



# 四川大学

## 本科生毕业设计（技术报告）



题    目 企业信息聚合平台

学    院 软件学院

专    业 软件工程

学生姓名 汪航

学    号 2013141463189 年级 2013

指导教师 何军

教务处制表  
二〇一七年四月二十四日



# 企业信息聚合平台

软件工程

学生 汪航 指导老师 何军

**[摘要]**企业信息聚合平台本质上是一个信息共享平台。是以 web 端为管理端，Android 端为客户端的平台。企聚平台是一款自组织型网络社区平台，以 B2C 模式创建。企聚平台以为有了解各行各业和各商家公司、最近宣传活动的信息需求的个人用户，构造一个值得信赖、资讯确切、信息共享的社区平台为目的。在企业的角度而言，企聚平台是一个它发布企业最新动态以及活动的平台，同时也是企业自我展示，推广自身企业文化，推销自己的产品的平台。

该平台主要由两部分组成：一是拥有用户的登录注册，信息检索，添加编辑活动，活动地图定位等功能的 Android 客户端，二是管理员用于用户权限管理，数据库增删改查的 web 服务器端。客户端实现了企业会员用户和普通用户的区别登录注册。会员拥有添加自己公司地址信息，文字图片信息，发布公司活动的功能。通过高德地图的 API，用户可以在地图上查看自己所在地附近的活动和公司信息。也可以通过模糊检索来查找所有以富文本形式展示的活动。服务器端与客户端按照 JSON 的信息格式通信。管理端是典型的 j2ee 类型项目，model 层是基于 MYSQL 的数据持久化操作，view 层是基于 JSP 的页面渲染，controller 层是基于 HTTPSERVLET 的请求分发。

该课题涉及到信息聚合展示设计，手机地图定位，图层展示，移动端富文本编辑展示，JSON 格式低冗余通信，ajax 异步请求。

**[关键词]** 信息聚合；地图定位；并发请求队列；安卓；手机富文本编辑；



# Polymeric corporate Information platform

Software Engineering

Student: Hang Wang      Adviser: Jun He

**[Abstract]** Enterprise information aggregation platform is to aggregate the enterprise project information to the Android side for the client, the web side of the server-side platform. The platform is a network of spontaneous platforms, the goal is to need to know the activities of different industries and enterprises, the latest dynamic, information individuals, to create a reliable, accurate information and information sharing platform. For enterprises, through the platform to register, and improve the enterprise information to obtain the right to publish projects and activities in the platform, which created a self-display platform.

The platform is mainly composed of two parts: one is to have the user's login registration, information retrieval, add editing activities, activities, such as positioning the location of the Android client, the second is the administrator for user rights management, database increase the Web server The client implements the difference between the enterprise member user and the ordinary user. Members have the ability to add their own company address information, text image information, publishing company activities. With the high-map API, users can view their activities and company information near their location on the map. You can also use fuzzy search to find all activities in rich text. Server-side communication through the JSON format with the client, the typical j2ee project, MVC structure, the model layer is based on MYSQL Based data persistence operation, view layer is JSP page display, control layer is based on SERVLET request distribution.

This topic involves information aggregation display design, mobile map positioning, layer display, mobile end Rich text editing display, JSON format low redundant communication, Ajax asynchronous request.

**[Key Words]** Polymeric information; Map position; Android; Phone RichEditText;



## 目 录

1 绪论	6
1.1 选题来源与意义	6
1.2 本文主要工作	7
1.3 本文的组织与结构	9
2 文本相关知识和技术	10
2.1 HTTP 并发请求队列	10
2.1.1 HTTP 并发请求队列简介	10
2.1.2 Volley 的基本架构	10
2.2 LRU 缓存技术	12
2.2.1 原理	12
2.2.2 实现	12
2.3 RICHEDITOR 的实现	15
2.3.1 简述	15
2.3.2 部分实现展示	15
3 系统的设计与实现	18
3.1 系统的概况	18
3.2 系统开发计划及阶段性完成情况	18
3.3 系统的详细设计与实现	21
3.3.1 企业信息聚合平台 Android 移动客户端	21
3.3.2 企业信息聚合平台 web 后台管理端	37
3.4 本章小结	48
4 方案选择与结果分析	48
4.1 软件运行环境	48
4.2 硬件环境	49
4.3 企业信息聚合平台用户可接受测试分析	50
用户反馈测试用例:	50
搜索项目测试用例:	51
搜索企业测试用例:	52
发起项目测试用例:	53
地图定位搜索附近企业用例:	54



---

4.4 本章小结.....	55
5 讨论与体会.....	56
5.1 对项目开发过程的体会.....	56
5.2 对项目的评价.....	56
5.3 对项目成果用于实际意义的阐述.....	57
5.4 本章小结.....	58
参考资料.....	59
声    明 60	
致    谢 61	
附录 1 详细设计文档.....	62
附录 2 软件测试文档.....	89
附录 3 用户文档说明书.....	101



## 1 绪论

### 1.1 选题来源与意义

自从我们进入 21 世纪以来，移动通讯技术依靠快速的更迭，当之无愧的变成现今发展最快速，最受到社会希冀和认可的新兴产业。这些年来伴随互联网，移动互联网的兴起，我们的世界不知不觉已经进入到一个信息爆炸的时代，互联网上出现了浩瀚无垠的信息。人们面对呼啸汹涌而来的巨量无所适从，想迅捷、高效的从广大繁多的信息中找寻自己所需要的信息，成为一件异常的艰难的工作。

诸如社会上一些企业，公益组织的活动、宣传信息等，不仅庞杂无穷，并且由于虚假、诈骗信息的存在，许多关注公司企业信息的人员无法获取到比较完善系统化的企业信息。基于这一点，我开发一款专门为企业、公益组织与个人信息服务的企业信息聚合平台（以下简称企聚平台）。企聚平台是一款自组织型网络社区平台，以 B2C 模式创建。企聚平台以为有了解各行各业和各商家公司、最近宣传活动的信息需求的个人用户，构造一个值得信赖、资讯确切、信息共享的社区平台为目的。即个人能通过该平台，借助该平台在某些领域内聚合网络资源，扩张他个人的人际辐射力和影响力。一方面，以最短的时间获取企业最为准确的、最新动态的信息；另一方面，它也是一个企业和企业客户之间的一个重要的及时的交流平台。

和于传统地通过网上搜索并且获取企业信息的方法相比较，这种企业信息聚合平台将给个人带来更多的便捷，同时个人也可以横向了解和比较同一行业的不同企业相应的信息。和另一种客户与企业之间通过电子邮件的方式对比，该平台也更具有及时性，拥有手机客户端的信息聚合平台，会愈加顺应移动互联网朝代的工作时间的零碎化，工作地点多元化，工作方式自由化的趋势。

关于企业来说，企聚平台是一个它发布企业最新动态以及活动的平台，商家可以经过在平台注册，并且填写企业信息，然后取得在本平台发布项目的权利。本平台为企业和商家提供了一个发布活动的平台，同时也是企业自我展示，推广自身企业文化，推销自己的产品的平台。



## 1.2 本文主要工作

本文主要为了清晰的说明本人开发的《企业信息聚合平台》所有实现的功能和所设计的项目结构。对项目的模块进行一个全面的描述，可以作为一个后续扩展开发的参考文件。

本项目为了实现使用者和管理者的区分，将只为普通使用者的提供服务的移动客户端，管理者提供 web 管理端。这两个都为客户端，共同与服务器端进行信息交涉。当然，移动端也有少量通过 API 与第三方平台的服务器端进行信息交互的情况，如移动端的地图定位，地址搜索时，我使用了高德地图所提供的服务。

在 web 管理端方面，考虑到管理端功能实现的简洁性和稳定性，直接采用的是最基本的 J2EE 项目，基于 servlet 和 java server page 等技术。在 web 管理端的功能划分方面，我主要分为三个模块：企业资源管理，活动项目资源管理，用户信息管理。三个模块都是由 servlet 来接收页面上传来的请求，并做一些安全性的逻辑处理，然后直接通过 JDBC，获取 MYSQL 数据库的连接，直接向 database 数据库的表进行 CRUD 操作。管理员端的设计，考虑到管理员用户的数量，和操作的频率次数较少的情况，使用数据库连接池会显得比较鸡肋，反而降低代码执行效率。所以没有采用传统的数据库连接池的方法与数据库进行交互，而是使用原生 JDBC 编码。

移动客户端的功能较 web 管理端更为复杂，用户也是面向社会大众。所以对界面 UI 的刷新；信息交互时接发；图片，html 页面等大数据传输等方面都有一定的设计与优化。客户端的 UI 融合了 Google 在 Android5.0 提倡的 material design 风格。静态上，文本框、图片、按钮、菜单等 UI 元素，在色彩统一、视觉扁平的基础上，具有一定阴影效果，动态上，数据加载弹框，提示弹窗，界面切换，下拉刷新都有生动的动画。移动端的网络相比 PC 端更加不稳定，所以移动端的信息发送和接收有需要优化。首先，必须使用异步的 HTTP 请求，手机网络的不稳定，不能保证严格的实时性，同步的情况下，一旦长时间没有接收到 response，导致线程长时间阻塞，会对手机性能造成严重的浪费。其次我使用了请求队列的模式来处理请求的发送，批量发送请求在流量消耗，请求速度方面有着明显的优势。针对大文件的传输，加载情况，添加了线程池加载和使用 LRU 缓存策略，根据内存缓存、磁盘缓存、服务器磁盘的顺序进行检索，只有在本层缓存缺失的情况，才会向下一级继续检索。内存缓存对，在移动端需要大量加载图片和 html 页面的情况，有着十倍以上性能提升，同时也减少了发送请求对流量和电量的消耗。

本文主要完成了以下工作：





1) 研究并选择适合 web 管理端实现模式：查阅不同的 web 应用实现模式，比如 SSH, EJB 等开发框架等。对比各个方案，最终采用了 SERVLET - JSP 的实现模式，并结合 Ajax 异步请求和 JDBC 数据库驱动来开发项目，一个请求对应一个 servlet，使业务简单的项目，需求明确，结构不至于臃肿。

2) 实现 MD(Material Design)风格 UI：通过研究学习 Google 提供的设计 demo 和大量官方文档、技术博客。使用了 github 上面的一些开源库，例如 Google 的 support-design, material dialog library, gallery final, circle image view 等。界面以蓝色的基调，沉浸式状态栏，抽屉式侧滑菜单，阴影按钮，扁平化选择弹框，都是很有特色的手机界面，由于性能的优化，流程的操作体验更容易打动用户。

3) 研究高德地图 AMAP 第三方服务：由于 Google 地图的退出，我只考察了百度地图和高德地图两个第三方的地图服务。综合比较，由于百度地图经常出现，地图升级后不兼容低版本的问题，这会给使用的 API 的应用造成较大的不稳定。所以采用了具有详细具体开发文档，并且拥有 GPS，基站，混合三种定位模式的 AMAP。研究了 AMAP 的 POI 绘制基本方法，图层分类，和三种定位的调用与电量消耗情况、定位精度。

4) 使用了本地的 LRU CAHCE 策略：根据 LRU（最近最少使用策略），来缓存较大数据量的图片、文件。避免了每次滑动到图片，都需要重新向服务器发请求的情况。内存的缓存区大小是根据当前剩余内存的空间，取其三分之一，既保证图片的正常缓存，又防止内存溢出，程序崩溃。在实现的过程中，当上下滑动列表时，列表的图片很容易有二次渲染的情况，通过监听界面布局的状态，决定是否开始加载图片，避免了二次渲染造成的滑动卡顿。

5) 实现运用并发请求消息队列：Volley 的消息循环采用具有优先级排序功能的 BlockingQueue 来处理。在 RequestQueue 类中含有一个优先级 BlockingQueue 的实例，所有的消息都将添加到这个 BlockingQueue 当中，而后根据不同的情况选择分配给两个对象——NetworkDispatcher 或者 CacheDispatcher，由它们来 Handle 这些消息。Volley 的 Priority BlockingQueue 是来自于 java 的 Concurrent 包下面的 BlockingQueue 类。Java1.5 版本之后 Concurrent 提供的 BlockingQueue 类很好的解决了在多线程（Multi-Threading）情况下，怎样才能让数据高效安全传输的问题。有了这些高效并且线程安全（Thread Safety）的集合类，我们可以快速方便的搭建高质量的 Multi-Threading 程序。

6) 系统的测试与优化：在完成系统基本功能后对企业信息聚合平台进行测试并根据结果来进行调整，以提高程序的稳定性和鲁棒性，以及响应速度。





## 1.3 本文的组织与结构

第一部分：绪论。主要介绍了企业信息聚合平台产生的社会背景与现状，创新点与先进性，以及本技术报告的研究背景和研究工作安排；

第二部分：相关知识和技术简介。主要对整个企业信息聚合平台中所用到的一些算法原理，编码技术基础和理论背景知识进行了简要介绍。介绍了什么是 material design UI 的设计风格，LRU 缓存原理，BlockingQueue 在移动端 HTTP 请求中的应用，第三方地图服务的 API 模型等；

第三部分：项目工作和开发结果。分别介绍了本项目的两个主要组成部分，web 管理端和移动 Android 端的需求功能、模块分类还有业务流程；

第四部分：方案选择与结果分析。总结了本项目如何选择分析第三方的地图定位方案，使用并发请求消息队列对请求响应时间的提升，使用缓存技术对 UI 界面渲染的性能的提高结果分析和评价对；

第五部分：讨论与体会。总结了本项目的的工作情况，全开发过程和进度安排。列举了再开发过程中出现的问题及其解决。

## 2 文本相关知识和技术

### 2.1 HTTP 并发请求队列

#### 2.1.1 HTTP 并发请求队列简介

HTTP 协议是网络世界上最常用的通信协议之一。几乎每种语言都对其有着自己的封装与应用。Java 的 HTTP 协议官方的封装是 `HttpConnection` 这个类，它提供了最基础的 Http 协议支持，与请求接发。但是，其封装过于基础，在某些应用场景下，无法适应软件的需求，显得有些力不从心。所以有些许多第三方组织和个人，对其有了二次封装，提供了令人欣喜的功能。并发的队列请求就是其中之一。

一般来说，发送 HTTP 请求，每次只能发送一个请求（request），包含一个头部（requests header）和正文（requests body），然后等待收到响应体（response body）。这就算完成了一次请求。但是如果在需要高频率发送 HTTP 请求时，每次都重新建立 HTTP 请求，是一个很低效且很消耗计算机资源的事情。Volley 库中所提供的以创建一个全局的 `BlockQueue` 的方式，来构造一个 HTTP 请求队列，并且创建的一个线程来管理这个队列的网络调度，和创建了另一个线程来管理缓存的请求队列。这样大大提高了同时间多个请求发送接收的效率。

#### 2.1.2 Volley 的基本架构

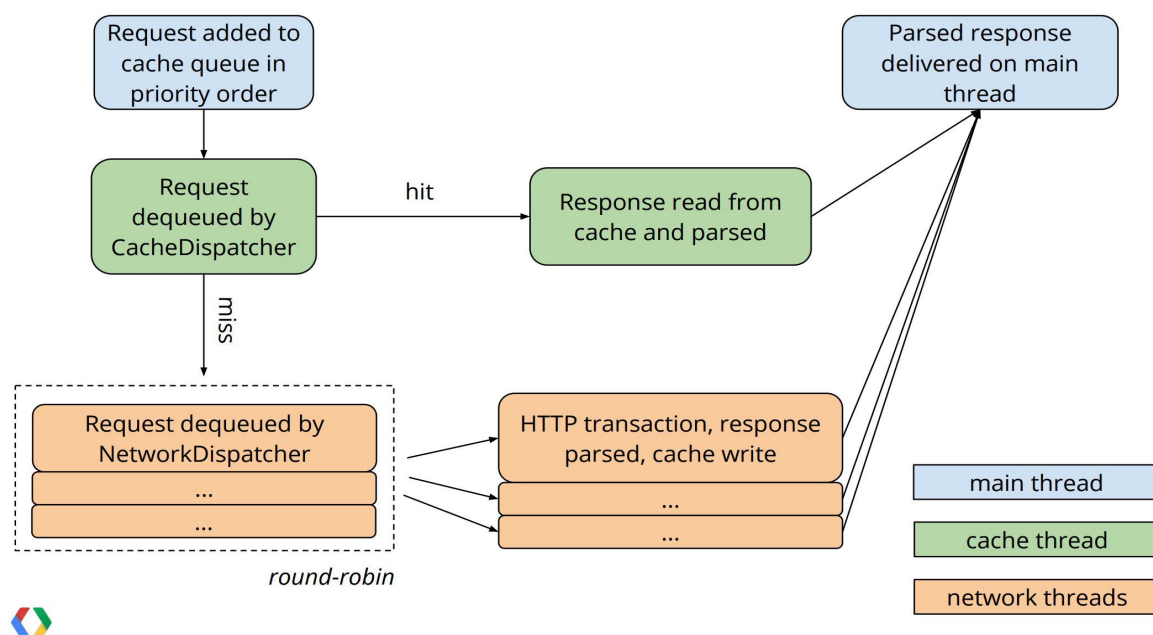


图 2.1.2-1 Volley 的架构图



如 2.1 图所示，Google 官方给出的这一份图中，Volley 启用了三线程，分别为 1 个蓝色的 UI 线程（主线程），1 个绿色的缓存 cache 线程，多个黄色的网络 network 线程（缺省值 5）。UI 线程承担往请求队列添加请求，展示请求的 response；缓存线程的工作是去检测匹配 Cache 区域，Cache 命中后会直接把匹配到的结果发送到主线程，让主线程做处理；Network 线程其实不止一个线程，其实是一个线程数为某一个值的线程池。这些线程会在同一时刻启动，并连续不断的从 RequestQueue 中获取待执行的包含一个 network request 的 task，task 执行完成后，会将请求的 response 分发到 UI 线程。

Volley 的一些核心类的简要分析：

**Request:** 表示一个请求的接口，它下面对不同数据类型如 json，字符串，图片有着不同的具体子类实现，细分为某一个特定类型的 Network Request。

**RequestQueue:** 整个框架最基础的类之一，表示 HTTP 请求的队列，所以请求都会进入它。而这个类内部拥有 ResponseDelivery，CacheDispatcher，多个 NetworkDispatcher，三种的实例，分别承担分发返回的结果，为需要存进 Cache 的请求进行分发的线程，为网络的请求进行分发的线程的功能。当 RequestQueue 启动时，网络和缓存的两个 Dispatcher 线程也会被开启。CacheDispatcher，NetworkDispatcher 都是基于 BlockQueue 的实现，他们有各自的队列，一旦有请求添加到队列中来就会立马执行出队的操作（根据请求的优先级），也就是取出请求，去获得返回体。

当在程序中调用了这行代码：

```
application.getRequestQueue().add(jsonrequest); //将请求加入到队列
```

Add 方法会将 Json 的请求添加进到 RequestQueue 里。然后 CacheDispatcher 线程会从这个 queue 中获取到请求，检查缓存中是否有请求的返回体。与 Cache 中的返回头匹配到后，CacheDispatcher 会直接将结果返回到 UI 线程中，完成请求。如果没有命中，则会将“处理权”转让给 NetworkDispatcher，由他从 queue 中取出请求，进行请求发送传输，接收解析，写进 Cache，把请求的 response 分发到 UI 线程。

根据以上流程，我们可以看出，当我们数据量不大的频繁网络操作时，会不断向请求队列添加请求，同时，Volley 也会不断地根据请求优先级，通过 NetworkDispatcher 进行请求传输，把返回写入缓存。请求数量越多，每个请求数据量越小，会让缓存的响应越多，那么缓存命中率就越高。直接从磁盘缓存中获取响应体，比在不稳定的移动网络

中获取响应体，拥有更好的实时性，更适应 Android 移动端这种轻量级数据传输，性能要求高的场景。

## 2.2 LRU 缓存技术

### 2.2.1 原理

LRU（Least recently used）算法中文名是：最近最少使用优先算法。算法基于局部性原理来排除数据。它的主要概念可以这样描述，如果某一段数据最近的时间被访问过，那么未来临近的时间被访问的概率会比较大，所以会优先保留这段数据。

LFU 算法全称——Least Frequently Used，和 LRU 算法的不同之处，LRU 算法的排除规则是依据时间长短排除，对比 LFU 算法是依据访问的次数的。LRU 是著名的最常用于“页面置换算法”的，在分页的虚拟内存上面应用的最多。计算机领域有个公认的原理——局部性原理：在一段时间内，如果前面有几个指令被频繁的执行，那么这几条指令有很大概率在临近的一段时间内执行。相反，如果一个指令较长时间没有被使用，则会很可能在将来很大一截时间内会不被用到。计算机之所以存在速度在 memory 之上的 cache，也是基于此。

### 2.2.2 实现

下面简单的介绍一下算法领域最常见的 LRU 的实现思路：

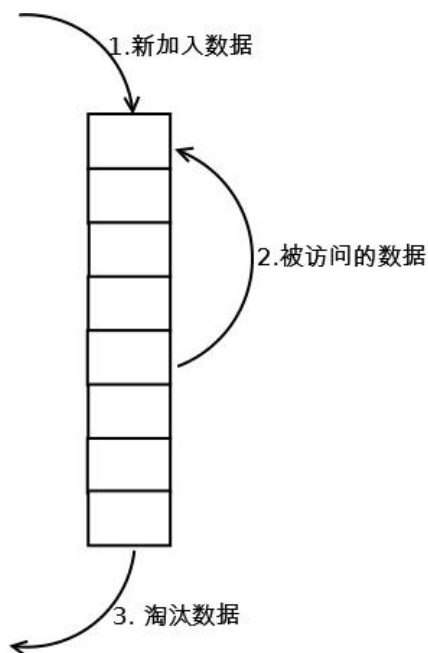


图 2.2.2-1 LRU 算法原理



1. 每当有新的数据来，把它添加到链表头；
2. 每当缓存区命中时（即 Cache 区域的数据和添加的数据相同），则将这段数据转移到链表头；
3. 如果链表溢出了，就把表尾溢出的节点移除。

然而 LRU 算法也是有着其自身的缺点。首先，它需要遍历链表，而且将数据转移到头部这些操作；其次，当出现周期性质的操作时，LRU 会出现较大的 miss 情况，有着很巨大的缓存污染。所以应用到具体场景时，需要考虑到是否能够规避这些弊端。

LRU 伪代码实现：

```
public class LRULinkedHashMap<K, V> extends LinkedHashMap<K, V> {
    private final int maxCapacity;

    private static final float DEFAULT_LOAD_FACTOR = 0.75f;

    private final Lock lock = new ReentrantLock();

    public LRULinkedHashMap(int maxCapacity) {
        super(maxCapacity, DEFAULT_LOAD_FACTOR, true);
        this.maxCapacity = maxCapacity;
    }

    @Override
    protected boolean removeEldestEntry(java.util.Map.Entry<K, V> eldest) {
        return size() > maxCapacity;
    }

    @Override
    public boolean containsKey(Object key) {
        try {
            lock.lock();
            return super.containsKey(key);
        } finally {
            lock.unlock();
        }
    }
}
```

```
@Override
public V get(Object key) {
    try {
        lock.lock();
        return super.get(key);
    } finally {
        lock.unlock();
    }
}

@Override
public V put(K key, V value) {
    try {
        lock.lock();
        return super.put(key, value);
    } finally {
        lock.unlock();
    }
}

public int size() {
    try {
        lock.lock();
        return super.size();
    } finally {
        lock.unlock();
    }
}

public void clear() {
    try {
        lock.lock();
        super.clear();
    } finally {
        lock.unlock();
    }
}

public Collection<Map.Entry<K, V>> getAll() {
    try {
        lock.lock();
        return new ArrayList<Map.Entry<K, V>>(super.entrySet());
    } finally {
        lock.unlock();
    }
}
}
```

图 2.2.2-2 LRU 算法代码





## 2.3 RichEditor 的实现

### 2.3.1 简述

本项目开发时遇到了比超出设计之初所预料到的难点。比如在处理手机端图文混排时，就遇到了难点。因为本项目的 Android 端不只是为了展示类似于网页的图片文字混排的功能，还需要能够编辑发布它。当时在手机 APP 领域，还没有较好的实现方法，只有微信的公众号文章和简书 APP 的文章有这个实现。但是他们都是互联网行业里面的技术巨头公司，才以良好的用户体验实现了这个功能，而且没有开源他们各自的实现的思路与方案。我只有查阅大量资料，了解他们具体怎么实现和寻找第三方开源的实现项目。

根据查阅的资料，发现微信和简书是各自使用的不同的实现方式。微信它有自己的框架，是基于 H5 技术来处理移动端的页面问题。微信公众号的文章，都是利用它自己的渲染框架，转换成手机网页界面，在 Android 手机端的一个 webview 控件中显示，这样完成了类似于浏览器一样的图片、文字混合排列的效果。而简书这款应用，则是开发了一个支持 Markdown 语法的移动端编辑器。利用 Markdown 语言某些标签，嵌入图片的 URL，添加文字的样式来显示图文混排效果。我要从头实现类似于他们的功能，需要巨大的工作量和良好的技术设计，才能完成这种移动端的编辑器。

虽然本人以前掌握了一些 Android 方面的开发经历，但是这种框架性质的开发，还是给了我不少的困难。但是，困难总是要克服的，不能因为难就不去研究，不去实现。在网上不断搜索，我根据一个国外的博客找到了思路。他在 github 上提供了一个初级的 RichEditor 的实现模板。思路是编写一个 RichEditor 来重新实现 Android 的 webView 控件。运用属于前端领域的 JavaScript 和 CSS 技术，在 webview 里面渲染出图文混排的效果。字体的样式种类，文本的操作，事先在 RichEditor 中定义好。然后在底部创建一个控制栏，根据控制栏中的不同功能的点击按钮，去调用 RichEditor 的展示不同的样式字体。RichEditor 在利用保存的图片 URL，向服务器请求图片数据，进行展示。这样就基本实现了 Android 端的 RichEditorView 的编辑功能和浏览功能。

### 2.3.2 部分实现展示





图 2.3.2-1 RichEditor 底部工具栏

RichEditor 底部工具类的种类对应代码实现。

//富文本支持样式的类型

```
public enum Type {  
    BOLD,  
    ITALIC,  
    SUBSCRIPT,  
    SUPERScript,  
    STRIKETHROUGH,  
    UNDERLINE,  
    H1,  
    H2,  
    H3,  
}
```



```
H4,  
H5,  
H6  
}  
  
// 对 webview 组件的继承扩展  
public class RichEditor extends WebView  
  
// 对文本改变的  
public interface OnTextChangeListener {  
  
    void onTextChange(String text);  
}  
  
// 初始化检测接口  
public interface AfterInitialLoadListener {  
  
    void onAfterInitialLoad(boolean isReady);  
}
```

图 2.3.2-1 RichEditor 部分代码展示



## 3 系统的设计与实现

### 3.1 系统的概况

开发供企业可以在网络上发布动态，用户可以搜索、浏览、评论的稳定系统平台和管理员可以在后端审查、修改的后台应用，操作简单，能良好的运行。本应有的使用人员应该是，希望能在一个平台上获知不同企业、不同行业信息的人员和发布企业信息的人员。维护人员：熟悉该平台的运作流程，熟悉计算机的各项操作，拥有敏锐的洞察力。

社会上一些企业，公益组织的活动、宣传信息等，不仅庞杂无穷，并且由于虚假、诈骗信息的存在，许多关注公司企业信息的人员无法获取到比较完善系统化的企业信息。即使是通过搜索引擎检索，也无法保证其时效性，所以建立一个企业和个人之间一个 B2C 的交流社区，有待挖掘的商业领域。

该平台为一个网络自发型平台，以 B2C 模式创建。目标是为需要获知不同行业和企业活动、最近动态的信息的个人，创建一个可靠、信息准确以及信息共享交流的平台。即个人能通过该平台，借助该平台在某些领域内聚合网络资源，扩张他个人的人际辐射力和影响力。一方面，以最短的时间获取企业最为准确的、最新动态的信息；另一方面，它也是一个企业和企业客户之间的一个重要的及时的交流平台。

本项目为了实现使用者和管理者的区分，将只为使用者的提供移动客户端，管理者提供 web 管理端。这两个都为客户端，共同与服务器端进行信息交互。当然，移动端也有少量通过 API 与第三方平台的服务器端进行信息交互的情况，如移动端的地图定位，地址搜索时，我使用了高德地图所提供的服务。

在 web 管理端方面，考虑到管理端功能实现的简洁性和稳定性，直接采用的是最基本的 J2EE 项目，基于 servlet 和 java server page 等技术。在 web 管理端的功能划分方面，我主要分为三个模块：企业资源管理，活动项目资源管理，用户信息管理。

### 3.2 系统开发计划及阶段性完成情况

表 3.2-1 企业信息聚合平台开发计划表

N O	任 务	预计开 始日期	预计 完成 日期	天 数	2016 年			2017 年				
					10	11	12	1	2	3	4	5
1	市场 调研	10.1	10.20	20	√							
2	需求	10.20	10.30	10	√							



	分析											
3	技术分析 及架构设计	10.30	11.9	10	√	√						
4	服务器环境搭建	11.9	11.19	10		√						
5	研究 第三 方 API 调用	11.19	12.19	30		√	√					
6	数据采集	12.19	12.25	6			√					
7	界面设计	12.25	12.30	5			√					
8	界面实现	12.30	1.10	10			√	√				
9	系统功能设计	1.10	1.15	5				√				
10	系统功能实现	1.15	2.15	30				√	√			
11	数据库实现	2.15	2.20	5					√			
12	各模块整合	2.20	3.5	15					√	√		



1 3	测试 分析	3.5	3.30	25						√		
1 4	项目 总结	3.30	4.5	5						√	√	
1 5	论文 研写	4.5	5.1	25							√	√

表 3.2-2 短信过滤系统项目阶段性完成情况

N O	任 务	预计开 始日期	预计 完成 日期	天 数	2015 年			2016 年				
					10	11	12	1	2	3	4	5
1	市场 调研	10.1	10.15	15	√							
2	需求 分析	10.15	10.23	8	√							
3	技术 分析 及架 构设 计	10.23	11.2	10	√	√						
4	服务 器环 境搭 建	11.2	11.11	9		√						
5	研究 第三 方 API 调用	11.11	12.20	40		√	√					
6	数据	12.20	12.25	5			√					



	采集											
7	界面设计	12.25	1.5	10			√	√				
8	界面实现	1.5	1.17	12			√	√				
9	系统功能设计	1.17	1.25	8				√				
10	系统功能实现	1.25	2.27	32				√	√			
11	数据库实现	2.27	3.4	6					√	√		
12	各模块整合	3.4	3.9	5						√		
13	测试分析	3.9	3.29	20						√		
14	项目总结	3.29	4.2	4						√	√	
15	论文研写	4.2	4.28	26							√	

### 3.3 系统的详细设计与实现

本项目为了实现使用者和管理者的区分，一是将只为普通使用者的提供搜索、浏览、评论等功能的稳定系统移动客户端，二是为管理者提供可以在后端审查、修改的后台应用，操作简单，能良好的运行 web 管理端。

#### 3.3.1 企业信息聚合平台 Android 移动客户端

移动客户端的功能较为复杂，用户也是面向社会大众。主要是让普通用户可以在这个网络自发型平台上，根据自己需要，获知不同行业和企业的活动、最近动态的信息，



并且加入参与活动，通过评论进行反馈。创建一个可靠、信息准确以及信息共享交流的平台，也是一个对于企业来说，通过在平台注册，并且完善企业信息获得在该平台发布项目的权力，这给企业创造了一个自我展示的平台。

### 3.3.1.1 系统主要功能

经过资料查阅，分析了如何建立企业信息聚合平台的移动端的功能，与导师进行研究探讨后，将整个 Android 移动客户端的主要功能需求和流程抽象描述如下：

- 1.普通用户和会员用户的注册和登录。
- 2.活动项目的浏览：分别有项目列表查看，模糊查询，详情查看。
- 3.对活动项目的支持：分为收藏项目和评论项目。
- 4.在地图上查看附近的企业，及其发起的活动项目内容及位置地址。
- 5.搜索注册的企业信息，简介。
- 6.个人信息管理：姓名，年龄，性别，是否会员，上传个人头像。
- 7.注册企业会员功能：发布新项目，添加自己所属企业基本信息。

根据以上主要功能需求和流程抽象分析，给出其现在的主要的功能。如图 3.1.1 所示的客户端部分：



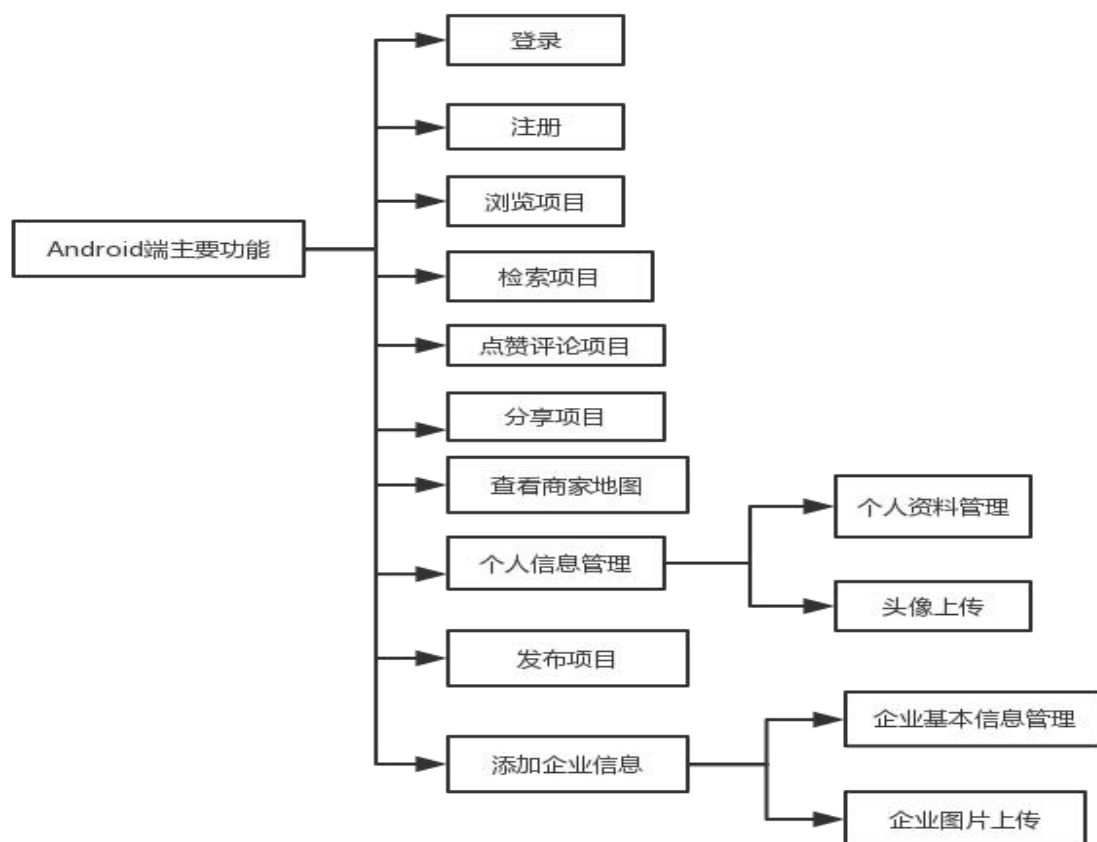


图 3.3.1.1-1.Android 移动客户端功能划分

### 3.3.1.2 系统模块功能结构化描述

表 3.3.1.2 -1 系统模块功能结构化描述

编号	名称	简述	输入数据	处理（算法）	输出
SRS-001	浏览所有项目列表	普通用户浏览所有项目	无	根据时间日期进行排序	显示项目列表，包括每个项目的活动展示照片、活动发起人、活动时间、点赞、评论等



编号	名称	简述	输入数据	处理（算法）	输出
SRS-002	浏览单个项目详情	普通用户浏览单个项目详情	所选项目编号	基于移动端富文本框图文混排	显示活动发起人、活动时间、和活动主题以及活动详情（使用图文混排界面），下面是点赞数，以及评论
SRS-003	搜索项目	普通用户通过模糊搜索	模糊搜索：项目名称或项目类别	模糊匹配算法	符合搜索条件的项目列表，包括每个项目的活动展示照片、活动发起人、活动时间、点赞、评论等
SRS-004	分享项目	普通用户可分享某个项目至微信、微博等第三方社交平台	所选项目编号、所选社交平台名称	无	跳转到相应第三方社交平台的页面，并显示已成功分享用户的内容
SRS-005	用户注册	普通用户注册成为注册用户	用户名、密码、昵称、真实姓名、地址、性别、上传头像等注册信息	将用户提交的信息写入数据库	文字提示是否注册成功
SRS-006	用户登录	注册用户登录	用户名，密码	验证该用户名有效性	登录是否成功
SRS-007	发布项目	注册用户可发布项目	包括每个项目的活动主题、活动展示照片、活动时间和活动地点	将用户提交的信息写入数据库	文字提示是否提交成功



编号	名称	简述	输入数据	处理（算法）	输出
SRS-008	评论项目	注册用户可评论项目	评论项目编号、评论人编号、评论内容、评论时间	根据评论项目编号在数据库中找到对应的项目，并将评论写入数据库	文字提示是否评论成功
SRS-009	支持项目	注册用户可为项目“点赞”来支持项目	支持项目编号	支持的项目上支持数目加一	显示支持的项目上支持数目已加一
SRS-0010	个人信息管理	注册用户查看自己以及编辑个人信息	用户编号	自增处理	用户个人信息
SRS-0011	添加企业信息	注册用户可以添加自己公司的信息	公司的名称、公司地址、所属行业、简介、企业展示界面	无	提示是否添加成功
SRS-0012	查看商家地图	注册用户可以通过地图上表示的商家进行查找	通过地图上选择商家所在位置	调用地图接口	显示企业的详情的界面
SRS-0013	搜索企业	注册用户可以通过输入关键字查找	主要是企业名称进行查找	模糊搜索	显示查找商家的列表
SRS-0014	用户管理	后台管理员对注册用户进行增删改查操作	相应的后台网页操作		



编号	名称	简述	输入数据	处理（算法）	输出
SRS-0015	项目管理	后台管理员对项目进行增删改查操作	相应的后台网页操作		
SRS-0016	企业管理	后台管理员对企业进行增删改查操作	相应的后台网页操作		

### 3.3.1.3 对性能的规定

#### 精度

- 传输过程中的精度：数字的小数点之后面保全三位有效字；
- 输出数据精度：数字的小数点之后面保全三位有效字；
- 输入数据精度：数字的小数点之后面保全三位有效字；

#### 时间特性要求

- 实时反馈时间：不多于 1s。即用户在软件的每个功能函数的点击、触摸、键盘输入等等动作事件的返回时长，应该规定在 1s 以内，以给用户舒适的体验；
- 更新处理时间：不超过 1s；
- 数据的变换和传输时间：不多于 1s；

#### 灵活性

- 软件在最初的概要设计上和具体完成上，均需要考虑到其在不同运行环境的状况。要求能在不同的运行环境稳定执行，即在 Android 4.0 及其以上版本里，运行时能够兼容；
- 当输入整数能自动转换为系统规范的三位小数的精度。如有错误类型数据，应及时提示错误信息
- 软件应满足开闭原则，以适应将来有可能会出现的更改
- 软件系统进行升级时保证用户数据的安全性



### 3.3.1.4 输入输出要求

#### 输入

该软件在客户端支持以下两种输入方式：触摸输入和文字输入

- 触摸输入：用户用手指在屏幕上触摸，点击，滑动，如果该位置是系统允许的有效区域，则定义为用户输入
- 文字输入：用户在指定文本框中以文本的形式输入，被系统校验为有效输入后，定义为用户输入

#### 输出

##### 1) 用户登录及注册服务

- 用户输入无效时，系统自动查询，返回相似的有效输入提示
- 系统友好地输出是否登录注册成功

##### 2) 项目浏览服务

- 对所有项目列表
- 对单个项目详情

##### 3) 项目查询服务

- 用户输入无效时，系统自动查询，返回相似的有效输入提示
- 列表显示查询结果

##### 4) 项目评论及支持服务

- 用户输入无效时，系统自动查询，返回相似的有效输入提示
- 系统友好地输出是否评论或支持成功

##### 5) 分享服务

- 跳转到相应社交平台的分享页面，并输出展示已成功分享的用户内容。



### 3.3.1.5 核心接口展示

注册请求接口设计:

表 3.3.1.5-1 注册函数功能设计

NO	接口名	说明
1	StringRequest()	向服务器发送请求进行注册

```
StringRequest(  
Request.Method.POST,  
url,  
new Response.Listener<String>(){} ,  
new Response.ErrorListener() {}  
)
```

侧滑界面初始化函数:

表 3.3.1.5-2 侧滑界面初始化函数功能设计

NO	函数名	说明
1	initSlidingMenu()	进入主界面加载侧滑布局菜单，根据屏幕宽度，灵活设置菜单宽度占4/5.

```
public void initSlidingMenu() {  
    slidingMenu = new SlidingMenu(MainActivity.this);  
    slidingMenu.setMode(SlidingMenu.LEFT);  
    slidingMenu.setTouchModeAbove(SlidingMenu.TOUCHMODE_MARGIN);  
    //值越大，侧滑越窄  
    slidingMenu.setBehindOffset(200);  
    slidingMenu.setBehindScrollScale(1);  
    slidingMenu.setShadowDrawable(R.drawable.xlistview_arrow);  
    slidingMenu.setFadeDegree(1);  
    slidingMenu.attachToActivity(this, SlidingMenu.SLIDING_CONTENT);  
    slidingMenu.setMenu(R.layout.slide_layout);  
    // 设置专场动画效果  
    android.support.v4.app.Fragment slideFragment = new SlidingFragment(MainActivity.this);  
    getSupportFragmentManager().beginTransaction()  
        .replace(R.id.frag_slide, slideFragment).commit();  
}
```



图片初始化函数：

表 3.3.1.5-3 图片批量选择及添加上传初始化函数功能设计

NO	函数名	说明
2	initImageLoader(Context context)	初始化图片加载器，使用多线程加载机制，使用内存缓存机制，使用MD5摘要算法生成图片名称

```
//初始化 imageloader
private void initImageLoader(Context context) {

    ImageLoaderConfiguration.Builder config = new
    ImageLoaderConfiguration.Builder(context);
    config.threadPriority(Thread.NORM_PRIORITY - 2);
    config.denyCacheImageMultipleSizesInMemory();
    config.diskCacheFileNameGenerator(new Md5FileNameGenerator());
    config.diskCacheSize(50 * 1024 * 1024); // 50 MiB
    config.tasksProcessingOrder(QueueProcessingType.LIFO);
    config.writeDebugLogs(); // Remove for release app
    // Initialize ImageLoader with configuration.
    ImageLoader.getInstance().init(config.build());
}
```

### 3.3.1.6 表设计

系统逻辑结构设计

User 表

用户信息（User 编号，用户名，密码，姓名，性别，公司所在地址，所属服务队，头像地址，创建时间，更新时间）

Company 表

公司（Company 编号，所属用户编号，公司名称，地址进度，地址纬度，经营范围，所属行业，公司介绍详情界面，创建时间）

Project 表

项目（Project 编号，项目标题，活动时间，发起者编号，点赞数，封面照片，详情界面，创建时间）

Comment 表

评论（Comment 编号，对应 project 编号，对应 User 编号，评论内容，创建时间）





## 表解析

User 用户表  
Company 公司表  
Project 项目表  
Common 评论表

### 用户表解析

表 3.3.1.6-1 用户表解析

字段名	类型	允许为空	默认值	说明
id	INT	FALSE		用户 id
username	VARCHAR(45)	FALSE		用户名
password	VARCHAR(100)	FALSE		密码
header	VARCHAR(100)	TRUE		头像地址
name	VARCHAR(20)	TRUE		姓名
sex	INT	TRUE		性别
address	VARCHAR(45)	TRUE		地址
create_time	VARCHAR(50)	TRUE		创建时间
update_time	DATETIME	TRUE		更新事件

### 公司表解析

表 3.3.1.6-2 公司表解析

字段名	类型	允许为空	默认值	说明
id	INT	FALSE		公司 id
user_id	VARCHAR(45)	FALSE		所属用户的 id
company_name	VARCHAR(100)	FALSE		公司名称
address_longitude	VARCHAR(45)	FALSE		地址经度
address_latitude	VARCHAR(45)	FALSE		地址纬度
business_scope	VARCHAR(45)	FALSE		经营范围
industry	INT	FALSE		所属行业
show_page	LONGTEXT	FALSE		公司介绍详情界面 (html 文件)
create_time	DATETIME	FALSE		创建时间



## 项目表解析

表 3.3.1.6-3 项目表解析

字段名	类型	允许为空	默认值	说明
id	INT	FALSE		项目（活动）id
title	VARCHAR(100)	FALSE		项目标题
time	VARCHAR(45)	FALSE		活动时间
launcher_id	INT	FALSE		发起者 id
favorite	INT	TRUE		点赞数
cover_image	LONGTEXT	TRUE		封面照片
details_page	LONGTEXT	FALSE		详情界面（html）

## 评论表解析

表 3.3.1.6-3 评论表解析

字段名	类型	允许为空	默认值	说明
id	INT	FALSE		评论 id
project_id	INT	FALSE		对应的项目 id
user_id	INT	FALSE		对应的用户 id
content	LONGTEXT	FALSE		评论内容
create_time	VARCHAR(45)	FALSE		创建时间

## 概念模型

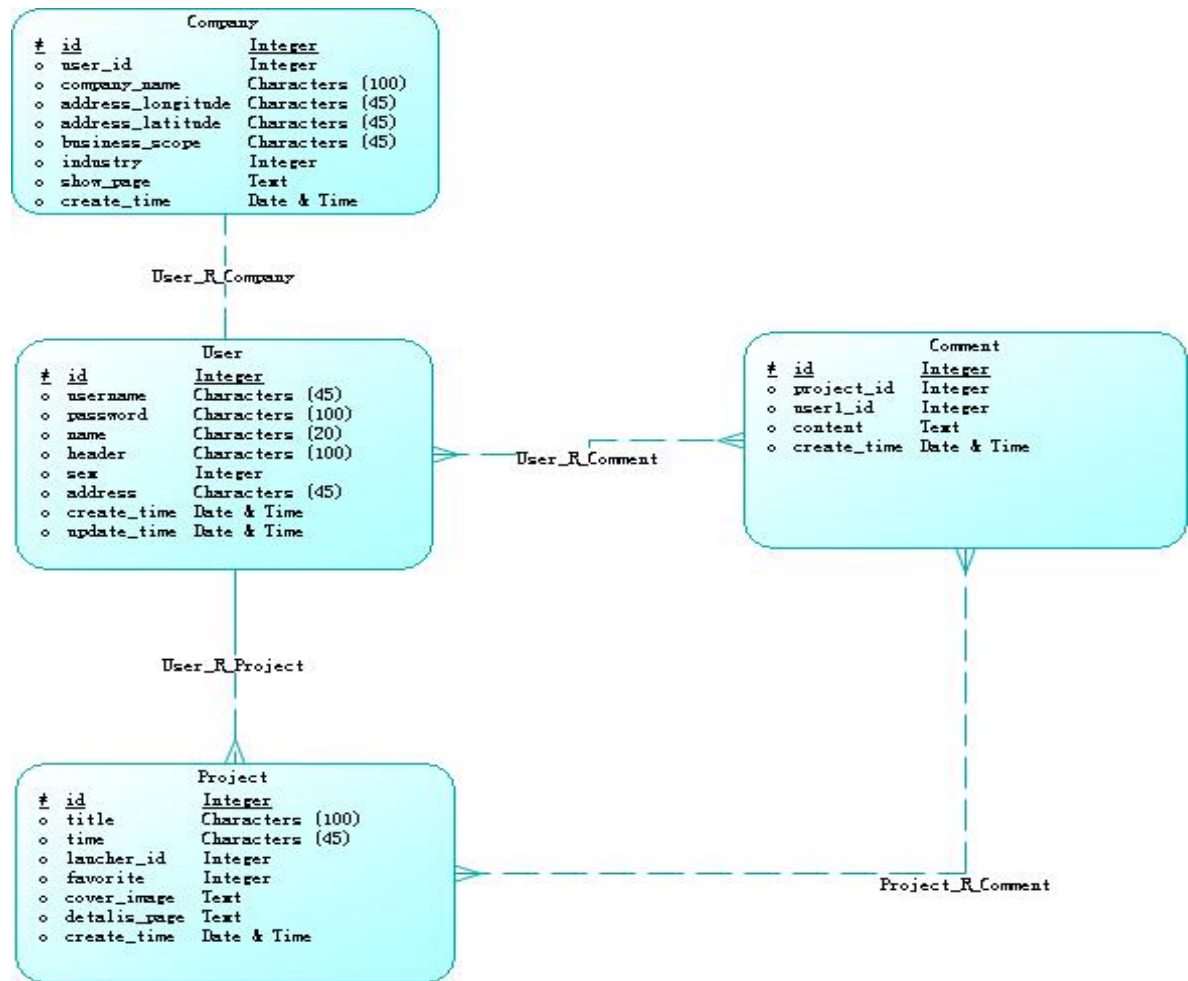


图 3.3.1.6-1 逻辑概念上的项目的实体关系图

## 3.3.1.7 系统业务流程

普通用户注册后，登录 Android 端，系统就会根据用户名密码进行检测校验，如果验证成功，应用将会进入主界面。主界面是通过查询服务器，获取最近发布的 20 个项目，更加日期相关程度进行列表展示。

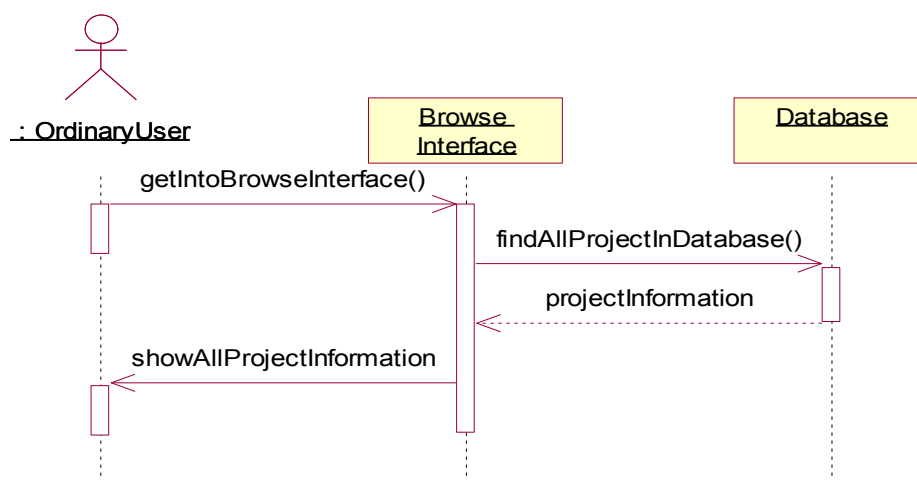


图 3.3.1.7-1 普通用户登录之后的函数调用流程图

用户进入地图界面，立即调用调用 GPS 定位获取当前位置，然后根据当前位置坐标，向服务器

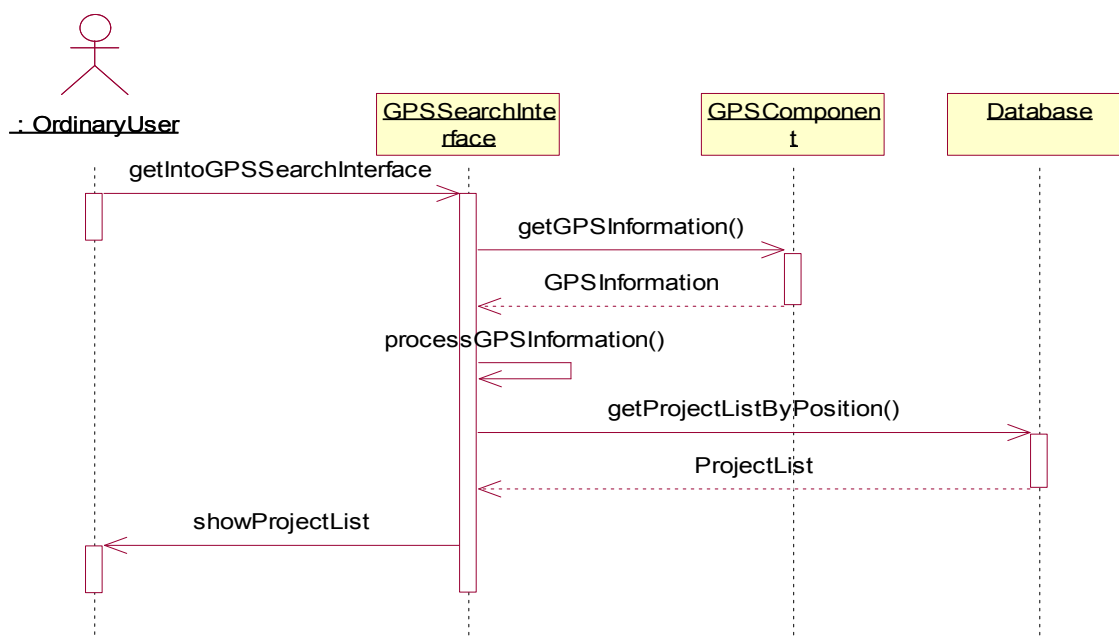


图 3.3.1.7-2 普通用户通过 GPS 查找商家流程图

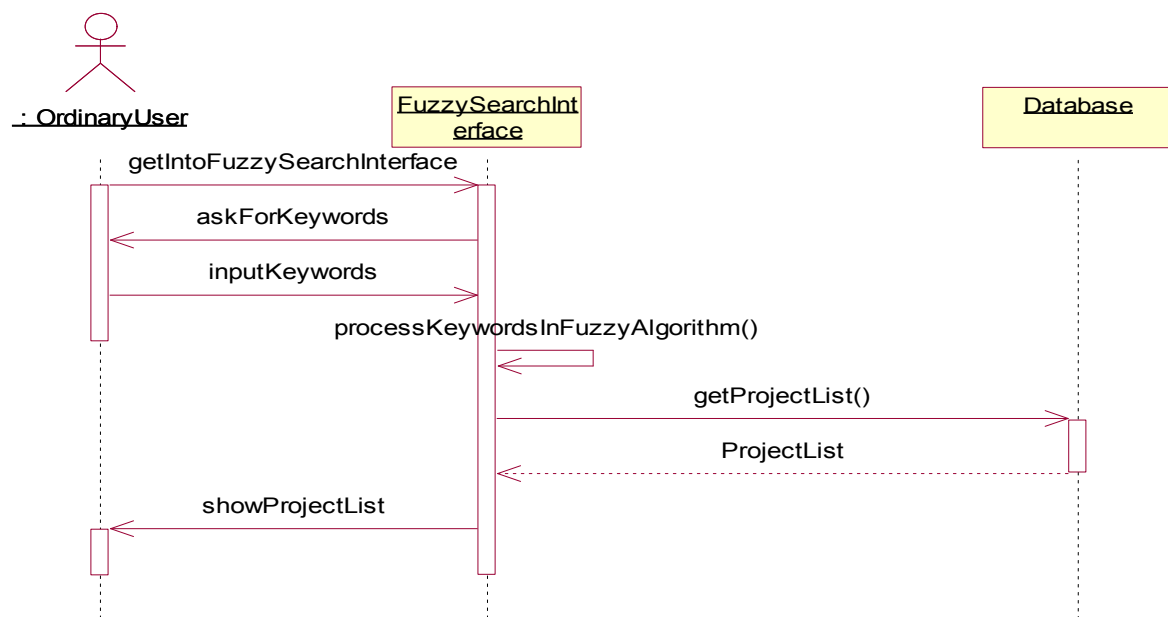


图 3.3.1.7-3 普通用户通过模糊搜索查找项目图

### 3.3.1.8 项目 UI 界面展示



图 3.3.1.8-1 企业信息聚合移动端主界面



图 3.3.1.8-2 个人资料展示界面



图 3.3.1.8-3 会员用户发布活动界面

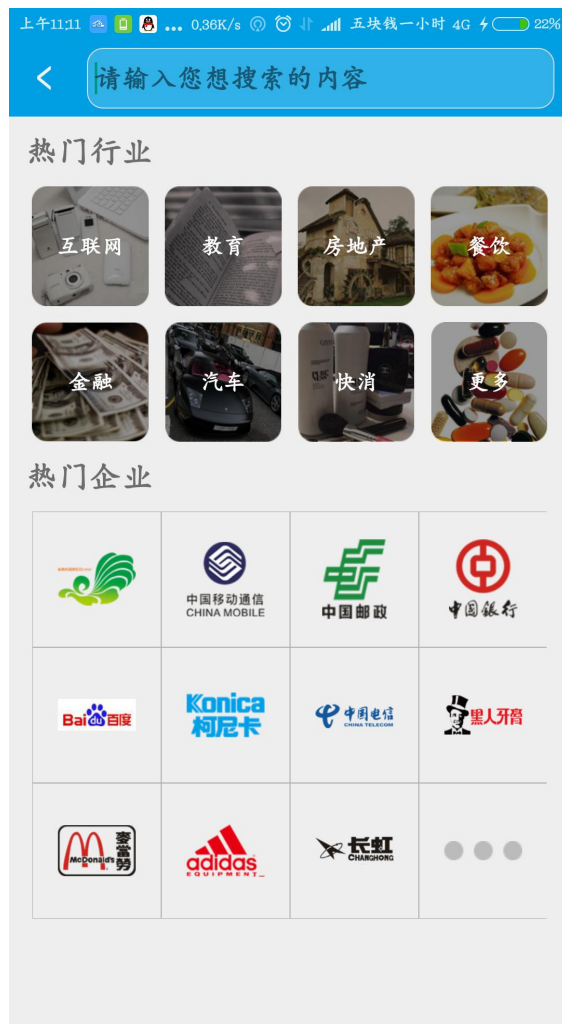


图 3.3.1.8-4 企业检索界面





### 3.3.2 企业信息聚合平台 web 后台管理端

在 web 管理端方面，考虑到管理端功能实现的简洁性和稳定性，直接采用的是最基本的 J2EE 项目，基于 servlet 和 java server page 等技术。在 web 管理端的功能划分方面，我主要分为三个模块：企业资源管理，活动项目资源管理，用户信息管理。三个模块都是由 servlet 来接收页面上传来的请求，并做一些安全性的逻辑处理，然后直接通过 JDBC，获取 MYSQL 数据库的连接，直接向 database 数据库的表进行 CRUD 操作。Web 管理端和后台服务程序运行在同一个服务器上，都拥有数据库的访问权限。Web 端所用的环境是 Apatch 和 tomcat 所组合的 web 容器。

#### 3.3.2.1 web 端主要功能

如同 Android 移动端一样，经过详细的查阅资料，再根据 web 管理端所提供的服务，对其功能有着如下的划分：

- 1.项目管理
- 2.企业管理
- 3.用户管理

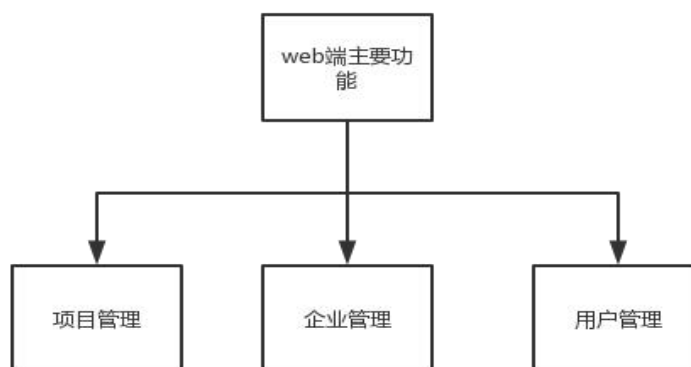


图 3.3.2.1-1 现行 web 管理端主要功能描述



### 3.3.2.2 web 端结构化描述

网页的界面的跳转操作都是封装在 `servlet` 包里，通过对数据的处理以及数据库的连接实现跳转条件，跳转到相应的界面。

**AddUserServlet:** 当管理员在网页选择添加用户选项时，通过该类中的方法进行跳转到添加用户的网页。

**DeleteUserServlet:** 当管理员在网页点击删除某个特定用户选项时，通过该类中的方法进行相应的操作，并且删除特定用户，更新数据库。

**CheckUserServlet:** 当管理员在网页点击查看用户选项时，通过该类并且查询数据库，显示所有用户。

**ModifyUserServlet:** 当管理员在网页点击修改用户选项时，通过该类跳转到修改用户信息界面。

**AddCompanyServlet:** 当管理员在网页点击添加企业选项时，通过该类中的方法进行跳转到添加企业的网页。

**DeleteCompanyServlet:** 当管理员在网页点击删除某个特定企业选项时，通过该类中的方法进行相应的操作，并且删除特定企业，更新数据库。

**CheckCompanyServlet:** 当管理员在网页点击查看企业选项时，通过该类并且查询数据库，显示所有企业。

**ModifyCompanyServlet:** 当管理员在网页点击修改企业选项时，通过该类跳转到修改企业信息界面。

**AddProjectServlet:** 当管理员在网页点击添加项目选项时，通过该类中的方法进行跳转到添加项目的网页。

**DeleteProjectServlet:** 当管理员在网页点击删除某个特定项目选项时，通过该类中的方法进行相应的操作，并且删除特定项目，更新数据库。

**CheckProjectServlet:** 当用户在网页点击查看项目选项时，通过该类并且查询数据库，显示所有项目。

**ModifyProjectServlet:** 当管理员在网页点击修改项目选项时，通过该类跳转到修改项目信息界面。

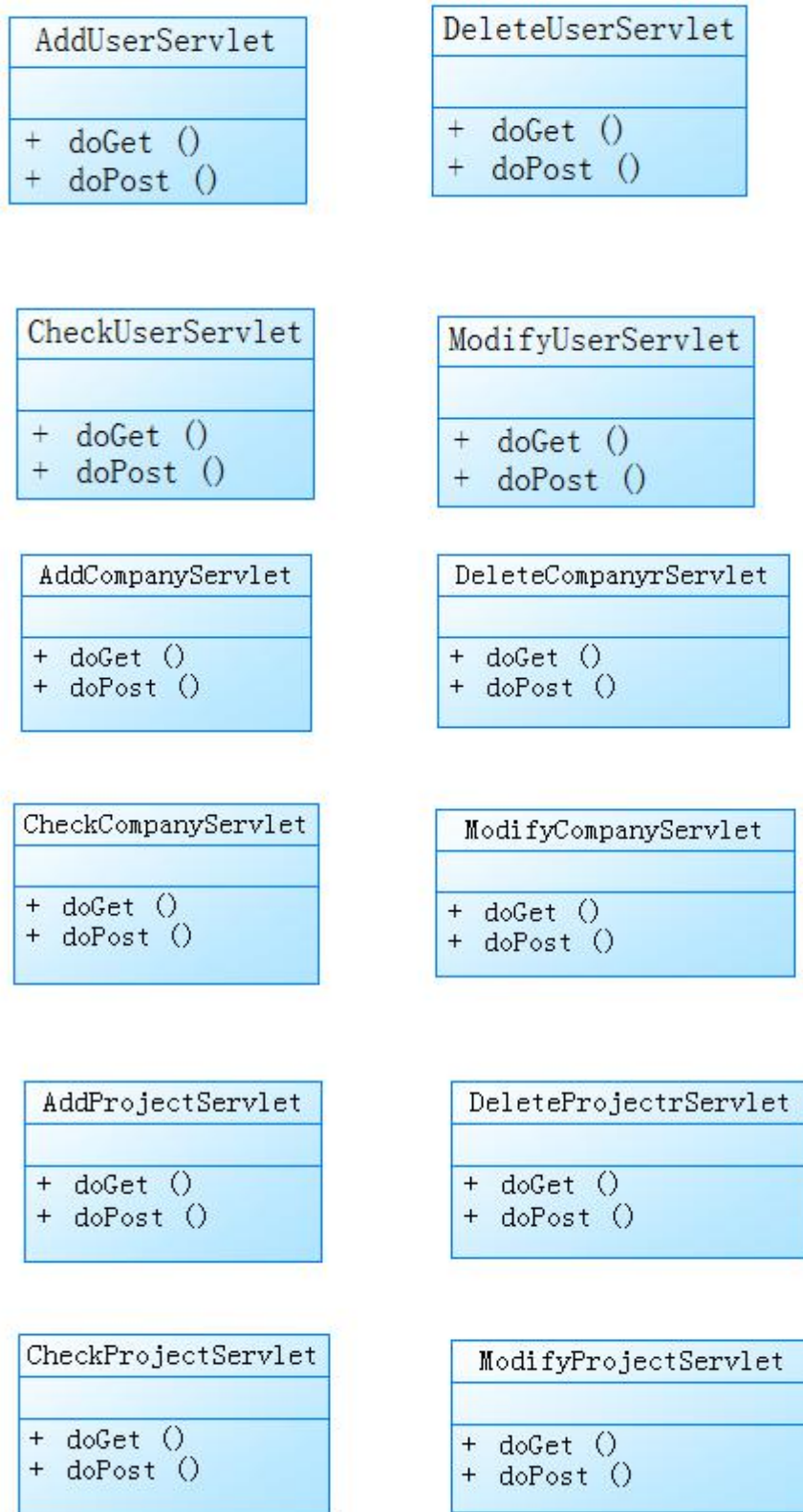


图 3.3.2.2-2 注册企业检索和会员用户发布活动界面



### 3.3.1.2.1 UI 模块

#### 模块概述

UI 模块即界面模块，主要负责控制 web 端的界面显示、实现人机交互部分。采用用户普遍喜爱的蓝色作为主基调，使用现流行网站都普遍采用的 fragment 布局，使用户更容易上手并习惯该系统。

UI 主要分为 4 个界面，登录成功后的主页(dashboard)页面，企业资源管理，活动项目资源管理，用户信息管理。其中企业管理界面，项目管理界面，用户管理界面采用列表的形式，可以复用同一个 UI 模块。而且三个界面的业务逻辑大致相同，所以采用同构控制逻辑，可以复用 UI 背后的行为控制模块。

#### 界面设计



图 3.3.1.2.1-1 web 端管理主页界面设计



The image shows a user registration form with the following fields and controls:

- 用户名: [text input field]
- 密码: [password input field]
- 姓名: [text input field]
- 性别: ☐ 男 ☐ 女
- 地址: [text input field]
- 增加: [blue button]

图 3.3.1.2.1-2 增加用户界面设计

### 3.3.1.2.2 项目管理模块

#### 模块概述

这一模块是担任对活动项目的数据进行管理。本模块从数据库中获取 project 表的信息，经过 JDBC 的调用以便进行后续操作。由于项目管理可以增加和修改项目，所以，web 端也实现了，富文本文档编辑、项目关键字模糊匹配、图片上传等功能，以匹配移动端发起项目的模块。

页面的前端是使用 jQuery 和 css3 来完成的，可以适配 IE8 也上的浏览器和 chrome 浏览器。JQuery 是相当轻量级的，使用 Packer 压缩后，体积不到 30KB。本项目所使用的 Ajax 异步请求，也是基于 jQuery 封装的一个函数 \$.ajax() 里，这个封装避免了浏览器的兼容性问题 and 某些 Request 对象的管理问题，帮助代码人员把注意力放在业务逻辑的实现上。



## 函数功能设计

表 3.3.1.2.2-1 项目管理模块函数功能表

NO	函数名	说明
1	public Project getProjectByID(int id)	Project 的 DAO 层单次 query 数据库操作，根据 project 的 ID 字段获取 project 的详细信息
2	public boolean add(String title, String time, int launcherId, int favorite, String coverImage, String detailsPage, String createTime)	Project 的插入数据库操作，根据输入的项目信息参数，向数据库指定表插入一行记录值。

### 3.3.1.2.3 企业管理模块

#### 模块概述

本模块负责对企业信息的管理。企业信息是指在，本平台注册的企业，所填写的必要信息，包括公司名，地址的经纬度，所属行业，创建时间，展示图片。由于公司数据和项目的数据有关联的成分，公司 ID 是作为项目表的外键。所以，我已经考虑到了，当公司更改之后，相应的项目数据一会同时做更新处理。

#### 函数功能设计

表 3.3.1.2.3-1 基于企业管理模块函数功能表

NO	函数名	说明
1	public Company getCompanyByID(int id)	根据企业 ID 获取企业的详细信息。
2	public void doPost(HttpServletRequest request, HttpServletResponse response)	删除公司的 POST 请求业务逻辑处理，请求转接，输入输出流获取。



### 3.3.1.2.4 用户管理模块

#### 模块概述

本模块负责对用户信息的管理。用户信息是指在，本平台注册的普通用户和会员用户。包含了 ID 编号，用户名，密码，头像地址，姓名，性别，地址，所属服务队，创建时间，更新时间等信息。用户信息是本应用的关键信息，特别是用户名和登录密码等，在数据操作的 DAO 层对于数据变更的情况，均有数据库事务的运用，防止因为异常导致出现坏数据的情况，影响程序正常运行。事务管理和回滚机制的使用，加强了程序鲁棒性和数据的安全性。

#### 函数功能设计

表 3.3.1.2.4-1 基于内容的过滤模块函数功能表

NO	函数名	说明
1	public void doPost(HttpServletRequest request, HttpServletResponse response)	该函数对用户信息的更新请求做了处理
2	public boolean update(String sql, Object... obj)	该函数在 DAO 层直接对 MYSQL 数据库进行了更新操作

#### 部分代码展示

数据库优化，数据库事务操作和回滚机制：

```
public boolean update(String sql, Object... obj) {  
    boolean flag = false;  
    con = getCon();  
    try {  
        con.setAutoCommit(false);  
        ps = con.prepareStatement(sql);  
        if (obj != null) {  
            for (int i = 0; i < obj.length; i++) {  
                ps.setObject(i + 1, obj[i]);  
            }  
        }  
    }  
}
```





```
flag = ps.executeUpdate() > 0 ? true : false;

} catch (Exception e) {
    e.printStackTrace();
    try {
        con.rollback();
        con.commit();
    } catch (SQLException e1) {
        e1.printStackTrace();
    }
} finally {
    closeAll();
}
return flag;
}
```

### 3.3.1.2.5 DAO 层逻辑展示

#### 模块概述

DAO 这一层级负责直接与本项目的数据库表打交涉。考虑到可扩展性原则，DAO 层有两个模块，一是定义的基本操作的接口类，二是对所定义的接口函数的给出具体化实现类。具体实现类可添加处理增删改查这种基本操作之外的，复杂操作如登录验证，获取所有用户列表等。

#### 函数功能设计

表 3.3.1.2.5-1 垃圾短信管理模块函数功能表

NO	函数名	说明
1	login()	查询数据库，添加实体
2	update()	修改用户信息，并且更新数据库信息
3	getUserByID()	根据用户的 id 号来获得用户实体
4	getAllCompany ()	根据 SQL 语句获取所有企业实体

## DAO 层类图展示及描述

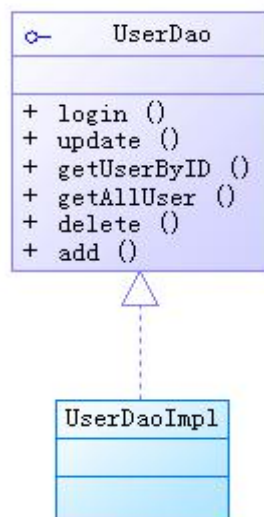


图 3.3.1.2.5-1 DAO 层类图展示

`login()` 查询数据库，添加实体

`update()` 修改用户信息，并且更新数据库信息

`getUserById()` 根据用户的 id 号来获得用户实体

`getAllUser()` 根据 SQL 语句获取所有用户实体

`delete()` 根据 id 号来删除实体

`add()` 增加用户，根据 SQL 语言写入数据库

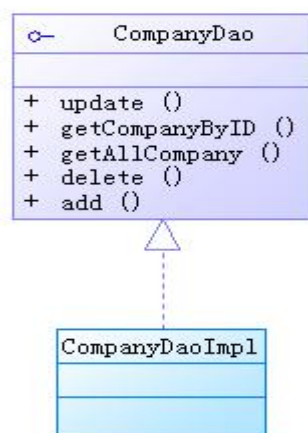


图 3.3.1.2.5-2 DAO 层类图展示

update() 修改企业信息，并且更新数据库信息  
getCompanyByID() 根据企业的 id 号来获得用户实体  
getAllCompany () 根据 SQL 语句获取所有企业实体  
delete() 根据 id 号来删除企业实体  
add() 增加企业，根据 SQL 语言写入数据库

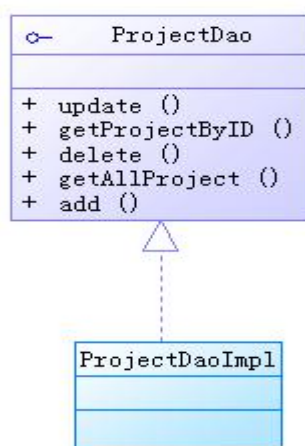


图 3.3.1.2.5-3 DAO 层类图展示

update() 修改项目信息，并且更新数据库信息  
getProjectByID() 根据企业的 id 号来获得项目实体  
getAllProject () 根据 SQL 语句获取所有项目实体  
delete() 根据 id 号来删除项目实体  
add() 增加项目，根据 SQL 语言写入数据库



### 3.3.2.3 web 端业务流程

在管理员注册登录后，进入主页页面，根据需求选择企业，项目，用户页面进入，点击页面对应的按钮，操作相应的数据，将数据的变动更新都持久化到数据库，然后再刷新页面进行展示，这一个循环的流程，就是 web 管理端的业务流程。

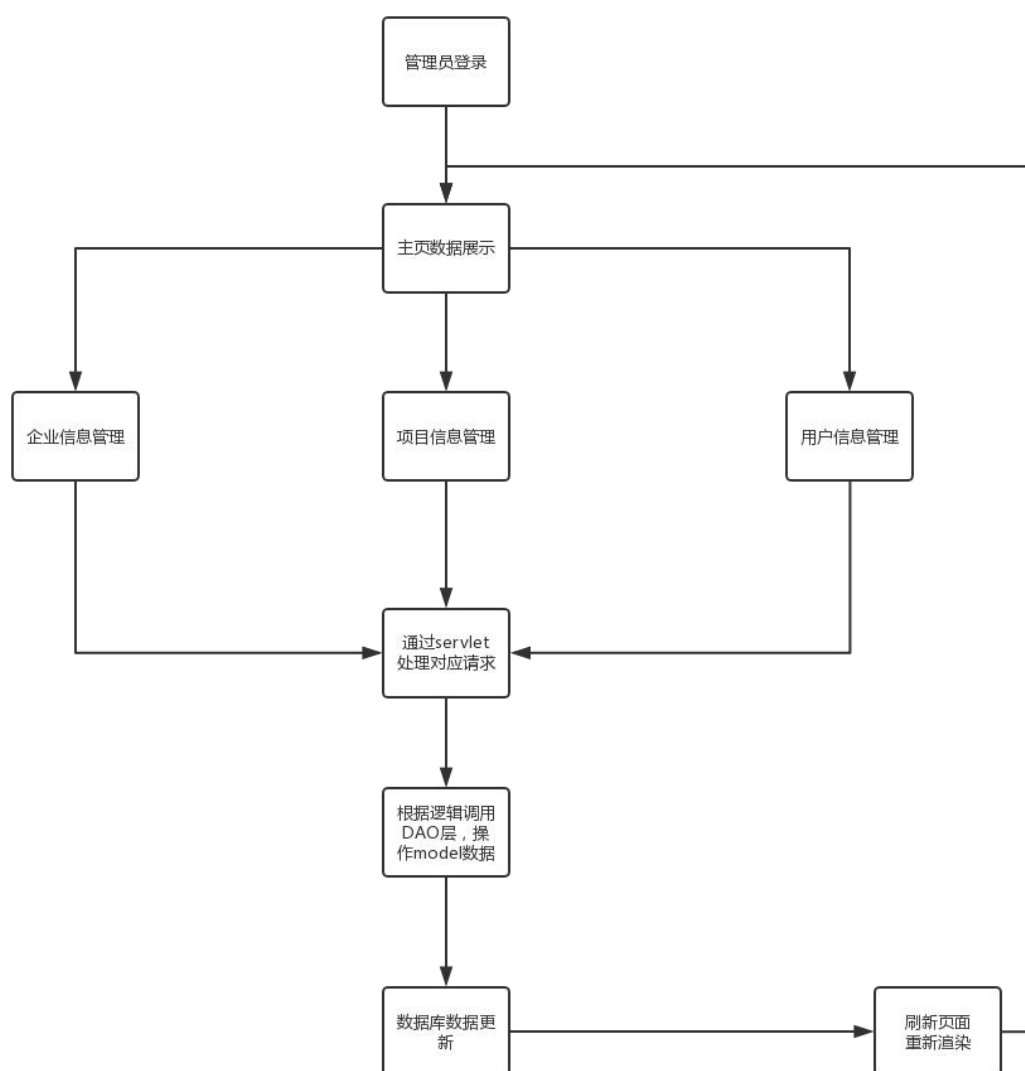


图 3.3.2.3-1 客户端垃圾短信分类处理流程



### 3.4 本章小结

本章介绍了本系统的两个主要组成部分，本项目为了实现使用者和管理者的区分，一是将只为普通使用者的提供搜索、浏览、评论等功能的稳定系统移动客户端，二是为管理者提供可以在后端审查、修改的后台应用，操作简单，能良好的运行 web 管理端。

分别在各自模块对主要功能，性能规定，输入输出，代码展示，业务流程，UI 界面进行了描述和图形化展示。并且列表解释了模块中使用的重要函数的具体作用以及用处。诸如表结构设计，异常处理机制，数据回滚，请求重发等封装后不易展现的功能，也做了代码展示和专门叙述。

## 4 方案选择与结果分析

### 4.1 软件运行环境

表 4.1-1 软件运行环境

	名称	版本	语种
操作系统	Android	4.0 以上	
操作系统的附加功能			
数据库平台	Mysql	5.0	
语言	JDK	1.5 以上	
邮件系统			
客户端软件			



## 4.2 硬件环境

表 4.1-2 硬件环境

硬件配置	最低配置	推荐配置
服务器	1CPU: P4 2.0G	1CPU: P4 2.8G
	Mem: 1G	Mem: 2G
	HD: 40G	HD: 120G
客户端	1CPU: P4 1.0G	1CPU: P4 2.0G
	Mem: 512M	Mem: 1G
	HD: 40G	HD: 120G



### 4.3 企业信息聚合平台用户可接受测试分析

用户反馈测试用例：

表 4.3-1 用户反馈用例

测试用例	用户反馈用例	
测试用例 ID	Test1	
输入/动作	期望的输出/相应	实际情况
点击用户反馈	出现输入反馈信息的输入框	出现输入反馈信息的输入框
测试用例 ID	Test2	
输入/动作	期望的输出/相应	实际情况
输入反馈内容，点击提交	提示反馈成功	提示反馈成功
测试用例 ID	Test3	
输入/动作	期望的输出/相应	实际情况
输入纯数字反馈内容，点击提交	提示用户重新输入	提示反馈成功
测试用例 ID	Test4	
输入/动作	期望的输出/相应	实际情况
输入无意义反馈内容，点击提交	提示用户重新输入	提示反馈成功
测试用例 ID	Test5	
输入/动作	期望的输出/相应	实际情况
不输入反馈内容，点击提交	提示用户输入信息	提示反馈成功
测试用例 ID	Test6	
输入/动作	期望的输出/相应	实际情况
输入反馈内容，点击取消	回到首页	回到首页
测试用例 ID	test7	
输入/动作	期望的输出/相应	实际情况
不输入反馈内容，点击取消	回到首页	回到首页
测试用例 ID	Test8	
输入/动作	期望的输出/相应	实际情况
点击返回上一步	回到首页	回到首页





搜索项目测试用例：

表 4.3-2 搜索项目用例

测试用例	搜索项目用例		
测试用例 ID	Test1		
输入/动作	期望的输出/相应	实际情况	
用户点击搜索按钮	进入搜索界面弹出键盘 监听输入	进入搜索界面弹出键盘 监听输入	
测试用例 ID	Test2		
输入/动作	期望的输出/相应	实际情况	
输入内容为空字符时， 点击开始搜索	提示输入内容为空，请 输入内容	提示输入内容为空，请 输入内容	
测试用例 ID	Test3		
输入/动作	期望的输出/相应	实际情况	
用户输入无意义的内容	根据用户内容无法匹配 出相应的项目，项目列 表为空	根据用户内容无法匹配 出相应的项目，项目列 表为空	
测试用例 ID	Test4		
输入/动作	期望的输出/相应	实际情况	
用户输入多个项目共同 具有的内容	根据模块匹配原则，根据 时间的相关度，在项目 列表里列出所有有关 的项目	根据模块匹配原则，根据 时间的相关度，在项目 列表里列出所有有关 的项目	
测试用例 ID	Test5		
输入/动作	期望的输出/相应	实际情况	
用户不输入内容，点击 返回按钮	关闭搜索界面，回到首 页	关闭搜索界面，回到首 页	
测试用例 ID	Test6		
输入/动作	期望的输出/相应	实际情况	
用户输入内容之后，点 击返回按钮	关闭搜索界面，回到首 页	关闭搜索界面，回到首 页	
测试用例 ID	test7		
输入/动作	期望的输出/相应	实际情况	
用户在搜索到内容之 后，点击返回	关闭搜索界面，回到首 页	关闭搜索界面，回到首 页	



搜索企业测试用例：

表 4.3-3 搜索企业用例

测试用例	搜索企业用例	
测试用例 ID	Test1	
输入/动作	期望的输出/相应	实际情况
用户点击搜索按钮	进入搜索界面弹出键盘监听输入	进入搜索界面弹出键盘监听输入
测试用例 ID	Test2	
输入/动作	期望的输出/相应	实际情况
输入内容为空字符时，点击开始搜索	提示输入内容为空，请输入内容	提示输入内容为空，请输入内容
测试用例 ID	Test3	
输入/动作	期望的输出/相应	实际情况
用户输入无意义的内容	根据用户内容无法匹配出相应的企业，企业列表为空	根据用户内容无法匹配出相应的企业，企业列表为空
测试用例 ID	Test4	
输入/动作	期望的输出/相应	实际情况
用户输入多个企业共同具有的内容	根据模块匹配原则，根据时间的相关度，在企业列表里列出所有有关的企业信息	根据模块匹配原则，根据时间的相关度，在企业列表里列出所有有关的企业信息
测试用例 ID	Test5	
输入/动作	期望的输出/相应	实际情况
用户不输入内容，点击返回按钮	关闭搜索界面，回到首页	关闭搜索界面，回到首页
测试用例 ID	Test6	
输入/动作	期望的输出/相应	实际情况
用户输入内容之后，点击返回按钮	关闭搜索界面，回到首页	关闭搜索界面，回到首页
测试用例 ID	test7	
输入/动作	期望的输出/相应	实际情况
用户在搜索到内容之后，点击返回	关闭搜索界面，回到首页	关闭搜索界面，回到首页



发起项目测试用例：

表 4.3-4 用户反馈用例

测试用例		发起项目用例	
测试用例 ID	Test1		
输入/动作		期望的输出/相应	实际情况
会员用户点击发起项目按钮		进入项目编辑界面，可以编辑发布项目	进入搜索界面弹出键盘监听输入
测试用例 ID	Test2		
输入/动作		期望的输出/相应	实际情况
非会员用户点击发起项目按钮		提示需要认证的企业会员才能发布活动项目	提示需要认证的企业会员才能发布活动项目
测试用例 ID	Test3		
输入/动作		期望的输出/相应	实际情况
发起项目时没有填写活动标题，点击发布按钮		提示项目信息填写不完整	提示项目信息填写不完整
测试用例 ID	Test4		
输入/动作		期望的输出/相应	实际情况
发起项目时，没有选择项目封面，点击发布按钮		提示项目信息填写不完整	提示项目信息填写不完整
测试用例 ID	Test5		
输入/动作		期望的输出/相应	实际情况
发起项目时，没有填写项目详细介绍，点击发布按钮		提示项目信息填写不完整	提示项目信息填写不完整
测试用例 ID	Test6		
输入/动作		期望的输出/相应	实际情况
发起项目时，已经填写项目信息的情况下，点击返回按钮		关闭搜索界面，回到首页	关闭搜索界面，回到首页



地图定位搜索附近企业用例：

表 4.3-5 用户反馈用例

测试用例	地图定位搜索附近企业用例		
测试用例 ID	Test1		
输入/动作	期望的输出/相应	实际情况	
用户连接 WiFi，点击商家地图按钮	后台启动服务开开始定位，进入地图界面，显示附近的商家企业	后台启动服务开开始定位，进入地图界面，显示附近的商家企业	
测试用例 ID	Test2		
输入/动作	期望的输出/相应	实际情况	
用户连接数据流量，点击商家地图按钮	后台启动服务开开始定位，进入地图界面，显示附近的商家企业	后台启动服务开开始定位，进入地图界面，显示附近的商家企业	
测试用例 ID	Test3		
输入/动作	期望的输出/相应	实际情况	
用户没有与互联网取得任何连接	提示没有网络，无法实现定位及搜索企业服务	提示没有网络，无法实现定位及搜索企业服务	
测试用例 ID	Test4		
输入/动作	期望的输出/相应	实际情况	
有网络情况，点击地图上的附近的某一个企业图标	进入该企业的企业详情界面	进入该企业的企业详情界面	
测试用例 ID	Test5		
输入/动作	期望的输出/相应	实际情况	
有网络情况开始搜某一个注册企业	在地图上显示该企业的位置	在地图上显示该企业的位置	
测试用例 ID	Test6		
输入/动作	期望的输出/相应	实际情况	
有网络情况开始搜某一个未注册企业	地图仍旧显示用户所在位置，并提示没有找到该企业	地图仍旧显示用户所在位置，并提示没有找到该企业	
测试用例 ID	test7		
输入/动作	期望的输出/相应	实际情况	
搜索到企业后，点击地图上改企业的图标	进入该企业的企业详情界面	进入该企业的企业详情界面	



## 4.4 本章小结

这一章接结是对项目整体的运行环境要求，和在真实环境下的 UAT 用户可以接受用例方案的讨论与分析，是对项目整体质量的检验。保证功能与需要的契合。



## 5 讨论与体会

### 5.1 对项目开发过程的体会

开发一个项目是对所学知识最好的检验。本项目开发时遇到了比超出设计之初所预料到的难点。比如在处理手机端图文混排时，就遇到了难点。因为本项目的 Android 端不只是一是要展示类似于网页的图片文字混排的功能，还需要能够编辑发布它。当时在手机 APP 领域，还没有较好的实现方法，只有微信的公众号文章和简书 APP 的文章有这个实现。但是他们都是互联网行业里面的技术巨头公司，才以良好的用户体验实现了这个功能，而且没有开源他们各自的实现的思路与方案。我只有查阅大量资料，了解他们具体怎么实现和寻找第三方开源的实现项目。

根据查阅的资料，发现微信和简书是各自使用的不同的实现方式。微信它有自己的框架，是基于 H5 技术来处理移动端的页面问题。微信公众号的文章，都是利用它自己的渲染框架，转换成手机网页界面，在 Android 手机端的一个 webview 控件中显示，这样完成了类似于浏览器一样的图片、文字混合排列的效果。而简书这款应用，则是开发了一个支持 Markdown 语法的移动端编辑器。利用 MARKDOWN 语言某些标签，嵌入图片的 URL，添加文字的样式来显示图文混排效果。我要从头实现类似于他们的功能，需要巨大的工作量和良好的技术设计，才能完成这种移动端的编辑器。

虽然本人以前掌握了一些 Android 方面的开发经历，但是这种框架性质的开发，还是给了我不小的困难。但是，困难总是要克服的，不能因为难就不去研究，不去实现。在网上不断搜索，我根据一个国外的博客找到了思路。他在 github 上提供了一个初级的 RichEditor 的实现模板。思路是编写一个 RichEditor 来重新实现 Android 的 webView 控件。运用属于前端领域的 JavaScript 和 CSS 技术，在 webview 里面渲染出图文混排的效果。字体的样式种类，文本的操作，事先在 RichEditor 中定义好。然后在底部创建一个控制栏，根据控制栏中的不同功能的点击按钮，去调用 RichEditor 的展示不同的样式字体。RichEditor 在利用保存的图片 URL，向服务器请求图片数据，进行展示。这样就基本实现了 Android 端的 RichEditorView 的编辑功能和浏览功能。

设计与实现的出现差别，在我看来是正常的事情。开发时勇敢的去尝试，设计时细心的规划，是能够让两者之间的差距渐渐变小的。思维枯竭时，“柳暗花明又一村”的灵感，克服困难后的成就感，都会让我拥有我变成一个优秀开发者的不可多得品质。

### 5.2 对项目的评价

现在，个人和企业之间的交流仅仅局限于 58 同城，赶集网这种大分类的门户网站。





和本平台相比，本平台是更加细分的一个专业领域，不像他们那样庞杂。它们的作用只是像找工作，买二手等这些突发性的，偶然性的需求。没有形成一个长期，稳定的行业内交流社区。这恰恰是本项目的优点，也是我开发这个平台的一个初衷。

同时，我在这个平台里还加了自己一些创新点 idea，比如商家地图功能。商家地图，参考了附近的人的模式，把注册的商家根据与用户的距离，显示在地图上，让用户按照自己的喜好，查看浏览，促进企业与个人的交流。这打破了传统企业的老旧土气的展示形象，顺应了时代的趋势。

当然，本项目本平台也会存在不足之处。既然是一个内容交流的社区平台，就需要一有人维护内容社区的健康积极向上，有人来运营管理。这些都没有适合的组织团队来维护。还有，项目的商业利益方面，除了考虑到向注册商家收取平台管理费和引入广告机制获取盈利之外，没有其他策划内的方式。一个平台的如果要投入市场商业，还需要添加充分的盈利功能，才有可能成为一个有竞争力，生命力的产品。

### 5.3 对项目成果用于实际意义的阐述

自从我们进入 21 世纪以来，移动通讯技术依靠快速的更迭，当之无愧的变成现今发展最快速，最受到社会希冀和认可的新兴产业。这些年来伴随互联网，移动互联网的普及，我们的世界不知不觉已经进入到一个信息爆炸的时代，网络上出现了浩瀚无垠的信息。人们面对呼啸汹涌而来的巨量无所适从，想迅捷、高效的从广大繁多的信息中找寻自己所需要的信息，成为一件异常的艰难的工作。

诸如社会上一些企业，公益组织的活动、宣传信息等，不仅庞杂无穷，并且由于虚假、诈骗信息的存在，许多关注公司企业信息的人员无法获取到比较完善系统化的企业信息。基于这一点，我开发一款专门为企业、公益组织与个人信息服务的企业信息聚合平台（以下简称企聚平台）。企聚平台是一款自组织型网络社区平台，以 B2C 模式创建。企聚平台以为有了解各行各业和各商家公司、最近宣传活动的信息需求的个人用户，构造一个值得信赖、资讯确切、信息共享的社区平台为目的。即个人能通过该平台，借助该平台在某些领域内聚合网络资源，扩张他个人的人际辐射力和社会影响力。一方面，以最短的时间获取企业最为准确的、最新动态的信息；另一方面，它也是一个企业和企业客户之间的一个重要的及时的交流平台。

和于传统地通过网上搜索并且获取企业信息的方法相比较，这种企业信息聚合平台将给个人带来更多的便捷，同时个人也可以横向了解和比较同一行业的不同企业相应的信息。和另一种客户与企业之间通过电子邮件的方式对比，该平台也更具有及时性，拥



有手机客户端的信息聚合平台，会愈加顺应移动互联网朝代的工作时间的碎片化，工作地点多元化，工作方式自由化的趋势。

关于企业来说，企聚平台是一个它发布企业最新动态以及活动的平台，商家可以经过在平台注册，并且填写企业信息，然后取得在本平台发布项目的权利。本平台为企业和商家提供了一个发布活动的平台，同时也是企业自我展示，推广自身企业文化，推销自己的产品的平台。

## 5.4 本章小结

本章总结了该项目开发过程中的一些体会，列举了一些遇到的困难，和对困难的克服过程。对该项目的创新性与先进性，对社会的积极意义，商业应用前景都进行的恰当的叙述。





## 参考资料

- [1]Roger S. Pressman. 《软件工程——实践者的研究方法》[M]. 2008.
- [2]Java 核心技术 《Java core I》 《Java core II》
- [3]阳慧. LRU 算法的研究及实现[J]. 计算机时代, 2004(2):28-29.
- [4]Agarwal S, Dunagan J, Jain N, et al. Volley: Automated Data Placement for Geo-Distributed Cloud Services[C]// Usenix Symposium on Networked Systems Design and Implementation, NSDI 2010, April 28-30, 2010, San Jose, Ca, Usa. DBLP, 2010:17-32.
- [5]吴亚峰, 杜化美, 苏亚光. Android 编程典型实例与项目开发[M]. 电子工业出版社, 2011.
- [6]ZigurdMednieks, LairdDornin, G.BlakeMeike, 等. Android 编程[M]. 东南大学出版社, 2013.
- [7]Qi Y, Cao M, Zhang C, et al. A Design of Network Behavior-Based Malware Detection System for Android[M]// Algorithms and Architectures for Parallel Processing. Springer International Publishing, 2014:590-600.
- [8]Kirkegaard, C., & Ller, A. (2006). Static analysis for java servlets and JSP. International Conference on Static Analysis (Vol.4134, pp.336-352). Springer-Verlag.



## 声 明

本人声明所呈交的学位论文是本人在导师指导下进行的研究工作及取得的研究成果。据我所知，除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得四川大学或其他教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示谢意。

本学位论文成果是本人在四川大学读书期间在导师指导下取得的，论文成果归四川大学所有，特此声明。

学位论文作者（签名） \_\_\_\_\_

论文指导教师（签名） \_\_\_\_\_

2017 年 04 月 24 日



## 致 谢

首先我要感谢我的父母，谢谢你们一直以来支持我自己的选择，让我能自由选择自己的发展道路，成功的找到一份好的工作。

感谢我的导师，一个很钦佩喜爱何军老师的耐心与认真的教学态度。在这段时间里我能圆满顺利地完成我的毕业设计，既离不开我自身的努力，更离不开何军老师的指点。也感谢在这段时间里和我讨论，在遇见困难时鼓励我不气馁，督促并帮助我完成实验的学长。

感谢我的室友，一直以来我们一起奋斗的日子令人难以忘却，到现在我们的努力都没有白费，最终大家都找到了满意的出路。毕业以后我们也要互相督促，共同进步。

感谢学校的社团、俱乐部，让我不仅在志愿者生活中实现自我价值，还让我在举办各种活动的过程中锻炼自己的能力，更丰富了我的见识和经验。

最后感谢四川大学和软件学院为我提供了除学日常上课学习以外的很多的机会，让我在小学期课程上受到很多外校优秀的老师的教导，也让我有机会去参加各种编程比赛项目，这些经验不仅提高了我的学术水平，更开阔了我的眼界，让我更加全面地发展。

天下没有不散的筵席，满载回忆的大学四年生活面临结束，但是这四年里我学习到的知识和获得的经验将使我终生受益。

作者签名：

2017 年 04 月 24 日

指导教师签名：

2017 年 04 月 24 日



## 附录 1 详细设计文档

### 编写目的

编写目的本需求规格说明书编写的目的是为了清晰地说明本小组开发《企业信息聚合平台》要实现的所有功能。同时也为软件使用者和软件开发者之间建立共同的需求目标，进行一定程度的交流沟通。同时对软件所需实现的功能模块做了全面的描述。它是后续开发工作和验收工作的依据文件。软件使用者和软件开发者以此为共同的基础。

本说明书的预期读者为：软件开发小组的相关人员，软件开发小组的产品实现成员应该阅读和参考本说明进行代码编写、测试。

### 背景

自从我们进入 21 世纪以来，移动通讯技术依靠快速的更迭，当之无愧的变成现今发展最快速，最受到社会希冀和认可的新兴产业。这些年来伴随互联网，移动互联网的普及，我们的世界不知不觉已经进入到一个信息爆炸的时代，网络上出现了浩瀚无垠的信息。人们面对呼啸汹涌而来的巨量无所适从，想迅捷、高效的从广大繁多的信息中找寻自己所需要的信息，成为一件异常的艰难的工作。

诸如社会上一些企业，公益组织的活动、宣传信息等，不仅庞杂无穷，并且由于虚假、诈骗信息的存在，许多关注公司企业信息的人员无法获取到比较完善系统化的企业信息。基于这一点，我开发一款专门为企业、公益组织与个人信息服务的企业信息聚合平台（以下简称企聚平台）。企聚平台是一款自组织型网络社区平台，以 B2C 模式创建。企聚平台以为有了解各行各业和各商家公司、最近宣传活动的信息需求的个人用户，构造一个值得信赖、资讯确切、信息共享的社区平台为目的。即个人能通过该平台，借助该平台在某些领域内聚合网络资源，扩张他个人的人际辐射力和影响力。一方面，以最短的时间获取企业最为准确的、最新动态的信息；另一方面，它也是一个企业和企业客户之间的一个重要的及时的交流平台。

和于传统地通过网上搜索并且获取企业信息的方法相比较，这种企业信息聚合平台将给个人带来更多的便捷，同时个人也可以横向了解和比较同一行业的不同企业相应的信息。和另一种客户与企业之间通过电子邮件的方式对比，该平台也更具有及时性，拥有手机客户端的信息聚合平台，会愈加顺应移动互联网朝代的工作时间的碎片化，工作地点多元化，工作方式自由化的趋势。

关于企业来说，企聚平台是一个它发布企业最新动态以及活动的平台，商家可以经过在平台注册，并且填写企业信息，然后取得在本平台发布项目的权利。本平台为企业和商家提供了一个发布活动的平台，同时也是企业自我展示，推广自身企业文化，推销自己的产品的平台。

### 范围

适用于社会上所有用户。

### 术语定义

### 参考资料

- [1] Roger S. Pressman. 《软件工程——实践者的研究方法》[M]. 2008.
- [2] Java 核心技术
- [3] 阳慧. LRU 算法的研究及实现[J]. 计算机时代, 2004(2):28-29.



## 目标

开发供企业可以在网络上发布动态，用户可以浏览评论的稳定系统平台，操作简单，能良好的运行。

## 用户的特点

本软件最终用户具有以下特点：

- 使用人员：希望能在一个平台上获知不同企业、不同行业信息的人员和发布企业信息的人员。
- 维护人员：熟悉该平台的运作流程，熟悉计算机的各项操作，拥有敏锐的洞察力。
- 预期使用频度：10000 次/日

## 假定和约束

- 规定了一个公司只能由一个用户注册
- 规定了只有注册公司的用户才能发布项目

## 运行环境

.NET Framework 3.5 或更高版本

Android4.0 或更高版本

## 软件环境

	名称	版本	语种
操作系统	Windows	XP, 2003	
操作系统的附加功能			
数据库平台	Mysql	5.0	
语言	JDK	5.0 以上	
邮件系统			
客户端软件			

## 硬件环境

硬件配置	最低配置	推荐配置
服务器	1CPU: P4 2.0G	1CPU: P4 2.8G
	Mem: 1G	Mem: 2G
	HD: 40G	HD: 120G
客户端	1CPU: P4 1.0G	1CPU: P4 2.0G
	Mem: 512M	Mem: 1G
	HD: 40G	HD: 120G



## 接口

### 1.7.3.1 外部接口

外部接口的用户界面部分应遵循如下规则：

- 界面对用户友好，且必须分别对鼠标和键盘，触摸提供支持；
- 界面要具有一致性；
- 提供简单的错误处理；
- 设计良好的联机帮助

## 2 命名规则

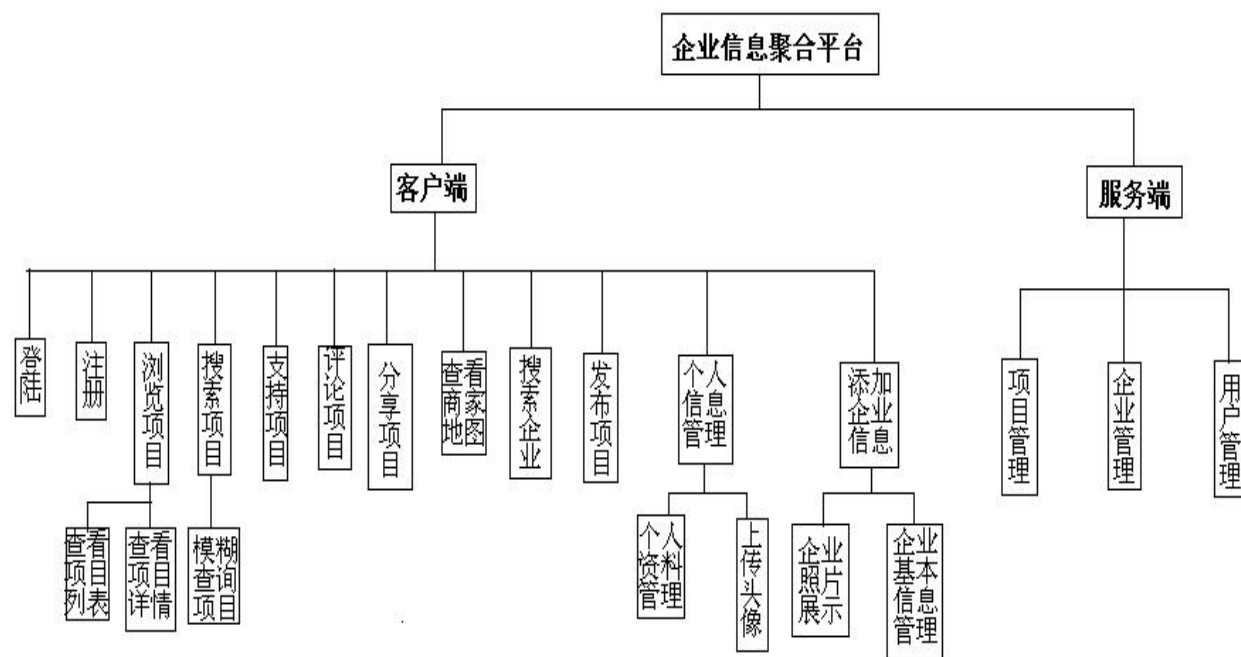
(1) 项目命名：Sunshine+项目功能且首字母大写

(2) 包命名：com.etc+所属项目功能小写+包功能小写

(3) 类命名：驼峰命名法，根据类的功能命名，如果类名只有一个单词则单词的首字母大写，其余都使用小写；如果类名由多个单词构成，则每个单词的首字母大写，其余都使用小写。

## 3 系统功能及描述

### 3.1 系统功能层次模块图



### 3.2 服务器模块功能的描述

编号	名称	简述	输入数据	处理（算法）	输出
SRS-001	浏览所有项目列表	普通用户浏览所有项目	无	无	显示项目列表，包括每个项目的活动展示照片、活动发起人、活动时间、点赞、评论等



编号	名称	简述	输入数据	处理（算法）	输出
SRS-002	浏览单个项目详情	普通用户浏览单个项目详情	所选项目编号	无	显示活动发起人、活动时间、和活动主题以及活动详情（使用图文混排界面），下面是点赞数，以及评论
SRS-003	搜索项目	普通用户通过模糊搜索	模糊搜索：项目名称或项目类别	模糊匹配算法	符合搜索条件的项目列表，包括每个项目的活动展示照片、活动发起人、活动时间、点赞、评论等
SRS-004	分享项目	普通用户可分享某个项目至微信、微博等第三方社交平台	所选项目编号、所选社交平台名称	无	跳转到相应第三方社交平台的页面，并显示已成功分享用户的内容
SRS-005	用户注册	普通用户注册成为注册用户	用户名、密码、昵称、真实姓名、地址、性别、上传头像等注册信息	将用户提交的信息写入数据库	文字提示是否注册成功
SRS-006	用户登录	注册用户登录	用户名，密码	验证该用户名有效性	登录是否成功
SRS-007	发布项目	注册用户可发布项目	包括每个项目的活动主题、活动展示照片、活动时间和活动地点	将用户提交的信息写入数据库	文字提示是否提交成功
SRS-008	评论项目	注册用户可评论项目	评论项目编号、评论人编号、评论内容、评论时间	根据评论项目编号在数据库中找到对应的项目，并将评论写入数据库	文字提示是否评论成功
SRS-009	支持项目	注册用户可为项目“点赞”来支持项目	支持项目编号	支持的项目上支持数目加一	显示支持的项目上支持数目已加一



编号	名称	简述	输入数据	处理（算法）	输出
SRS-0010	个人信息管理	注册用户查看自己以及编辑个人信息	用户编号	无	用户个人信息
SRS-0011	添加企业信息	注册用户可以添加自己公司的信息	公司的名称、公司地址、所属行业、简介、企业展示界面	无	提示是否添加成功
SRS-0012	查看商家地图	注册用户可以通过地图上表示的商家进行查找	通过地图上选择商家所在位置	调用地图接口	显示商家的详情的界面
SRS-0013	搜索企业	注册用户可以通过输入关键字查找	主要是企业名称进行查找	模糊搜索	显示查找商家的列表
SRS-0014	用户管理	后台管理员对注册用户进行增删改查操作	相应的后台网页操作		
SRS-0015	项目管理	后台管理员对项目进行增删改查操作	相应的后台网页操作		
SRS-0016	企业管理	后台管理员对企业进行增删改查操作	相应的后台网页操作		

### 3.4 对非功能性的需求

#### 3.4.1 对性能的规定

##### (1)精度

传输过程中的精度：数字的小数点之后面保全三位有效字；

输出数据精度：数字的小数点之后面保全三位有效字；

输入数据精度：数字的小数点之后面保全三位有效字；

##### (2)时间特性要求

- 实时反馈时间：不多于 1s。即用户在软件的每个功能函数的点击、触摸、键盘输入等等动作事件的返回时长，应该规定在 1s 以内，以给用户舒适的体验；





- 更新处理时间：不超过 1s；
- 数据的变换和传输时间：不多于 1s；

### (3)灵活性

1. 软件在最初的概要设计上和具体完成上，均需要考虑到其在不同运行环境的状况。要求能在不同的运行环境稳定执行，即在 Android 4.0 及其以上版本里，运行时能够兼容；
2. 当输入整数能自动转换为系统规范的三位小数的精度。如有错误类型数据，应及时提示错误信息
3. 软件应满足开闭原则，以适应将来有可能会出现的需求更改
4. 软件系统进行升级时保证用户数据的安全性输入输出要求

#### 3.4.2 输入输出要求

##### 3.3.1 输入

该软件在客户端支持以下两种输入方式：触摸输入和文字输入

- 触摸输入：用户用手指触摸屏幕，如果该位置是系统允许的有效目标则记录为用户输入
- 文字输入：用户在指定文本框中输入文字，被系统确认为有效后记录为用户输入

##### 3.3.2 输出

###### 6) 用户登录及注册服务

- 用户输入无效时，系统自动查询，返回相似的有效输入提示
- 系统友好地输出是否登录注册成功

###### 7) 项目浏览服务

- 对所有项目列表
- 对单个项目详情

###### 8) 项目查询服务

- 用户输入无效时，系统自动查询，返回相似的有效输入提示
- 列表显示查询结果

###### 9) 项目评论及支持服务

- 用户输入无效时，系统自动查询，返回相似的有效输入提示
- 系统友好地输出是否评论或支持成功

###### 10) 分享服务

- 跳转到相应社交平台的分享页面，并输出展示已成功分享的用户内容。

#### 3.4.3 数据管理能力要求

数据库中的数据处理关系简单，但是由于需要存放不断更新的用户信息和项目信息，数据库中的数据量相对较多。因此，对数据管理能力要求相对较高。

- 项目信息的存储：将已审核的项目信息准确的存储到数据库中，还包括对数据的增、删、改等操作



基本数据的设定：设置合理的基本数据，保证该平台的正常运行。

### 3.4.4 故障处理要求

- 手机电量不足：软件提供定时保存机制，每隔一定时间自动对信息进行保存，从而保证用户数据的安全，当用户在重新给手机供电后，该软件可正常使用；
- 手机内存不足：该软件尽可能保持稳定，如果解决内存不足的故障之后，软件又可正常使用；
- 服务器无法连接到数据库：软件不能从数据库中获取相关的用户资料和项目信息，系统会友好地提示暂时无法提供服务，并引导用户告知软件维护人员错误，由软件维护人员进行处理；
- 软件依赖的数据库损毁：数据库会定期备份，当发生数据库损毁，系统会友好亲切地提示数据信息损毁或缺失等具体错误信息，并引导用户告知软件维护人员错误，由软件维护人员来处理。

### 3.4.5 其他专门要求

- 可用性：当系统内部、外部发生错误，系统能较快地检测到错误事件并记录故障，通知用户或系统或根据已定义的规则禁止故障源等；用户输入的错误数据都有提示信息，具有较好的容错能力。
- 性能：事件触发状态，系统的反馈时间为秒级，性能较高；
- 安全性：对于系统的重要数据都有加密处理，未经授权不能访问数据或服务，用户和管理员都根据计划来提交数据或服务；加强审核机制，能保证交易各方是所声称的人；能在系统内部跟踪系统活动，具有较强的安全性；
- 易用性：用户学习成本低，可以有效地快速地使用系统，软件能适应用户的自身需求，提高用户舒适度和满意度。
- 运行环境可转换性：兼容 Android4.0 以上系统，能对不同的运行环境提供支持

## 4 系统逻辑结构设计

### 4.1 系统逻辑结构设计

User 表：

用户信息（User 编号，用户名，密码，姓名，性别，公司所在地址，所属服务队，头像地址，创建时间，更新时间）

Company 表

公司（Company 编号，所属用户编号，公司名称，地址进度，地址纬度，经营范围，所属行业，公司介绍详情界面，创建时间）

Project 表

项目（Project 编号，项目标题，活动时间，发起者编号，点赞数，封面照片，详情界面，创建时间）

Comment 表

评论（Comment 编号，对应 project 编号，对应 User 编号，评论内容，创建时间）



## 4.2 表设计

根据数据字典的设计，进一步细化为数据库表的设计。表的设计包括以下内容：

### 4.3.1 数据库表名解析

User 用户表  
Company 公司表  
Project 项目表  
Common 评论表

### 4.3.2 用户表解析

字段名	类型	允许为空	默认值	说明
id	INT	FALSE		用户 id
username	VARCHAR(45)	FALSE		用户名
password	VARCHAR(100)	FALSE		密码
header	VARCHAR(100)	TRUE		头像地址
name	VARCHAR(20)	TRUE		姓名
sex	INT	TRUE		性别
address	VARCHAR(45)	TRUE		地址
create_time	VARCHAR(50)	TRUE		创建时间
update_time	DATETIME	TRUE		更新事件

### 4.3.3 公司表解析

字段名	类型	允许为空	默认值	说明
id	INT	FALSE		公司 id
user_id	VARCHAR(45)	FALSE		所属用户的 id
company_name	VARCHAR(100)	FALSE		公司名称
address_longitude	VARCHAR(45)	FALSE		地址经度
address_latitude	VARCHAR(45)	FALSE		地址纬度
business_scope	VARCHAR(45)	FALSE		经营范围
industry	INT	FALSE		所属行业
show_page	LONGTEXT	FALSE		公司介绍详情界面（html 文件）
create_time	DATETIME	FALSE		创建时间

### 4.3.4 项目表解析



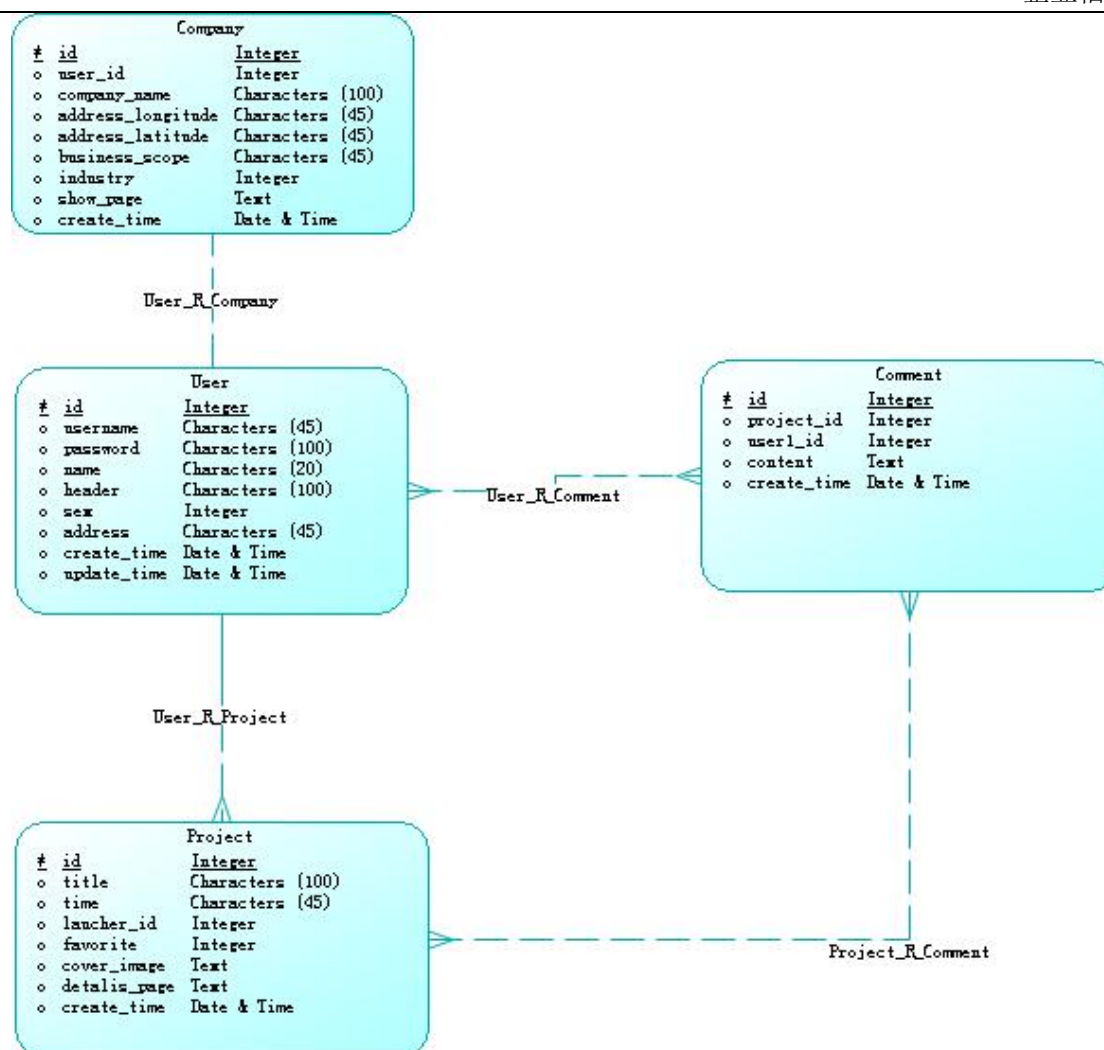
字段名	类型	允许为空	默认值	说明
id	INT	FALSE		项目（活动）id
title	VARCHAR(100)	FALSE		项目标题
time	VARCHAR(45)	FALSE		活动时间
launcher_id	INT	FALSE		发起者 id
favorite	INT	TRUE		点赞数
cover_image	LONGTEXT	TRUE		封面照片
details_page	LONGTEXT	FALSE		详情界面 (html)
create_time	VARCHAR(45)	FALSE		创建时间

#### 4.3.5 评论表解析

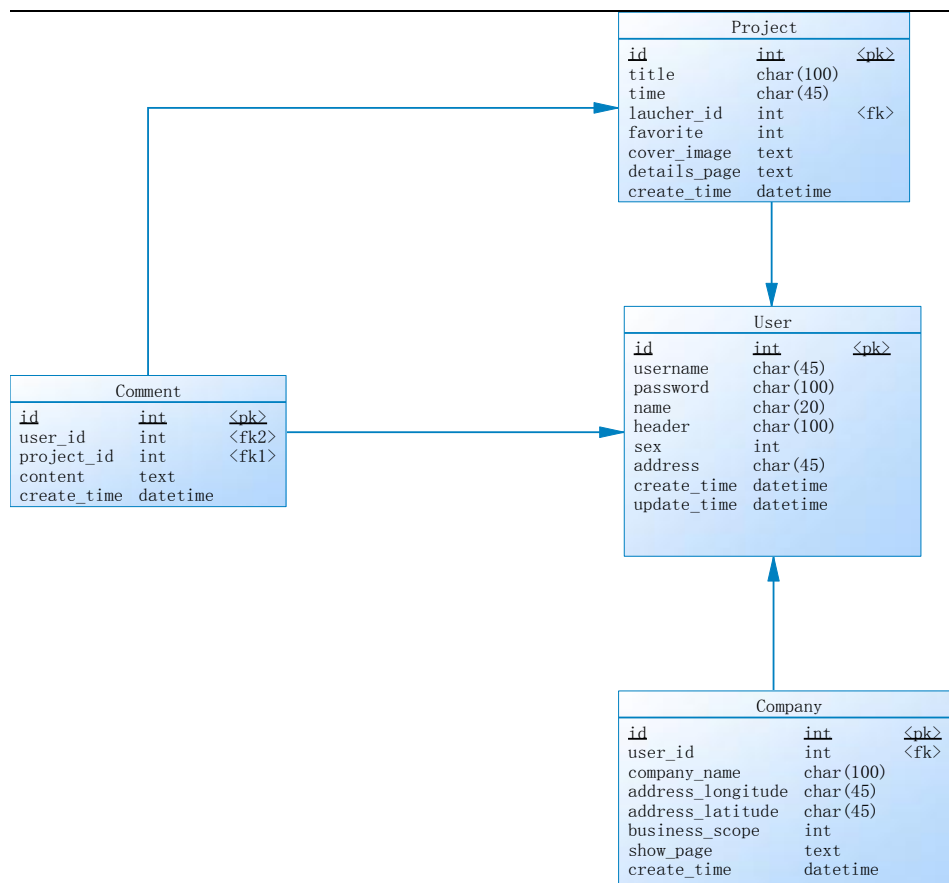
字段名	类型	允许为空	默认值	说明
id	INT	FALSE		评论 id
project_id	INT	FALSE		对应的项目 id
user_id	INT	FALSE		对应的用户 id
content	LONGTEXT	FALSE		评论内容
create_time	VARCHAR(45)	FALSE		创建时间

### 4.4 表之间的关联设计

#### 4.4.1 概念模型

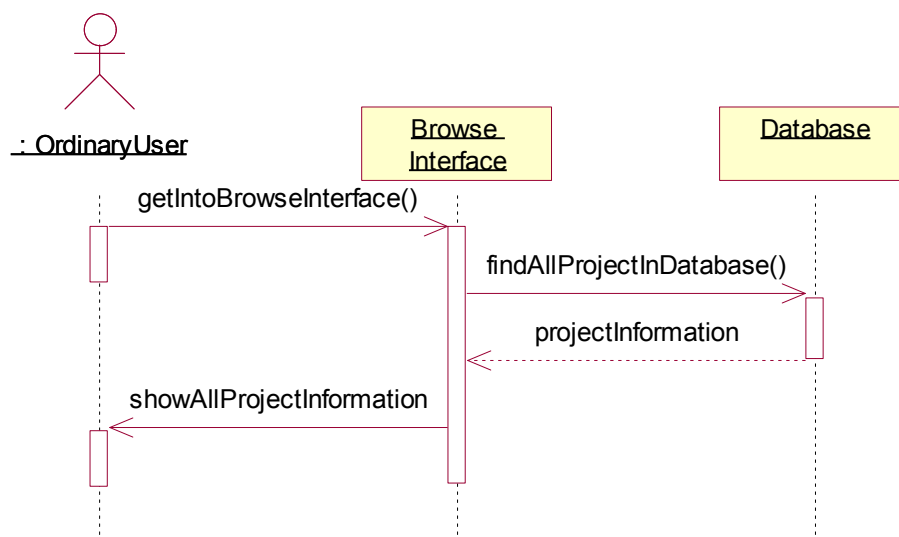


#### 4.4.2 物理模型



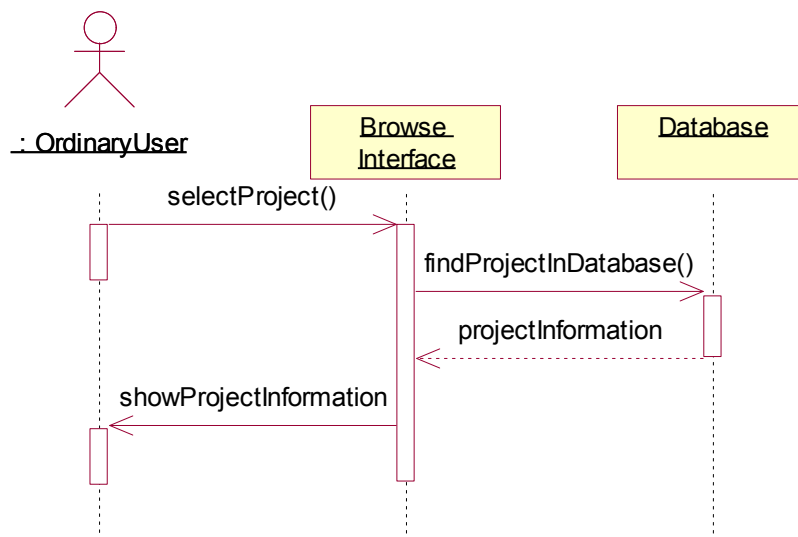
## 4.5 时序图

### 4.5.1 浏览所有项目列表

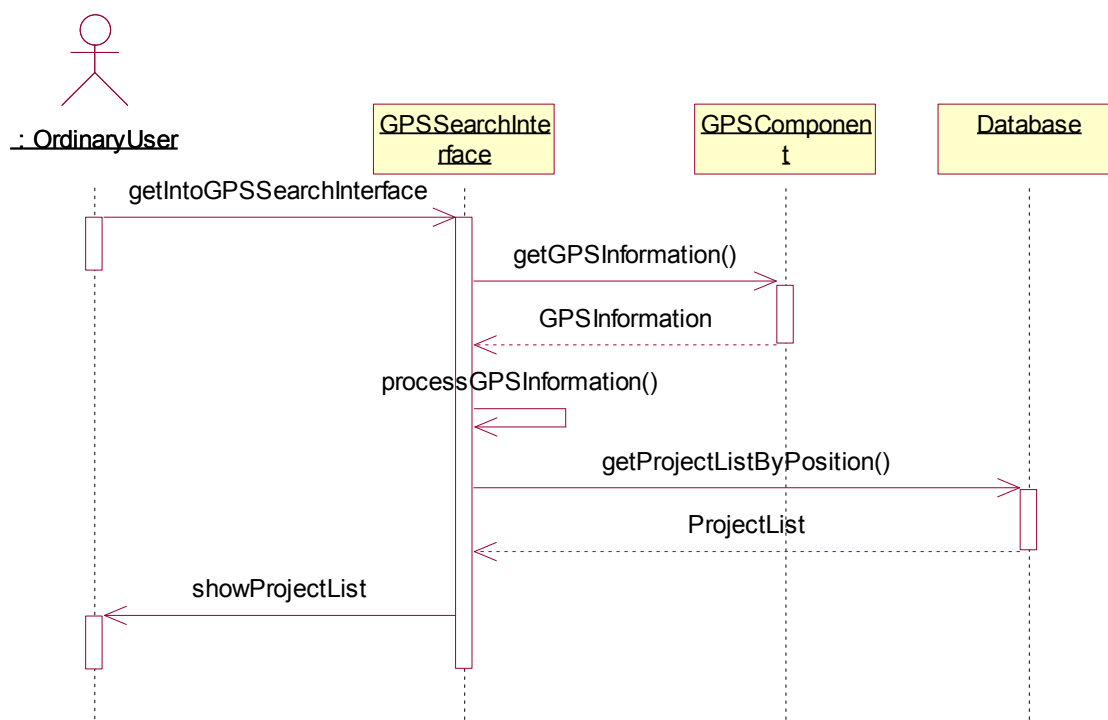




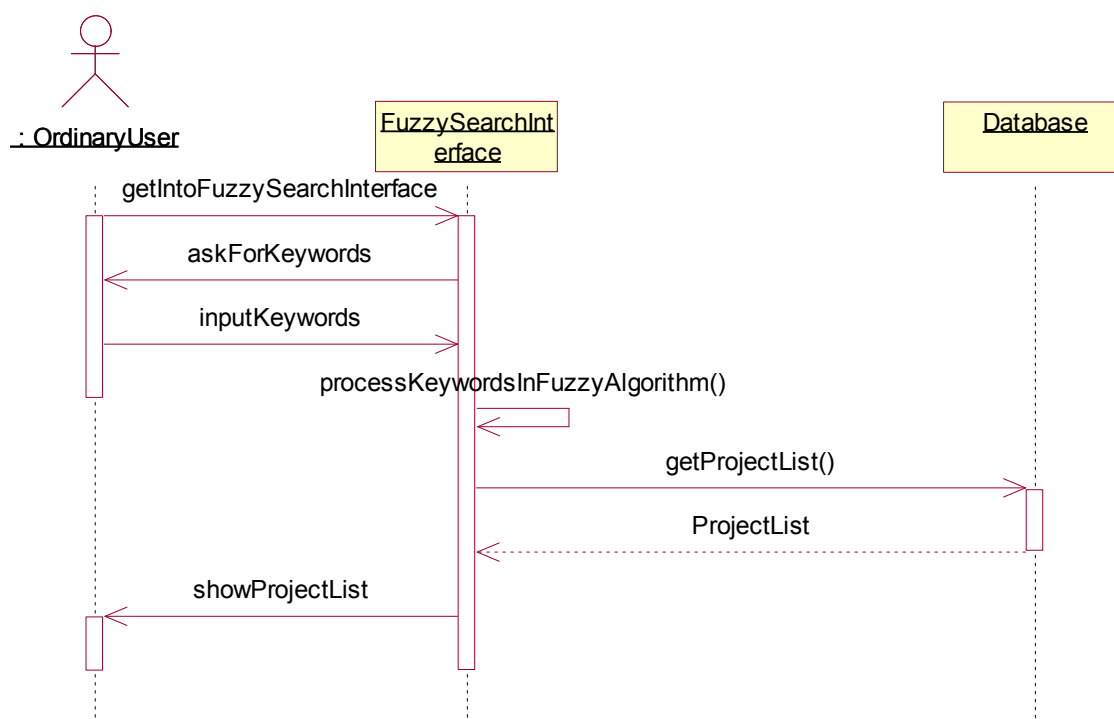
#### 4.5.2 浏览单个项目详情



#### 4.5.3 通过 GPS 查找商家

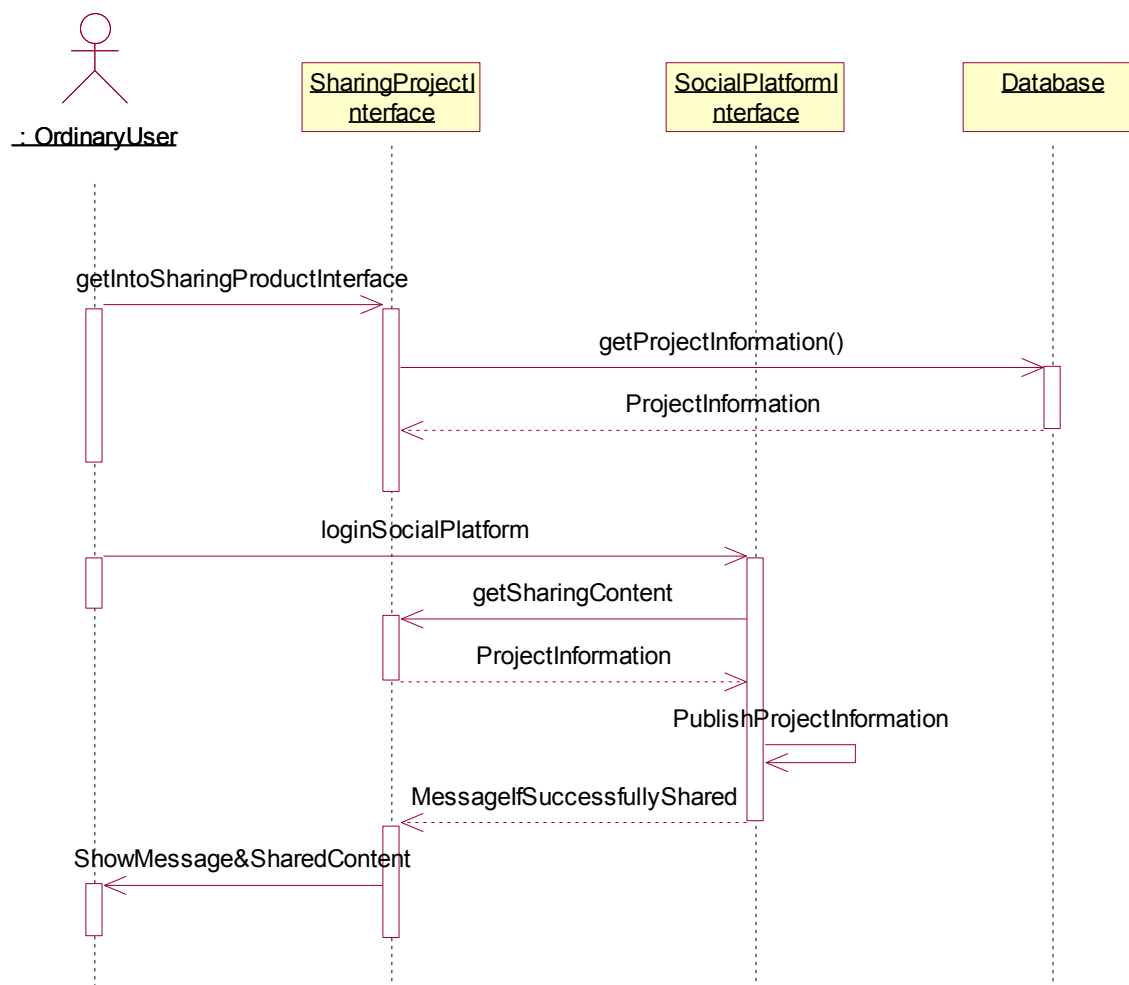


#### 4. 5. 4 通过模糊搜索查找项目

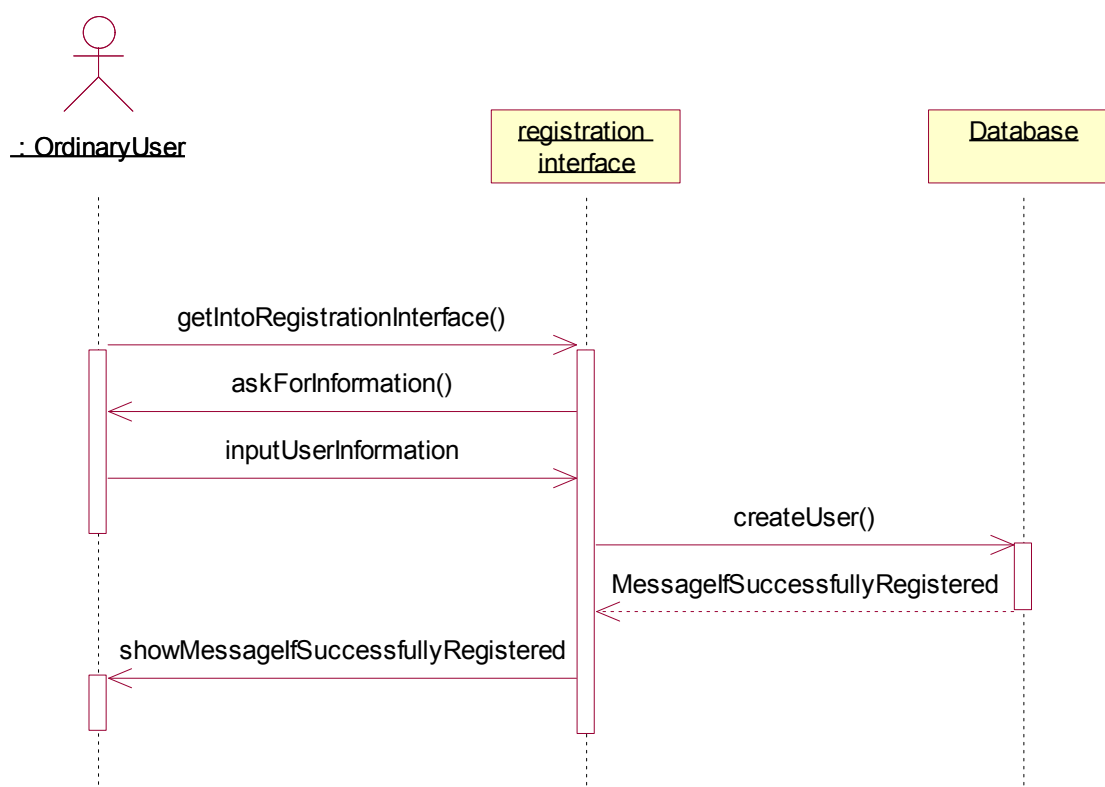


#### 4. 5. 5 分享项目

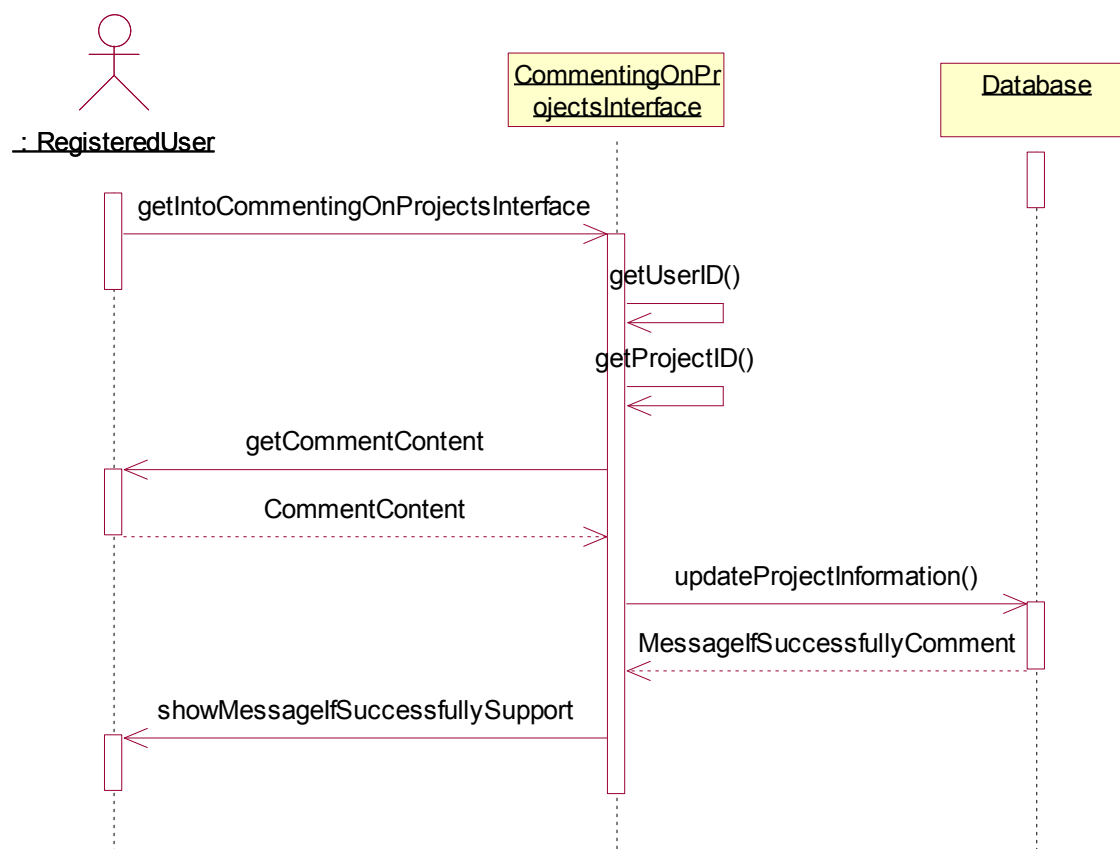




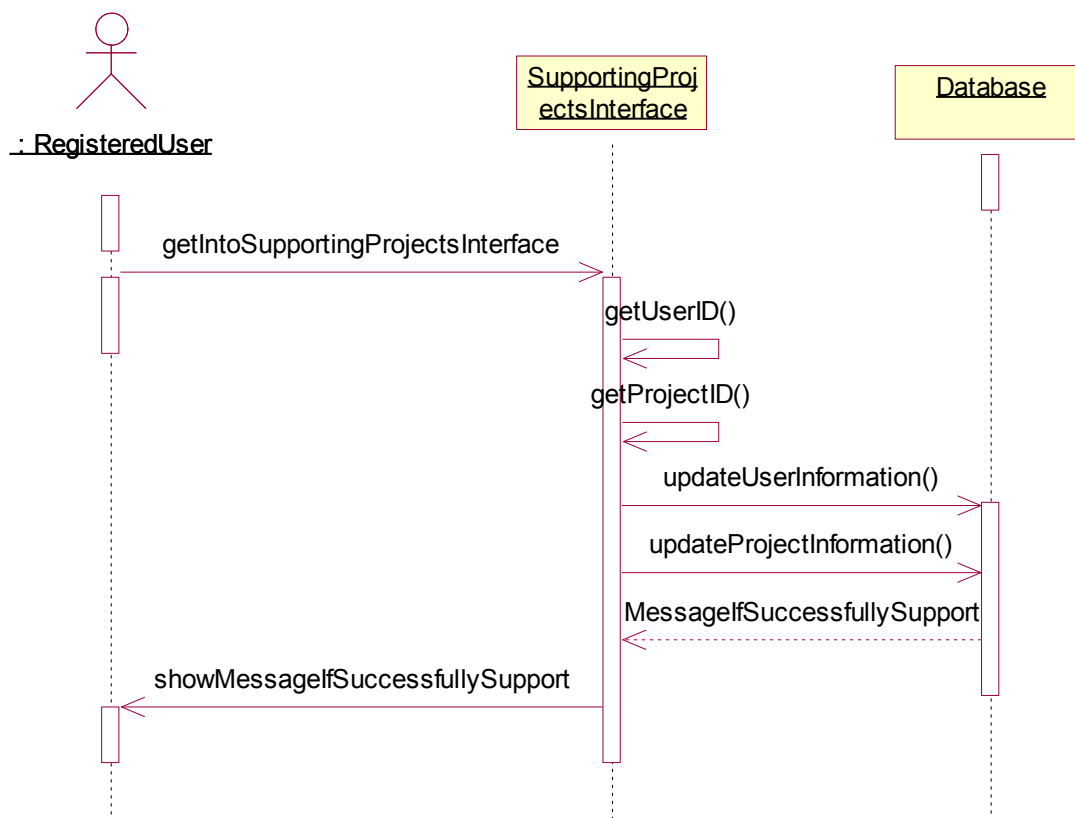
#### 4.5.6 用户注册



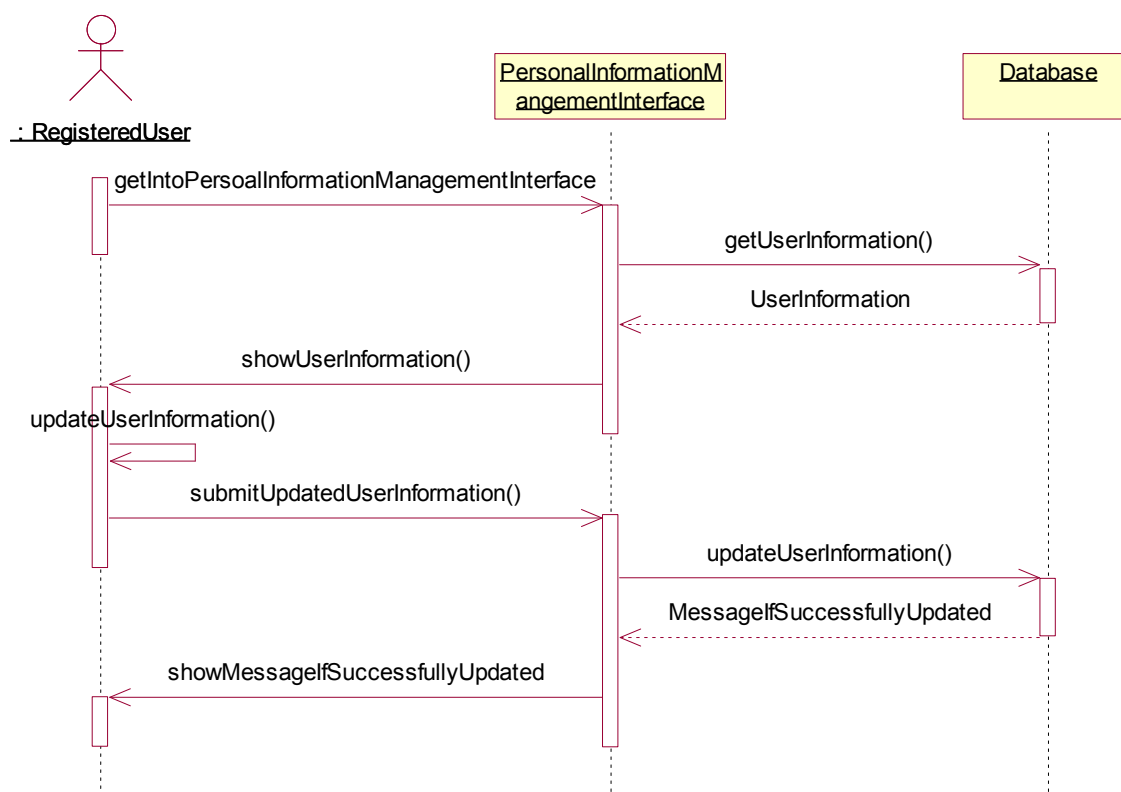
#### 4.5.7 评论项目



#### 4.5.8 支持项目



#### 4.5.9 个人信息管理



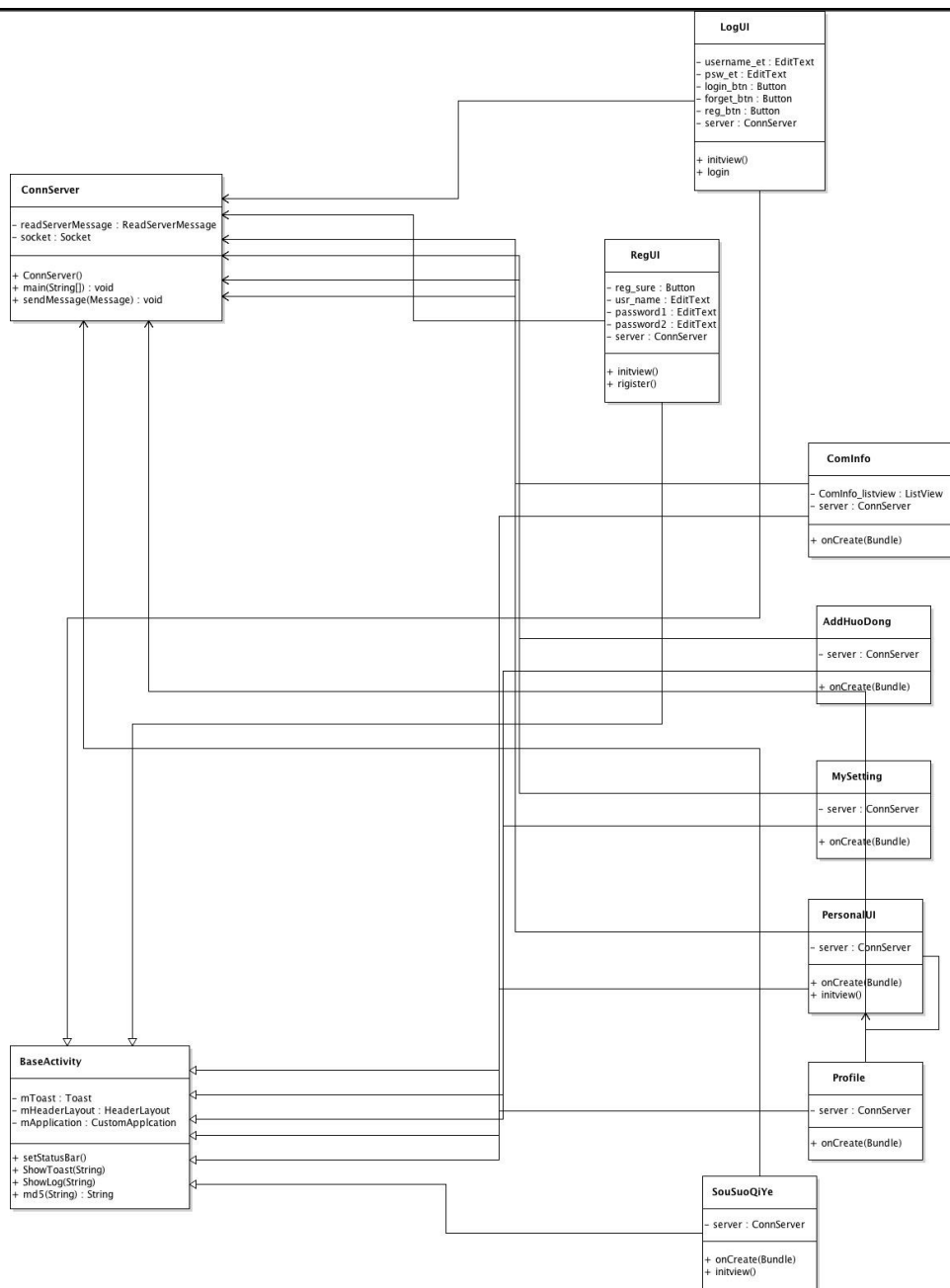
## 5 系统模块设计

### 5.1 客户端

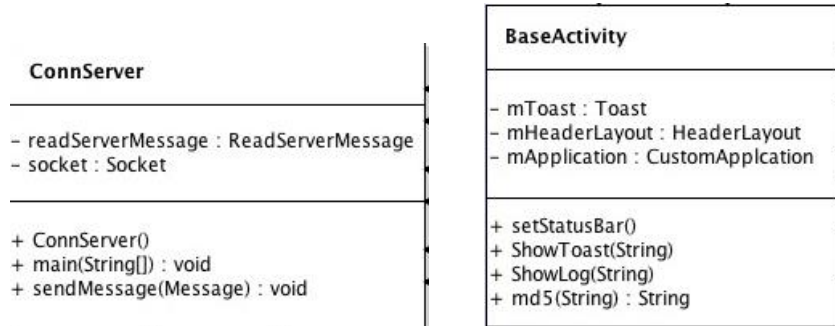
模块名称	客户端
功能描述	实现 用户登陆 注册 搜索 添加活动等



实体类图



业务逻辑类





<p>界面 ACTION</p>	<div><div><div><b>SouSuoQiYe</b><div><div>- server : ConnServer</div><div>+ onCreate(Bundle) + initview()</div></div></div><div><b>AddHuoDong</b><div><div>- server : ConnServer</div><div>+ onCreate(Bundle)</div></div><div><b>PersonalUI</b><div><div>- server : ConnServer</div><div>+ onCreate(Bundle) + initview()</div></div></div></div><div><div><b>RegUI</b><div><div>- reg_sure : Button - usr_name : EditText - password1 : EditText - password2 : EditText - server : ConnServer</div><div>+ initview() + rigister()</div></div><div><b>ComInfo</b><div><div>- ComInfo_listview : ListView - server : ConnServer</div><div>+ onCreate(Bundle)</div></div><div><b>Profile</b><div><div>- server : ConnServer</div><div>+ onCreate(Bundle)</div></div></div></div><div><div><b>LogUI</b><div><div>- username_et : EditText - psw_et : EditText - login_btn : Button - forget_btn : Button - reg_btn : Button - server : ConnServer</div><div>+ initview() + login</div></div><div><b>MySetting</b><div><div>- server : ConnServer</div><div>+ onCreate(Bundle)</div></div></div></div></div></div></div></div></div>
<p>接口与 属性</p>	<p><b>5.1.1 业务逻辑类说明:</b></p> <p>login () 处理登陆信息包</p> <p>regist() 处理注册信息包</p> <p>getProjectData() 处理活动数据信息包</p> <p>getProjectDetails() 处理特定项目具体信息包</p> <p>getCompanyData() 处理公司数据信息包</p> <p>getCompanyDetails() 处理特定公司数据信息包</p> <p>Volley 向服务器发送请求库</p> <p>GSON 解析服务器返回的 JSON 数据</p> <p><b>5.1.2ACTION 说明:</b></p> <p>无</p>
<p>补充说明</p>	



## 5.2 网页端

模块名称	网页管理端
功能描述	实现管理员登入，对用户、项目、公司的增、删、改、查
实体类图	<pre>classDiagram     class Administrator {         - username : String         - password : String     }     class Company {         - id : int         - user_id : int         - company_name : String         - address_longitude : String         - address_latitude : String         - business_scope : String         - industry : int         - show_page : String         - create_time : String     }     class User {         - id : int         - username : String         - password : String         - header : String         - name : String         - sex : int         - address : String         - create_time : String         - update_time : String     }     class Project {         - id : int         - launcher_id : int         - title : String         - time : String         - favorite : int         - cover_image : String         - details_page : String         - create_time : String     }     class Comment {         - id : int         - project_id : int         - user_id : int         - content : String         - create_time : String     }     Administrator "1..1" -- "1..1" User     Administrator "1..1" -- "1..1" Comment     Company "0..*" -- "0..1" User     User "0..*" -- "0..1" Project     User "0..1" -- "0..*" Comment     Project "0..1" -- "0..*" Comment</pre>
界面 ACTION	





AddUserServlet

+ doGet ()  
+ doPost ()

DeleteUserServlet

+ doGet ()  
+ doPost ()

CheckUserServlet

+ doGet ()  
+ doPost ()

ModifyUserServlet

+ doGet ()  
+ doPost ()

AddCompanyServlet

+ doGet ()  
+ doPost ()

DeleteCompanyrServlet

+ doGet ()  
+ doPost ()

CheckCompanyServlet

+ doGet ()  
+ doPost ()

ModifyCompanyServlet

+ doGet ()  
+ doPost ()

AddProjectServlet

+ doGet ()  
+ doPost ()

DeleteProjectrServlet

+ doGet ()  
+ doPost ()

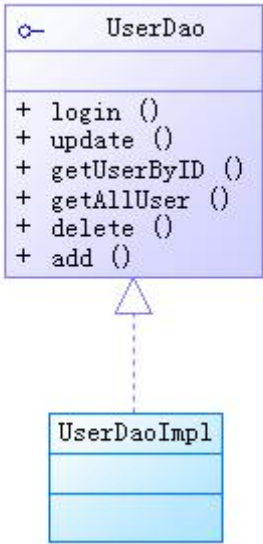
CheckProjectServlet

+ doGet ()  
+ doPost ()

ModifyProjectServlet

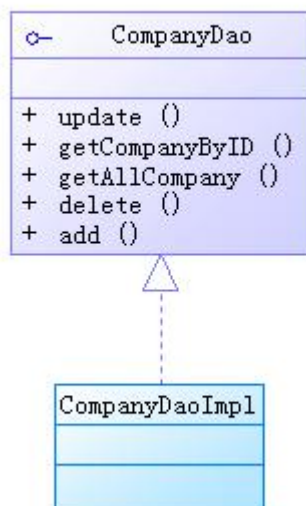
+ doGet ()  
+ doPost ()



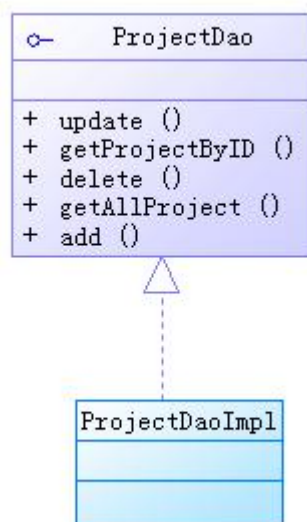
接口与属性	<p><b>5.2.1 ACTION 说明:</b></p> <p>网页的界面的跳转操作都是封装在 servlet 包里，通过对数据的处理以及数据库的连接实现跳转条件，跳转到相应的界面。</p> <p>AddUserServlet: 当管理员在网页选择添加用户选项时，通过该类中的方法进行跳转到添加用户的网页。</p> <p>DeleteUserServlet: 当管理员在网页点击删除某个特定用户选项时，通过该类中的方法进行相应的操作，并且删除特定用户，更新数据库。</p> <p>CheckUserServlet: 当管理员在网页点击查看用户选项时，通过该类并且查询数据库，显示所有用户。</p> <p>ModifyUserServlet: 当管理员在网页点击修改用户选项时，通过该类跳转到修改用户信息界面。</p> <p>AddCompanyServlet: 当管理员在网页点击添加企业选项时，通过该类中的方法进行跳转到添加企业的网页。</p> <p>DeleteCompanyServlet: 当管理员在网页点击删除某个特定企业选项时，通过该类中的方法进行相应的操作，并且删除特定企业，更新数据库。</p> <p>CheckCompanyServlet: 当管理员在网页点击查看企业选项时，通过该类并且查询数据库，显示所有企业。</p> <p>ModifyCompanyServlet: 当管理员在网页点击修改企业选项时，通过该类跳转到修改企业信息界面。</p> <p>AddProjectServlet: 当管理员在网页点击添加项目选项时，通过该类中的方法进行跳转到添加项目的网页。</p> <p>DeleteProjectServlet: 当管理员在网页点击删除某个特定项目选项时，通过该类中的方法进行相应的操作，并且删除特定项目，更新数据库。</p> <p>CheckProjectServlet: 当用户在网页点击查看项目选项时，通过该类并且查询数据库，显示所有项目。</p> <p>ModifyProjectServlet: 当管理员在网页点击修改项目选项时，通过该类跳转到修改项目信息界面。</p>
补充说明	<p><b>5.2.2 DAO 层类图:</b></p>  <pre>classDiagram     class UserDao {         +login()         +update()         +getUserByID()         +getAllUser()         +delete()         +add()     }     class UserDaoImpl {     }     UserDao &lt; -- UserDaoImpl</pre>



login() 查询数据库，添加实体  
update() 修改用户信息，并且更新数据库信息  
getUserByID() 根据用户的 id 号来获得用户实体  
getAllUser() 根据 SQL 语句获取所有用户实体  
delete() 根据 id 号来删除实体  
add() 增加用户，根据 SOL 语言写入数据库



update() 修改企业信息，并且更新数据库信息  
getCompanyByID() 根据企业的 id 号来获得用户实体  
getAllCompany () 根据 SQL 语句获取所有企业实体  
delete() 根据 id 号来删除企业实体  
add() 增加企业，根据 SOL 语言写入数据库





	<p>update() 修改项目信息，并且更新数据库信息</p> <p>getProjectByID() 根据企业的 id 号来获得项目实体</p> <p>getAllProject () 根据 SQL 语句获取所有项目实体</p> <p>delete() 根据 id 号来删除项目实体</p> <p>add() 增加项目，根据 SOL 语言写入数据库</p>
--	--

## 6 服务器格式设计

### 6.1 查看用户格式

查看用户					
Show <input type="text" value="10"/> entries			Search: <input type="text"/>		
账号	姓名	性别	地址	申请日期	
11141	1412	女	1412412	2016-03-11	
14141	124124	女	141241	2016-03-11	
21548	小李	男	江西南昌	2016-03-08	
234234	23423	女	234234	2016-03-11	
2423525	2523	女	2342352	2016-03-11	
25418	Jane	女	天津	2016-03-01	
425015	Bob	男	福建省宁德	2016-01-01	
523523	235235	女	252523	2016-03-11	
5264	小明	男	福建宁德	2016-03-06	
54586	夏夏	男	江苏省南京	2015-02-26	
账号	姓名	性别	地址	申请日期	
Showing 1 to 10 of 21 entries					
First Previous <b>1</b> 2 3 Next Last					



## 6.2 增加用户格式

用户名：

密码：

姓名：

性别：

☐男☐女

地址：

增加

## 6.3 删除用户格式



## 删除用户

Show 10 entries

Search:

账号	姓名	性别	申请日期	操作
11141	1412	女	2016-03-11	删除
14141	124124	女	2016-03-11	删除
21548	小李	男	2016-03-08	删除
234234	23423	女	2016-03-11	删除
2423525	2523	女	2016-03-11	删除
25418	Jane	女	2016-03-01	删除
425015	Bob	男	2016-01-01	删除
523523	235235	女	2016-03-11	删除
5264	小明	男	2016-03-06	删除
54586	夏夏	男	2015-02-26	删除
账号	姓名	性别	申请日期	操作

Showing 1 to 10 of 22 entries

First

Previous

1

2

3

Next

Last

### 6.4 修改用户格式



## 修改用户

Show 10 entries

Search:

账号	姓名	性别	申请日期	操作
11141	1412	女	2016-03-11	修改
14141	124124	女	2016-03-11	修改
21548	小李	男	2016-03-08	修改
234234	23423	女	2016-03-11	修改
2423525	2523	女	2016-03-11	修改
25418	Jane	女	2016-03-01	修改
425015	Bob	男	2016-01-01	修改
523523	235235	女	2016-03-11	修改
5264	小明	男	2016-03-06	修改
54566	夏夏	男	2015-02-26	修改
账号	姓名	性别	申请日期	操作

Showing 1 to 10 of 21 entries

First

Previous

1

2

3

Next

Last

## 附录 2 软件测试文档

### 1.1 编写目的

本测试文档是为了发现“企业信息聚合平台”中的 bug 而编写。测试的目的是在软件发布之前，尽可能多的发现软件中的错误。本文档然是保证软件质量的关键步骤，是对软件规格说明、设计和编码的最后复审，也是必不可少的关键步骤。由于在开发软件的过程中，开发人员会面对极其错综复杂的





问题，我们开发设计人员，用户等的主观认识不可能完全符合客观现实，甚至会出现分歧或是对同一事物理解的偏差。因此，在软件生命周期的每个阶段都不可避免地会产生差错。因此，软件测试对整个软件的开发是不可或缺的，而一份完整的测试计划在开发流程里也是必须的。此份软件测试计划的读者包括：软件开发设计人员，项目管理人员，软件测试人员。

### 1.2 项目背景

自从我们进入 21 世纪以来，移动通讯技术依靠快速的更迭，当之无愧的变成现今发展最快速，最受到社会希冀和认可的新兴产业。这些年来伴随互联网，移动互联网的普及，我们的世界不知不觉已经进入到一个信息爆炸的时代，网络上出现了浩瀚无垠的信息。人们面对呼啸汹涌而来的巨量无所适从，想迅捷、高效的从广大繁多的信息中找寻自己所需要的信息，成为一件异常的艰难的工作。

诸如社会上一些企业，公益组织的活动、宣传信息等，不仅庞杂无穷，并且由于虚假、诈骗信息的存在，许多关注公司企业信息的人员无法获取到比较完善系统化的企业信息。基于这一点，我开发一款专门为企业、公益组织与个人信息服务的企业信息聚合平台（以下简称企聚平台）。企聚平台是一款自组织型网络社区平台，以 B2C 模式创建。企聚平台以为有了解各行各业和各商家公司、最近宣传活动的信息需求的个人用户，构造一个值得信赖、资讯确切、信息共享的社区平台为目的。即个人能通过该平台，借助该平台在某些领域内聚合网络资源，扩张他个人的人际辐射力和影响力。一方面，以最短的时间获取企业最为准确的、最新动态的信息；另一方面，它也是一个企业和企业客户之间的一个重要的及时的交流平台。

和于传统地通过网上搜索并且获取企业信息的方法相比较，这种企业信息聚合平台将给个人带来更多的便捷，同时个人也可以横向了解和比较同一行业的不同企业相应的信息。和另一种客户与企业之间通过电子邮件的方式对比，该平台也更具有及时性，拥有手机客户端的信息聚合平台，会愈加顺应移动互联网朝代的工作时间的碎片化，工作地点多元化，工作方式自由化的趋势。

关于企业来说，企聚平台是一个它发布企业最新动态以及活动的平台，商家可以经过在平台注册，并且填写企业信息，然后取得在本平台发布项目的权利。本平台为企业和商家提供了一个发布活动的平台，同时也是企业自我展示，推广自身企业文化，推销自己的产品的平台。

### 1.3 系统说明

#### 运行环境

.NET Framework 3.5 或更高版本

Android4.0 或更高版本

#### 软件环境

	名称	版本	语种
操作系统	Windows	XP, 2003	
操作系统的附加功能			
数据库平台	Mysql	5.0	
语言	JDK	5.0 以上	
邮件系统			
客户端软件			

#### 硬件环境





硬件配置	最低配置	推荐配置
服务器	1CPU: P4 2.0G	1CPU: P4 2.8G
	Mem: 1G	Mem: 2G
	HD: 40G	HD: 120G
客户端	1CPU: P4 1.0G	1CPU: P4 2.0G
	Mem: 512M	Mem: 1G
	HD: 40G	HD: 120G

## 接口

### 1.7.3.1 外部接口

外部接口的用户界面部分应遵循如下规则：

- 界面对用户友好，且必须对鼠标和键盘提供支持；
- 界面要具有一致性；
- 提供简单的错误处理；
- 提供信息反馈，用多种信息提示用户当前系统运行状态、系统界面；
- 操作可逆，其动作可以是单个的操作，或者是一个相对独立的操作序列；
- 设计良好的联机帮助

## 1.4 测试设计说明

用户注册模块测试：



测试用例	注册用例	
测试用例 ID	test1	
输入/动作	期望的输出/相应	实际情况
用户名：123，密码：123456	提示注册成功，跳转至登录页面	提示注册成功，跳转至登录页面
测试用例 ID	test2	
输入/动作	期望的输出/相应	实际情况
用户名:abc，密码不输入	提示请输入密码	提示请输入密码
测试用例 ID	test3	
输入/动作	期望的输出/相应	实际情况
用户名不输入，密码：abcdef	提示请输入用户名	提示请输入用户名
测试用例 ID	test4	
输入/动作	期望的输出/相应	实际情况
用户名不输入，密码不输入	提示请输入用户名和密码	提示请输入用户名和密码
测试用例 ID	test5	
输入/动作	期望的输出/相应	实际情况
用户名：123，密码123456	提示用户名已存在	提示用户名已存在
测试用例 ID	test6	
输入/动作	期望的输出/相应	实际情况
用户名：..，密码：123456	提示注册成功，跳转至登录页面	提示注册成功，跳转至登录页面
测试用例 ID	test7	
输入/动作	期望的输出/相应	实际情况
用户名：12..ab，密码：123456	提示注册成功，跳转至登录页面	提示注册成功，跳转至登录页面
测试用例 ID	test8	
输入/动作	期望的输出/相应	实际情况
正常未注册电话号码	提示注册成功，跳转至登录页面	提示注册成功，跳转至登录页面

测试用例 ID	Test9	
输入/动作	期望的输出/相应	实际情况
电话号码已注册	提示电话号码已经注册	提示电话号码已经注册



测试用例 ID	Test10		
输入/动作	期望的输出/相应	实际情况	
电话号码不符合规范或者位数不够	提示电话号码格式不对	提示电话号码格式不对	
测试用例 ID	Test11		
输入/动作	期望的输出/相应	实际情况	
注册后未填写个人资料如姓名，年龄，所属行业信息	提示填写之后才能进入主页面	提示填写之后才能进入主页面	
测试用例 ID	Test12		
输入/动作	期望的输出/相应	实际情况	
注册后填写了基本个人信息，但未添加头像	提示填写之后才能进入主页面	提示填写之后才能进入主页面	
测试用例 ID	Test13		
输入/动作	期望的输出/相应	实际情况	
注册后未填写个人资料如姓名，年龄，所属行业信息及上传了头像	提示注册成功，跳转至登录页面	提示注册成功，跳转至登录页面	

用户登录模块测试:



测试用例	登录用例		
测试用例 ID	Test1		
输入/动作	期望的输出/相应	实际情况	
输入用户名：123，密码：123456，按登录	用户登录成功，跳转至首页	用户登录成功，跳转至首页	
测试用例 ID	Test2		
输入/动作	期望的输出/相应	实际情况	
输入用户名：12，密码：123456，按登录	用户名或密码错误	用户名或密码错误	
测试用例 ID	Test3		
输入/动作	期望的输出/相应	实际情况	
输入用户名：123，密码：123，按登录	用户名或密码错误	用户名或密码错误	
测试用例 ID	Test4		
输入/动作	期望的输出/相应	实际情况	
输入用户名：123，不输入密码，按登录	提示登录失败	提示登录失败	
测试用例 ID	Test5		
输入/动作	期望的输出/相应	实际情况	
不输入用户名，输入密码 123456，按登录	提示登录失败	提示登录失败	
测试用例 ID	Test6		
输入/动作	期望的输出/相应	实际情况	
不输入用户名，不输入密码，按登录	提示登录失败	提示登录失败	

用户反馈用例：



测试用例	用户反馈用例	
测试用例 ID	Test1	
输入/动作	期望的输出/相应	实际情况
点击用户反馈	出现输入反馈信息的输入框	出现输入反馈信息的输入框
测试用例 ID	Test2	
输入/动作	期望的输出/相应	实际情况
输入反馈内容，点击提交	提示反馈成功	提示反馈成功
测试用例 ID	Test3	
输入/动作	期望的输出/相应	实际情况
输入纯数字反馈内容，点击提交	提示用户重新输入	提示反馈成功
测试用例 ID	Test4	
输入/动作	期望的输出/相应	实际情况
输入无意义反馈内容，点击提交	提示用户重新输入	提示反馈成功
测试用例 ID	Test5	
输入/动作	期望的输出/相应	实际情况
不输入反馈内容，点击提交	提示用户输入信息	提示反馈成功
测试用例 ID	Test6	
输入/动作	期望的输出/相应	实际情况
输入反馈内容，点击取消	回到首页	回到首页
测试用例 ID	test7	
输入/动作	期望的输出/相应	实际情况
不输入反馈内容，点击取消	回到首页	回到首页
测试用例 ID	Test8	
输入/动作	期望的输出/相应	实际情况
点击返回上一步	回到首页	回到首页

搜索项目用例：



测试用例	搜索项目用例	
测试用例 ID	Test1	
输入/动作	期望的输出/相应	实际情况
用户点击搜索按钮	进入搜索界面弹出键盘监听输入	进入搜索界面弹出键盘监听输入
测试用例 ID	Test2	
输入/动作	期望的输出/相应	实际情况
输入内容为空字符时，点击开始搜索	提示输入内容为空，请输入内容	提示输入内容为空，请输入内容
测试用例 ID	Test3	
输入/动作	期望的输出/相应	实际情况
用户输入无意义的内容	根据用户内容无法匹配出相应的项目，项目列表为空	根据用户内容无法匹配出相应的项目，项目列表为空
测试用例 ID	Test4	
输入/动作	期望的输出/相应	实际情况
用户输入多个项目共同具有的内容	根据模块匹配原则，根据时间的相关度，在项目列表里列出所有有关的项目	根据模块匹配原则，根据时间的相关度，在项目列表里列出所有有关的项目
测试用例 ID	Test5	
输入/动作	期望的输出/相应	实际情况
用户不输入内容，点击返回按钮	关闭搜索界面，回到首页	关闭搜索界面，回到首页
测试用例 ID	Test6	
输入/动作	期望的输出/相应	实际情况
用户输入内容之后，点击返回按钮	关闭搜索界面，回到首页	关闭搜索界面，回到首页
测试用例 ID	test7	
输入/动作	期望的输出/相应	实际情况
用户在搜索到内容之后，点击返回	关闭搜索界面，回到首页	关闭搜索界面，回到首页
测试用例 ID	Test8	
输入/动作	期望的输出/相应	实际情况
用户在搜索到项目内容以后，点击搜索列表里的某个项目	关闭搜索界面，进入项目主页	关闭搜索界面，进入项目主页



搜索企业用例：

测试用例	搜索企业用例		
测试用例 ID	Test1		
输入/动作	期望的输出/相应	实际情况	
用户点击搜索按钮	进入搜索界面弹出键盘监听输入	进入搜索界面弹出键盘监听输入	
测试用例 ID	Test2		
输入/动作	期望的输出/相应	实际情况	
输入内容为空字符时，点击开始搜索	提示输入内容为空，请输入内容	提示输入内容为空，请输入内容	
测试用例 ID	Test3		
输入/动作	期望的输出/相应	实际情况	
用户输入无意义的内容	根据用户内容无法匹配出相应的企业，企业列表为空	根据用户内容无法匹配出相应的企业，企业列表为空	
测试用例 ID	Test4		
输入/动作	期望的输出/相应	实际情况	
用户输入多个企业共同具有的内容	根据模块匹配原则，根据时间的相关度，在企业列表里列出所有有关的企业信息	根据模块匹配原则，根据时间的相关度，在企业列表里列出所有有关的企业信息	
测试用例 ID	Test5		
输入/动作	期望的输出/相应	实际情况	
用户不输入内容，点击返回按钮	关闭搜索界面，回到首页	关闭搜索界面，回到首页	
测试用例 ID	Test6		
输入/动作	期望的输出/相应	实际情况	
用户输入内容之后，点击返回按钮	关闭搜索界面，回到首页	关闭搜索界面，回到首页	
测试用例 ID	test7		
输入/动作	期望的输出/相应	实际情况	
用户在搜索到内容之后，点击返回	关闭搜索界面，回到首页	关闭搜索界面，回到首页	
测试用例 ID	Test8		
输入/动作	期望的输出/相应	实际情况	
用户在搜索到项目内容以后，点击搜索列表里的某个项目	关闭搜索界面，进入项目主页	关闭搜索界面，进入项目主页	



发起项目用例:





测试用例	发起项目用例	
测试用例 ID	Test1	
输入/动作	期望的输出/相应	实际情况
会员用户点击发起项目按钮	进入项目编辑界面，可以编辑发布项目	进入搜索界面弹出键盘监听输入
测试用例 ID	Test2	
输入/动作	期望的输出/相应	实际情况
非会员用户点击发起项目按钮	提示需要认证的企业会员才能发布活动项目	提示需要认证的企业会员才能发布活动项目
测试用例 ID	Test3	
输入/动作	期望的输出/相应	实际情况
发起项目时没有填写活动标题，点击发布按钮	提示项目信息填写不完整	提示项目信息填写不完整
测试用例 ID	Test4	
输入/动作	期望的输出/相应	实际情况
发起项目时，没有选择项目封面，点击发布按钮	提示项目信息填写不完整	提示项目信息填写不完整
测试用例 ID	Test5	
输入/动作	期望的输出/相应	实际情况
发起项目时，没有填写项目详细介绍，点击发布按钮	提示项目信息填写不完整	提示项目信息填写不完整
测试用例 ID	Test6	
输入/动作	期望的输出/相应	实际情况
发起项目时，已经填写项目信息，点击返回按钮	关闭搜索界面，回到首页	关闭搜索界面，回到首页
测试用例 ID	test7	
输入/动作	期望的输出/相应	实际情况
发起项目时，没有填写任何项目信息，点击返回按钮	关闭搜索界面，回到首页	关闭搜索界面，回到首页
测试用例 ID	Test8	
输入/动作	期望的输出/相应	实际情况
编辑项目时，使用富文本编辑器所提供的功能，为项目详情添加粗体带颜色的文字，插入图片	能够在项目详情浏览界面，正常显示带格式的文字，和图片。	能够在项目详情浏览界面，正常显示带格式的文字，和图片。



地图定位搜索附近企业用例：

测试用例	地图定位搜索附近企业用例		
测试用例 ID	Test1		
输入/动作	期望的输出/相应	实际情况	
用户连接 WiFi，点击商家地图按钮	后台启动服务开开始定位，进入地图界面，显示附近的商家企业	后台启动服务开开始定位，进入地图界面，显示附近的商家企业	
测试用例 ID	Test2		
输入/动作	期望的输出/相应	实际情况	
用户连接数据流量，点击商家地图按钮	后台启动服务开开始定位，进入地图界面，显示附近的商家企业	后台启动服务开开始定位，进入地图界面，显示附近的商家企业	
测试用例 ID	Test3		
输入/动作	期望的输出/相应	实际情况	
用户没有与互联网取得任何连接	提示没有网络，无法实现定位及搜索企业服务	提示没有网络，无法实现定位及搜索企业服务	
测试用例 ID	Test4		
输入/动作	期望的输出/相应	实际情况	
有网络情况，点击地图上的附近的某一个企业图标	进入该企业的企业详情界面	进入该企业的企业详情界面	
测试用例 ID	Test5		
输入/动作	期望的输出/相应	实际情况	
有网络情况开始搜某一个注册企业	在地图上显示该企业的位置	在地图上显示该企业的位置	
测试用例 ID	Test6		
输入/动作	期望的输出/相应	实际情况	
有网络情况开始搜某一个未注册企业	地图仍旧显示用户所在位置，并提示没有找到该企业	地图仍旧显示用户所在位置，并提示没有找到该企业	
测试用例 ID	test7		
输入/动作	期望的输出/相应	实际情况	
搜索到企业后，点击地图上改企业的图标	进入该企业的企业详情界面	进入该企业的企业详情界面	
测试用例 ID	Test8		
输入/动作	期望的输出/相应	实际情况	
搜索到企业后，不点击地图上改企业的图标，直接返回	关闭搜索界面，进入主界面	关闭搜索界面，进入主界面	



## 附录 3 用户文档说明书

### 目的

本说明书目的在于明确说明企业信息聚合平台 APP 的系统功能的使用方法，为用户提供操作指南。

### 适用范围

该系统是基于 Android 平台上的移动应用，因此该系统的用户是拥有智能手机的人。

### 术语和缩略语

术语、缩略语	解 释
Android	2007 年 Google 公司发起的 OHA 联盟推出的一种开源手机平台。
Eclipse	IDE，在这里用来搭建 Android 开发环境。
MySql	建立数据库
TomCat	一个免费的开放源代码的 Web 应用服务器
MyEclipse	IDE，用于搭建后台业务逻辑

### 使用说明

功能点名称：登陆注册

#### 功能介绍

本模块针对用户

#### 操作界面图示



3G 0.1K/s 09:28 97%

< 设置

退出登录

3G 0K/s 09:27 97%

< 注册

手机号码

密 码

确认密码

确认注册

忘记密码

点击注册



## 操作指南

点击侧拉框，选择设置，进入设置界面，点击登录，进入登录界面

登陆界面：输入账号及密码，点击登陆即可。

如无账号，可点击注册，进入注册界面，输入相关注册信息点击注册即可

## 功能点名称：查看商家地图

### 功能介绍

用户查看商家地图，并且可以在地图中进行搜索

### 操作界面图示



## 操作指南

侧拉框点击商家地图进入地图界面即可进行查看与搜索。