食堂点单小程序项目计划书

# 锦食江安

第3小组

**修订表**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 编号 | 生成版本 | 修订人 | 修订章节与内容 | 修订日期 |
| 1 | V1.0 | ALL | 初步确定计划书内容 | 2025/2/26 |
| 2 | V2.0 | ALL | 完善修改计划书内容 | 2025/3/5 |

目录

[食堂点单小程序项目计划书 1](#_Toc20817)

[第3小组 1](#_Toc6352)

[1 引言 4](#_Toc11836)

[1.1 编写目的 4](#_Toc21681)

[1.2 背景 4](#_Toc28119)

[1.3 定义 4](#_Toc31425)

[1.4 项目内容 4](#_Toc4384)

[2 项目概述 4](#_Toc1054)

[2.1 项目目标 4](#_Toc132)

[2.2 产品目标与范围 4](#_Toc25630)

[2.3 应交付成果 5](#_Toc25629)

[2.3.1 需完成的软件 5](#_Toc1065)

[2.3.2 需提交用户的文档 5](#_Toc8861)

[2.3.3 须提交内部的文档 5](#_Toc28519)

[2.3.4 运营时提供的服务 5](#_Toc25795)

[2.4 项目开发环境 5](#_Toc13553)

[3 项目团队组织 5](#_Toc10578)

[3.1 人员分工 6](#_Toc31838)

[4 实施计划 6](#_Toc5599)

[4.1 风险评估及对策 6](#_Toc26595)

[4.2 总体进度计划 6](#_Toc21233)

[4.3 项目里程碑计划 6](#_Toc30824)

[5 预算 7](#_Toc30284)

[5.1 人员成本 7](#_Toc2872)

[5.2 设备成本 7](#_Toc21594)

[5.3 其它经费预算 7](#_Toc1589)

[第三方服务（支付接口、地图API）：2.0万元 7](#_Toc6668)

[参考文献 7](#_Toc11966)

**1. 引言**

* 1. **编写目的**

本计划书旨在明确“食堂点单小程序”项目的开发目标、功能范围、实施路径及资源分配，为团队提供清晰的开发指南。通过系统化的需求分析、技术方案设计和风险管理，确保项目高效推进，实现以下核心目标：

·指导开发：定义功能模块优先级与技术实现路径；

·协调资源：明确团队分工与预算分配，避免资源浪费；

·控制风险：提前预判技术难点与用户接受度问题，制定应对策略；

·统一共识：为开发团队、食堂管理方及用户提供一致的项目认知。

* 1. **背景**

传统高校食堂普遍存在三大痛点：

·效率低下：高峰期排队时间长，人工点餐易出错；

·信息不透明：人流量分布、菜品余量无法实时查询；

·反馈缺失：学生对菜品的评价缺乏有效收集渠道，食堂改进滞后。

以四川大学为例，江安校区日均就餐人次超2万，但学生常因排队错过课程，食堂窗口也存在“冷热不均”现象。本项目通过开发智能化点单小程序，打通“用户-食堂”双向数据链路，实现精准就餐决策与数字化管理升级，助力打造“智慧校园”标杆案例。

* 1. **定义**

|  |  |
| --- | --- |
| 食堂点单小程序 | 一款基于微信平台的轻量化应用，支持用户在线浏览食堂窗口、下单支付、查看实时人流量及评价菜品，旨在优化校园就餐流程 |
| 在线点单系统 | 用户通过小程序选择菜品、提交订单并完成支付的数字化流程，支持堂食自提与预约取餐 |
| 实时人流量数据 | 通过用户定位、订单密度或传感器采集的食堂区域人流信息，以可视化形式（如热力图）展示，辅助用户决策 |
| 菜品评价模块 | 用户对已购菜品进行评分（1-5星）及文字/图片反馈的功能模块，数据用于优化食堂运营 |
| 智能推荐系统 | 基于用户历史订单、评价数据及热门榜单，自动推荐新品或高匹配度菜品的算法模块 |
| 多食堂管理 | 支持同一小程序内切换不同校区或楼层食堂的功能，实现跨区域就餐信息整合 |
| AI客服 | 支持自然语言处理（NLP），与学生进行流畅对话，解答关于菜品、食堂窗口、营业时间等常见问题。  示例场景：学生输入“今天有什么推荐菜？” → AI根据历史订单和热门菜品推荐3-5个选项。 |

* 1. **项目内容**

开发一款集成食堂窗口浏览、在线点单、人流量查询、评价系统及新品推荐功能的微信小程序，支持多食堂管理，覆盖用户端与管理端。

用户端核心功能

智能导览：分校区、分楼层展示食堂窗口，支持菜品搜索与收藏；

无接触点单：在线下单（堂食/自提）、多方式支付（微信/校园卡）、取餐码通知；

实时决策：查看人流量热力图、菜品库存预警、窗口评分排行榜；

互动社区：发布带图评价、参与新品投票、领取优惠券。

AI客服：与学生进行流畅对话，解答关于菜品、食堂窗口、营业时间等常见问题。

管理端核心功能

数据驾驶舱：实时监控订单量、人流量峰值、菜品好评率等关键指标；

动态调控：调整窗口营业状态、更新菜品信息、推送促销活动；

反馈处理：自动归类用户评价，生成改进建议报告。

技术覆盖范围

前端：微信小程序（用户端）

后端：Springboot微服务架构+Java；

数据：MySQL存储业务数据，Redis缓存高并发请求（如人流量更新）。

1. **项目概述**

**2.1 项目目标**

1.全流程数字化就餐服务

实现用户从浏览食堂窗口、在线点单、支付到评价的闭环流程，替代传统人工点餐模式。

关键成果：订单处理效率提升50%，支付成功率≥99%。

2.动态人流量决策支持

基于实时数据生成食堂人流量热力图，并提供历史高峰时段分析，帮助用户避开拥堵。

关键成果：用户平均排队时间缩短至5分钟以内。

3.食堂管理智能化升级

搭建可视化后台管理系统，支持菜品上下架、订单处理、用户反馈分析及数据报表生成。

关键成果：食堂档口资源利用率提升30%，菜品迭代周期缩短至1周。

**2.2 产品目标与范围**

目标用户 ：在校师生、企业员工等高频食堂就餐人群。

|  |  |  |
| --- | --- | --- |
| 模块 | 用户端功能 | 管理端功能 |
| 基础服务 | - 分校区/楼层切换浏览窗口  - 菜品详情页（价格、成分、过敏原提示） | - 多食堂数据统一接入  - 权限分级管理 |
| 订单管理 | - 购物车多窗口合并下单  - 堂食预约（选择时间/座位）  - 订单状态实时通知 | - 订单异常监控（退单、超时）  - 财务报表导出 |
| 数据可视化 | - 实时人流量热力图（红/黄/绿三色标注）  - 窗口评分排行榜（按口味/性价比分类） | - 用户行为分析看板  - 菜品销量趋势预测 |
| 互动生态 | - 新品投票活动 | - 评价关键词提取  - 活动效果追踪 |

**2.3 应交付成果**

### 2.3.1 需完成的软件

微信小程序（用户端）：在线点单、实时查看、评价打分、新品推荐

食堂管理后台（Web端）：订单管理

### 2.3.2 需提交用户的文档

《用户操作手册》：

结构：图文分步指南（含扫码点餐、预约取餐、评价流程）、常见问题QA、客服联系方式。

交付形式：小程序内嵌PDF+纸质版放置食堂入口。

《隐私政策与服务协议》：

重点条款：用户数据加密标准（AES-256）、定位权限使用范围、评价内容审核规则。

### 2.3.3 须提交内部的文档

《数据库设计说明书》：

内容：ER图、表结构（如dishes表包含spicy\_level辣度字段）、索引优化方案。

《系统接口文档》

### 2.3.4 运营时提供的服务

运维支持：

7×24小时服务器监控（Zabbix）、月度性能报告（响应时间、故障率）。

用户反馈机制：

紧急问题（如支付失败）2小时内响应，功能建议48小时内回复并纳入迭代排期。

**2.4 项目开发环境**

前端：微信开发者工具

后端：Java+Springboot

数据库：MySQL + Redis（缓存）

1. **项目团队组织**

**3.1 人员分工**

|  |  |  |
| --- | --- | --- |
| **角色** | **职责** | **人员** |
| 项目经理 | 需求分析、进度把控、资源协调 | 刘宗毅 |
| 前端开发工程师 | 小程序界面开发、管理端UI实现 | 黄正阳 |
| 后端开发工程师 | API接口开发、数据库设计与优化 | 夏弋昕 |
| 测试工程师 | 功能测试、性能测试、安全测试 | 唐纳 |
| 运维工程师 | 服务器部署、后期运维支持 | 刘小瑜 |

**3.2项目管理**

进度：

质量：

成本：

1. **实施计划**

**4.1 风险评估及对策**

|  |  |  |
| --- | --- | --- |
| **风险类型** | **风险描述** | **应对措施** |
| 技术风险 | 人流量数据采集精度不足 | 采用用户手动上报+定时更新的混合模式 |
| 进度风险 | 前后端联调延迟 | 模块化开发，每日同步进度 |
| 用户接受度风险 | 初期用户活跃度低 | 联合食堂推出“首单立减”活动 |

**4.2 项目里程碑计划（甘特图）**

|  |  |  |
| --- | --- | --- |
| **里程碑** | **时间节点** | **交付物** |
| 需求确认 | 第1、2周 | 需求说明书、UI原型图 |
| 核心功能开发完成 | 第10周 | 可运行的小程序Demo、管理端基础功能 |
| 测试验收 | 第11周 | 测试报告、修复后的稳定版本 |
| 正式上线 | 第12周 | 小程序发布、用户手册交付 |

1. **预算**
   1. **人员成本**

|  |  |  |
| --- | --- | --- |
| **角色** | **人数** | **周** |
| 项目经理 | 1 | 13 |
| 开发与测试团队 | 4 | 13 |
| 合计 | 5 | 65 |

**5.2 设备成本**

服务器租用：2000元

AI api端口调用：200元

**参考文献**

1. 《微信小程序开发文档》

2. 《Sprngboot后端开发实战》

3. 《MySQL数据库设计与优化》