目需求分析规约 | 团队审核 |
| 每个增量的概要设计完成时 | 项目概要设计规约 | 团队审核 |
| 每个增量的代码详细设计完成时 | 项目详细设计规约-代码部分 | 团队审核 |
| 每个增量的测试详细设计完成时 | 项目详细设计规约-测试部分 | 团队审核 |
| 每个增量按照模块伪代码完成时 | 伪代码 | 单人审核 |
| 每个增量按照模块代码完成时 | 代码 | 单人审核 |
| 每个增量测试代码完成时 | 测试代码 | 单人审核 |
| 每个增量单元测试结束时 | 测试结果和代码 | 单人审核 |
| 每个增量修改和回归测试结束时 | 测试报告、结果和代码 | 团队审核 |
| 每个增量部署交付结束时 | 交付报告 | 团队审核 |
| 整体项目部署交付结束时 | 交付报告 | 团队审核 |

# 风险管理

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 风险描述 | 可能性 | 影响度 | 风险暴露指数 | 风险缓解计划 |
| 1 | 团队中成员开发经验较少，可能会遇到未知的问题 | 0.7 | 9 | 6.3 | 寻找市场上的类似产品，学习竞品的运作方式与业务逻辑；咨询专业的社交网站运营人员。 |
| 2 | 开发人员缺乏经验，遇到技术难题会导致进度变慢 | 0.9 | 5 | 4.5 | 制定针对性的培训计划；聘请技术顾问；合理调节项目进度表。 |
| 3 | 没有独立的测试团队，测试工作全部由开发人员完成 | 0.7 | 4 | 2.8 | 招收新的团队成员，并指定专人进行软件测试工作；或者将软件测试外包给第三方进行。 |
| 4 | 因为人手有限，管理人员也需要参与一部分开发工作,导致工作量增大，进度变慢。 | 0.5 | 3 | 1.5 | 合理安排工作量，减少不必要的工作流程等。 |
| 5 | 项目涉及许多有关图片的操作，可能会出现数据量较大的情况，可能需要做特别处理，而开发人员无相关经验 | 0.6 | 7 | 3.5 | 学习相关技术知识，或者找技术人员进行交流 |

项目跟踪

# 测量计划

|  |  |  |
| --- | --- | --- |
| 需要手机的数据 | 数据单位 | 所用工具 |
| 规模 | LOC, FP | XcodeCoverage |
| 人力 | 人天 | MS Project |
| 缺陷 | 个数 | Bugzilla |
| 时刻表 | 时长 | MS Project |

# 任务追踪

|  |  |
| --- | --- |
| 活动 | 过程 |
| 任务时间计划 | 用MS Project做时间计划，并且在必要时调整时间计划来适应最新情况 |
| 任务布置 | MS Project软件导出任务提醒文件，团队成员可以自行下载 |
| 任务状态追踪 | 每天都进行任务状态更新 |
| 项目会议 | 每2天进行一次站会，每周组会。可以根据进度进行同行评审。 |
| 详细分析会议 | 每次迭代结束进行详细分析会议。 |

# 问题追踪

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 问题类型 | 日志位置 | 日志创建许可人 | 审核人安排/审核时间 | 处理时长 |
| 线上问题 | GitHub issue | 任何人 | 由项目经理指定修改，每日更新 | 2天 |
| 开发技术问题 | 技术文档 | 任何人 | 由项目经理指定修改，每日更新 | 1-2天 |
| 用户反馈 | UserFeedback.xls | 业务人员 | 由项目经理指定修改，每日更新 | 7天 |
| 业务管理问题 | 周报 | 业务经理 | 由项目经理指定修改，每日更新 | 4天 |
| 服务支持问题 | Request Tracker | 任何人 | 由项目经理指定修改，每日更新 | 2天 |

# 用户反馈

|  |  |
| --- | --- |
| 事项 | 日志和追踪过程 |
| 用户反馈 | 业务经理每周统计用户反馈信息 |
| 客户抱怨 | 业务经理及时答复客户，处理事件 |

# 质量追踪

|  |  |
| --- | --- |
| 质量活动 | 行动 |
| 缺陷追踪 | 用Github Issue记录并跟踪缺陷 |
| 审核（需求审核，概要设计审核，详细设计审核，代码审核） | 依据质量计划进行检验，按时召开同行评审会议，以走查的方式，对伪代码，代码，文档等内容进行审核。 |
| 独立单元测试 | 生成测试报告并交由开发者做代码重构 |
| 连续集成测试 | 测试不通过自动发布报告 |

# 需要高级管理员审核的内容

|  |  |  |
| --- | --- | --- |
| 序列号 | 审核事项 | 审核频率 |
| 1 | 时刻表 | 版本更新时审核 |
| 2 | 项目计划 | 重大计划变动时审核 |
| 3 | 里程碑报告 | 每个里程碑时间点 |

# 状态报告

|  |  |
| --- | --- |
| 报告对象 | 频率 |
| 业务经理 | 每周一邮件报告 |
| 项目经理 | 每日例会报告 |

# 里程统计偏移

|  |  |  |
| --- | --- | --- |
| 实际与预估比较 | 前五个历程 | 其余里程 |
| 人力投入 | 10% | 5% |
| 时刻表 | 10% | 5% |
| 缺陷数 | 20% | 20% |

# 客户报告

开发团队即为需求提出者，因此无需进行开发进度汇报，及时按照进程，对需求进行总结回顾即可。

# 业务经理报告

客户反馈

里程碑和周报告

问题报告

需求改动

计划重大改变

项目团队

参考

Addison Wesley. Software Project Management in Practice, 2003

Imagine 项目团队. Imagine 需求规约(说明书), 2016

Imangine 项目团队. Imagine 需求分析规约(说明书), 2016

Imangine 项目团队. Imagine 概要设计规约(说明书), 2016

Imangine 项目团队. Imagine 详细设计规约(说明书), 2016