餐谋长设计规格说明

Ver:1.0

**目 录**

[1 引言 3](#_Toc38271035)

[1.1 目标 3](#_Toc38271036)

[1.2 文档范围 3](#_Toc38271037)

[1.3 术语和缩略语 3](#_Toc38271038)

[1.4 参考资料 3](#_Toc38271039)

[1.5 系统目标和约束 3](#_Toc38271040)

[2 系统设计 4](#_Toc38271041)

[2.1 系统架构概述 4](#_Toc38271042)

[2.2 对象模型 4](#_Toc38271043)

[2.3 特性实现 4](#_Toc38271044)

[2.4 错误代码 8](#_Toc38271045)

[2.5 错误日志 8](#_Toc38271046)

[2.6 部署视图 8](#_Toc38271047)

[3 数据库设计 10](#_Toc38271048)

[3.1 逻辑模型 10](#_Toc38271049)

[3.2 物理模型 10](#_Toc38271050)

[4 质量及其他方面 11](#_Toc38271051)

[4.1 可维护性 11](#_Toc38271052)

[4.2 安全性 11](#_Toc38271053)

[4.3 可扩展性 11](#_Toc38271054)

[4.4 可靠性 11](#_Toc38271055)

[4.5 可用性 11](#_Toc38271056)

[5 附录 11](#_Toc38271057)

[5.1 附件 11](#_Toc38271058)

[5.2 修过记录 12](#_Toc38271059)

# 引言

## 目标

在完成软件开发前期的准备工作如项目需求等，结合《餐谋长项目章程》、《餐谋长范围说明书》，项目小组提出了这份软件设计说明书。

此概要设计说明书对餐谋长APP的功能分配，模块划分，程序的总体结构，运行设计，数据结构设计及出错设计等方面作了全面的概括性的说明，为软件详细设计奠定了基础，同时作为系统分析员工作的阶段性总结和程序员进行开发及未来测试的重要文档资料。

## 文档范围

本文档包括系统的架构设计、数据库设计、安全性、可靠性、可用性等方面的规划和设计。

## 术语和缩略语

参谋长：指本次团队开发的能够让用户搜索查看菜谱并购买菜品的Android APP。

## 参考资料

部分内容参考了《餐谋长需求确认书》

## 系统目标和约束

系统目标：项目需要完成餐谋长用户（买家和卖家）的注册登录功能，查看餐谋长提供的菜谱、搜索菜谱、添加菜品购物车、社区分享、评论、查看商家菜品等功能。

系统的约束：因为时间和成本限制，该APP只实现最基本的功能，运行速度和效率暂时不做优化，服务端部署在个人电脑上，数据安全性可能不高。

# 系统设计

## 系统架构概述

该系统的架构采用两层架构的模式：展示层（视图+业务逻辑）和数据访问层。采用面向对象的设计方法。

## 对象模型

该系统采用面向对象的设计方法。

## 特性实现

1. 启动页展示



1. 买家、卖家注册
   1. 注册页面
   2. 注册成功，可选择进入登录页面

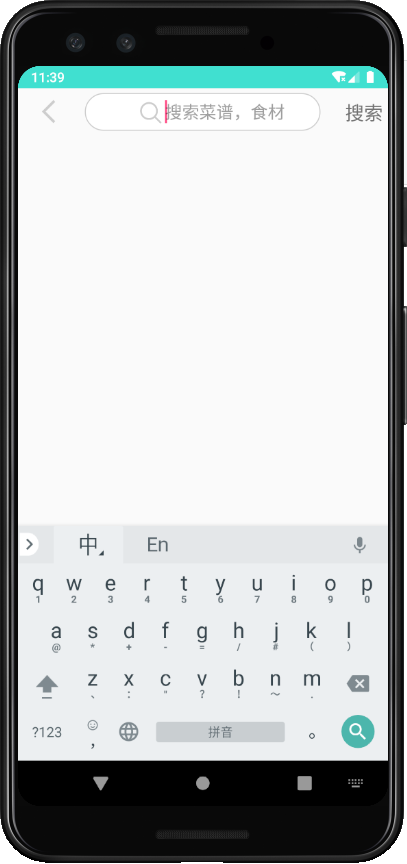




1. 用户浏览首页



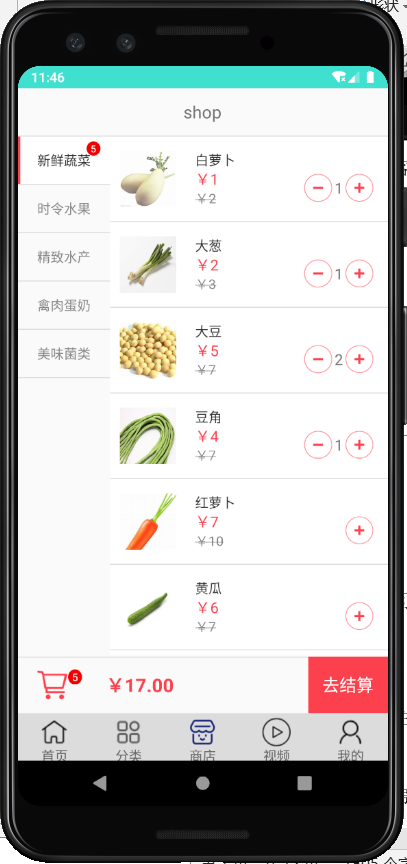
1. 用户搜索菜谱
   1. 用户点击主界面上方搜索栏
   2. 输入所需菜谱名称
   3. 后台自动搜索匹配的菜品



1. 用户通菜品分类查找所需菜品



1. 用户查看商家所提供的菜品
   1. 用户点击菜品的加减按钮选择菜品以及数量
   2. 菜品保持在购物车中方便下单结算



## 错误代码

500 服务器遇到意外情况，而无法完成请求。

## 错误日志

网络错误，用户的网络未连接或不通畅

服务器压力过大，系统崩溃

## 部署视图

1. 硬件环境：
   1. 需要互联网
   2. 开发人员的电脑作为服务器
2. 软件：
   1. Android Studio
   2. Eclipse
   3. MySql
   4. Android 5.0或更高版本
   5. 以Tomcat作为WEB服务器。
3. 配置：
   1. 下载并安装JDK
   2. 配置电脑环境变量，

变量 JAVA\_HOME 值为JDK的安装目录

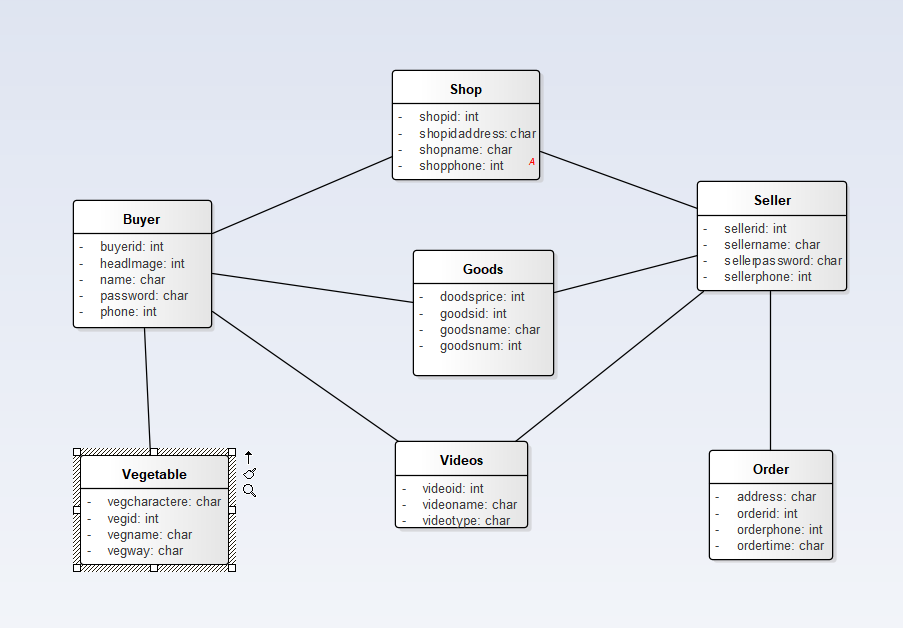
变量 Path 值为 % JAVA\_HOME %\bin;% JAVA\_HOME %\jre\bin;

* 1. 下载并安装Eclipse软件
  2. 下载并安装Android Studio软件
  3. 下载Tomcat并解压
  4. 在Eclipse中配置Tomcat服务器
  5. 下载并安装MySQL。

1. 部署：
   1. 下载Tomcat并解压
   2. 在Eclipse中配置Tomcat服务器。

# 数据库设计

## 逻辑模型



## 物理模型

1. 为每个表初始设置50M的存储空间，以10%的大小扩展。
2. 为用户上传的头像图片预留空间。
3. 为用户发布的社区文章及图片预留空间。

# 质量及其他方面

## 可维护性

1. 通过程序注释等方式增加代码的可读性和可维护性。
2. 将数据访问层分离，做成一个个函数，由其它层调用，以增加代码的可维护性。
3. 数据库有日志记录，系统一旦出现故障有恢复到故障之前的信息和数据的能力

## 安全性

1. 密码使用md5加密
2. 对用户的输入进行验证
3. 对用户的输入的特殊字符进行转义，防止sql注入攻击

## 可扩展性

1. 可以通过增加硬件资源的方式提高系统的响应速度。
2. 可以通过修改代码，扩充系统的功能。

## 可靠性

1. 数据库按时备份，有日志记录

## 可用性

1、做好充分的前期工作，系统一旦投入使用，尽量减少服务器宕机的次数和时间

# 附录

## 附件

墨刀原型

https://next.modao.cc/app/9e92e9b5a7a089f777d77f28dc27449169b3d3aa#screen=sk32ort883x8ydw

## 修过记录

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 版本 | 修改人 | 修改时间 | 修改内容 | 核准 |
| 1.0 | 袁世航、张金乾 | 2020-4-19 |  |  |