# 项目采用的方法、技术和可行性说明

## 一、概述

本项目总体上将采用C/S和B/S架构（前者用于为支付用户提供服务，后者为服务提供商提供数据分析支持），遵照MVC的设计模式，用户手机上将运行客户端，后台服务器则运行在服务提供商（银行、超市）。客户端主要用于完成商品信息获取、条形码识别和结算等功能。服务器端的任务则集中在实现数据持久化，数据分析和提供结算接口等功能。

## 二、完整的业务流程：

1. 用户进入商场，用户客户端登陆；

2. 客户端获取并显示当前商场的概况，促销信息等内容；

3. （此步骤为可选）用户通过客户端搜索特定商品，系统返回商品的确切位置；

4. 用户找到商品，使用条形码识别，系统返回商品信息（包括商品的厂家、价格、生产日期等自然信息，此外还包括其他用户对它的评价）；

5. 用户将想要买的商品装入购物车，返回步骤3继续购物，直至用户把想买的商品都买完；

6. 用户通过支付系统进行支付；

7. 用户离开商场。

## 三、采取的方法和技术

### 1. 手机客户端部分

根据业务流程，客户端承担了和用户的所有交互任务。为用户提供商品信息，提供条形码识别和结算功能。根据这些要求，目前，我们选取Android为系统的客户端平台（今后会向iOS拓展）。本项目除了使用Android的基本控件外，将依赖摄像头模块完成核心功能“条形码识别”。此外，GPS、Google Map API也是客户端将使用的技术。

### 2. 服务提供商客户端部分

此部分软件为Browser（浏览器），服务提供商可以登录查看其用户的支付统计数据，系统为其提供基本的数据分析图表。

页面使用JQuery编写。

### 3. 服务器部分

服务器的任务集中在业务逻辑处理和数据持久化。

前者体现在：根据用户输入和用户特性提供特定商品的信息，完成结算业务；

后者体现在：存取商品的信息，存取用户的账户信息，存取每一笔交易的信息等。

因此，服务端将基于J2EE，使用Spring 3.x 框架作为总体框架。Spring适合企业级网络应用，拥有良好的分层结构，稳定性和并发性优良。服务端程序运行在Tomcat服务器上，接收客户端的各种请求做出相应的回应。

关于数据持久化，我们采用MySql数据库管理系统，并以Hibernate框架来完成数据的存储和访问。

### 4. 服务器和客户端间的通信

两者间的通信使用HTTP协议，数据格式为JSON文本。考虑到支付系统的安全敏感性，本项目采取的安全策略为https和MD5加密。

## 四、可行性分析

总体可行性：

本项目技术组成员均为软件工程系在读本科生，熟练掌握Java语言，掌握基本的网络知识，有过Android、J2EE开发经验。在实践过程中遇到技术难题不可避免，但成员间的互帮互助和学习可以克服这些困难。

### 1. 手机客户端

1) Android开发日趋成熟，有大量的开发资料可以参考，本项目小组也有部分成员曾有Android软件（特别是移动互联网类软件）开发经验。此外Android本身的开放性以及和Google服务的无缝结合性都为客户端的开发提供了有力保障。

2) 条形码识别技术已经被运用于Android手机，有开源类库使用，这在降低开发难度的同时也为我们节省了不少精力。

3) 当前的Android硬件设备普遍配置较高，能胜任条形码识别等任务。

### 2. 服务提供商后台页面

1) jQuery作为JavaScript最为流行的框架，久经考验，成功案例数不胜数。它兼容CSS3，还兼容各种浏览器。jQuery使用户能更方便地处理HTML documents、events、实现动画效果，并且方便地为网站提供AJAX交互。

2) jQuery的文档说明很全，而且各种应用也说得很详细，同时还有许多成熟的插件可供选择。jQuery能够使用户的html页保持代码和html内容分离，也就是说，不用再在html里面插入一堆js来调用命令了，只需定义id即可。

### 3. 服务端

1) Spring 3.x框架为企业级网络应用提供了一整套工具，分层架构允许使用者选择使用哪一个组件。其稳定性，并发性也为软件的顺利运行提供了保障。

2) Hibernate是一个开放源代码的对象关系映射框架，它对JDBC进行了非常轻量级的对象封装，使得Java程序员可以随心所欲的使用对象编程思维来操纵数据库，完成数据持久化的重任。

### 4. 网络

当前各大ISP提供的网络环境越来越好，3G兴起让移动互联网的稳定性上了一个台阶。此外各大商场Wi-fi覆盖率也越来越大，为网络通信提供了一定保障。