软件工程课程设计

“曲辰智慧点餐系统”需求报告

**队长：丁开锋**

**团队成员：季文昊、刘诗妮、刘卓霖**

目 录

[1.引言 1](#_Toc14525)

[1.1项目背景 1](#_Toc25259)

[1.2项目目标 1](#_Toc21212)

[1.3适用范围 1](#_Toc6282)

[1.4主要特点 2](#_Toc9261)

[2系统概述 2](#_Toc19695)

[2.1系统功能 2](#_Toc2714)

[2.1.1用户管理模块 2](#_Toc6387)

[2.1.2菜品管理模块 2](#_Toc2634)

[2.1.3用户点餐模块 2](#_Toc18893)

[2.1.4订单管理模块 3](#_Toc22172)

[2.2假设与依赖关系 3](#_Toc31838)

[2.2.1系统假设 3](#_Toc23546)

[2.2.2系统依赖关系 3](#_Toc27178)

[2.3系统交互示例 4](#_Toc6529)

[3.系统需求 5](#_Toc6905)

[3.1用户类型 5](#_Toc20611)

[3.2功能需求 6](#_Toc2240)

[3.2.1用户管理模块 6](#_Toc361)

[3.2.2菜品管理模块 6](#_Toc28539)

[3.2.3点餐模块 6](#_Toc10387)

[3.2.4订单管理模块 7](#_Toc4375)

[3.2.5优惠券系统 7](#_Toc787)

[3.3非功能性需求 7](#_Toc13748)

[3.3.1安全需求 7](#_Toc8835)

[3.3.2性能需求 8](#_Toc12977)

[4.外部接口需求 8](#_Toc4163)

[4.1软件接口 8](#_Toc22376)

[4.2硬件接口 9](#_Toc988)

[5.其他需求 10](#_Toc21637)

[5.1前端开发工具 10](#_Toc3431)

[5.2后端开发工具 10](#_Toc7005)

[5.3开发环境 10](#_Toc23579)

# 1.引言

## 1.1项目背景

随着餐饮行业的数字化转型，越来越多的餐厅开始采用信息化管理手段来提高服务质量和运营效率。传统的点餐方式存在诸多弊端，如手工记录、订单容易遗漏、等待时间长等问题。通过本系统的实施，不仅能解决传统点餐中的痛点，还能提高顾客的就餐体验，帮助餐厅提高效率，降低运营成本。

在此背景下，基于微信小程序构建的智能餐厅点餐系统将成为现代餐厅经营的有效工具，并能促进餐饮行业的智能化和信息化发展。

## **1.2项目目标**

本项目旨在开发一款**智能餐厅点餐系统**，旨在实现餐厅管理和顾客点餐流程的自动化与智能化。该系统将使餐厅能够高效地处理顾客订单和菜品信息，同时为顾客提供卓越的用户体验。为了满足现代餐饮业的需求，系统特别利用**微信小程序**作为用户入口，确保顾客和餐厅管理员都能轻松访问和使用。

**通过实施本系统，我们期望达到以下目标：**

**【1】**提高餐厅管理的自动化程度，减少人工操作，从而提升整体工作效率。

**【2】**创建一个用户友好的点餐平台，以增强顾客的用餐体验。

**【3】**支持订单管理、数据统计分析以及报表生成，帮助餐厅管理者做出更明智的决策并优化业务流程。

**【4】**确保系统的高可用性和可扩展性，使其能够在高并发环境下稳定运行。

这样的系统不仅能够提升餐厅的运营效率，还能通过提供便捷的服务来吸引和保留顾客，从而为餐厅带来长期的竞争优势。

## **1.3适用范围**

该智能餐厅点餐系统主要适用于餐饮行业，具体适用范围包括：

**【1】餐厅管理**：为餐厅管理员提供一个便捷的后台管理平台，支持对菜品、订单等信息的集中管理。管理员可以通过系统监控实时订单，更新菜单内容，管理顾客信息，并生成统计报表等。

**【2】顾客点餐**：为顾客提供一个方便的在线点餐平台，顾客可以通过微信小程序浏览餐厅的菜品、查看推荐菜品、提交订单以及查看订单状态。顾客在用餐结束后可以对菜品进行评价，提供反馈信息。

此外，系统的功能还可以根据餐厅的需求进行扩展和定制。例如，餐厅可以根据实际需求设置不同的权限管理，支持多语言界面，以及个性化菜品推荐等。

## **1.4主要特点**

**【1】用户友好界面：**界面简洁易用，顾客可以快速找到想要的菜品并下单，餐厅管理员可以方便地管理后台内容。

**【2】实时订单管理：**餐厅管理员可以实时查看订单情况，进行状态更新，而顾客也能随时查看自己的订单状态，提供更加透明的服务。

**【3】数据统计与分析：**系统可以自动统计顾客的点餐数据，为餐厅管理者提供有价值的报表，帮助餐厅做出更加精准的经营决策。

# **2系统概述**

## **2.1系统功能**

### **2.1.1用户管理模块**

**【1】用户注册与登录：**支持通过微信小程序注册与登录，验证用户身份，确保数据的安全性。用户可以通过微信账户绑定系统，便捷快捷地实现登录。

**【2】个人信息管理：**用户可以管理自己的个人信息，包括查看、修改昵称、密码、联系方式等。

**【3】用户收藏功能：**用户可以收藏自己喜欢的菜品，方便日后快速找到并下单。收藏的菜品将保存至用户的个人页面，便于随时浏览和点餐。

### **2.1.2菜品管理模块**

**【1】菜品信息管理：**管理员可以对菜品信息进行增、删、改、查操作，更新菜品的名称、价格、描述、图片、分类等信息，确保顾客看到最新的菜品信息。

**【2】菜品展示：**顾客可以查看所有菜品的详细信息，包括菜品图片、价格、评价等信息。

**【3】套餐定义功能（管理员功能）：**管理员可以定义套餐，设置套餐的内容、价格等信息。顾客可以选择套餐，享受更优惠的组合套餐，提升点餐体验。

### **2.1.3用户点餐模块**

**【1】在线点餐：**顾客可以选择菜品并加入购物车，提交订单。每道菜品都包含详细信息，顾客可以看到菜品的评价、价格、推荐量等。

**【2】菜品评价：**顾客可以在用餐后对菜品进行评价，评分和评论将展示给其他用户，帮助他们做出选择。

**【3】扫码点餐功能：**顾客可以扫描餐桌上的二维码，直接进入对应的菜单进行点餐，免去手动搜索的麻烦。每个二维码与餐桌号关联，确保点餐信息准确无误。

### **2.1.4订单管理模块**

**【1】订单状态管理：**顾客可以查看自己订单的实时状态，例如“待支付”、“已付款”、“待配送”、“已完成”等。管理员可以查看和更新所有订单的状态，确保餐厅能够及时处理订单。

**【2】订单修改与取消：**顾客可以在订单未处理前修改或取消订单。管理员也可以根据实际情况修改订单。

**【3】优惠券系统（顾客与管理员共同参与）：**管理员可以设置并发放优惠券，顾客可以在结算时使用优惠券享受折扣。优惠券的设置包括金额、有效期、使用条件等。

## **2.2假设与依赖关系**

### **2.2.1系统假设**

系统能够支持微信小程序用户注册、登录和扫码点餐功能。微信小程序会与系统后端进行数据交互。

用户能够在微信小程序访问系统，并通过手机设备进行扫码操作。

系统支持每个餐桌生成唯一二维码，顾客扫码后能够获取到对应餐桌的点餐菜单。

系统能够支持用户收藏功能，并能存储顾客的历史收藏记录。

### **2.2.2系统依赖关系**

系统依赖于一个后端数据库来存储用户数据、菜品数据、订单数据以及其他如优惠券、收藏菜品等数据。数据库需要具备高效的读写性能，支持大并发处理。

系统需要依赖于支付接口，顾客可以通过线上支付方式进行付款，管理员可以确认支付状态。

系统需要依赖于第三方的微信小程序SDK或类似平台，确保微信小程序的兼容性与稳定性。

二维码生成与识别：系统需要支持生成与解析二维码的功能，确保扫码点餐的顺利进行。

## **2.3系统交互示例**

**为了让系统设计更加清晰，可以通过以下几个交互示例来帮助理解：**

**【1】顾客点餐：**顾客进入系统后，可以在主界面浏览菜品（个性推荐），同时可以在个人中心查看收藏菜品，给店家建议，查看优惠券，顾客可以点击菜品可以查看菜品详细信息和所有评价和对该菜品做出评价。顾客可通过扫描餐桌二维码来直接进入点餐界面。顾客选择套餐时，系统会推荐套餐并显示优惠价格。完成点餐后，顾客可以使用优惠券进行支付，支付成功后，顾客可以在订单界面查看订单状态。

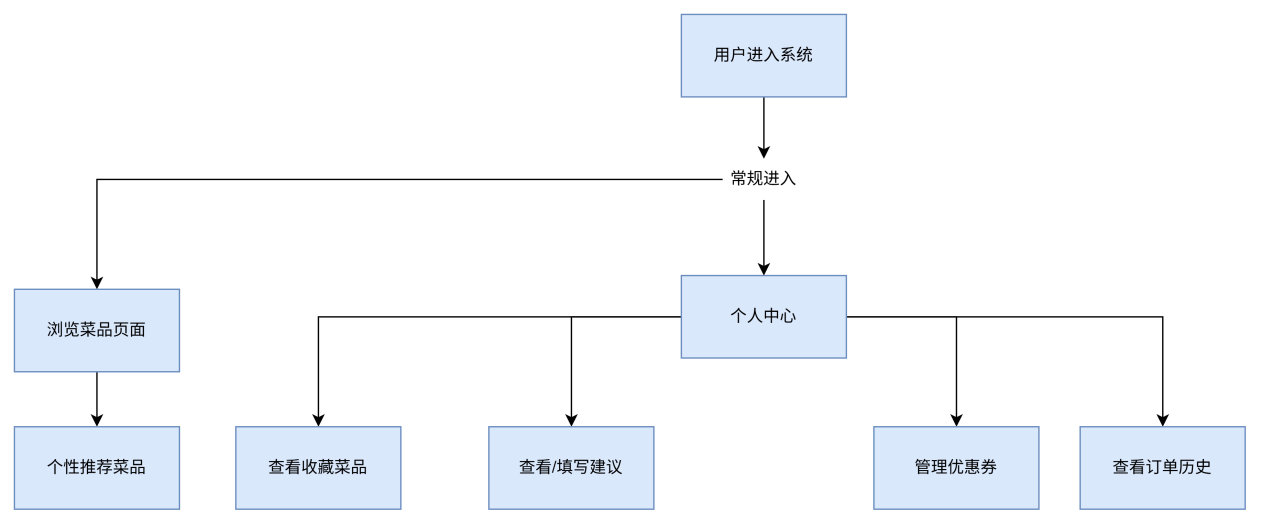


图1 顾客前台常规进入功能模块

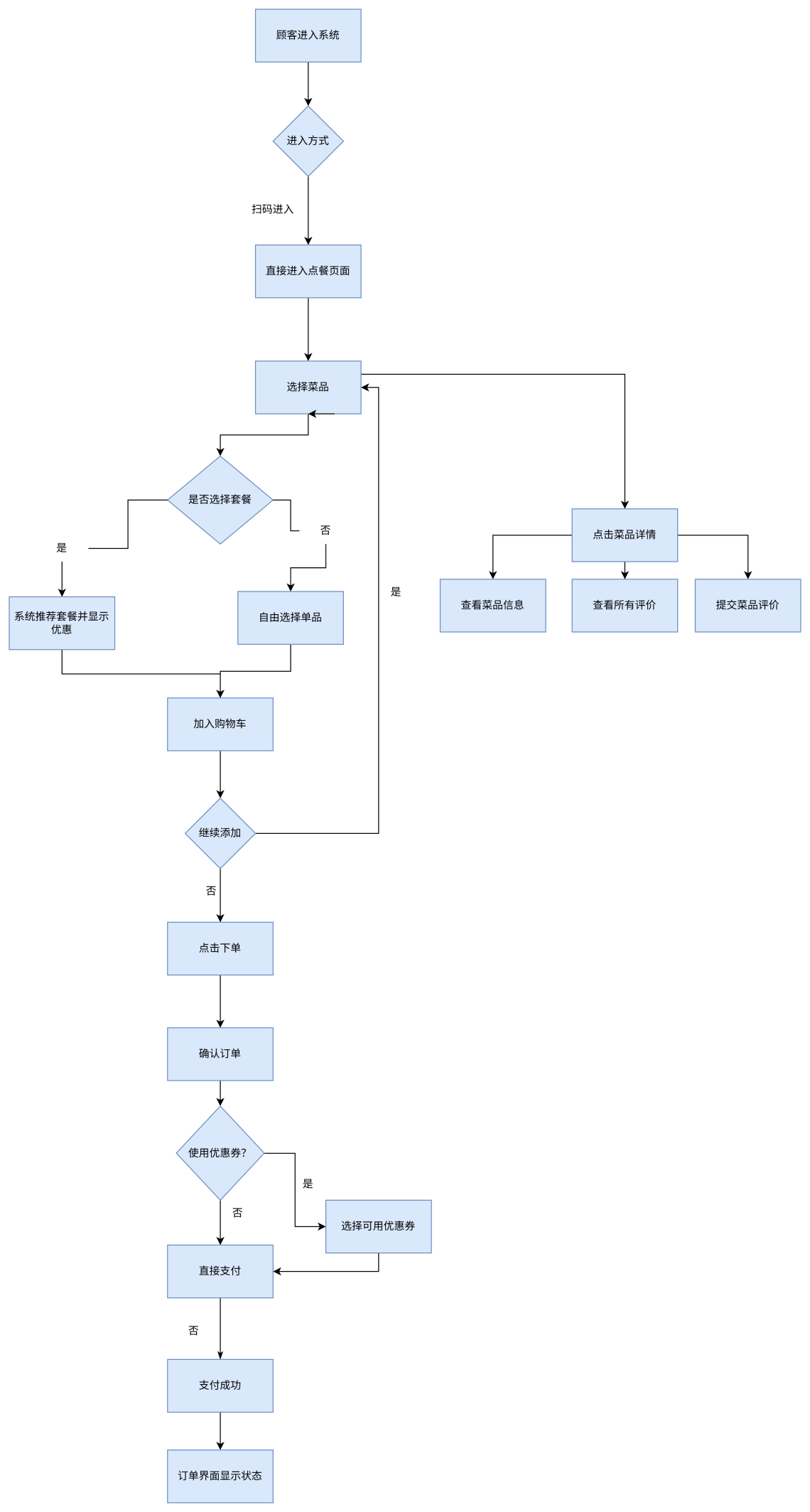


图2 顾客前台扫码进入功能模块

**【2】管理员管理后台：**管理员可以在后台管理界面增加、删除或更新菜品

信息。在系统中创建套餐时，管理员可以选择多个菜品组成一个套餐并设置套餐价格。管理员还可以通过后台发放优惠券，设定优惠券的使用规则和有效期。管理员可以在后台查看所有顾客信息，在线订单信息，历史订单信息，财务统计报告，处理订单，查看顾客评价。

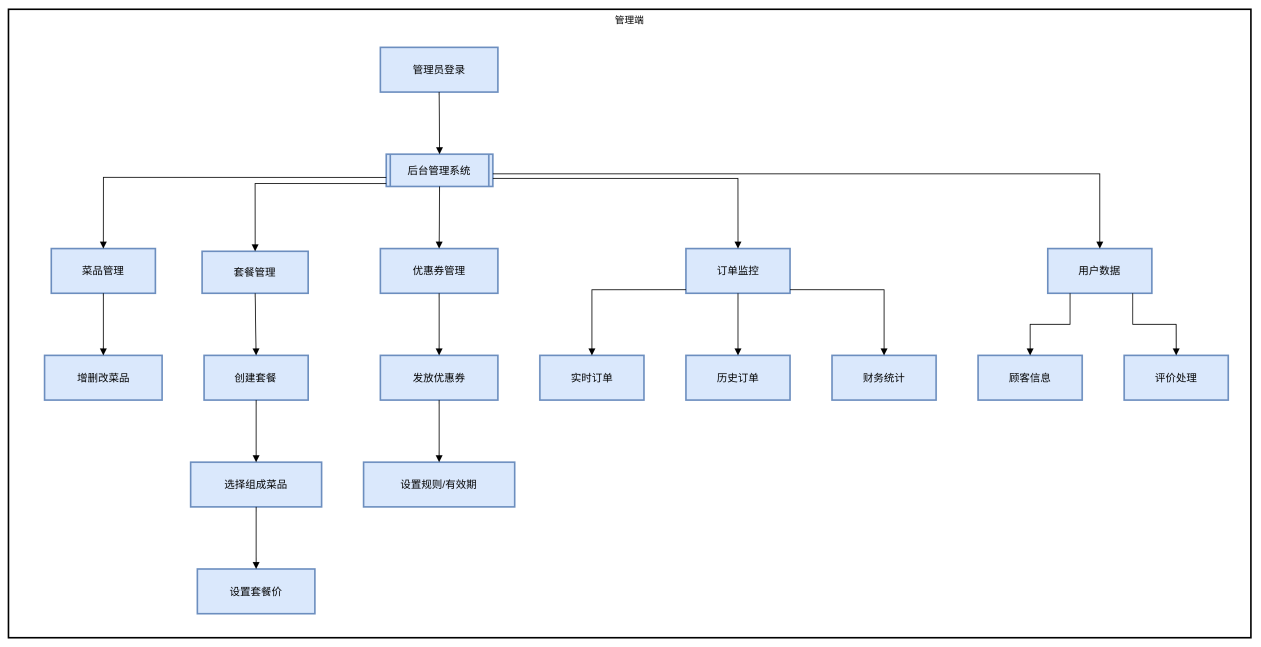


图2 管理员后台功能模块

# **3.系统需求**

## **3.1用户类型**

**【1】顾客：**

**浏览菜品与下单：**顾客可以浏览菜单。

**查看订单与状态：**顾客可以随时查看自己的订单历史和当前订单状态。

**收藏菜品：**顾客可以收藏自己喜爱的菜品，并快速下单。

**评价菜品：**顾客可以对菜品进行评价，发表评分和评论。

**扫码点餐：**顾客可以通过扫描桌子上的二维码，快速进入点餐界面并选择菜品。

**【2】管理员：**

**菜品信息管理：**管理员可以添加、删除或更新菜品信息。

**订单管理：**管理员可以查看和更新所有订单状态，处理顾客的订单需求。

**套餐管理：**管理员可以创建和编辑套餐内容及其价格，便于顾客选择套餐。

**优惠券发放与管理：**管理员可以设置不同种类的优惠券并发放给顾客，支持优惠券的使用与管理。

## **3.2功能需求**

### **3.2.1用户管理模块**

**【1】用户注册与登录：**

顾客可以通过微信账户或注册创建新账号登录。

注册时需要进行密码验证，确保用户信息的准确性和安全性。

系统支持通过微信授权登录，便于用户快速进入系统。

支持密码重置。

**【2】用户信息管理：**

用户可以查看和修改个人信息，如昵称、联系方式、收货地址等。

支持用户头像上传，个性化用户界面。

### **3.2.2菜品管理模块**

**【1】管理员管理菜品信息：**

管理员可以通过后台界面添加新菜品、修改现有菜品、删除不再提供的菜品。

每个菜品的信息包括：名称、价格、描述、图片、菜品类别（如主菜、配菜、饮品等）。

管理员可以对菜品设置推荐标识，以便在菜品展示中进行优先展示。

**【2】菜品展示：**

顾客可以浏览所有菜品，查看详细信息（名称、价格、图片、介绍、评价等）。

支持根据菜品类别、价格、评价等进行筛选。

### **3.2.3点餐模块**

**【1】在线点餐功能：**

顾客可以选择菜品，加入购物车并继续浏览，完成点餐。

顾客能够查看已选菜品的总价、数量、每项菜品的详细信息等。

支持套餐选择，顾客可以选择预设的菜品套餐，并享受优惠价格。

支持菜品数量修改和删除，购物车中的菜品可以随时进行编辑。

**【2】扫码点餐功能：**

顾客可以扫描餐桌二维码，自动跳转到对应餐桌的菜单页面，直接开始点餐。

**【3】菜品评价：**

顾客可以对已点菜品进行评价，给出星级评分并发表文字评论。

餐厅可以根据评价内容优化菜品质量，顾客评价会在菜品页面展示。

### **3.2.4订单管理模块**

**【1】顾客查看订单状态：**

顾客可以查看订单的实时状态，包括“待支付”、“已支付”、“准备中”、“配送中”、“已完成”等。

**【2】订单修改与取消：**

顾客可以在订单未发货前修改或取消订单。

系统会限制顾客修改的时间窗口，例如只能在支付前进行修改或取消。

**【3】管理员查看订单：**

管理员可以查看所有顾客的订单，检查订单的详细信息（如顾客信息、订单内容、支付状态等）。

管理员可以修改订单状态（如标记为“已发货”或“已完成”）。

**【4】订单历史：**

顾客可以查看自己的订单历史，包括每笔订单的状态、评价等。

### **3.2.5优惠券系统**

**【1】优惠券发放与管理：**

管理员可以设置不同种类的优惠券，定义优惠券的金额、有效期和使用条件。

顾客可以在结账时使用优惠券，享受折扣。

**【2】优惠券使用限制：**

系统应支持设置每个用户可用的优惠券数量限制，确保优惠券的使用不超过设定的条件。

## **3.3非功能性需求**

### 3.3.1安全需求

**【1】身份验证与授权：**

系统应提供用户身份验证，包括微信授权登录和账号密码登录两种方式。

顾客和管理员需要通过不同的权限验证来区分操作权限，管理员具有更高的权限，可以管理菜品和订单等。

所有的用户登录信息、订单信息等应通过安全协议（如HTTPS）进行加密传输。

**【2】密码保护与加密存储：**

用户的密码在系统中加密存储，如使用哈希加密算法（例如：bcrypt、SHA-256等）来保护用户密码，避免泄露。

系统不应存储明文密码，只保存加密后的密码信息。

**【3】数据保护：**

顾客的个人信息（如联系方式、地址等）应受到严格保护，避免泄露。

所有订单交易信息、优惠券信息、支付数据等必须加密存储，确保系统的安全性。

**【4】网络安全：**

用户与管理员之间的所有数据传输都应通过HTTPS协议加密，确保通信的安全性。

### **3.3.2性能需求**

**【1】系统响应时间：**

系统在接收用户请求后，所有操作（如菜品查询、下单、支付等）应在5秒以内完成响应。

页面加载时间不应超过3秒，以保证顾客流畅的浏览体验。

**【2】并发性能：**

系统应支持至少50名用户同时在线，在高并发情况下，仍能保持系统的稳定性和响应速度。

系统能够承受每日订单量的波动，保证餐厅在忙碌时段的正常运营。

**【3】数据存储：**

系统应采用数据库分区、缓存等技术，确保在高并发情况下，订单、用户信息、菜品数据等能够快速读取和写入。

# **4.外部接口需求**

## **4.1软件接口**

**【1】通信协议：**

小程序与后端API之间的通信必须通过HTTPS协议进行，确保所有数据的传输过程都经过加密，保障用户隐私和数据的安全性。

**【2】数据格式：**

系统使用JSON格式作为数据交换格式。所有请求和响应数据应以标准化的JSON格式传输，便于不同平台和系统进行解析与处理。

JSON格式应具备良好的可扩展性，以便在未来增加新的数据字段或功能时，确保与现有功能的兼容性。

**【3】API设计：**

后端API应遵循RESTful API设计原则，每个接口的路径应能清晰表达其资源和操作，例如：

POST /api/register：用户注册接口

POST /api/login：用户登录接口

GET /api/menu：获取菜品菜单接口

POST /api/order：提交订单接口

PUT /api/order/{order\_id}：修改订单状态接口

所有API接口需确保接口文档清晰，包含每个接口的请求方式（GET、POST、PUT、DELETE）、参数、返回值、状态码等详细说明。

**【4】接口验证：**

接口应对错误请求和无效请求进行友好的错误处理，返回规范化的错误信息（如：404 未找到、500 内部服务器错误等）。

## **4.2硬件接口**

**【1】服务器硬件：**

**CPU：**服务器需配备至少四核处理器。

**内存：**至少16GB RAM，以确保多用户并发情况下系统的响应速度。

**存储：**至少500GB的存储空间，用于保存菜品信息、用户数据、订单记录、支付信息等。可以使用分布式存储系统（如：分布式文件系统）来扩展存储能力，适应数据增长。

**网络带宽：**服务器需支持至少100Mbps的网络带宽，确保多个并发用户时系统仍能稳定运行。

**【2】数据库硬件要求：**

系统的数据库需要支持高并发写入和读取操作，因此需要高性能的数据库存储设备（如SSD硬盘）。

如果预期数据量较大，建议使用数据库分库分表策略来优化查询性能。

# **5.其他需求**

## **5.1前端开发工具**

**【1】微信小程序开发工具：**用于开发微信小程序的前端部分，提供开发、调试、测试和部署等一站式服务。

**【2】开发语言：**微信小程序支持使用JavaScript（JS）编写逻辑，WXML（微信标记语言）和WXSS（微信样式表）用于UI布局和样式设计。

**【3】开发框架：**可以考虑使用Taro或UniApp等跨平台框架进行开发，提升开发效率。

## **5.2后端开发工具**

**【1】开发语言：**采用Python进行后端开发，因其在Web开发中的高效性和广泛的社区支持。

**【2】Web框架：**使用Django框架构建后端服务，Django是一个高效且功能强大的Web框架，适用于快速开发和部署Web应用。

**【3】Django的内置功能，**如ORM（对象关系映射）、身份认证系统等，可以减少开发者的工作量，快速实现功能。可以使用**Django Rest Framework（DRF）**来开发RESTful API接口，简化API设计与数据处理。

**【4】数据库：**使用MySQL或PostgreSQL等关系型数据库系统，确保系统的数据管理高效稳定。

**【5】版本控制工具：**使用Git进行代码版本控制，确保多人协作开发时的代码管理和协作。

## **5.3开发环境**

本地开发环境：开发者使用Visual Studio Code作为主要的开发IDE，结合Python、Django和Git插件，提供便捷的开发体验。