**SJTU公司**

**立项建议书**

项目名称：边走边拍app

项目组：

|  |  |  |  |
| --- | --- | --- | --- |
| 学号 | 姓名 | 手机 | 电子邮箱 |
| 515030910284 | 于航 | 13162093291 | yuhang012@sjtu.edu.cn |
| 515030910287 | 唐天成 | 18918817890 | 1025151692@qq.com |
| 515030910282 | 刘瑾旭 | 13369050626 | 1010878473@qq.com |
| 515030910292 | 李新宇 | 13966447393 | 1091161455@qq.com |

**2017 年 5月**

1. 项目的必要性

目前，拍照和分享已经成为我们生活的一部分。人们对拍照与照片分享平台的需求越来越大、要求越来越高。起初的相机、图片分享社区渐渐难以满足这种需求，以至于发展至今，已经产生能够对图片做各种修饰和修改并加以分享的平台。近年来，相关app在国内做得风生水起。倘若能在这个方向投入精力，并有所创新和改进，一定能够在市场竞争中占据优势地位。然而如何创新和改进，一直是各公司关注的重要内容。

即便如今各产品已经发展得较为完善，但仍有不足。因此，改造传统的拍照分享类app，建立一个功能强大的、能让使用者在步行、骑车、驾车、登山和旅游过程中，拍下身边的照片，并在地图上沿路展示，进行分享的app，是顺应时代发展并应该能够有所成就的。

1. 项目目标和特性

目标特性：

本项目所开发的软件目标是提供方便优质的图片社交的功能，构建起一个开放的图片社交生态圈，帮助用户分享在步行、骑车、驾车、登山和旅游过程中而拍摄的照片。总体目标是：博采众家之长，开发适合于我国用户的图片社交软件，支持本项目软件内和其他社交软件的图片分享，有效解决用户社交需求，为用户提供优质服务。

预计实现特性：

注册与登陆（高）

拍摄照片（高）

选择其中一些照片进行分享，可以是私享、好友分享和大众分享（高）

在地图上标注上这些照片（高）

在地图上动态展示行走轨迹和相关的照片（中）

分享到微信群中（中）

其他用户可以评论和交流（高）

对照片进行美化和装饰（低）

可撰写美文或小诗（低）

不同时间上同一位置的照片对比（中）

用户管理（高）

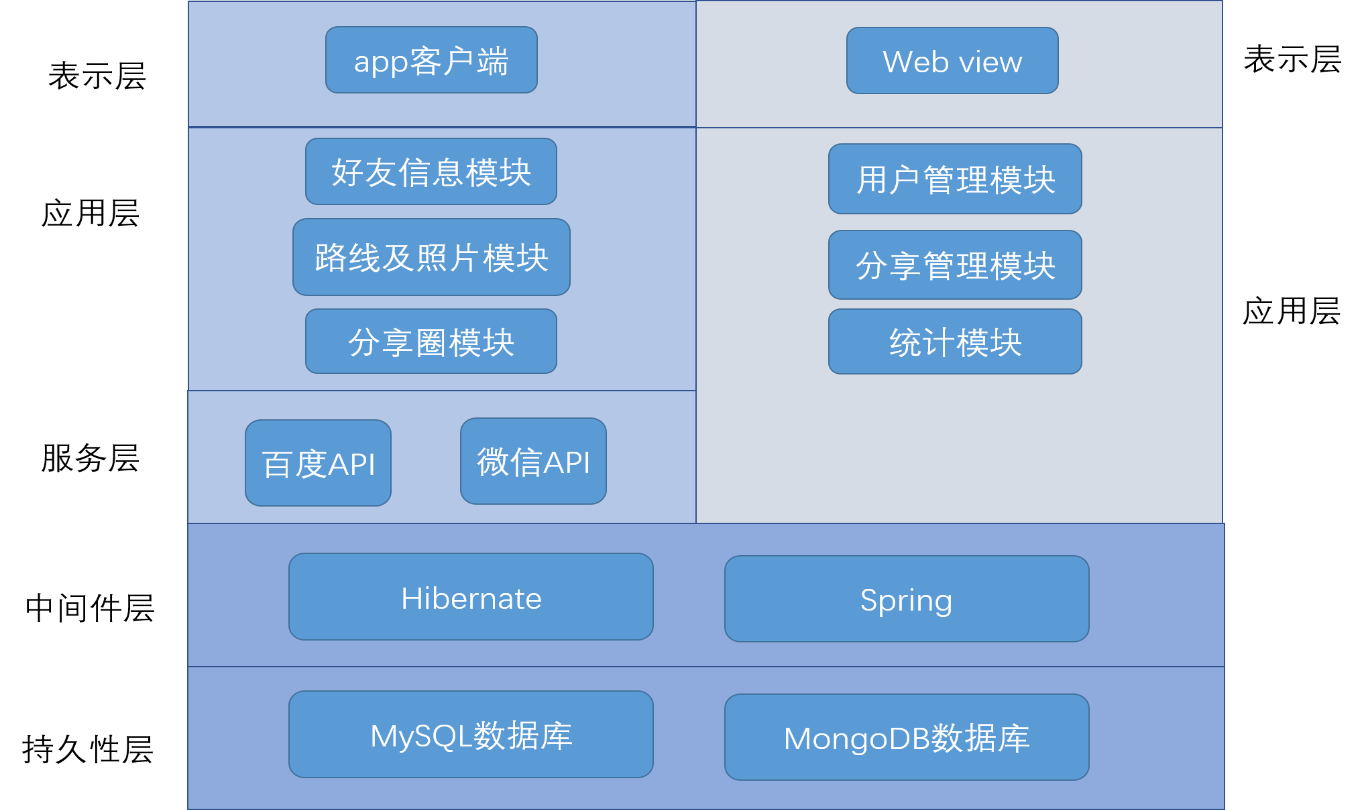
对照片和用户进行简单的统计分析（中）

根据评论和浏览次数，选出最佳的分享（低）

就本项目所开发软件所实现的功能而言，拥有相似功能的软件有Instagram（国外）、in（国内）。这两款APP都具有拍照美化然后上传分享（图片社交）的功能，并且它们拥有庞大的用户人群，尤其是Instagram。相比于这两者，本项目的特点在于可以在地图上标注这些照片、动态展示行走轨迹和相关照片并且可以在对不同时间同一位置上的照片进行对比，这是它们所不具备的功能。这些功能也有效避免了同质化的问题，使本项目软件具备独特性，拓展了细分市场。

1. 项目技术方案

技术架构图



我们采用分层模式，最低层的持久性层由MySQL和MongoDB组成，MySQL用于存储用户信息以及好友信息，MongoDB用于存储图片。中间件层采用Hibernate和Spring的技术用于ORM和OGM实现OOP。服务层包含前端app所采用的其他应用的服务接口。应用层分为app端和web端，app端大致分为三个模块—好友信息模块、路线及照片模块、分享圈模块，实现我们这个app的三大核心功能，web端也分为三个模块，实现管理功能。最上层的表示层则简单分为app客户端和Web view，分别用于与用户交互和与管理员交互。因为我们的app主要涉及CRUD，所以可将这些操作对象抽象为资源，在前端app和后端的应用服务之间采用RESTful接口进行通信，用以简化开发。

设计约束

* 模式：C/S，B/S
* 编程语言：JAVA，JS，HTML
* 数据库：Mysql，MongoDB
* 建模工具：Power Designer
* 建模语言：UML
* 开发工具： Android studio/Eclipse
* 运行环境：App运行于Android系统上，web系统运行于Apache/Tomcat（通过浏览器访问）

四、项目风险分析和里程碑计划

分析边走边拍APP的风险，最大的2个风险如下：

(1) 第一大风险：技术风险。应采用什么架构，是否要采用Web service技术，本项目小组没有相关经验。同时小组成员还要进行相关技术的学习和培训。

(2) 第二大风险：进度风险。整个项目的开发进度较为紧迫。按照当前掌握的需求，需要一到两个月完成。但是由于可能存在的意外原因，开发时间可能还会减少一些。

根据上述风险分析拟进行如下迭代安排。

(1) 第一次迭代，确定软件开发计划，解决技术风险和架构风险，开发架构原型，并通过测试确保所选的技术和架构是合理的，实现注册登录功能以证明架构合理性。

(2) 第二次~第五次迭代，解决进度风险。将功能需求排出优先级，将优先级高的需求放在第二次迭代完成，构建系统第一个版本，进行第一次移交；在第三次迭代进行第二次移交。在第四次迭代中完成优先级低的需求，构建系统第三个版本，进行第三次移交；第五次迭代（产品化迭代）时进行第四次移交，此时产品已经可以上线，每次迭代历时1-2周。。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **阶段** | **迭代** | | **任务描述** | **成果** |
| 先启阶段 | **-** | | **-** | **-** |
| 精化阶段 | 架构迭代——确定架构原型  6月24日~7月6日 | | 确定软件开发计划；开发架构原型，实现注册登录功能，并通过测试确保所选技术和架构的合理性 | 软件开发计划  架构原型（实现注册登录功能） |
| 构建阶段 | 第一次开发迭代  7月7日~7月15日 | | 实现并测试（单元测试，系统测试）R1需求，构建系统第一个版本 | 测试报告；完成第一次开发迭代的编码，移交第一个版本 |
| 第二次开发迭代  7月16日~7月23日 | | 实现并测试（单元测试，系统测试）R2需求，构建系统第二个版本，改进第一个版本的缺陷 | 测试报告；完成第二次开发迭代的编码，移交第二个版本 |
| 第三次开发迭代  7月24日~7月29日 | | 实现并测试（单元测试，系统测试）R3需求，构建系统第三个版本，改进第二个版本的缺陷 | 测试报告；完成第三次开发迭代的编码，移交第三个版本 |
| 产品化阶段 | | 产品化迭代  7月30日~8月31日 | 测试处于发布准备中的产品，基于用户反馈进行微小调整；探索产品新功能 | 测试报告；  发布产品 |

R1需求：

1) 注册与登陆

2) 用户管理

3) 拍摄照片

4) 选择其中一些照片进行分享，可以是私享、好友分享和大众分享

5) 在地图上标注上这些照片

6) 其他用户可以评论和交流

R2需求：

1) 分享到微信群中

2) 在地图上动态展示行走轨迹和相关的照片

3) 不同时间上同一位置的照片对比

4) 对照片和用户进行简单的统计分析

5) 根据评论和浏览次数，选出最佳的分享

R3需求：

1) 对照片进行美化和装饰

2) 可撰写美文或小诗

1. 项目预期成果

项目验收时至少应交付以下成果：

* 《项目计划》
* 《迭代计划》（每个迭代开始前编写迭代计划）
* 《迭代评估报告》（每个迭代结束后编写迭代评估报告）
* 《SRS文档》和用例模型（.oom）
* 《软件架构文档》和分析设计模型（.oom）
* 《测试用例》和《测试报告》
* 《项目总结报告》
* 源代码和可执行代码
* 演示视频文件（包括安装、运行、功能等）
* 演示PPT
* 分工说明