娄底职业技术学院

《基于Android开发框架基础》

实训报告

|  |
| --- |
| 题目：基于Android的网上订餐系统 |

类型：☑产品设计 □工艺设计 □方案设计

|  |  |
| --- | --- |
| 学 号： | 201702050343 |
| 班 级： | 17级软件3班 |
| 姓 名： | 郭非凡 |
| 指导老师： | 邓伟华 |

摘要

当下，网络的迅速发展影响到了人们的方方面面。越来越多的人习惯通过网络处理各种事务。基于Android的网上订餐系统是在Android手机平台基础上的网上订餐系统，能很好的适应现代化商业的需求。它能提高服务效率，减少服务等待时间，为人们省去拥挤的烦恼。

基于Android的网上订餐系统分为布局文件、Activity文件和服务器端，布局文件主要功能包括：界面布局、图像、动画；Activity文件主要功能包括：跳转、连接服务器、计算；服务器端主要功能包括：数据存储、Json文件。这个系统采用JAVA,并利用Android SDK、Tomcat、Json等开发环境或者开发工具。

**关键词**：Android；Java；Tomcat；移动电子商务

目 录

[第1章 系统概述 1](#_Toc11745967)

[1.1 设计背景 1](#_Toc11745968)

[1.2 基础技术分析 1](#_Toc11745969)

[1.2.1 Android Studio简介 1](#_Toc11745970)

[1.2.2 Tomcat简介 2](#_Toc11745971)

[1.2.3 Json简介 2](#_Toc11745972)

[1.2.4 Java简介 3](#_Toc11745973)

[1.3 开发环境 3](#_Toc11745974)

[1.4 任务安排 3](#_Toc11745975)

[1.4.1 前期任务 3](#_Toc11745976)

[1.4.2 中期任务 4](#_Toc11745977)

[1.4.3 后期任务 4](#_Toc11745978)

[1.5 模块说明 4](#_Toc11745979)

[1.6 实施步骤及进度安排 5](#_Toc11745980)

[第2章 系统分析 6](#_Toc11745981)

[2.1 需求分析 6](#_Toc11745982)

[2.2 数据分析 6](#_Toc11745983)

[2.3 逻辑分析 6](#_Toc11745984)

[第3章 系统设计 7](#_Toc11745985)

[3.1 服务器设计 7](#_Toc11745986)

[3.2 界面设计（部分主要代码） 8](#_Toc11745987)

[3.2.1 返回键标题栏 8](#_Toc11745988)

[3.2.2 显示店铺列表 9](#_Toc11745989)

[3.2.3 店铺列表中的店铺信息 10](#_Toc11745990)

[3.2.4 店铺详情界面 12](#_Toc11745991)

[3.2.5 购物车详细信息 13](#_Toc11745992)

[第4章 系统实现（部分主要代码） 17](#_Toc11745993)

[4.1 连接服务器 17](#_Toc11745994)

[4.2 解析服务器Json数据 17](#_Toc11745995)

[4.3 获取服务器数据 18](#_Toc11745996)

[4.4 实现菜单显示与购物车功能 21](#_Toc11745997)

[4.5 实现订单显示与支付功能 30](#_Toc11745998)

[第5章 系统测试与部署 33](#_Toc11745999)

[5.1 店铺首页 33](#_Toc11746000)

[5.2 店铺详情与菜单界面 34](#_Toc11746001)

[5.3 菜品详情界面 35](#_Toc11746002)

[5.4 购物车界面与确认清空购物车弹窗 36](#_Toc11746003)

[5.5 订单界面 37](#_Toc11746004)

[5.6 支付二维码界面 38](#_Toc11746005)

[5.7 打包过程 39](#_Toc11746006)

[总结 40](#_Toc11746007)

[参考文献 41](#_Toc11746008)

[致谢 42](#_Toc11746009)

# 第1章 系统概述

## 1.1 设计背景

随着手机应用的迅猛发展，其应用已经普及到经济和社会生活的各个领域。为了适应高效的现代生活，使订餐信息的发布更为及时、高效，应用网络信息化手段对订餐系统进行管理已经显得非常必要。 当学生们忙碌了整个上午后，享用丰富的午餐成了基本的要求，而网上订餐系统正是为了帮助人们寻找和发现新的菜种，品尝不同风格和不同种类的菜品。因此，面向大学生的网上订餐系统的设计有着良好的现实意义，网上订餐业务在校园也有极大的发展空间和良好的发展前景。

本系统在开发过程中力求系统的全面性、通用性，主要利用了Android Studio 作为开发环境，在Tomcat上搭建一个小型服务器，实现了一个简单的网上订餐系统。

## 1.2 基础技术分析

### 1.2.1 Android Studio简介

Android Studio是Google于2013 I/O大会针对Android开发推出的新的开发工具，目前很多开源项目都已经在采用，Google的更新速度也很快，明显能感觉到这是Android开发的未来。

Google推出的：毫无疑问，这个是它的最大优势，Android Stuido是Google推出，专门为Android“量身订做”的，是Google大力支持的一款基于IntelliJ IDEA改造的IDE，这个应该能说明为什么它是Android的未来。

速度更快：Eclipse的启动速度、响应速度、内存占用一直被诟病，相信大家这点应该深有体会，而且经常遇到卡死状态。Studio不管哪一个方面都全面领先Eclipse。

UI更漂亮：I/O上演示的那款黑色主题真是太棒了，极客范，Stuido自带的Darcula主题的炫酷黑界面实在是高大上，相比而言Eclipse下的黑色主题太low了。

更加智能：提示补全对于开发来说意义重大， Studio则更加智能，智能保存，从此再也不用每次都 Ctrl + S了。熟悉Studio以后效率会大大提升。

整合了Gradle构建工具：Gradle是一个新的构建工具，自Studio亮相之处就支持Gradle，可以说Gradle集合了Ant和Maven的优点，不管是配置、编译、打包都非常棒。

强大的UI编辑器：Android Studio的编辑器非常的智能，除了吸收Eclipse+ADT的优点之外，还自带了多设备的实时预览，相对这对Android开发者来说简直是神器啊。

内置终端：Studio内置终端，这对于习惯命令行操作的人来说简直是福音啊，再也不用来回切换了，一个Studio全部搞定。

更完善的插件系统：Studio下支持各种插件，如Git、Markdown、Gradle等等，你想要什么插件，直接搜索下载。

完美整合版本控制系统：安装的时候就自带了如GitHub, Git, SVN等流行的版本控制系统，可以直接check out你的项目。

### 1.2.2 Tomcat简介

Tocat是由Apache 开发的一个Servlet 容器实现了对Servlet 和 JSP 的支持，并提供了作为Web服务器的一些特有功能，体现了最新的Servlet 和JSP 规范性，由于 Tomcat 本身也内含了一个 HTTP 服务器，它也可以被视作一个单独的 Web 服务器。但是，不能将 Tomcat 和 Apache HTTP 服务器混淆，Apache HTTP 服务器是一个用 C 语言实现的 HTTP Web 服务器；这两个 HTTP web server 不是捆绑在一起的。Tomcat 包含了一个配置管理工具，也可以通过编辑XML格式的配置文件来进行配置。Tomcat 的性能极佳、技术先进，拥有轻量级应用服务器，有一个免费的开放源代码的Web 应用服务器，不依赖系统单独运行。

### 1.2.3 Json简介

[JSON](https://baike.baidu.com/item/JSON)([JavaScript](https://baike.baidu.com/item/JavaScript) Object Notation, JS 对象简谱) 是一种轻量级的数据交换格式。它基于 [ECMAScript](https://baike.baidu.com/item/ECMAScript) (欧洲计算机协会制定的js规范)的一个子集，采用完全独立于编程语言的文本格式来存储和表示数据。简洁和清晰的层次结构使得 JSON 成为理想的数据交换语言。 易于人阅读和编写，同时也易于机器解析和生成，并有效地提升网络传输效率。

### 1.2.4 Java简介

java是一种高级的面向对象的程序设计语言 。

面向对象：java语言的基础。java将一切问题都看做对象与对象之间的交互，将对象抽象成方法与属性的集合。

分布性：包含操作分布性与数据分布性两个方面。操作分布性是指由多个主机共同完成一项功能，数据分布性是分布在多台主机上的数据当做一个完成的整体处理。

跨平台：java语言编写的应用程序，可以由一种平台迁移到另一种平台。

解释型：使用java语言编写的源码被转化为字节码，JVM将字节码翻译成机器码就可以被计算机执行。

安全性：java语言的底层设计可以有效避免非法操作。

健壮性：java提供了许多机制防止运行时出现严重错误，如编译时类型检查、异常处理。

多线程：java支持进程内部多个线程同时工作。

## 1.3 开发环境

操作系统： windows10

开发工具：Android Studio 3.2

服务器：Apache-Tomcat-9.0.0.M9-windows-x64

API版本：Android API 27

开发语言：java

数据：Json

## 1.4 任务安排

### 1.4.1 前期任务

（1）查阅相关资料，以网上点餐系统为基础，经过调研、分析现有的管理模式和已有的管理软件，来建立最佳的系统模型。

（2）完成软件界面设计、Json设计、Activity的逻辑结构设计

（3）完成界面设计，主要包括：店铺页、店铺详情页、购物车、订单支付页这四个页面。

（4）编写Json文件编写，完成服务器搭建。

### 1.4.2 中期任务

（1）完成Activity主要模块功能：界面跳转、购物车金额自动计算

（2）完成跟与服务器连接，把Json文件的数据显示在相关的页面。

### 1.4.3 后期任务

（1）完善程序并解决错误

（2）编写实训报告

## 1.5 模块说明

网上订餐项目主要分为两大功能模块，分别为店铺和订单，这两个模块的结构如图1-5所示。

由图1-5可知，店铺模块包含店铺列表界面与店铺详情界面，店铺列表界面用于显示各个店铺的信息，店铺详情界面不仅显示店铺的详细信息，还显示各店铺中的菜单列表信息与购物车列表信息。订单模块包含确认订单界面与支付界面，确认订单界面用于显示购物车中己添加的商品信息，支付界面用于显示付款的二维码信息。

订餐系统

订单

店铺

确认订单

店铺详情

店铺列表

购物车

支付

图1-5 项目模块结构

## 1.6 实施步骤及进度安排

设计选题、查阅资料、了解需求、设计方案 06.01

搭建服务器，编写Json文件 06.02

设计界面布局并编写文件 06.03-06.04

Activity文件编写、功能实现 06.05-06.06

完善程序编写实训报告 06.07

# 第2章 系统分析

## 2.1 需求分析

确定系统的功能要求：功能需求并不是软件开发过程中的唯一需求。在保证功能需求的同时，还得满足性能需求、可靠性需求、可用性需求、接口需求、约束、逆向需求以及将来可能提出的软件升级的需求。同时系统反应速度较快，界面应美观大方，风格统一，突出特色，功能设计应体现人性化思想，方便用户操作。

## 2.2 数据分析

对系统数据进行分析：大多数的软件系统主要用来处理一些信息，在整个系统中，信息的处理起着至关重要的作用，它关系着整个系统的数据显示。对整个软件设计有深远的影响。

## 2.3 逻辑分析

实现设计中的逻辑模型：系统分析员往往把需要完成的系统转化成相应的逻辑模型，通过实现逻辑模型来完成系统。

# 第3章 系统设计

完成软件系分析后，我们就开始做系统设计，我们认真的构思了系统的整体结构，在哪一阶段需要做什么东西，规划出详细的蓝图。

## 3.1 服务器设计

编写Json文件：shop\_list\_data.json（部分代码）

[{

"id": 1,

"shopName": "蛋糕房",

"saleNum": 996,

"offerPrice": 100,

"distributionCost": 5,

"welfare": "进店可免费获得一个冰淇淋",

"time": "配送约2-5小时",

"shopPic": "http://192.168.43.135:8080/order/img/shop/shop1.png",

"foodList": [{

"foodId": "1",

"foodName": "招牌丰收硕果12寸",

"taste": "水果、奶油、面包、鸡蛋",

"saleNum": "50",

"price": 198,

"count": 0,

"foodPic": "http://192.168.43.135:8080/order/img/food/food1.png"

},

]

},

{

"id": 2,

"shopName": "黑潮奶茶",

"saleNum": 1000,

"offerPrice": 15,

"distributionCost": 2,

"welfare": "注册会员第一杯免费",

"time": "配送约1小时",

"shopPic": "http://192.168.43.135:8080/order/img/shop/shop2.png",

## 3.2 界面设计（部分主要代码）

"foodList": [{

"foodId": "1",

"foodName": "招牌奶茶",

"taste": "水、奶、茶",

"saleNum": "50",

"price": 10,

"count": 0,

"foodPic": "http://192.168.43.135:8080/order/img/food/food4.png"

}

]

}

]

### 3.2.1 返回键标题栏

title\_bar.xml

*<?***xml version="1.0" encoding="utf-8"***?>*<**RelativeLayout  
 xmlns:android="http://schemas.android.com/apk/res/android"  
 android:id="@+id/title\_bar"  
 android:layout\_width="match\_parent"  
 android:layout\_height="50dp"  
 android:background="@android:color/transparent"**>  
 <**TextView  
 android:id="@+id/tv\_back"  
 android:layout\_width="50dp"  
 android:layout\_height="50dp"  
 android:layout\_alignParentLeft="true"  
 android:layout\_centerVertical="true"  
 android:background="@drawable/go\_back\_selector"** />  
 <**TextView  
 android:id="@+id/tv\_title"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_centerInParent="true"  
 android:textColor="@android:color/white"  
 android:textSize="20sp"** />  
</**RelativeLayout**>

### 3.2.2 显示店铺列表

activity\_shop.xml

*<?***xml version="1.0" encoding="utf-8"***?>*<**LinearLayout  
 xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:background="@android:color/white"  
 android:orientation="vertical"**>  
 <**include layout="@layout/title\_bar"** />  
 <**ScrollView  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:background="#fcf0c0"  
 android:scrollbars="none"**>  
 <**LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:orientation="vertical"**>  
 <**ImageView  
 android:id="@+id/iv\_img"  
 android:layout\_width="match\_parent"  
 android:layout\_height="180dp"  
 android:scaleType="fitXY"  
 android:src="@drawable/banner"** />  
 <**cn.itcast.order.views.ShopListView  
 android:id="@+id/slv\_list"  
 android:layout\_width="fill\_parent"  
 android:layout\_height="fill\_parent"  
 android:layout\_marginLeft="8dp"  
 android:layout\_marginRight="8dp"  
 android:layout\_marginTop="8dp"  
 android:dividerHeight="8dp"  
 android:scrollbars="none"** />  
 </**LinearLayout**>  
 </**ScrollView**>  
</**LinearLayout**>

### 3.2.3 店铺列表中的店铺信息

shop\_car.xml

*<?***xml version="1.0" encoding="utf-8"***?>*<**RelativeLayout  
 xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"**>  
 <**FrameLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="65dp"  
 android:layout\_alignParentBottom="true"**>  
 <**RelativeLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="50dp"  
 android:layout\_marginTop="15dp"  
 android:background="@color/car\_gray\_color"  
 android:paddingLeft="60dp"**>  
 <**TextView  
 android:id="@+id/tv\_money"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="25dp"  
 android:layout\_marginLeft="4dp"  
 android:layout\_marginTop="4dp"  
 android:gravity="center"  
 android:text="未选购商品"  
 android:textColor="@color/light\_gray"  
 android:textSize="16sp"** />  
 <**TextView  
 android:id="@+id/tv\_distribution\_cost"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_below="@id/tv\_money"  
 android:layout\_marginLeft="4dp"  
 android:textColor="@android:color/white"  
 android:textSize="12sp"  
 android:textStyle="bold"  
 android:visibility="gone"** />  
 <**TextView  
 android:id="@+id/tv\_settle\_accounts"  
 android:layout\_width="100dp"  
 android:layout\_height="match\_parent"**

**android:layout\_alignParentRight="true"  
 android:layout\_centerVertical="true"  
 android:background="@color/account\_color"  
 android:gravity="center"  
 android:text="去结算"  
 android:textColor="@android:color/white"  
 android:textSize="16sp"  
 android:visibility="gone"** />  
 <**TextView  
 android:id="@+id/tv\_not\_enough"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="match\_parent"  
 android:layout\_alignParentRight="true"  
 android:background="@color/account\_gray\_color"  
 android:gravity="center"  
 android:padding="8dp"  
 android:textColor="@color/light\_gray"  
 android:textSize="16sp"** />  
 </**RelativeLayout**>  
 <**include layout="@layout/car"** />  
 </**FrameLayout**>  
</**RelativeLayout**>

### 3.2.4 店铺详情界面

activity\_shop\_detail.xml

*<?***xml version="1.0" encoding="utf-8"***?>*<**FrameLayout  
 xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"**>  
 <**LinearLayout  
 android:id="@+id/ll\_intro"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:background="@color/shop\_bg\_color"  
 android:orientation="vertical"**>  
 <**include layout="@layout/title\_bar"** />  
 <**LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:layout\_marginTop="65dp"  
 android:background="@android:color/white"  
 android:orientation="vertical"  
 android:paddingTop="60dp"**>  
 <**TextView  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:padding="10dp"  
 android:text="菜单"  
 android:textColor="@android:color/black"  
 android:textSize="16sp"** />  
 <**View  
 android:layout\_width="match\_parent"  
 android:layout\_height="1dp"  
 android:layout\_marginLeft="10dp"  
 android:layout\_marginRight="10dp"  
 android:background="@color/light\_gray"** />  
 </**LinearLayout**>  
 </**LinearLayout**>  
 <**include layout="@layout/shop\_detail\_head"** />  
 <**RelativeLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:layout\_marginTop="220dp"**>

<**ListView  
 android:id="@+id/lv\_list"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:layout\_alignParentBottom="true"  
 android:layout\_marginBottom="50dp"** />  
 <**include layout="@layout/car\_list"** />  
 <**include layout="@layout/shop\_car"** />  
 </**RelativeLayout**>  
</**FrameLayout**>

### 3.2.5 购物车详细信息

shop\_car.xml

*<?***xml version="1.0" encoding="utf-8"***?>*<**RelativeLayout  
 xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"**>  
 <**FrameLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="65dp"  
 android:layout\_alignParentBottom="true"**>  
 <**RelativeLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="50dp"  
 android:layout\_marginTop="15dp"  
 android:background="@color/car\_gray\_color"  
 android:paddingLeft="60dp"**>  
 <**TextView  
 android:id="@+id/tv\_money"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="25dp"  
 android:layout\_marginLeft="4dp"  
 android:layout\_marginTop="4dp"  
 android:gravity="center"  
 android:text="未选购商品"  
 android:textColor="@color/light\_gray"  
 android:textSize="16sp"** />

<**TextView  
 android:id="@+id/tv\_distribution\_cost"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_below="@id/tv\_money"  
 android:layout\_marginLeft="4dp"  
 android:textColor="@android:color/white"  
 android:textSize="12sp"  
 android:textStyle="bold"  
 android:visibility="gone"** />  
 <**TextView  
 android:id="@+id/tv\_settle\_accounts"  
 android:layout\_width="100dp"  
 android:layout\_height="match\_parent"  
 android:layout\_alignParentRight="true"  
 android:layout\_centerVertical="true"  
 android:background="@color/account\_color"  
 android:gravity="center"  
 android:text="去结算"  
 android:textColor="@android:color/white"  
 android:textSize="16sp"  
 android:visibility="gone"** />  
 <**TextView  
 android:id="@+id/tv\_not\_enough"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="match\_parent"  
 android:layout\_alignParentRight="true"  
 android:background="@color/account\_gray\_color"  
 android:gravity="center"  
 android:padding="8dp"  
 android:textColor="@color/light\_gray"  
 android:textSize="16sp"** />  
 </**RelativeLayout**>  
 <**include layout="@layout/car"** />  
 </**FrameLayout**>  
</**RelativeLayout**>

3.2.6 订单界面

order\_head.xml

*<?***xml version="1.0" encoding="utf-8"***?>*<**LinearLayout  
 xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:background="@color/type\_gray"  
 android:orientation="vertical"**>  
 <**include layout="@layout/title\_bar"** />  
 <**LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="50dp"  
 android:layout\_marginTop="15dp"  
 android:background="@android:color/white"**>  
 <**TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginLeft="10dp"  
 android:text="收货地址："  
 android:textColor="@color/color\_gray"  
 android:textSize="16sp"** />  
 <**EditText  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:background="@android:color/white"  
 android:textColor="@android:color/black"  
 android:textSize="16sp"** />  
 </**LinearLayout**>  
 <**ListView  
 android:id="@+id/lv\_order"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="10dp"  
 android:background="@android:color/white"** />  
 <**View  
 android:layout\_width="match\_parent"  
 android:layout\_height="1dp"  
 android:background="@color/type\_gray"** />  
 <**RelativeLayout  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"**

**android:background="@android:color/white"**>  
 <**TextView  
 android:id="@+id/tv\_cost"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_alignParentRight="true"  
 android:padding="10dp"  
 android:textColor="@color/price\_red"  
 android:textSize="16sp"  
 android:textStyle="bold"** />  
 <**TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="2dp"  
 android:layout\_toLeftOf="@id/tv\_cost"  
 android:padding="10dp"  
 android:text="小计"  
 android:textColor="@android:color/black"  
 android:textSize="14sp"** />  
 <**TextView  
 android:id="@+id/tv\_distribution\_cost"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_alignParentRight="true"  
 android:layout\_below="@id/tv\_cost"  
 android:padding="10dp"  
 android:textColor="@android:color/black"  
 android:textSize="14sp"  
 android:textStyle="bold"** />  
 <**TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_below="@id/tv\_cost"  
 android:layout\_toLeftOf="@id/tv\_distribution\_cost"  
 android:padding="10dp"  
 android:text="配送费"  
 android:textColor="@android:color/black"  
 android:textSize="14sp"** />  
 </**RelativeLayout**>  
</**LinearLayout**>

# 第4章 系统实现（部分主要代码）

## 4.1 连接服务器

|  |
| --- |
| **package** cn.itcast.order.utils; **public class** Constant  {  **public static final** String ***WEB\_SITE*** = **"http://192.168.43.135:8080/order"**;*//内网接口* **public static final** String ***REQUEST\_SHOP\_URL*** = **"/shop\_list\_data.json"**; *//店铺列表接口*  } |

Constant.Java

## 4.2 解析服务器Json数据

JsonParse.Java

|  |
| --- |
| **package** cn.itcast.order.utils;  **import** com.google.gson.Gson; **import** com.google.gson.JsonArray; **import** com.google.gson.reflect.TypeToken; **import** org.json.JSONArray; **import** org.json.JSONObject; **import** java.lang.reflect.Type; **import** java.util.List; **import** cn.itcast.order.bean.ShopBean;  **public class** JsonParse {  **private static** JsonParse *instance*;  **private** JsonParse() {  }  **public static** JsonParse getInstance() {  **if** (*instance* == **null**) {  *instance* = **new** JsonParse();  }  **return** *instance*;  }  **public** List<ShopBean> getShopList(String json) {  Gson gson = **new** Gson(); *// 使用gson库解析JSON数据  // 创建一个TypeToken的匿名子类对象，并调用对象的getType()方法* Type listType = **new** TypeToken<List<ShopBean>>() {  }.getType();  *// 把获取到的信息集合存到shopList中* List<ShopBean> shopList = gson.fromJson(json, listType);  **return** shopList;  } } |

## 4.3 获取服务器数据

ShopActivity.Java

|  |
| --- |
| **package** cn.itcast.order.activity; **import** android.os.Bundle; **import** android.os.Handler; **import** android.os.Message; **import** android.support.v7.app.AppCompatActivity; **import** android.util.Log; **import** android.view.KeyEvent; **import** android.view.View; **import** android.widget.RelativeLayout; **import** android.widget.TextView; **import** android.widget.Toast; **import** java.io.IOException; **import** java.util.List; **import** cn.itcast.order.R; **import** cn.itcast.order.adapter.ShopAdapter; **import** cn.itcast.order.bean.ShopBean; **import** cn.itcast.order.utils.Constant; **import** cn.itcast.order.utils.JsonParse; **import** cn.itcast.order.views.ShopListView; **import** okhttp3.Call; **import** okhttp3.Callback; **import** okhttp3.OkHttpClient; **import** okhttp3.Request; **import** okhttp3.Response; **public class** ShopActivity **extends** AppCompatActivity {  **private** TextView **tv\_back**,**tv\_title**; *//返回键与标题控件* **private** ShopListView **slv\_list**; *//列表控件* **private** ShopAdapter **adapter**; *//列表的适配器* **public static final int *MSG\_SHOP\_OK*** = 1; *//获取数据* **private** MHandler **mHandler**;  **private** RelativeLayout **rl\_title\_bar**;  @Override  **protected void** onCreate(Bundle savedInstanceState) {  **super**.onCreate(savedInstanceState);  setContentView(R.layout.***activity\_shop***);  **mHandler**=**new** MHandler();  initData();  init();  }  */\*\*  \* 初始化界面控件  \*/* **private void** init(){  **tv\_back** = (TextView) findViewById(R.id.***tv\_back***);  **tv\_title** = (TextView) findViewById(R.id.***tv\_title***);  **tv\_title**.setText(**"店铺"**);  **rl\_title\_bar** = (RelativeLayout) findViewById(R.id.***title\_bar***);  **rl\_title\_bar**.setBackgroundColor(getResources().getColor(R.color.***blue\_color***));  **tv\_back**.setVisibility(View.***GONE***);  **slv\_list**= (ShopListView) findViewById(R.id.***slv\_list***);  **adapter**=**new** ShopAdapter(**this**);  **slv\_list**.setAdapter(**adapter**);  }  **private void** initData() {  OkHttpClient okHttpClient = **new** OkHttpClient();  Request request = **new** Request.Builder().url(Constant.***WEB\_SITE*** +  Constant.***REQUEST\_SHOP\_URL***).build();  Call call = okHttpClient.newCall(request);  Log.*i*(**"ShopActivity"**,call.toString());  *// 开启异步线程访问网络* call.enqueue(**new** Callback() {  @Override  **public void** onResponse(Call call, Response response) **throws** IOException {  String res = response.body().string(); *//获取店铺数据* Message msg = **new** Message();  msg.**what** = ***MSG\_SHOP\_OK***;  msg.**obj** = res;  **mHandler**.sendMessage(msg);  }  @Override  **public void** onFailure(Call call, IOException e) {  Log.*i*(**"ShopActivity"**,e.getMessage());  }  });  }  */\*\*  \* 事件捕获  \*/* **class** MHandler **extends** Handler {  @Override  **public void** dispatchMessage(Message msg) {  **super**.dispatchMessage(msg);  **switch** (msg.**what**) {  **case *MSG\_SHOP\_OK***:  **if** (msg.**obj** != **null**) {  String vlResult = (String) msg.**obj**;  *//解析获取的JSON数据* List<ShopBean> pythonList = JsonParse.*getInstance*().  getShopList(vlResult);  **adapter**.setData(pythonList);  }  **break**;  }  }  }  **protected long exitTime**;*//记录第一次点击时的时间* @Override  **public boolean** onKeyDown(**int** keyCode, KeyEvent event) {  **if** (keyCode == KeyEvent.***KEYCODE\_BACK*** && event.getAction() == KeyEvent.***ACTION\_DOWN***) {  **if** ((System.*currentTimeMillis*() - **exitTime**) > 2000) {  Toast.*makeText*(ShopActivity.**this**, **"再按一次退出订餐应用"**,  Toast.***LENGTH\_SHORT***).show();  **exitTime** = System.*currentTimeMillis*();  } **else** {  ShopActivity.**this**.finish();  System.*exit*(0);  }  **return true**;  }  **return super**.onKeyDown(keyCode, event);  } } |

## 4.4 实现菜单显示与购物车功能

ShopDetailActivity.Java

|  |
| --- |
| **package** cn.itcast.order.activity; **import** android.app.Dialog; **import** android.content.Intent; **import** android.graphics.Color; **import** android.os.Bundle; **import** android.os.Handler; **import** android.os.Message; **import** android.support.v7.app.AppCompatActivity; **import** android.view.MotionEvent; **import** android.view.View; **import** android.view.animation.Animation; **import** android.view.animation.AnimationUtils; **import** android.widget.ImageView; **import** android.widget.ListView; **import** android.widget.RelativeLayout; **import** android.widget.TextView; **import** com.bumptech.glide.Glide; **import** java.io.Serializable; **import** java.math.BigDecimal; **import** java.util.ArrayList; **import** java.util.Iterator; **import** java.util.List; **import** cn.itcast.order.R; **import** cn.itcast.order.adapter.CarAdapter; **import** cn.itcast.order.adapter.MenuAdapter; **import** cn.itcast.order.bean.FoodBean; **import** cn.itcast.order.bean.ShopBean;  **public class** ShopDetailActivity **extends** AppCompatActivity **implements** View.  OnClickListener{  **private** ShopBean **bean**;  **private** TextView **tv\_shop\_name**, **tv\_time**, **tv\_notice**, **tv\_title**, **tv\_back**,  **tv\_settle\_accounts**, **tv\_count**, **tv\_money**, **tv\_distribution\_cost**,  **tv\_not\_enough**, **tv\_clear**;  **private** ImageView **iv\_shop\_pic**, **iv\_shop\_car**;  **private** ListView **lv\_list**, **lv\_car**;  **public static final int *MSG\_COUNT\_OK*** = 1;*// 获取购物车中商品的数量* **private** MHandler **mHandler**;  **private int totalCount** = 0;  **private** BigDecimal **totalMoney**; *//购物车中菜品的总价格* **private** List<FoodBean> **carFoodList**; *//购物车中的菜品数据* **private** MenuAdapter **adapter**;  **private** CarAdapter **carAdapter**;  **private** RelativeLayout **rl\_car\_list**;  @Override  **protected void** onCreate(Bundle savedInstanceState) {  **super**.onCreate(savedInstanceState);  setContentView(R.layout.***activity\_shop\_detail***);  **bean** = (ShopBean) getIntent().getSerializableExtra(**"shop"**); *//获取店铺详情数据* **if** (**bean** == **null**) **return**;  **mHandler** = **new** MHandler();  **totalMoney** = **new** BigDecimal(0.0);*//初始化变量totalMoney* **carFoodList** = **new** ArrayList<>(); *//初始化集合carFoodList* initView(); *//初始化界面控件* initAdapter(); *//初始化adapter* setData(); *//设置界面数据* }  */\*\*  \* 初始化界面控件  \*/* **private void** initView() {  **tv\_back** = (TextView) findViewById(R.id.***tv\_back***);  **tv\_title** = (TextView) findViewById(R.id.***tv\_title***);  **tv\_title**.setText(**"店铺详情"**);  **tv\_shop\_name** = (TextView) findViewById(R.id.***tv\_shop\_name***);  **tv\_time** = (TextView) findViewById(R.id.***tv\_time***);  **tv\_notice** = (TextView) findViewById(R.id.***tv\_notice***);  **iv\_shop\_pic** = (ImageView) findViewById(R.id.***iv\_shop\_pic***);  **lv\_list** = (ListView) findViewById(R.id.***lv\_list***);  **tv\_settle\_accounts** = (TextView) findViewById(R.id.***tv\_settle\_accounts***);  **tv\_distribution\_cost** = (TextView) findViewById(R.id.***tv\_distribution\_cost***);  **tv\_count** = (TextView) findViewById(R.id.***tv\_count***);  **iv\_shop\_car** = (ImageView) findViewById(R.id.***iv\_shop\_car***);  **tv\_money** = (TextView) findViewById(R.id.***tv\_money***);  **tv\_not\_enough** = (TextView) findViewById(R.id.***tv\_not\_enough***);  **tv\_clear** = (TextView) findViewById(R.id.***tv\_clear***);  **lv\_car** = (ListView) findViewById(R.id.***lv\_car***);  **rl\_car\_list** = (RelativeLayout) findViewById(R.id.***rl\_car\_list***);  *//点击购物车列表界面外的其他部分会隐藏购物车列表界面* **rl\_car\_list**.setOnTouchListener(**new** View.OnTouchListener() {  @Override  **public boolean** onTouch(View v, MotionEvent event) {  **if** (**rl\_car\_list**.getVisibility() == View.***VISIBLE***) {  **rl\_car\_list**.setVisibility(View.***GONE***);  }  **return false**;  }  });  *//设置返回键、去结算按钮、购物车图片、清空购物车按钮的点击监听事件* **tv\_back**.setOnClickListener(**this**);  **tv\_settle\_accounts**.setOnClickListener(**this**);  **iv\_shop\_car**.setOnClickListener(**this**);  **tv\_clear**.setOnClickListener(**this**);  }  */\*\*  \* 初始化adapter  \*/* **private void** initAdapter(){  **carAdapter** = **new** CarAdapter(**this**, **new** CarAdapter.OnSelectListener() {  @Override  **public void** onSelectAdd(**int** position, TextView tv\_food\_count, TextView  tv\_food\_price) {  *//添加菜品到购物车中* FoodBean bean = **carFoodList**.get(position); *//获取当前菜品对象* tv\_food\_count.setText(bean.getCount() + 1 + **""**); *//设置该菜品在购物车中的数量* BigDecimal count = BigDecimal.*valueOf*(bean.getCount() + 1);  tv\_food\_price.setText(**"￥"** + bean.getPrice().multiply(count));*//菜品总价格* bean.setCount(bean.getCount() + 1);*//将当前菜品在购物车中的数量设置给菜品对象* Iterator<FoodBean> iterator = **carFoodList**.iterator();  **while** (iterator.hasNext()) {*//遍历购物车中的数据* FoodBean food = iterator.next();  **if** (food.getFoodId() == bean.getFoodId()) {*//找到当前菜品* iterator.remove(); *//删除购物车中当前菜品的旧数据* }  }  **carFoodList**.add(position, bean); *//将当前菜品的最新数据添加到购物车数据集合中* **totalCount** = **totalCount** + 1; *//购物车中菜品的总数量+1  //购物车中菜品的总价格+当前菜品价格* **totalMoney** = **totalMoney**.add(bean.getPrice());  carDataMsg();*//将购物车中菜品的总数量和总价格通过Handler传递到主线程中* }  @Override  **public void** onSelectMis(**int** position, TextView tv\_food\_count, TextView  tv\_food\_price) {  FoodBean bean = **carFoodList**.get(position); *//获取当前菜品对象* tv\_food\_count.setText(bean.getCount() - 1 + **""**);*//设置当前菜品的数量* BigDecimal count = BigDecimal.*valueOf*(bean.getCount() - 1);  *//设置当前菜品总价格，菜品价格=菜品单价\*菜品数量* tv\_food\_price.setText(**"￥"** + bean.getPrice().multiply(count));  minusCarData(bean, position);*//删除购物车中的菜品* }  });  **adapter** = **new** MenuAdapter(**this**, **new** MenuAdapter.OnSelectListener() {  @Override  **public void** onSelectAddCar(**int** position) {  *//点击加入购物车按钮将菜添加到购物车中* FoodBean fb = **bean**.getFoodList().get(position);  fb.setCount(fb.getCount() + 1);  Iterator<FoodBean> iterator = **carFoodList**.iterator();  **while** (iterator.hasNext()) {  FoodBean food = iterator.next();  **if** (food.getFoodId() == fb.getFoodId()) {  iterator.remove();  }  }  **carFoodList**.add(fb);  **totalCount** = **totalCount** + 1;  **totalMoney** = **totalMoney**.add(fb.getPrice());  carDataMsg();  }  });  **lv\_list**.setAdapter(**adapter**);  }  @Override  **public void** onClick(View v) {  **switch** (v.getId()){  **case** R.id.***tv\_back***: *//返回按钮的点击事件* finish();  **break**;  **case** R.id.***tv\_settle\_accounts***: *//去结算按钮的点击事件  //跳转到订单界面* **if** (**totalCount** > 0) {  Intent intent = **new** Intent(ShopDetailActivity.**this**, OrderActivity.**class**);  intent.putExtra(**"carFoodList"**, (Serializable) **carFoodList**);  intent.putExtra(**"totalMoney"**, **totalMoney** + **""**);  intent.putExtra(**"distributionCost"**, **bean**.getDistributionCost() + **""**);  startActivity(intent);  }  **break**;  **case** R.id.***iv\_shop\_car***: *//购物车的点击事件* **if** (**totalCount** <= 0) **return**;  **if** (**rl\_car\_list** != **null**) {  **if** (**rl\_car\_list**.getVisibility() == View.***VISIBLE***) {  **rl\_car\_list**.setVisibility(View.***GONE***);  } **else** {  **rl\_car\_list**.setVisibility(View.***VISIBLE***);  *//创建一个从底部滑出的动画* Animation animation = AnimationUtils.*loadAnimation*(  ShopDetailActivity.**this**, R.anim.***slide\_bottom\_to\_top***);  **rl\_car\_list**.startAnimation(animation);*//将动画加载到购物车列表界面* }  }  **carAdapter**.setData(**carFoodList**);  **lv\_car**.setAdapter(**carAdapter**);  **break**;  **case** R.id.***tv\_clear***:*//清空按钮的点击事件* dialogClear(); *//弹出确认清空购物车的对话框* **break**;  }  }  */\*\*  \* 弹出清空购物车的对话框  \*/* **private void** dialogClear() {  *//创建一个对话框Dialog* **final** Dialog dialog = **new** Dialog(ShopDetailActivity.**this**, R.style.***Dialog\_Style***);  dialog.setContentView(R.layout.***dialog\_clear***); *//将布局文件加载到对话框中* dialog.show(); *//显示对话框* TextView tv\_clear = dialog.findViewById(R.id.***tv\_clear***); *//获取对话框清除按钮* TextView tv\_cancel = dialog.findViewById(R.id.***tv\_cancel***);*//获取对话框取消按钮* tv\_cancel.setOnClickListener(**new** View.OnClickListener() {  @Override  **public void** onClick(View view) {  dialog.dismiss();*//关闭对话框* }  });  tv\_clear.setOnClickListener(**new** View.OnClickListener() {  @Override  **public void** onClick(View view) {  **if** (**carFoodList** == **null**) **return**;  **for** (FoodBean bean : **carFoodList**) {  bean.setCount(0);*//设置购物车中所有菜品的数量为0* }  **carFoodList**.clear();*//清空购物车中的数据* **carAdapter**.notifyDataSetChanged(); *//更新界面* **totalCount** = 0; *//购物车中菜品的数量设置为0* **totalMoney** = BigDecimal.*valueOf*(0.0);*//总价格设置为0* carDataMsg(); *//通过Handler更新购物车中菜品的数量和总价格* dialog.dismiss(); *//关闭对话框* }  });  }  */\*\*  \* 将购物车中菜品的总数量和总价格通过Handler传递到主线程中  \*/* **private void** carDataMsg() {  Message msg = Message.*obtain*();  msg.**what** = ***MSG\_COUNT\_OK***;  Bundle bundle = **new** Bundle();*//创建一个Bundler对象  //将购物车中的菜品数量和价格放入Bundler对象中* bundle.putString(**"totalCount"**, **totalCount** + **""**);  bundle.putString(**"totalMoney"**, **totalMoney** + **""**);  msg.setData(bundle); *//将Bundler对象放入Message对象* **mHandler**.sendMessage(msg); *//将Message对象传递到MHandler类* }  */\*\*  \* 删除购物车中的数据  \*/* **private void** minusCarData(FoodBean bean, **int** position) {  **int** count = bean.getCount() - 1; *//将该菜品的数量减1* bean.setCount(count); *//将减后的数量设置到菜品对象中* Iterator<FoodBean> iterator = **carFoodList**.iterator();  **while** (iterator.hasNext()) { *//遍历购物车中的菜* FoodBean food = iterator.next();  **if** (food.getFoodId() == bean.getFoodId()) {*//找到购物车中当前菜的Id* iterator.remove(); *//删除存放的菜* }  }  *//如果当前菜品的数量减1之后大于0，则将当前菜品添加到购物车集合中* **if** (count > 0) **carFoodList**.add(position, bean);  **else carAdapter**.notifyDataSetChanged();  **totalCount** = **totalCount** - 1; *//购物车中菜品的数量减1  //购物车中的总价钱=总价钱-当前菜品的价格* **totalMoney** = **totalMoney**.subtract(bean.getPrice());  carDataMsg(); *//调用该方法更新购物车中的数据* }  */\*\*  \* 事件捕获  \*/* **class** MHandler **extends** Handler {  @Override  **public void** dispatchMessage(Message msg) {  **super**.dispatchMessage(msg);  **switch** (msg.**what**) {  **case *MSG\_COUNT\_OK***:  Bundle bundle = msg.getData();  String count = bundle.getString(**"totalCount"**, **""**);  String money = bundle.getString(**"totalMoney"**, **""**);  **if** (bundle != **null**) {  **if** (Integer.*parseInt*(count) > 0) {*//如果购物车中有菜品* **iv\_shop\_car**.setImageResource(R.drawable.***shop\_car***);  **tv\_count**.setVisibility(View.***VISIBLE***);  **tv\_distribution\_cost**.setVisibility(View.***VISIBLE***);  **tv\_money**.setTextColor(Color.*parseColor*(**"#ffffff"**));  **tv\_money**.getPaint().setFakeBoldText(**true**);*//加粗字体* **tv\_money**.setText(**"￥"** + money);*//设置购物车中菜品总价格* **tv\_count**.setText(count); *//设置购物车中菜品总数量* **tv\_distribution\_cost**.setText(**"另需配送费￥"** +  **bean**.getDistributionCost());  *//将变量money的类型转换为BigDecimal类型* BigDecimal bdm = **new** BigDecimal(money);  *//总价格money与起送价格对比* **int** result = bdm.compareTo(**bean**.getOfferPrice());  **if** (-1 == result) { *//总价格<起送价格* **tv\_settle\_accounts**.setVisibility(View.***GONE***);*//隐藏去结算按钮* **tv\_not\_enough**.setVisibility(View.***VISIBLE***); *//显示差价文本  //差价=起送价格-总价格* BigDecimal m = **bean**.getOfferPrice().subtract(bdm);  **tv\_not\_enough**.setText(**"还差￥"** + m + **"起送"**);  } **else** { *//总价格>=起送价格  //显示去结算按钮* **tv\_settle\_accounts**.setVisibility(View.***VISIBLE***);  **tv\_not\_enough**.setVisibility(View.***GONE***); *//隐藏差价文本* }  } **else** { *//如果购物车中没有菜品* **if** (**rl\_car\_list**.getVisibility() == View.***VISIBLE***) {  **rl\_car\_list**.setVisibility(View.***GONE***); *//隐藏购物车列表* } **else** {  **rl\_car\_list**.setVisibility(View.***VISIBLE***);*//显示购物车列表* }  **iv\_shop\_car**.setImageResource(R.drawable.***shop\_car\_empty***);  **tv\_settle\_accounts**.setVisibility(View.***GONE***);*//隐藏去结算按钮* **tv\_not\_enough**.setVisibility(View.***VISIBLE***); *//显示差价文本* **tv\_not\_enough**.setText(**"￥"** + **bean**.getOfferPrice() + **"起送"**);  **tv\_count**.setVisibility(View.***GONE***);*//隐藏购物中的菜品数量控件* **tv\_distribution\_cost**.setVisibility(View.***GONE***);*//隐藏配送费用* **tv\_money**.setTextColor(getResources().getColor(R.color.  ***light\_gray***));  **tv\_money**.setText(**"未选购商品"**);  }  }  **break**;  }  }  }  */\*\*  \* 设置界面数据  \*/* **private void** setData() {  **if** (**bean** == **null**) **return**;  **tv\_shop\_name**.setText(**bean**.getShopName()); *//设置店铺名称* **tv\_time**.setText(**bean**.getTime()); *//设置配送时间* **tv\_notice**.setText(**"公告："**+**bean**.getWelfare()); *//设置店铺公告* **tv\_not\_enough**.setText(**"￥"** + **bean**.getOfferPrice() + **"起送"**); *//设置起送价格* Glide.*with*(**this**)  .load(**bean**.getShopPic())  .error(R.mipmap.***ic\_launcher***)  .into(**iv\_shop\_pic**); *//设置店铺图片* **adapter**.setData(**bean**.getFoodList());*//将菜单列表数据传递到adapter中* } } |

## 4.5 实现订单显示与支付功能

OrderActivity.Java

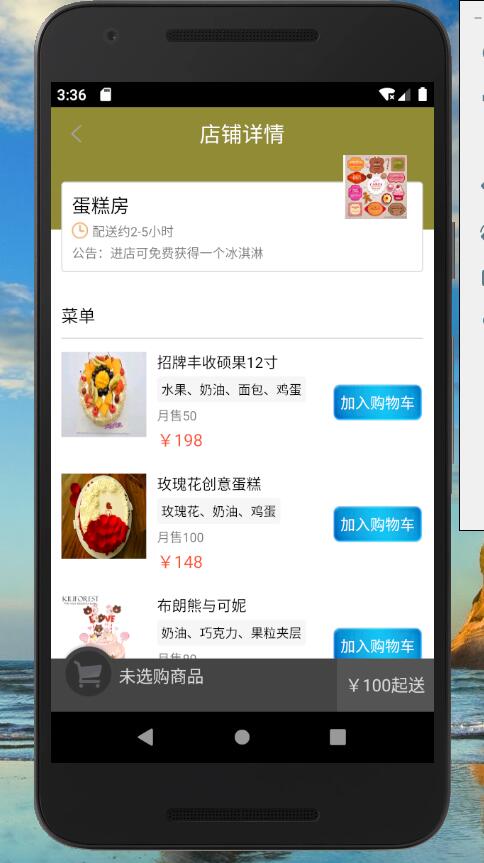
|  |
| --- |
| **package** cn.itcast.order.activity; **import** android.app.Dialog; **import** android.os.Bundle; **import** android.support.v7.app.AppCompatActivity; **import** android.view.View; **import** android.widget.ListView; **import** android.widget.RelativeLayout; **import** android.widget.TextView; **import** java.math.BigDecimal; **import** java.util.List; **import** cn.itcast.order.R; **import** cn.itcast.order.adapter.OrderAdapter; **import** cn.itcast.order.bean.FoodBean; **public class** OrderActivity **extends** AppCompatActivity {  **private** ListView **lv\_order**;  **private** OrderAdapter **adapter**;  **private** List<FoodBean> **carFoodList**;  **private** TextView **tv\_title**, **tv\_back**,**tv\_distribution\_cost**,**tv\_total\_cost**,  **tv\_cost**,**tv\_payment**;  **private** RelativeLayout **rl\_title\_bar**;  **private** BigDecimal **money**,**distributionCost**;  @Override  **protected void** onCreate(Bundle savedInstanceState) {  **super**.onCreate(savedInstanceState);  *//获取购物车中的数据* **carFoodList**= (List<FoodBean>) getIntent().getSerializableExtra(**"carFoodList"**);  *//获取购物车中菜的总价格* **money**=**new** BigDecimal(getIntent().getStringExtra(**"totalMoney"**));  *//获取店铺的配送费* **distributionCost**=**new** BigDecimal(getIntent().getStringExtra(  **"distributionCost"**));  setContentView(R.layout.***activity\_order***);   initView();  setData();  }  */\*\*  \* 初始化界面控件  \*/* **private void** initView(){  **tv\_title** = (TextView) findViewById(R.id.***tv\_title***);  **tv\_title**.setText(**"订单"**);  **rl\_title\_bar** = (RelativeLayout) findViewById(R.id.***title\_bar***);  **rl\_title\_bar**.setBackgroundColor(getResources().getColor(R.color.***blue\_color***));  **tv\_back** = (TextView) findViewById(R.id.***tv\_back***);  **lv\_order**= (ListView) findViewById(R.id.***lv\_order***);  **tv\_distribution\_cost** = (TextView) findViewById(R.id.***tv\_distribution\_cost***);  **tv\_total\_cost** = (TextView) findViewById(R.id.***tv\_total\_cost***);  **tv\_cost** = (TextView) findViewById(R.id.***tv\_cost***);  **tv\_payment** = (TextView) findViewById(R.id.***tv\_payment***);  *// 返回键的点击事件* **tv\_back**.setOnClickListener(**new** View.OnClickListener() {  @Override  **public void** onClick(View v) {  finish();  }  });  **tv\_payment**.setOnClickListener(**new** View.OnClickListener() {  @Override  **public void** onClick(View view) { *//“去支付”按钮的点击事件* Dialog dialog = **new** Dialog(OrderActivity.**this**, R.style.***Dialog\_Style***);  dialog.setContentView(R.layout.***qr\_code***);  dialog.show();  }  });  }  */\*\*  \* 设置界面数据  \*/* **private void** setData() {  **adapter**=**new** OrderAdapter(**this**);  **lv\_order**.setAdapter(**adapter**);  **adapter**.setData(**carFoodList**);  **tv\_cost**.setText(**"￥"**+**money**);  **tv\_distribution\_cost**.setText(**"￥"**+**distributionCost**);  **tv\_total\_cost**.setText(**"￥"**+(**money**.add(**distributionCost**)));  } } |

# 第5章 系统测试与部署

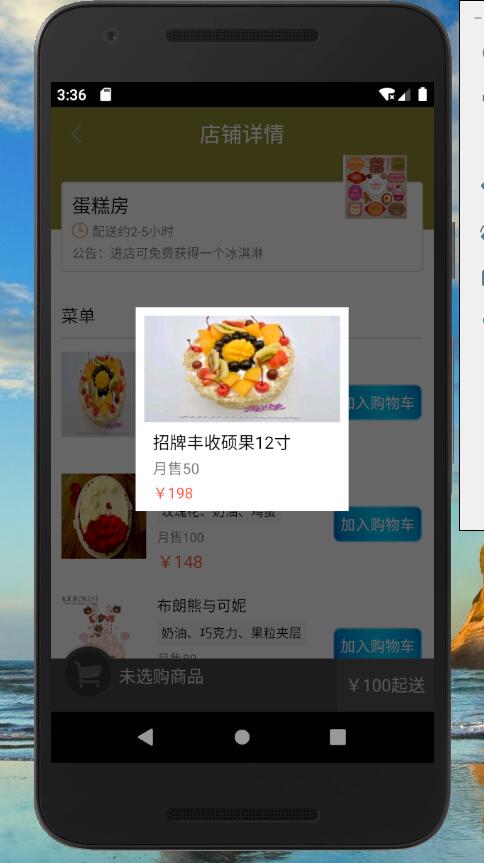
## 5.1 店铺首页



## 5.2 店铺详情与菜单界面



## 5.3 菜品详情界面

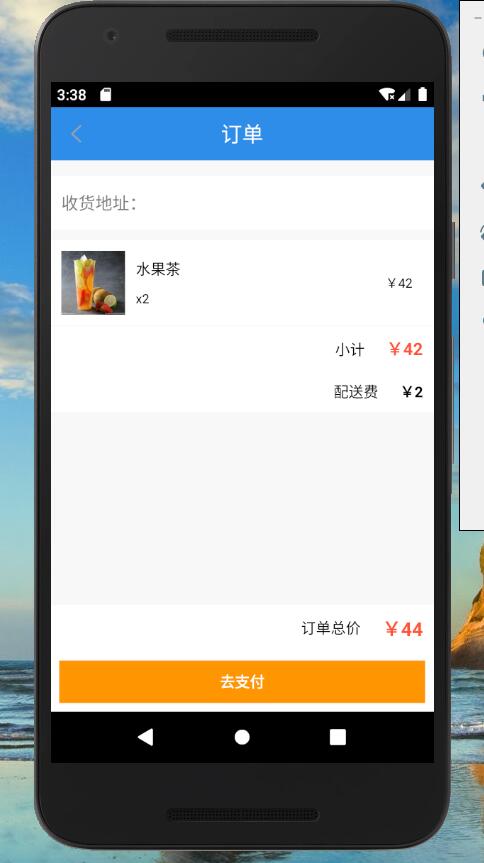


## 5.4 购物车界面与确认清空购物车弹窗





## 5.5 订单界面



## 5.6 支付二维码界面



## 5.7 打包过程

第一步：点击菜单栏中的Build，选择Generate Signed Bundle/APK

第二步：点击弹出来的对话中的APK选项，然后点击Next

第三步：在Module中选择需要打包的项目，然后在Key store path中选择签名文件，如果没有签名文件则点击Create new创建，在key store password中输入签名文件的密码，keyalias中输入签名文件的别名，以及Key password中输入别名的密码，点击Next

第四步：在Destination Folder中选择打包成apk时存放的位置，以及在Signature Versions中选中V1(ar Signature)即可，V2也选中可能会造成部分安卓系统无法按钮，点击Finish即可，等待软件打包完成，在之前选择存放apk的文件夹中即可找到打包完成的apk文件

# 总结

经过一周的努力，网上订餐系统基本完成。在本次实训设计中，通过对Android Studio和一些系统工具的运用，基本掌握了所运用的模型，加深了对已有知识的理解，不再局限于课堂和书本的简单案例。在设计时，需要充分考虑个人能力，切实结合自身实际情况定度系统的基本功能框架，进而细化各个模块的功能，确定各个功能模块，设计相应的界面实现相应的功能，最后结合各模块内容排查设计中的缺陷与遗漏，完成系统。

在实践中，我对于编程方面感觉很吃力，有一方面原因是老师课堂上讲的东西我并没有完全弄懂，更不要说实际编程了，另一方面原因是我们课下并没有看关于专业的书籍，也没有在课下把老师讲的再复习一遍，基础知识不牢固。在此次的实训设计中，我们充分意识到了平常学习的不足之处，也明白理论和实践是不一样的，实践不仅要求我们熟练掌握专业理论知识，更是要求我们把理论知识转化到应用中，知道怎么应用我们学过的知识。

在以后的学习中，我会注重理论与实践结合，补充编程中经常遇到的课外知识，不断充实自己的知识库，以求弥补我在软件制作方面的不足，提高我的实际操作能力。在此，感谢老师的耐心指导与帮助,谢谢！

# 参考文献

【1】著作：黑马程序员.Android移动开发基础.北京：人民邮电出版社.2017.

【2】著作：黑马程序员.Android移动应用基础教程（Android Studio）（第二版）.北京：中国铁道出版社有限公司.2019.

# 致谢

在系统实现过程中遇到诸多大大小小的问题，尤其体现在编程部分。在此非常感谢实训过程中，邓老师给我的帮助。老师给我详细讲述了实训设计的做法，耐心给我解决实践过程中遇到的难题，并针对我的问题给我们推荐了一些很好的书籍，对我帮助很大。