

Imagine图片分享网站 项目管理计划

刘林青 1354361

杨安琪 1353005

刘雅骏 1352876

[1 项目总述](#)

[1.1 项目概要](#)

[1.2 项目成果](#)

[1.3 项目假设](#)

[2.1 项目过程](#)

[2.1.1 过程模型](#)

[2.1.2 裁减要点](#)

[2.1.3 需求变更管理过程](#)

[2.1.4 需求变更生命周期](#)

[2.1.5 需求变更可跟踪性](#)

[2.2 估算](#)

[2.2.1 系统抽象](#)

[2.2.2 功能点估算](#)

[2.2.3 用例估算](#)

[2.2.4 综合估算结果](#)

[2.3 时刻表](#)

[2.4 人员](#)

[2.5 开发环境](#)

[2.6 硬件与软件资源需求](#)

[2.7 工具](#)

[2.8 训练](#)

[2.9 质量计划](#)

[2.9.1 质量目标](#)

[2.9.2 检测缺陷评估](#)

[2.9.3 达到质量目标的策略](#)

[2.9.4 审核](#)

[2.10 风险管理](#)

[3. 项目跟踪](#)

[3.1 测量计划](#)

[3.2 任务跟踪](#)

[3.3 问题追踪](#)

[3.4 用户反馈](#)

[3.5 质量追踪](#)

[3.6 需要高级管理员审核的内容](#)

[3.7 状态报告](#)

[3.8 里程统计偏移](#)

[3.9 客户报告](#)

[3.10 业务经理报告](#)

[4 项目团队](#)

[4.1 项目组织](#)

[4.2 项目团队](#)

[4.3 角色与职责](#)

[5 参考](#)

[6 缩写说明](#)

1 项目总述

1.1 项目概要

人们遇到喜欢的图片时会希望保存到某地以备将来使用,并且希望能够网罗相关主题的图片 素材并且查 看志同道合的朋友的图片开阔视野,这一需求对于前端开发人员或者设计师来说尤为重要。我们希望通过 开发这个网站满足此类用户的需求,为用户提供分享和发现图片的平台。

本网站主要包括三大大模块:

I. 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,同时以管理员的 身份管理讨论小组。

II. 社交模块——用户可以收藏、评论、分享别人的图片,关注好友动态,并且针对感兴趣的话题发起 讨论小组。

III. 浏览模块——查看个人主页,并且查看网站反馈的图片资源。 IV. 账户管理模块——用户注册登录获得修改个人信息的权限。

项目代码	项目名称	客户
20160601	IMAGINE图片分享网站	网站用户

项目经理	配置经理	业务经理	项目副经理	配置副经理
杨安琪	刘林青	刘林青	杨安琪	杨安琪

项目类型	浏览器要求	项目阶段数
新开发	safari 6.2.1及以上版本 chrome 47.0.2526.10及以上版本 FireFox 43.0.2.533及以上版本	4

项目开始时间	项目截止时间	总预算预计
2016年3月	2016年9月	10万人民币

项目与客户联系人			
姓名	手机号	传真	邮件

刘林青	1801921598		1354361@ton gji.edu.cn
-----	------------	--	---------------------------

<p>项目边界</p> <p>外部接口描述</p> <p>1.1与第三方平台接口</p> <p>1.1.1 微信公众开发平台接口API</p> <p>用户可以选择把自己喜欢的图片分享到微信平台，网页可以自动跳转到微信登陆界面，用户授权登陆微信账号之后，可以把图片分享给微信好友，发送网站链接，或者直接分享到朋友圈。</p> <p>1.1.2 腾讯QQ接口API</p> <p>用户可以选择把自己喜欢的图片分享到QQ平台，网页可以自动跳转到QQ登陆界面，用户授权登陆QQ账号之后，可以把图片分享给QQ好友，发送网站链接，或者直接分享到个人空间。</p> <p>1.1.3 新浪微博接口API</p> <p>用户可以选择把自己喜欢的图片分享到新浪微博，网页可以自动跳转到微博登陆界面，把图片分享到新浪微博当中。</p> <p>1.2软件接口</p> <p>1.2.1 账户授权协议：OAuth 2.0</p> <p>客户端登陆授权层之后，服务提供商将根据令牌的权限范围和有效期，向“客户端”开放用户的资料。OAuth在"客户端"与"服务提供商"之间，设置了一个授权层（authorization layer）。"客户端"不能直接登录"服务提供商"，只能登录授权层，以此将用户与客户端区分开来。"客户端"登录授权层所用的令牌（token），与用户的密码不同。用户可以在登录的时候，指定授权层令牌的权限范围和有效期。</p> <p>1.2.2 网络访问协议：TCP/IP，HTTP</p> <p>浏览网页时，浏览器通过HTTP传输协议与WEB服务器交换信息。考虑到用户隐私数据的安全问题 必须防止中途窃听数据盗取用户隐私数据,改良为使用 HTTPS,在 HTTP 层之下建设 Secure Sockets Layer 来建立安全的传输连接。</p> <p>1.2.3 数据压缩：objective-zip</p> <p>在数据传输时，如果网络比较好、数据量较小，传输的方式可以忽略不计，可以不考虑优化。但是当网络条件不好或者数据量比较大时，需要将数据压缩后进行传输。</p>
--

项目增值服务
开发后期在线图片渲染、美化网页应用，方便用户使用图片。

公司目标
为前端开发人员、设计师和其他所有的图片爱好者分门别类网罗全面的图片素材，缩小图片搜索范围，与第三方平台的接口支持这些图片快速的转发 传播,使得用户在网站之外的地方也能获取网站中的图片资源。

1.2 项目成果

序号	里程碑日期	里程碑	交付物
1	2016-07-05	立项	
2	2016-07-13	可行性研究完成	可行性报告（市场分析、项目优势、面向人群、技术需求等）
3	2016-08-09	需求分析完成	需求规约（项目整体描述、功能性需求、非功能性需求）
4	2016-08-02	计划阶段完成	项目计划书（开发计划、测试计划、人员管理）
5	2016-08-25	概要设计完成	概要设计书（系统总体架构设计、系统软件结构设计、接口设计、界面设计、数据库设计、系统出错处理设置）
6	2016-09-02	详细设计完成	详细设计书（每一模块的输入项、输出项、逻辑流程图、接口、限制条件及出错处理）
7	2016-09-09	设计阶段完成	
8	2016-10-1	Construction 1: 代码实现与测试实现	主要页面demo（达到展示要求）
9	2016-08-01	Construction 2: 代码审核与重构	通过测试样例，达到覆盖率最高的项目
10	2016-08-31	Transition: 代码提交与合并	审核结果与通过审核的项目
11	2016-09-01	Deployment 1: 部署项目	部署项目，用户接受测试
12	2016-09-10	Deployment 2: 收集用户反馈	用户反馈报告
13	2016-09-15	Construction 3: 项目重构	恢复已知bug的项目
14	2016-09-20	Deployment 4: 项目上线更新	项目正式上线使用

其他成果	
序号	成果
1	该网站收集的图片数据和对应图片的文字描述数据均可作为开源数据资源
2	图片在线渲染、美化网页

1.3 项目假设

- 开发条件:成员自备开发使用电脑,并配置相关开发环境

- 设计成本: 由甲方承担
- 经费支持: 由甲方承担
- 开发效率: 采用增量开发模型, 开发冲刺期间避免外界打扰

2 项目计划

2.1 项目过程

2.1.1 过程模型

本项目采用了增量模型与RUP相结合的开发模型，增量模型将会被剪裁以适应RUP要求。

Imagine 图片分享平台的项目开发使用了增量模型的迭代式开发模型。由于本项目的三大核心需求与创新点已经明确，采用增量模型，在第一个迭代周期中首先进行核心功能的设计、编码、测试并发布，可以以本项目地创新点为亮点，快速在图片分享平台市场中抢占先机。其次，随着时间的推移，用户的对于图片搜索与分享的需求很可能进一步变化，采用增量模型，使得本项目能够很好地拥抱需求变更，减小维护成本，对于节约开发资金、降低投资风险有重大意义。

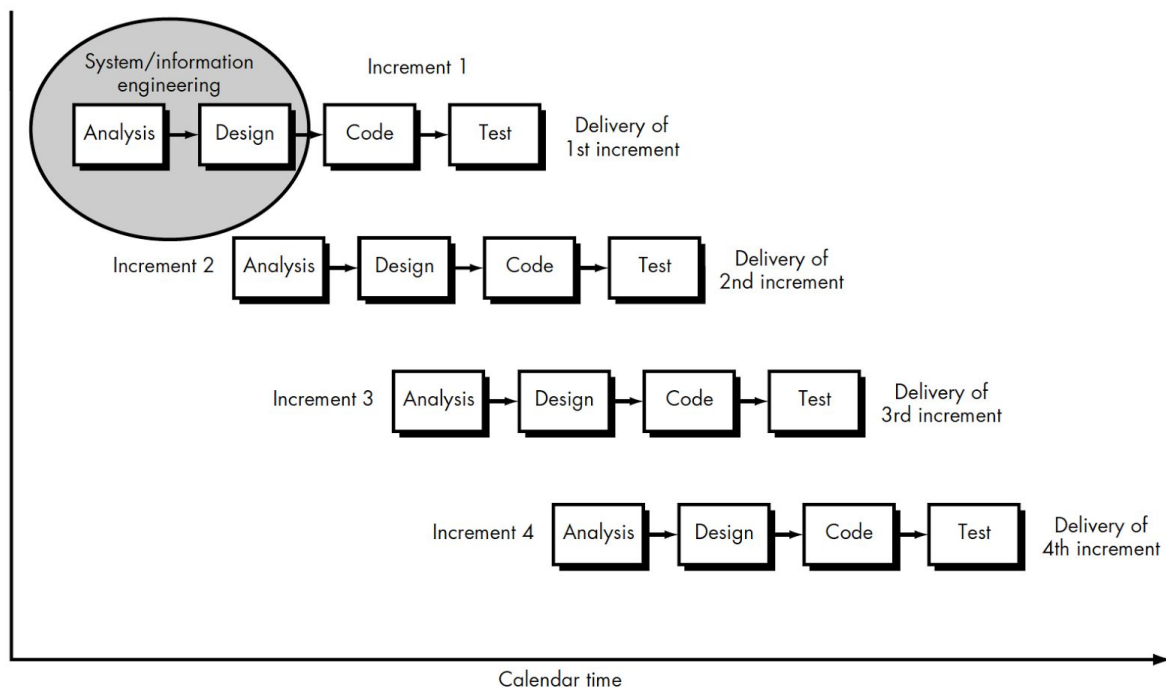


图2-1 增量模型

2.1.2 裁减要点

与标准模型的差异	增加／修改／删除	理由
相对于标准模型，更加注重开发过程和代码质量。	增加、修改	开发人员的平台开发经验不足，项目管理者的项目经验丰富，因此通过强化开发过程的质量审核过程来提高项目质量。
增加了代码审核与合并	增加	理由同上
细化了详细设计	修改	理由同上
增加了收集用户反馈	增加	理由同上
细化了测试步骤	修改	Imagine图片分享平台分为浏览、个人主页管理、社交、账户管理四大模块，且每个模块中用户的组合业务路径极多，因此细化测试步骤，对于保证本项目功能的正确性、提升用户体验具有重大意义。

2.1.3 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ChangeRequests.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。主要的考虑原则有：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、对整体日期和工作量的潜在引申影响等。
3. 估算修改变更所需的工作量。
4. 重新估算交付日程表。
5. 计算累计变更并分析影响。
6. 检查是否通过阈值。若工作量、日程、累计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。

2.1.4 需求变更生命周期

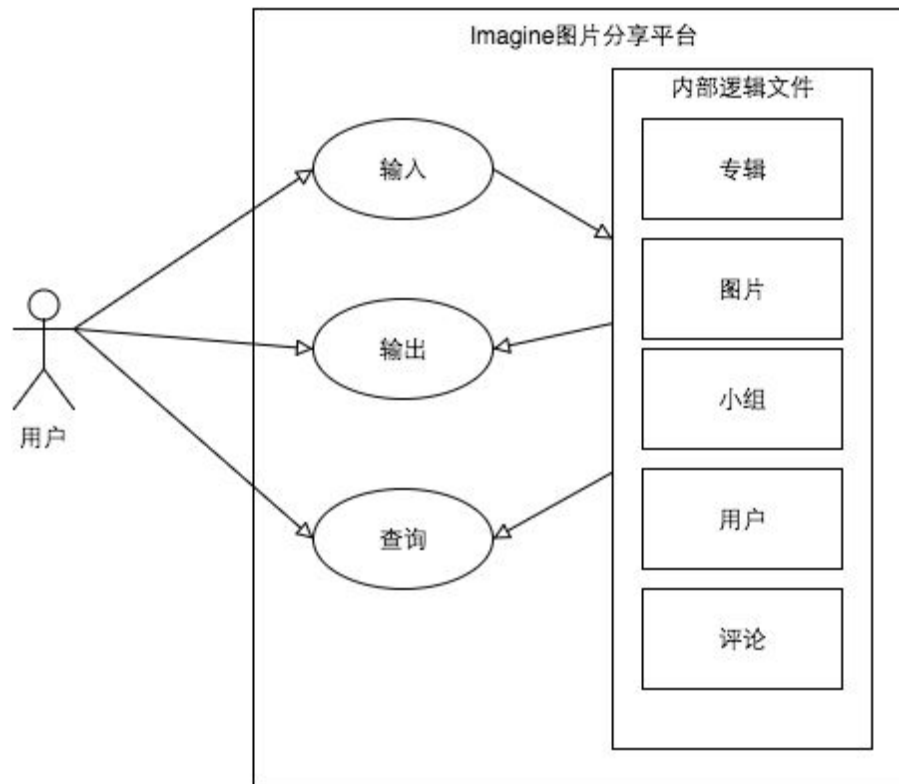
状态	说明
已登记	该需求变更已登记在ChangeRequest.xls
已批准	需求变更分析结果得到高级经理和客户的批准
返工中	得到批准后，按照项目经理制定的计划返工
验证中	返工完成后，需要验证各类规约是否修改一致
测试中	验证完成后，需要额外对代码进行测试
完结	测试通过
挂起	测试不通过，需进一步检查
Note: 状态之间可以流水跳转，也可以随机跳转。	

2.1.5 需求变更可跟踪性

利用需求管理工具 RequisitePro 进行跟踪。

2.2 估算

2.2.1 系统抽象



2.2.2 功能点估算

系统的内部逻辑文件（ILF）、外部逻辑文件（EIF）及其字段如下表：

专辑	图片	小组
专辑基本信息： <ul style="list-style-type: none">1. 专辑ID2. 创建人3. 专辑中的图片数量4. 创建日期5. 简述6. 主题7. 封面图片 专辑中的图片： <ul style="list-style-type: none">1. 图片ID	图片基本信息： <ul style="list-style-type: none">1. 图片ID2. 上传图片的用户3. 图片的本地地址4. 上传日期5. 标题6. 描述7. 标签	加入的小组： <ul style="list-style-type: none">1. 小组ID； 小组基本信息： <ul style="list-style-type: none">1. 小组ID2. 小组名称3. 发起小组的用户4. 成员数5. 发起日期6. 主题 小组中的专辑： <ul style="list-style-type: none">1. 专辑ID

用户	评论
用户基本信息： 1. 用户ID 2. 昵称 3. 密码 4. 用户头像 5. 邮箱 6. 所在城市 7. 性别 8. 生日 9. 个人描述 用户好友： 1. 好友ID 收藏的专辑： 1. 专辑ID 收藏的图片： 1. 图片ID 加入的小组： 2. 小组ID；	1. 评论ID 2. 图片／专辑／小组ID 3. 发起评论的用户 4. 评论日期 5. 评论内容

识别出的ILF和EIF功能点个数如下：

ILF内部逻辑文件	RET	DET	复杂度	未调整FP个数
专辑	2个	8个	低	7
图片	1个	7个	低	7
用户	5个	12个	低	7
小组	2个	7个	低	7
评论	1个	5个	低	7
EIF外部逻辑文件	RET	DET	复杂度	未调整FP个数
无	0个	0个	—	0
共计：35个				

由此，数据功能点个数 $DFP = ILF + EIF = 35$ 。

识别出的EI、EO、EQ个数如下表所示：

EI	FTR	DET个数	复杂度	未调整FP个数
注册	用户	用户ID为自动生成，共9个	低	3
登录	用户	共3个	低	3

修改个人信息	用户	共8个	低	3
新建小组	小组	小组ID为自动生成，共7个	低	3
修改小组信息	小组	共6个	低	3
删除小组	小组	共8个	低	3
新建专辑	专辑	专辑ID为自动生成，共8个	低	3
修改专辑信息	专辑	共7个	低	3
删除专辑信息	专辑	共8个	低	3
上传图片	图片	图片ID为自动生成，共7个	低	3
修改图片信息	图片	共6个	低	3
删除图片	图片	共7个	低	3
关注好友	用户	共1个	低	3
加入小组	小组	共1个	低	3
收藏图片	用户	共1个	低	3
收藏专辑	用户	共1个	低	3

共计：48个

EO	FTR	DET个数	复杂度	未调整FP个数
推荐图片	图片	共7个	低	4
推荐专辑	专辑	共8个	低	4
推荐小组	小组	共7个	低	4
分享图片	图片	共6个（图片ID不分享）	低	4
分享专辑	专辑	共7个（专辑ID不分享）	低	4

共计：20个

EQ	FTR	DET个数	复杂度	未调整FP个数
查询图片详情	图片	输入：图片、图片名称； 输出：图片基本信息，7个 共9个	低	3
查询专辑详情	专辑	输入：专辑封面、专辑名称； 输出：专辑基本信息，8个 共10个	低	3
查询小组详情	小组	输入：小组名称 输出：小组基本信息，7个 共8个	低	3

查看个人收藏图片	用户、图片	输入：“收藏图片”按钮； 输出：收藏的图片ID、图片基本信息，8个 共8个	中	4
查看个人上传图片	图片	输入：“上传图片”按钮； 输出：图片基本信息，7个 共9个	低	3
查看个人收藏专辑	用户、专辑	输入：“收藏专辑”按钮； 输出：收藏的专辑ID、专辑基本信息，9个 共10个	中	4
查看个人创建专辑	专辑	输入：“创建的专辑”按钮； 输出：专辑基本信息，8个 共9个	低	3
查看个人所在小组	用户、小组	输入：“所在的小组”按钮； 输出：发起的小组ID，小组基本信息，8个 共9个	中	4
查看个人发起小组	小组	输入：“发起的专辑”按钮； 输出：小组基本信息，7个 共8个	低	3
搜索图片	图片	输入：图片ID，名称，标题，上传用户，标签，5个 输出：图片基本信息，7个 共12个	低	3
共计：33个				

由此，事务功能点个数 $TFP = EI + EO + EQ = 48 + 20 + 33 = 101$ 个。

所以，未调整的功能点 $UFP = DFP + TFP = 35 + 101 = 136$ 个。

进一步考虑，本系统的系统特性及影响程度如下：

系统特性	分数
数据通讯	4
分布式数据处理	4
性能	3
大业务量配置	1
事务处理率	3

在线数据输入	3
最终用户效率	4
在线更新	2
复杂处理	1
可复用性	4
易安装性	0
易操作性	0
多场地	3
支持变更	3
合计：35	
调整因子：0.65+0.01*35=1	

由此，调整后的功能点 $AFP = UFP * 1 = 136$

根据日本IPA SEC 组织 (<http://sec.ipa.go.jp/>) 所提供的工作量估算共识：

$Effort = e^{0.542 * FP^{1.154}} = 498.30$ 人一天

假设日标准工作时间为8小时，工作量估计为 3986.41人一小时。

2.2.3 用例估算

用例分类一句如下表：

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

考虑项目中的用例：

用例	事务	复杂度
01 导航栏	29	高

浏览 模块	02 网站首页	27	高
	03 推荐图片	7	中
	04 推荐专辑	9	高
	05 推荐小组	11	高
	06 查询图片详情	2	低
	07 查询专辑详情	4	中
	08 查询小组详情	6	中
	09 个人主页	36	高
	10 查看个人收藏图片	4	中
	11 查看个人上传图片	4	中
	12 查看个人收藏专辑	6	中
	13 查看个人创建专辑	6	中
	14 查看个人所在小组	8	高
	15 查看个人发起小组	8	高
	16 搜索图片	3	低
账户 管理 模块	17 注册	9	高
	18 登录	4	中
	19 修改个人信息	3	低
个人 主页 管理 模块	20 新建小组	15	高
	21 修改小组信息	30	高
	22 删除小组	3	低
	23 新建专辑	10	高
	24 修改专辑信息	12	高
	25 删除专辑信息	3	低
	26 上传图片	4	中
	27 修改图片信息	3	低
	28 删除图片	3	低
	29 关注好友	1	低

社交模块	30 加入小组	1	低
	31 收藏	2	低
	32 分享	4	中
	33 评论	4	中

估算工作量			
用例类型	工作量（基于以往项目）	个数	总工作量（人一天）
低	1 人一天	10	10
中	5 人一天	11	55
高	8 人一天	12	96
共计	161		

计算未调整用例点UUCP=10 * 5 + 11 * 10 + 12 * 15 = 340

考虑技术复杂度因子TCP如下表，得分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	4
2	Response or throughput performance objectives	1	3
3	End-user efficiency (online)	1	4
4	Complex internal processing	1	1
5	Code must be reusable	1	4
6	Easy to install	0.5	0
7	Easy to use	0.5	0
8	Portable	2	0
9	Easy to change	1	3
10	Concurrent	1	3
11	Includes special security features	1	1
12	Provides direct access for third parties	1	5

13	Special user training facilities required	1	0
----	---	---	---

根据权重计算：

$$TFactor = 4 \times 2 + 3 \times 1 + 4 \times 1 + 1 \times 1 + 4 \times 1 + 0 + 0 + 0 + 3 \times 1 + 3 \times 1 + 1 \times 1 + 5 \times 1 + 0 = 32$$

$$TCP = 0.6 + (0.01 \times TFactor) = 0.92$$

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	5
2	Application experience	0.5	5
3	Object-oriented experience	1	5
4	Lead analyst capability	0.5	4
5	Motivation	1	4
6	Stable requirements	2	3
7	Part-time workers	-1	0
8	Difficult programming languages	-1	1

$$EFactor = 5 \times 1.5 + 5 \times 0.5 + 5 \times 1 + 4 \times 0.5 + 4 \times 1 + 3 \times 2 + 0 + 1 \times (-1) = 26$$

$$EF = 1.4 + (-0.03 \times EFactor) = 0.62$$

最后，根据技术复杂度和环境因素：

$$计算UCP = UUCP \times TCF \times EF = 340 \times 0.92 \times 0.62 = 193.94$$

根据标准工作量1个UCP对应20人一小时，得到工作量估算为 3878.72 人一小时

假设日工作时间为8小时，即需要 484.84 人一天。

2.2.4 综合估算结果

综合2.2.3和2.2.4两种估算方法得到的工作量估计，求其平均得到工作量为3932.60人一小时。

假设工作时间为8小时，即需要491.57人一天。

Phase-wise Effort Estimate		
Activity/Phase	Person-days	% of Total Effort
Requirements	49	10
Design	59	12
Build	143	29
Intergration testing	35	7
Regression testing	10	2
Acceptance testing	30	6
Project management	74	15
Configuration management	15	3
Training	49	10
Others	28	6
Estimated effort	492	100%

Effort Estimate by Iterations	Person-days	% of Total Effort
Project initiation	25	5
Inception phase	25	5
Elaboration phase: Iteration 1	44	9
Elaboration phase: Iteration 2	34	7
Construction phase: Iteration 1	25	5
Construction phase: Iteration 2	25	5
Construction phase: Iteration 3	20	4
Transition phase	108	22
Project closure	10	2
Project management	74	15
Configuration managements	15	3
Training	49	10

Others	38	8
Total estimated effort	492	100%

2.3 时刻表

详见附件：Imagine:Project.cpp

2.4 人员

按角色		
角色	需求人数	日期
项目经理		
配置管理		
开发者		
数据库管理员		
测试经理		
测试人员		
业务经理		
配置经理		
总计		

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	3	1	2
Java	4	1	3
MySQL	2	0	2
总计	9	2	7

人员需求计划

Month	Offshore	Onsite	Total

2.5 开发环境

硬件	软件
Macbook	Mac OS X El Capitan, IntelliJ IDEA
高配置服务器	Apache, MySQL

2.6 硬件与软件资源需求

描述	所需数量	日期
Macbook with 256 RAM	9	
16GB space on server	1	
IntelliJ IDEA	4	
MySQL	4	
Requisite Pro	1	
Microsoft Project 2010	1	
Bugzilla	4	

2.7 工具

UML 建模工具：<http://draw.io>

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 2010

测试工具：Apache JMeter

2.8 训练

训练领域	时长	免去条件
技术领域		
JavaEE开发	10天	有Java EE开发经验
前端开发	10天	有前端开发经验
MySQL	2小时	有MySQL使用经验
业务领域		
过程相关		
质量管理体系	3小时	有项目管理工作经验
配置管理	2小时	有项目配置管理经验
代码审核	3小时	有代码审核经验

2.9 质量计划

2.9.1 质量目标

项目质量目标		
目标	值	目标设置基础
注入的缺陷总数	132	0.033缺陷/人时。比类似的项目Synergy要好10%，Synergy的缺陷率是0.036。
质量（接受缺陷密度）	6	平均的缺陷移除率是93%，预计本项目的缺陷移除率是95%。则预计验收测试时的缺陷总数是总缺陷数的5%。
生产力	15 功能点/人月	4-18功能点/人月，平均10功能点/人月。
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%	Synergy质量成本为33%
------	-----	-----------------

2.9.2 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	26	20%	类似项目Synergy和PCB
代码走查	26	20%	类似项目Synergy和PCB
单元测试	53	40%	类似项目Synergy和PCB
集成测试和回归测试	23	17%	类似项目Synergy和PCB
验收测试	4	3%	类似项目Synergy和PCB
总计	132	100%	类似项目Synergy和PCB

2.9.3 达到质量目标的策略

策略	期望
按照缺陷预防纲领和过程来做缺陷预防：按照Synergy缺陷预防标准来编码。	降低15%缺陷注入，提高25%生产力。
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。

2.9.4 审核

审核点	审核对象	审核类型
项目计划结束	项目计划 项目时刻表	团队审核 SQA审核
原型需求讨论结束	需求规约 用例条目 界面框架	团队审核
概要设计结束	概要设计规约	团队审核

详细设计结束	详细设计规约	团队审核
功能实现结束	代码	单人审核
单元测试结束	测试结果和代码	单人审核
代码审核结束	审核结果和代码	单人审核
用户反馈报告生成结束	用户反馈报告	团队审核

2.10 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	对于网站的业务逻辑不清晰，团队中缺少具有社交网站运营经验的成员。	0.9	7	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项目延期。	0.5	5	2.5	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	5	3.5	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。

3. 项目跟踪

3.1 测量计划

需要收集的数据	数据单位	所用工具
规模	LOC, FP	XcodeCoverage

人力	人天	MSP
缺陷	个数	BugsBunny
时刻表	时长	MSP

3.2 任务跟踪

活动	过程
任务时间计划	用MSP做时间计划 在必要时调整时间计划以适应最新情况。
任务布置	MSP软件将最新的任务安排发布到团队成员，成员可以自行下载
任务状态追踪	每天都要追踪任务完成情况。
项目会议	每周一次组会
详细分析会议	每次迭代结束之后进行

3.3 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	任意项目成员	PM指定修改, 每天	2天
用户反馈	UserFeedBack.xls	PL指定人员	PL指定修改, 每天	2天
业务管理问题	每周进度报告	业务经理	PL指定修改, 每天	5天
业务支持问题	Request Tracker	任意项目成员	支持性服务, 每天	2天

3.4 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

3.5 质量追踪

质量活动	行动
缺陷追踪	用Github Issue记录缺陷，跟踪缺陷至缺陷解决
审核(需求, 概要设计, 详细设计)	依据质量计划里的项目目标进行审核
代码审核	在提交申请之后请同级开发者对代码进行审核，并提出改进意见报告。
独立单元测试	生成测试报告并交由开发者做代码重构
集成测试、系统测试	依据质量计划里的项目说明进行测试

3.6 需要高级管理员审核的内容

序号号	审核事项	审核频率
1	时刻表	每一次版本更新时
2	项目计划	重大计划变更时
3	里程碑报告	每个里程碑时间点

3.7 状态报告

报告对象	频率
业务经理	每周一邮件汇报
项目经理	每周一例会报告

3.8 里程统计偏移

实际与预算比较	前五个里程	其余里程
人力投入	10%	5%
时刻表	10%	5%
缺陷数	20%	20%

3.9 客户报告

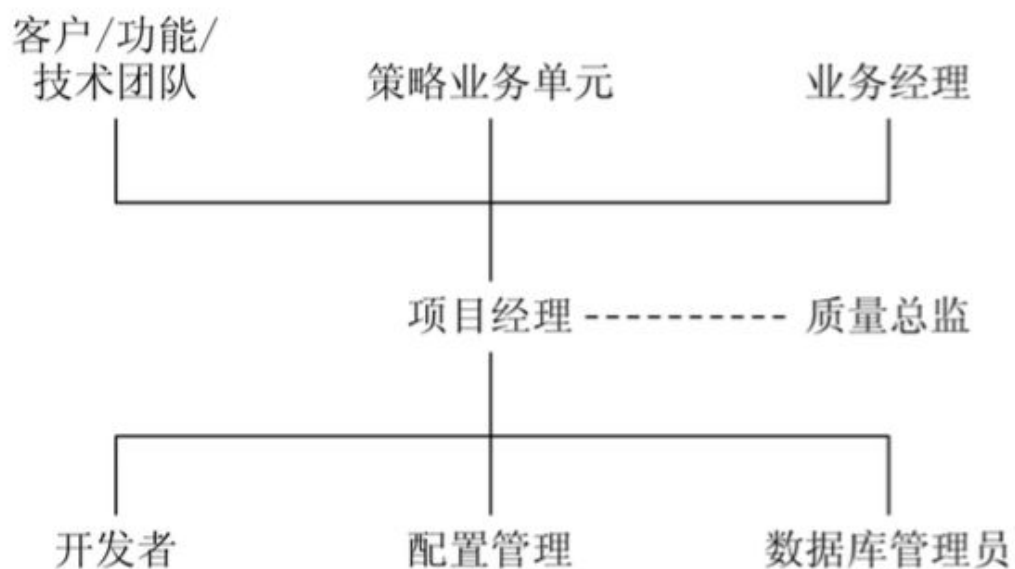
每周进行项目开发进度的汇报，每个里程碑点进行汇报，开发过程中发现的问题要汇报整个的追踪过程和进度。

3.10 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

4 项目团队

4.1 项目组织



4.2 项目团队

序号	职责	开始日期	预计结束日期
1	项目经理	2016-03-20	2016-09-20
2	质量总监	2016-03-20	2016-09-20
3	开发者	2016-07-01	2016-09-15
4	数据库管理员	2016-07-30	2016-09-15
5	测试经理	2016-07-01	2016-09-20
6	测试人员	2016-07-01	2016-09-15
7	业务经理	2016-03-20	2016-09-20
8	配置经理	2016-03-20	2016-09-20

4.3 角色与职责

角色	职责
项目经理	<ul style="list-style-type: none">● 组织项目组成员进行技术培训● 负责把握整个项目进度● 参与关键技术评审● 协调解决项目中发生的问题，保证按计划进行
质量总监	<ul style="list-style-type: none">● 审核质量计划● 检查质量维护按计划进行
开发者	<ul style="list-style-type: none">● 概要与详细设计● 代码实现与重构● 参与技术评审
数据库管理员	<ul style="list-style-type: none">● 维护数据库与后台
测试经理	<ul style="list-style-type: none">● 整体测试计划设计与监督
测试人员	<ul style="list-style-type: none">● 测试实现
业务经理	<ul style="list-style-type: none">● 与用户交流，获取需求● 整理用户反馈● 推广业务执行
配置经理	<ul style="list-style-type: none">● 软件设施准备● 人员配置与管理● 采购与预算计划

5 参考

Pankaj Jalote. *Software Project Management in Practice*, 2001

Imagine 项目团队. *Imagine* (), 2016

Imagine 项目团队. *Imagine* (), 2016

Imagine 项目团队. *Imagine* (), 2016

Imagine 项目团队. *Imagine* (), 2016

6 缩写说明

缩写	说明
MSP	Microsoft Project 2013
FP	Function Point Approach 功能点估算
UCP	Use Case Point Approach 用例点估算
UUCP	unadjusted use case points 未调整用例点
TCF	technical complexity factor 技术复杂度因子
EF	environment factor 环境因子