**基于Andriod的餐厅点餐系统体系结构设计**

目录

**[一、系统物理结构](#_Toc14073_WPSOffice_Level1)** **[1](#_Toc14073_WPSOffice_Level1)**

**[二、 系统数据流程图](#_Toc7588_WPSOffice_Level1)** **[1](#_Toc7588_WPSOffice_Level1)**

[2.1 客户端数据流程](#_Toc17754_WPSOffice_Level2) [2](#_Toc17754_WPSOffice_Level2)

[2.2 服务器端数据流程图](#_Toc10393_WPSOffice_Level2) [2](#_Toc10393_WPSOffice_Level2)

**[三、 类图设计](#_Toc17754_WPSOffice_Level1)** **[3](#_Toc17754_WPSOffice_Level1)**

**[四、 系统功能设计](#_Toc20393_WPSOffice_Level1)** **[4](#_Toc20393_WPSOffice_Level1)**

[4.1 系统功能架构](#_Toc17481_WPSOffice_Level2) [5](#_Toc17481_WPSOffice_Level2)

[4.2 功能模块设计](#_Toc18876_WPSOffice_Level2) [5](#_Toc18876_WPSOffice_Level2)

**[五、 手机客户端与服务器的交互设计](#_Toc10393_WPSOffice_Level1)** **[8](#_Toc10393_WPSOffice_Level1)**

[5.1](#_Toc31362_WPSOffice_Level2)[采用 http 的 Post 方法接收数据](#_Toc31362_WPSOffice_Level2) [8](#_Toc31362_WPSOffice_Level2)

[5.2 配置服务器端组件集成文件](#_Toc28766_WPSOffice_Level2) [9](#_Toc28766_WPSOffice_Level2)

## 一、系统物理结构

根据前文的需求分析可以知道，基于 Android 平台的点餐系统需满足两种不同类型的处理需求，第一种为顾客利用手机客户端完成点餐流程 ; 第二种为顾客在服务员的协助下完成就餐。而目前应用较多的架构为 C/S 模式，即客户端服务器模式，另一种为 B/S 模式，即浏览服务器模式。基于 Android 平台的点餐系统采用的是 C/S 和 B/ S 的混合模式结构，其中手机客户端的实现采用的是 C/S 模式，而服务器端基于 B/S 模式进行实现。本系统的系统物理结构如图 3-1 所示：

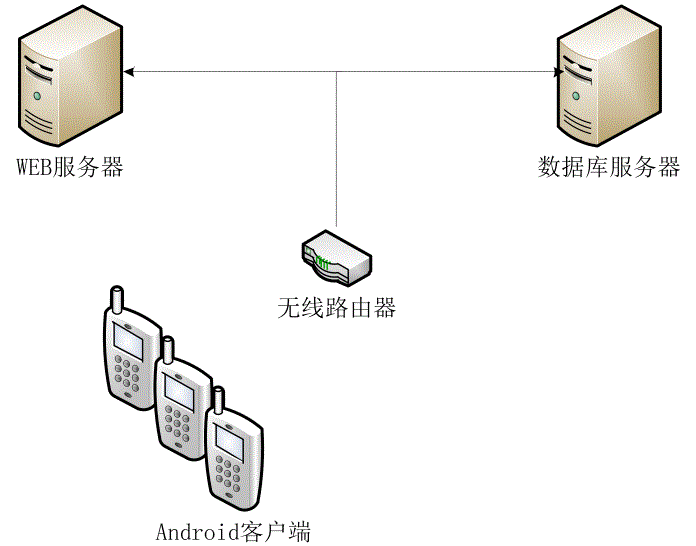


图 3-1 点餐系统物理结构

由点餐系统物理结构图 3-1 可以知道，餐厅提供两台服务器，一台为 web 服务器，用于运行点餐系统的后台，一台为数据库服务器，用于保存系统的数据信息。同时客户端 Android 系统设备通过无线网络访问后台服务器，如果需要数据访问，则访问后台数据库服务器。

## 二、 系统数据流程图

基于 Android 平台的点餐系统由客户端和服务器端两部分组成，两部分具备的功能是不一样，客户端主要负责点菜，服务器端负责信息管理。本节将从客户端数据流程和服务器端数据流程两部分对其进行介绍。

### 2.1 客户端数据流程

客户端的基本流程如下所示 :

(1) 用户输入正确的登录凭证进入系统，若登录凭证不对，系统会自动返回到登录页面，提示用户重新输入登录信息，直到用户成功登录进入系统。

(2) 用户成功进入系统后，可先进行查台操作，若餐厅存在空余座位，可进行开台操作。

(3) 开台完成后，用户可进行点菜，点菜完成由服务员进行下单操作。

(4) 就餐后，可进行结账。

根据上述流程可得到客户端的数据流程图，具体如图 3-2 所示：

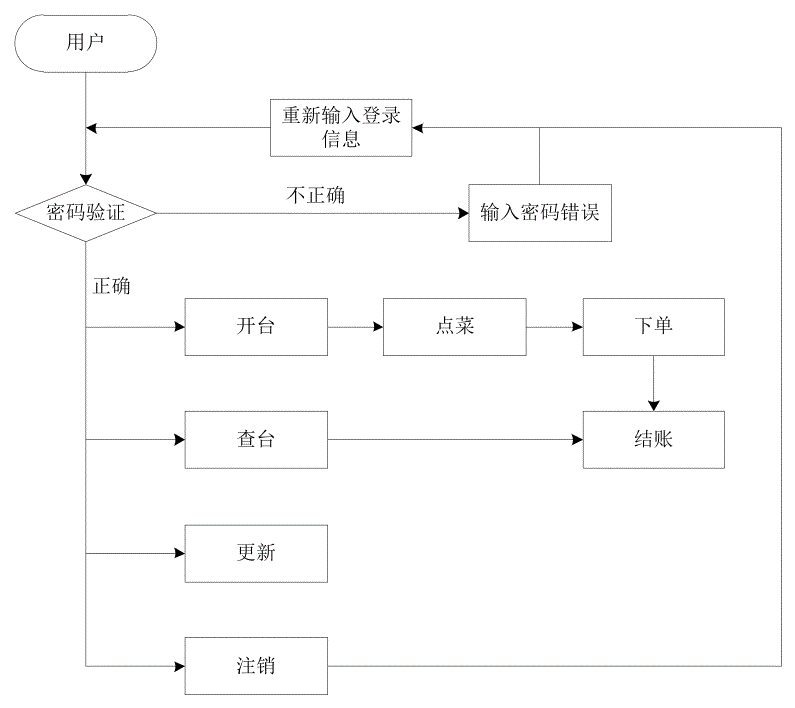


图 3-2 客户端数据流程图

### 2.2 服务器端数据流程图

服务器端的基本流程如下所示 :

(1) 用户输入正确的登录凭证进入系统，若登录凭证不对，系统会自动返回到登录页面，提示用户重新输入登录信息，直到用户成功登录进入系统。

(2) 用户成功进入系统后，可进行各类信息的管理，如餐桌信息管理、用户信息管理、订单管理以及菜谱管理等，其实就是对上述信息进行添加、删除以及修改操作。

为了简化流程图，本节以菜谱管理为例进行说明，不再将所有信息加入到流程图中，根据上述描述可得到点餐系统服务器端的数据流程图，具体如图 3-3 所示：

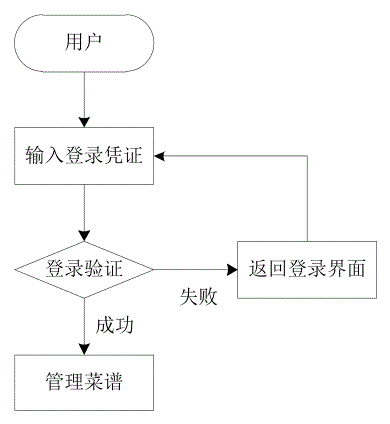


图 3-3 服务器端数据流程图

## 三、 类图设计

类图是面向对象系统建模中最常用的图，描述的是系统中类的静态结构。类图不仅定义了系统中的类、表现类与类之间的关联、依赖、类属、聚合、组合等关系，还描述了类的内部结构，包括类的属性和类的方法，此外，类图还描述了类集、接口集、协作以及他们之间的关系。因为类图所描述的是一种静态关系，所以它在系统的整个生命周期内都是有效的。类图是对象图、时序图、协作图、组件图、配置图的基础。在类图的基础之上、用例图、状态图、活动图进一步描述系统其他方面的特性。由于本系统包含的类比较多，同时每个类中实现的方法较多，在类图中不展示所有方法，只以重点方法为例来对其进行说明，点餐

系统的类图如图 3-4 所示：



图 3 - 4 点餐 系统类图

由点餐系统程序类图可以知道，首先系统中包含大量的类，如人员管理类、评价类以及选餐类等，同时各个类之间存在不同的关联关系，首先人员管理类实现对系统用户的管理，如厨师、经理以及服务员等，其次统计系统会调用收费系统中的方法来完成相关操作，除此以外，系统用户还会调用其他类的功能，如经理会调用打印报表类的打印方法。同时类图中包含很多实体，如订单，其包含属性包括菜单名、单价以及总价等，而收费系统只包含一个操作即收费功能。

## **四、 系统功能设计**

本节主要介绍基于 Android 平台的点餐系统的功能结构，具体从系统功能架构以及模块设计等两部分进行说明。

### 4.1 系统功能架构

本节的主要工作在于明确点餐系统的功能模块组成，并明确各子模块的功能组成。它是需求分析和系统实现的桥梁，也是系统设计阶段的最开始和最重要的阶段。

根据上一节的系统需求分析可知，基于 Android 平台的点餐系统由登录模块、点餐模块、结账模块、餐台管理模块以及更新模块等几大模块组成，每个模块的作用各不相同，并且彼此之间又存在依赖关系，通过分析上述模块之间的联系以及系统需求，可得到点餐系统的功能架构，具体如图 3-5 所示：

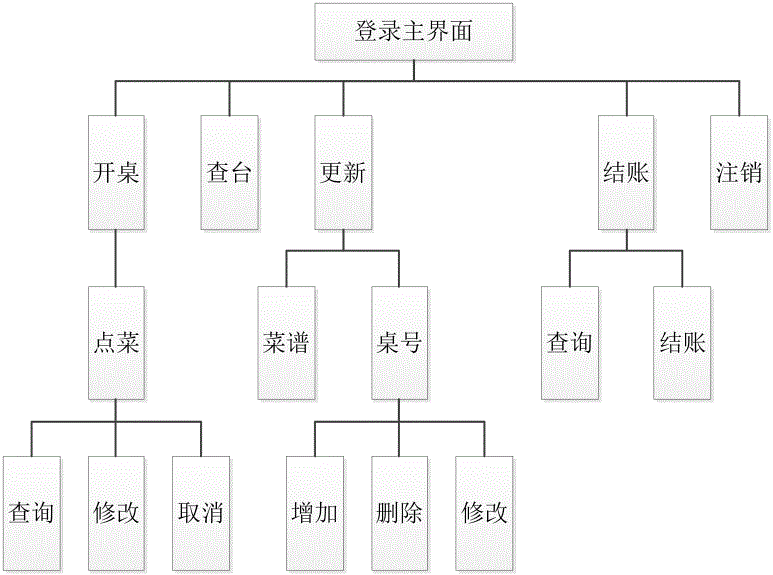


图 3-5 点餐系统功能架构

### 4.2 功能模块设计

由于点餐系统包含两个部分，客户端和服务器，下面分两部分对系统的功能模块进行介绍。

(1) 客户端

根据点餐系统的功能架构可以知道，客户端包括的主要功能有登录功能、系统主菜单功能、点餐功能、结账功能、查台功能以及更新功能等，下面对上述功能进行详细介绍。

①系统登录功能

任意用户在使用点餐系统前，必须提供正确的登录凭证才能进入系统，否则会拒绝用户使用系统相关功能，对于未注册的用户可事先进行注册，这样就可以提高系统的安全性，防止被恶意用户使用。同时，为了让系统有良好的人机交互界面，本软件系统用户界面布局应考虑到用户的操作，做到功能布局明显，空间分配平衡，以及界面结构和风格的一致。

②系统主菜单

用户在登录进入点餐系统后，系统会首先将主菜单展示给用户，如点菜以及菜谱查询等，点击菜单项，系统会自动跳转到对应的功能处理页面。本软件在进行设计时充分考虑到系统的简易操作性，具有操作简单，快捷的特点，并尽可能保证在操作上能够与通用的一些常用软件保持一致，以此缩短用户对系统操作的理解时间。同时并且减少用户发生错误选择的可能性。

③点餐功能

点餐功能是点餐系统中的核心功能，它主要处理用户的点餐要求，并由此生成菜单，在顾客点餐完成以及服务员下单后，系统会自动将生成的菜单信息传输到前台服务器，由经理和厨房进行处理。

此功能包括了三部分 : 开桌、点菜和下单。点菜部分的菜品展示可以按类别分开显示，类似书籍翻页的效果，方便顾客点菜。对提交的订单支持修改或取消。同时，尽可能地在界面中使用能适应用户本身的语言，而不能是设计者的语言。在设计界面时必须要考虑到人类大脑处理信息是不稳定并且有限度的。所以对于用户来说，浏览信息要比一记忆更加容易一些。这也是符合记忆负担最小化原则的。

④结账功能

顾客在就餐完成后可通过点击“结账”按钮来向前台提交结账请求，具体的结账流程如下所示 : 首先收银员接收结账请求后，可根据菜单详细列表来进行金额的计算，并以此为最终结果。由于涉及到金额，因此要求计算结果准确，同时不能出现数据混乱的情况，在出现数据库故障的情况下，可实现数据的恢复，防止数据丢失给餐厅带来不必要的损失。

⑤查台功能

当顾客进入餐厅时，服务员需进行查台操作，以此来确定当前餐厅是否有合适的空余座位。若没有空余座位，则需将相关情况告知顾客，请顾客耐心等待。若有空余座位，则引导客户到对应座位就座。

⑥更新功能

为了提高程序的运行效率，将服务器中菜品表 (menutbl) 和餐桌表 (tabletbl) 中的数据保存到客户端 SQLite 数据库中。因此，当服务器中的数据改变时需要及时将服务器中的数据动态地更新到客户端，更新功能就是为此而设计的。

(2) 服务器端

根据前文的分析可以知道，服务器端包括的功能有登录功能、用户管理、菜谱管理以及系统管理等，下面对上述功能进行详细介绍，在需求分析的时候我们已经决定使用公共服务层进行操作设计，具体构件设计如图 3-6 所示：



图 3-6 公共服务层构件图

①登录功能

任意用户进入服务器端，均需提供正确的用户名和密码，否则拒绝用户进入，由于服务器中存储着餐厅非常重要的信息，若用户信息以及菜谱信息等，若上述信息丢失或被窃取，均会给餐厅带来不必要的损失，由此可提高系统安全性。于是我们设计登录控制器实现安全控制，其作用方式如图 3-7 所示：



图 3-7 系统管理员登录顺序图

② 用户管理

用户管理的主要作用在于实现用户的添加、删除以及修改功能。管理员进入系统后可对恶意用户进行删除，也可手动添加管理员用户，或者修改个人信息，如密码等。

③菜谱管理

菜谱管理包含菜品添加、菜品删除以及菜品修改三个部分。管理员可利用菜品添加功能为餐厅新增菜品，丰富餐厅的菜系。当餐厅中某些菜被淘汰或者不再供应时，可对其进行删除操作。对于菜谱描述错误或需修改价格时，管理员可对菜品相关信息进行修改。

④系统管理

系统管理模块包含数据备份和数据恢复两个功能，其中数据备份的作用在于将点餐系统的数据导出到本地，当遇到数据库故障需要恢复数据库时可使用备份数据。数据恢复需使用到数据备份功能中备份的数据，以此来修复数据库的故障，该模块的作用在于提高系统的可靠性。

## **五、 手机客户端与服务器的交互设计**

手机客户端与服务器之间通过 WLAN 接口进行通讯，主要是在餐厅安装无线路由器来转换信号，下面对 Android 手机客户端与服务器的交互进行介绍。首先在 Android 手机终端发送请求后，接收请求的 web 服务器端会根据请求的数据信息进行运算，然后把运算结果响应给手机客户端。下面将对整个交互流程进行说明。

(1) 采用 http 的 Post 方法接收数据

该步骤包含四个部分，具体如下 :

①引用所需 API 在 Web 端开发应用程序时，首先需要利用 Sun 公司提供的必要类包 (API) ，包括数据流类 java.io.\* ，交互界面类 java.util. ，数据通信端口类 javax.servlet.\* ，通信协议类 j avax.servlet.http. ，用 Spring 架构通过 XML 文件注入数据对象类。

②创建 Servlet 类数据端口创建 Servlet 对象就是创建了 Web 应用服务器和连接互联网上所用客户端通信设备的数据接收和发送端口。

③访问持久层检索数据这里应用已经完善的 Spring 管理事务的对象方法来获取持久层的数据。

④创建输出流数据流对象这里创建的输出流对象主要是把响应的数据发送到手机端，由于可能是不同类型的数据，所以采用 Datao utputStream 对象来处理，用输出过滤流对象将不同类型的数据返回手机终端。

(2) 配置服务器端组件集成文件

Web 应用服务器端用到的是 SSH 架构技术，服务器的业务逻辑已经成熟，只需要将 Spring 的业务通过前述 Servlet 进行封装 . 就可以将所有的 Spring 管理的事物业务通过 Servlet 和 Android 手机进行交互，方法是在 Spring 的 ApplicationContext.xml 文件中增加相应的配置。