

PrecisePark | 瞬泊



瞬泊

智能停车场导航与管理系统

项目名称：	智能停车场导航与管理系统
项目类型：	软件工程课程设计
完成日期：	2025 年 9 月 24 日
团队成员：	梁炜琳 江子丹 罗文恺
成员学号：	2330502126 2330502119 2330502136
指导老师：	郭雷勇老师
学 院：	医药信息工程学院

目录

1	引言	2
1.1	编写目的	2
1.2	项目背景	2
1.3	研究范围与边界	2
2	总体描述	2
2.1	产品愿景	2
2.2	用户画像 (Personas)	3
2.3	约束与假设	3
3	具体需求	4
3.1	功能需求	4
3.1.1	用户故事列表 (User Stories)	4
3.1.2	数据流图 (DFD) (顶层图、0 层图、必要的 1 层图)	6
3.1.3	数据字典 (Data Dictionary) (对 DFD 中数据流、数据存储、处理逻辑 的详细定义)	10
	数据流定义 (Data Flows)10 数据存储定义 (Data Stores)12 处理逻辑定义 (Processes)14数 据元素定义 (Data Elements)16	
3.1.4	用例图 (Use Case Diagram)	18
3.1.5	活动图 (Activity Diagram) (对关键业务流程建模)	19
3.2	非功能性需求	21
3.2.1	性能需求 (响应时间、并发用户数)	21
3.2.2	安全性需求 (认证、授权、数据加密)	21
3.2.3	可用性需求 (用户界面友好性)	21
3.2.4	可靠性需求 (系统稳定性、故障恢复)	22
3.3	接口需求	22
3.3.1	用户接口	22
3.3.2	外部接口 (与其他系统或 AI 服务的交互)	23
4	运行环境规定	23

1 引言

1.1 编写目的

本文旨在系统阐述“智能停车场管理与预约系统”的研究动机、问题界定与研究路径，明确本课题的目标、范围、方法与预期贡献。通过对城市停车供需矛盾、停车场运营效率低与用户体验不佳等现实问题的剖析，本文提出一个以“预约—导航—入场—占位—离场—结算”闭环为核心的技术方案，并在学术与工程两个层面给出可验证的指标体系与评价方法。本文作为课程研究论文的开篇，将为后续的需求分析、总体设计、关键算法、系统实现与实验评估提供统一的术语、假设与方法论基础，保证全文论证的一致性与可复现性。

1.2 项目背景

随着城市经济的快速发展与居民生活节奏的加快，停车需求在时间与空间维度上呈现出显著的高峰拥挤与结构性失衡。传统“先到先停”的被动管理模式存在三方面结构性问题：其一，信息不对称导致车主难以及时获取空位，无法满足出行需求，“绕圈找位”进一步加剧城市道路拥堵；其二，市面上大部分停车系统在支付、对账与违规管理等环节相互割裂，缺乏与预约、入场、占位状态对接的统一信息接口。我们在现有基础上对相关功能进行整合与优化，使用户使用过程更加便捷高效。与此同时，地图导航、移动支付、机器视觉（车牌识别）、地锁/地磁等物联设备，以及基于 ETA（预计到达时间）的时空预测技术日趋成熟，为“预约驱动”的主动式停车管理提供了现实条件，并可在互联网平台的支撑下实现一体化联动。

1.3 研究范围与边界

本研究聚焦“单一城市—多停车场—多用户”的预约驱动场景，覆盖移动端（App/小程序）、云端服务（预约与调度）、边缘感知（车牌识别、地磁/地锁、入场道闸）与第三方地图/支付接口的端到端闭环。为保证在课程周期内的可实现性与可评估性，研究边界与假设如下：首先，优先采用通用硬件接口与开源软件栈，避免对专有设备的强依赖；其次，以典型商业停车场的入口道闸与车牌识别为基础，结合少量地锁或地磁传感进行占位校验；再次，导航采用第三方地图 ETA 并结合停车场内部的简化路径模型实现“场外到场—场内到位”的一体导航；数据隐私遵循“最小必要、端云分层、可审计与可撤回”的原则，不采集与停车无关的敏感数据，因能力有限，本项目主要以仿真测试，代码测试为主，无线下试点等操作。

2 总体描述

2.1 产品愿景

本软件旨在构建一个以“预约—导航—入场—占位—离场—结算”闭环为核心的智能停车场导航与管理系统，通过端到端的预约驱动机制与物联/感知融合，显著降低用户寻找停车位的时间、提升车位周转率并规范场内管理。系统将语音交互、第三方地图 ETA、车牌识别、

地锁/地磁与车位感应结合成一套可演示的线上仿真原型：对外提供移动端（Web/App）便捷查询与一键预约体验，对内提供云端的预约调度与风控、边缘侧的设备接入模拟以及运营端的监控与异常处置能力。设计上优先保证“端到端可演示、指标可复现”，在有限资源下采用开源技术栈与第三方服务（地图、语音、支付）快速实现 MVP，并保留向本地部署或云端+5G 物联网两类架构平滑演进的可能性，以便在后续从仿真验证过渡到真实场景部署。

2.2 用户画像 (Personas)

系统面向的首要用户群为具有移动互联网使用习惯的车主——以中青年为主、习惯用手机查询与语音交互，期待“到达即可有位、离场无感支付”的快捷体验；他们注重路线指引、预约可靠性与支付便捷性。其次是对数字交互敏感度较低的老年用户或语音识别使用弱势群体，这部分用户需要大字号、简化图形界面与一键求助/人工辅助通道以降低误操作概率。此外，停车场运营方与管理员是重要的企业用户，他们需要实时的预约监控、异常占用告警、策略配置与对账功能，以保障收益与运营秩序；对运营方而言，系统的可扩展性、数据安全与运维成本是首要决策指标。补充用户集合还包括新能源汽车车主（对“停车 + 充电”一体化有刚性需求）、场景中的支付服务方与第三方地图/语音供应商，整体使用场景假定终端兼容主流智能手机与浏览器，支持免密支付与常见 SDK 接入，从而覆盖绝大多数目标用户。

2.3 约束与假设

为保证在课程与演示周期内可行，本项目的范围、实现方式与运行环境基于若干明确约束与工程假设：首先，研究与演示限定在“单一城市—多停车场—多用户”的抽象场景，所有设备侧数据以模拟器或回放数据替代，用户数据以仿真数据为主，因而当前实现不依赖线下大规模试点；其次，技术实现优先采用通用、开源的软件栈（HTML/CSS/JavaScript 前端、Spring Boot 后端、MySQL 存储），并通过高德地图、科大讯飞语音与支付沙箱完成关键能力的集成，需考虑这些第三方 API 配额与降级方案（本地静态数据和手动输入）以保证演示鲁棒性。业务规则上设定若干硬性假设以简化一致性验证：预约车位最多锁定 10 分钟、锁定失效后 5 分钟可重新预约、车牌核验为发起地锁解锁的必要条件，且所有车位配备地锁/感应机制（地锁默认处于“放下/锁定”状态，满足预约并核验后升起），这些规则既是功能设计的核心也是自动化校验与仿真脚本的依据。并发与一致性方面假定后端使用事务与行锁处理竞态请求，同时接受在真实部署中可能需要引入更复杂的分布式锁与高可用策略；隐私与合规方面假设遵循“最小必要”原则对车牌等敏感信息进行脱敏与分层存储。以上约束与假设既限定了当前 MVP 的可交付边界，也为未来从仿真向生产环境迁移时需要重点验证的风险点（第三方依赖、网络稳定性、硬件同步、并发一致性与隐私合规）指明了方向。

3 具体需求

3.1 功能需求

3.1.1 用户故事列表 (User Stories)

- 角色：车主

表 1 数据流定义表

数据流名称	来源	去向	数据内容	说明
语音指令/手动操作	车主	语音交互处理	字符串：指令内容（如” 查询车位”” 预约车位”” 找车导航” 等）	车主向系统发起的操作指令，支持语音或手动输入
车位信息列表/导航路线/付费明细	智能停车系统（导航生成/车位信息管理等）	车主	结构体： - 车位信息列表（车位 ID、位置、状态） - 导航路线（起点、终点、路径点） - 付费明细（金额、时长、支付方式）	系统向车主反馈的核心信息集合
实时车位数据	停车场数据提供方	车位信息管理	结构体：停车场 ID、车位 ID、位置坐标、状态（空闲/占用）、更新时间戳	停车场实时同步给系统的车位状态数据
付费请求（模拟）	支付对接	支付系统	结构体：预约 ID、车主 ID、金额、支付方式（模拟）、订单时间	系统向支付系统发起的模拟付费请求
付费结果	支付系统	支付对接	结构体：预约 ID、支付状态（成功/失败）、交易号、支付时间	支付系统返回的付费结果
续表见下页				

续表 1 数据流定义				
数据流名称	来源	去向	数据内容	说明
数据查询/异常处理操作	系统管理员	管理员服务	字符串：操作类型（如” 查询预约记录”” 取消异常预约”” 查看告警日志” 等）	管理员对系统的管理操作指令
实时监控数据/异常告警日志	智能停车系统（各模块）	系统管理员	结构体： - 监控数据（车位占用率、系统负载） - 异常日志（时间、模块、异常类型、描述）	系统向管理员反馈的监控与异常信息
预约请求	语音交互处理	预约处理	结构体：车主 ID、车牌、目的地停车场 ID、预计到达时间	语音交互解析后传递给预约处理的预约请求数据
可用车位列表	车位信息管理	预约处理	结构体：停车场 ID、车位 ID 列表、位置坐标、剩余时长（若有）	车位信息管理模块提供给预约处理的可预约车位集合
预约车位坐标	预约处理	导航生成	结构体：车位 ID、停车场 ID、位置坐标（x,y）	预约成功后传递给导航生成模块的车位位置信息
付费需求	预约处理	支付对接	结构体：预约 ID、车主 ID、金额、付费截止时间	预约处理模块触发支付对接的付费需求数据
异常取消指令	管理员服务	预约处理	结构体：预约 ID、取消原因、管理员操作 ID	管理员手动取消异常预约的指令
车位数据（读写）	车位信息管理	车位信息库	结构体：车位 ID、停车场 ID、位置、状态、更新时间	车位信息管理与车位信息库之间的读写数据
续表见下页				

续表 1 数据流定义				
数据流名称	来源	去向	数据内容	说明
预约数据（读写）	预约处理	预约记录库	结构体：预约 ID、车主 ID、车位 ID、停车场 ID、预约时间、锁定截止时间、支付状态	预约处理与预约记录库之间的读写数据

- 角色：系统管理员

表 2 系统管理员用户故事列表

用户故事 ID	用户故事描述	验收标准
US-006	作为管理员，我希望查看实时车位数据（各停车场使用率、预约记录），以便监控系统运行	1. 后台实时显示各停车场车位状态（空闲/占用/预约），数据延迟 10 秒； 2. 支持按日/周导出预约记录（含用户 ID、车牌、预约时间）； 3. 显示异常数据告警（如车位状态同步失败）
US-007	作为管理员，我希望处理异常预约（如长期锁定车位），以便释放车位资源	1. 后台标记”超时未使用””重复预约”的异常记录，支持手动取消预约； 2. 取消后 3 秒内更新车位状态为”空闲”； 3. 生成异常处理日志（操作人、时间、原因）

3.1.2 数据流图 (DFD) (顶层图、0 层图、必要的 1 层图)

- 顶层图

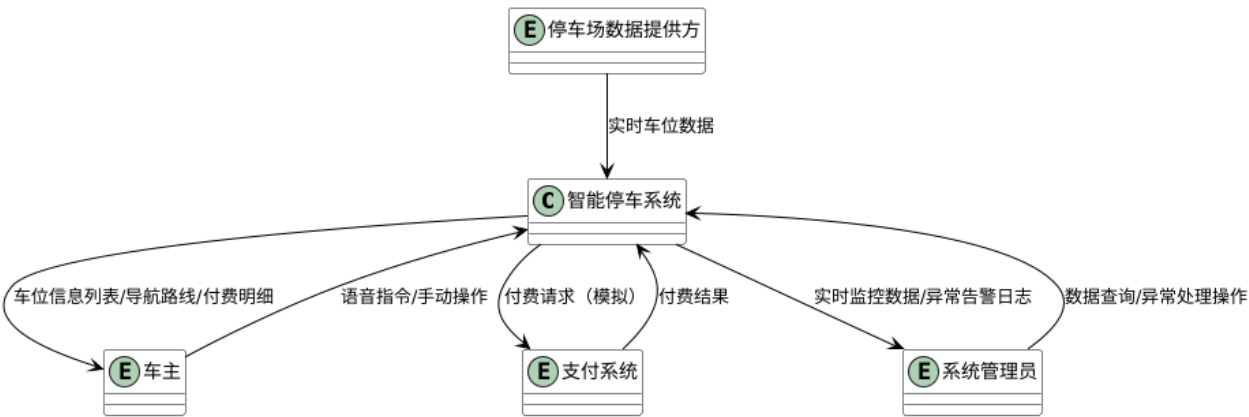


图 1 数据流图 - 顶层图

顶层图展示了智能停车系统与外部实体（车主、停车场设备、支付系统等）之间的数据交互关系。

- 0 层图

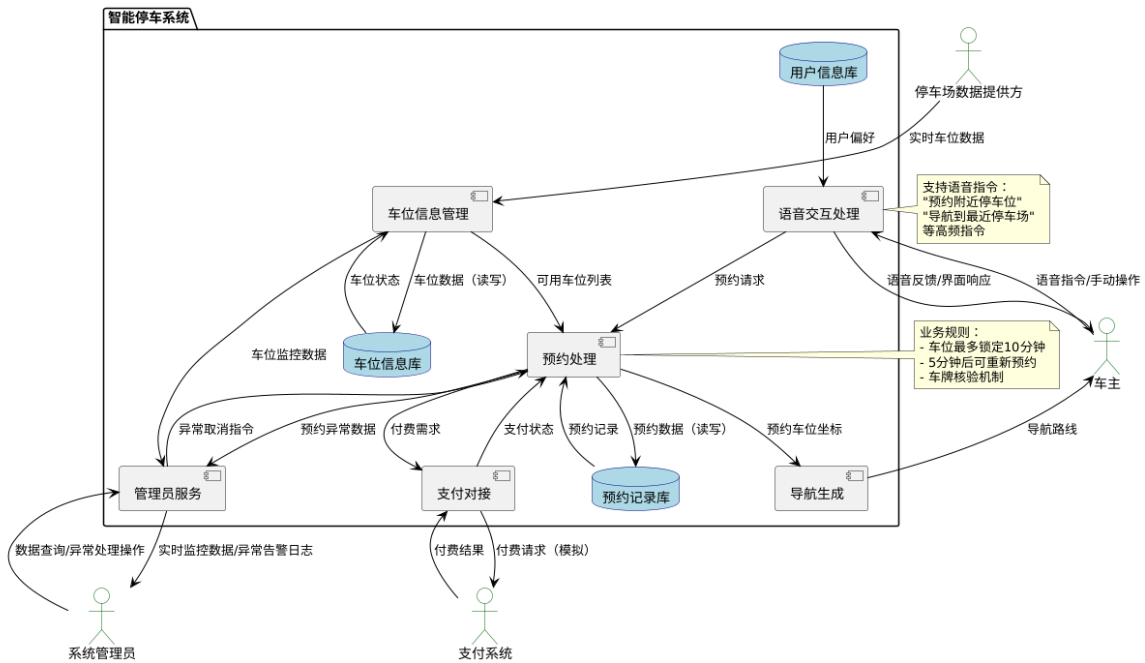


图 2 数据流图 - 0 层图

0 层图细化了系统的核心处理模块，包括车位查询、预约管理、导航引导和支付处理等功能。

- 1 层图

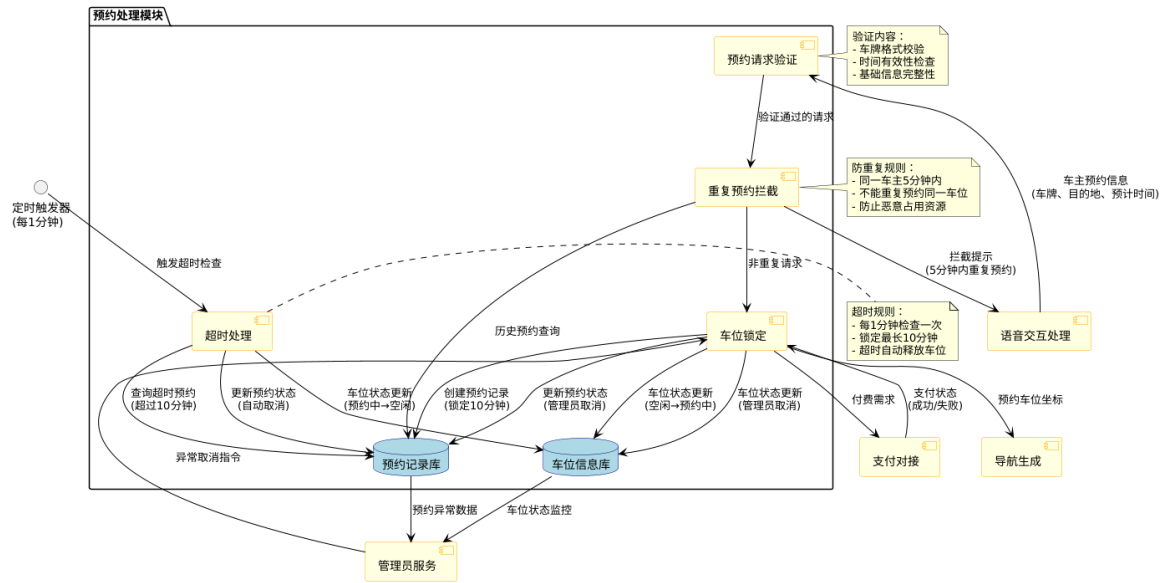


图 3 数据流图 - 1 层图

1 层图进一步详细描述了各功能模块的内部数据流和处理逻辑，特别是预约管理和支付处理的关键流程。

3.1.3 数据字典 (Data Dictionary) (对 DFD 中数据流、数据存储、处理逻辑的详细定义)

数据流定义 (Data Flows)

- 外部实体与系统间数据流

表 3 外部实体与系统间数据流

数据流名称	来源	去向	数据内容	数据类型	说明
语音指令/ 手动操作	车主	语音交互 处理	{指令类型: string, 指令内容: string, 指令时间: datetime, 用户 ID: string}	结构体	车 主 发 起 的 操 作 指 令, 支 持 语 音 或 手 动 输 入
车 位 信 息 列 表/ 导 航 路 线/ 付 费 明 细	系 统 各 模 块	车 主	{车位列表: array, 导航路线: object, 付费明细: object, 响应状态: string}	结构体	系 统 向 车 主 反 馈 的 核 心 信 息 集 合
实 时 车 位 数 据 提 供 方	停 车 场 数 据 提 供 方	车 位 信 息 管 理	{停车场 ID: string, 车位 ID: string, 位置坐标: object, 状态: enum, 更 新 时 间: datetime}	结构体	停 车 场 实 时 同 步 的 车 位 状 态 数 据
付 费 请 求 (模 拟)	支 付 对 接	支 付 系 统	{预约 ID: string, 车主 ID: string, 金额: decimal, 支 付 方 式: string, 订 单 时 间: datetime}	结构体	系 统 向 支 付 系 统 发 起 的 模 拟 付 费 请 求
付 费 结 果	支 付 系 统	支 付 对 接	{预约 ID: string, 支付状态: enum, 交 易 号: string, 支 付 时 间: datetime}	结构体	支 付 系 统 返 回 的 付 费 结 果
					续表见下页

续表 3 外部实体与系统间数据流

数据流名称	来源	去向	数据内容	数据类型	说明
数据查询/ 异常处理操作	系统 管理员	管理员 服务	{监控数据: object, 查询条件: object, 管理员 ID: string, 操作时间: datetime}	结构体	管理员对系 统的 管理指导
实时监控数据/ 异常告警日志	系统各 模块	系统 管理员	{监控数据: object, 异常日志: array, 系统状态: string, 生成时间: datetime}	结构体	系统向管理 员 反馈的监控 与 异常信息

- 系统内部数据流

表 4 系统内部数据流

数据流名称	来源	去向	数据内容	数据类型	说明
预约请求	语音 交互 处理	预约 处理	{车主 ID: string, 车牌号: string, 目的 地停车场 ID: string, 预计到达时间: datetime, 请求时间: datetime}	结构体	语音交互解 析 后的预约请 求 数据
可用车位列表	车位 信息 管理	预约 处理	{停车场 ID: string, 可用车位: array, 总数: integer, 查询时间: datetime}	结构体	可预约车位 集 合信息
预约车位坐标	预约 处理	导航 生成	{车位 ID: string, 停车场 ID: string, 位置坐标: object, 预约 ID: string}	结构体	预约成功的 车 位位置信息

续表见下页

续表 4 系统内部数据流					
数据流名称	来源	去向	数据内容	数据类型	说明
付费需求	预约处理	支付对接	{预约 ID: string, 车主 ID: string, 金额: decimal, 付费截止时间: datetime, 支付类型: string}	结构体	触发支付对接的付费需求数据
异常取消指令	管理员服务	预约处理	{预约 ID: string, 取消原因: string, 管理员操作 ID: string, 取消时间: datetime}	结构体	管理员手动取消异常预约的指令
车位数据 (读写)	车位信息管理	车位信息库	{操作类型: enum, 车位数据: object, 时间戳: datetime}	结构体	车位信息库的读写操作数据
预约数据 (读写)	预约处理	预约记录库	{操作类型: enum, 预约数据: object, 时间戳: datetime}	结构体	预约记录库的读写操作数据

数据存储定义 (Data Stores)

- 核心数据存储

表 5 核心数据存储 - 基本信息

存储名称	存储内容	主键	索引	说明
车位信息库	所有停车场基础信息	车位 ID	停车场 ID, 状态	核心数据存储, 支持车位查询、预约等核心功能
预约记录库	车位的预约及全生命周期	预约 ID	车位 ID, 锁定截止时间	记录预约全流程状态, 用于校验、超时处理等
用户信息库	车主的身份 (脱敏存储)	用户 ID	手机号, 车牌号	存储车主身份与关联信息, 用于个性化服务

表 6 核心数据存储 - 详细结构定义

存储名称	结构定义
车位信息库	车位 ID: string, 停车场 ID: string, 位置坐标: object, 状态: enum, 地锁状态: enum, 最后更新时间: datetime
预约记录库	预约 ID: string, 车主 ID: string, 车牌号: string, 车位 ID: string, 停车场 ID: string, 预约时间: datetime, 锁定开始时间: datetime, 锁定截止时间: datetime, 实际到达时间: datetime, 支付状态: enum, 预约状态: enum
用户信息库	用户 ID: string, 手机号: string, 车牌号: string, 支付方式: string, 用户偏好: object, 注册时间: datetime, 最后登录时间: datetime

- 辅助数据存储

表 7 辅助数据存储

存储名称	存储内容	结构定义	主键	说明
停车场信息库	停车场基础信息	{停车场 ID: string, 停车场名称: string, 位置坐标: object, 总车位数: integer, 收费标准: object, 营业时间: string}	停车场 ID	存储停车场基本属性信息
系统日志库	系统操作审计日志	{日志 ID: string, 操作类型: string, 操作内容: object, 操作人员: string, 操作时间: datetime, IP 地址: string}	日志 ID	记录所有关键操作用于审计

处理逻辑定义 (Processes)

- 主要处理逻辑

表 8 主要处理逻辑 - 流程信息

处理名称	输入数据流	输出数据流	性能要求
语音交互处理	语音指令/ 手动操作	预约请求, 车位查询请求等	p95 延迟 < 800ms
车位信息管理	实时车位数据, 车位数据 (读写)	可用车位列表, 车位数据 (读写)	支持 200+ 车位实时状态管理
预约处理	预约请求, 可用车位列表, 异常取消指令, 付费结果	预约车位坐标, 付费需求, 预约数据 (读写)	事务处理,行级锁控制

表 9 主要处理逻辑 - 功能规则

处理名称	处理描述	业务规则
语音交互处理	接收车主指令，解析为具体操作请求，传递给对应处理模块	<ol style="list-style-type: none"> 1. 支持自然语言识别 2. 固定指令识别率 > 95% 3. 响应时间 < 500ms
车位信息管理	接收实时车位数据，更新车位信息库；响应用户查询请求	<ol style="list-style-type: none"> 1. 数据延迟 < 10 秒 2. 状态更新原子性 3. 并发访问控制
预约处理	验证预约请求,锁定车位,管理预约全生命周期	<ol style="list-style-type: none"> 1. 车位锁定 10 分钟 2. 5 分钟内防重复预约 3. 车牌核验机制

- 预约处理子模块

表 10 预约处理子模块 - 流程信息

处理名称	输入数据流	输出数据流	处理描述
预约请求验证	车主预约信息	验证通过的预约请求/验证未通过反馈	校验车主输入的车牌格式、时间有效性等
车位锁定	验证通过的预约请求 可用车位列表	预约成功结果 预约数据 (写)	从可用车位列表中锁定目标车位，更新状态
超时处理	定时触发 预约记录库数据	车位状态更新 (空闲) 预约数据 (更新)	定时检查预约记录，对超时预约自动取消
重复预约拦截	验证未通过的请求 预约记录库数据	拦截提示/验证通过的请求	检查历史预约记录，防止恶意重复预约

表 11 预约处理子模块 - 规则算法

处理名称	业务规则	算法说明
预约请求验证	1. 车牌格式正则验证 2. 预计到达时间在 1 小时内 3. 必填字段完整性检查	使用正则表达式验证车牌格式
车位锁定	1. 选择最优可用车位 2. 更新状态为” 预约中” 3. 生成唯一预约 ID	基于距离和优先级的车位选择算法
超时处理	1. 每 1 分钟执行一次 2. 锁定超过 10 分钟自动取消 3. 释放车位资源	基于时间戳的定时任务调度
重复预约拦截	1. 同一车主 5 分钟内不能重复预约同一车位 2. 同一车牌号频率控制	基于车主 ID 和时间窗口的重复检查

数据元素定义 (Data Elements)

- 基本数据类型

表 12 基本数据类型

数据元素	数据类型	长度/格式	取值范围	说明
车位 ID	string	10 字符	PK + 数字序列	唯一标识一个车位
停车场 ID	string	8 字符	P + 数字序列	唯一标识一个停车场
预约 ID	string	16 字符	时间戳 + 随机数	唯一标识一次预约
用户 ID	string	12 字符	U + 数字序列	唯一标识一个用户
车牌号	string	7-8 字符	符合车牌号规则	车辆标识号码
续表见下页				

续表 12 基本数据类型				
数据元素	数据类型	长度/格式	取值范围	说明
位置坐标	object	{x decimal; y decimal}	经纬度坐标	车位或停车场的坐标位置
金额	decimal	10.2	0.00-999999.99	货币金额, 单位元

- 枚举类型定义

表 13 枚举类型定义

枚举名称	枚举值	说明
车位状态	空闲, 占用, 预约中	车位的当前使用状态
地锁状态	升起, 放下	地锁的物理状态
支付状态	未支付, 已支付, 已取消	预约的支付状态
预约状态	有效, 超时取消, 管理员取消	预约的业务状态
指令类型	查询车位, 预约车位, 找车导航, 取消预约	用户操作指令分类

3.1.4 用例图 (Use Case Diagram)

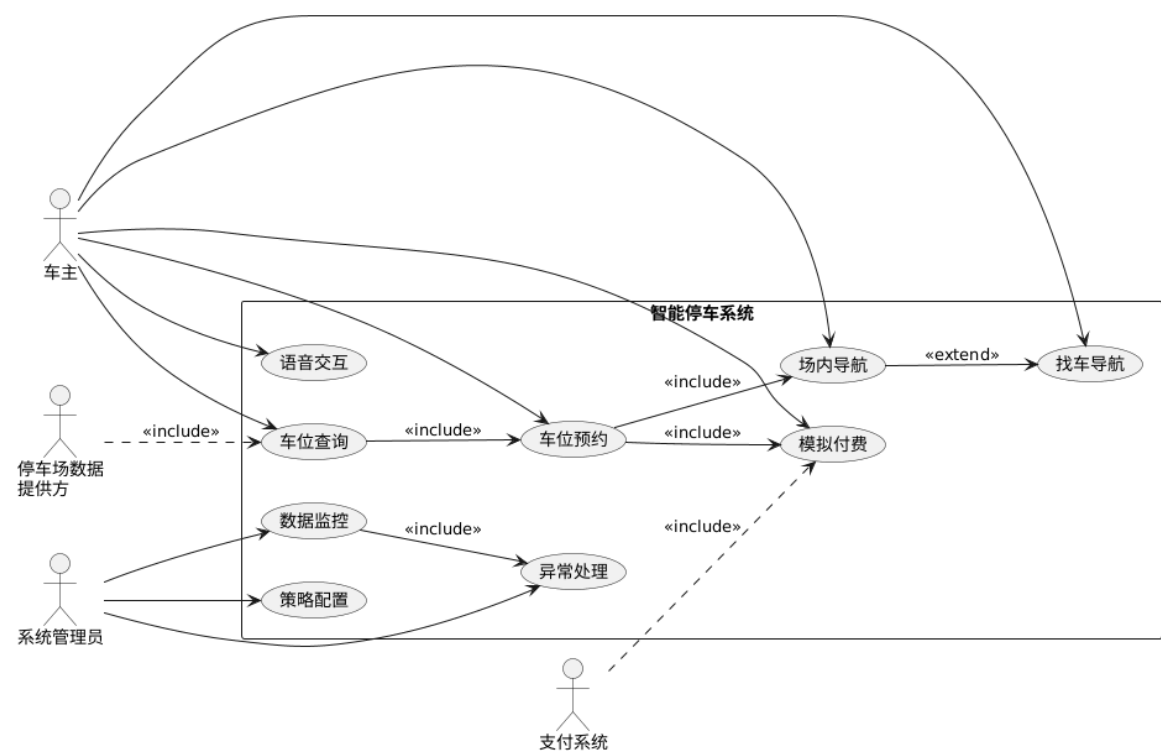


图 4 UML 用例图

3.1.5 活动图 (Activity Diagram) (对关键业务流程建模)

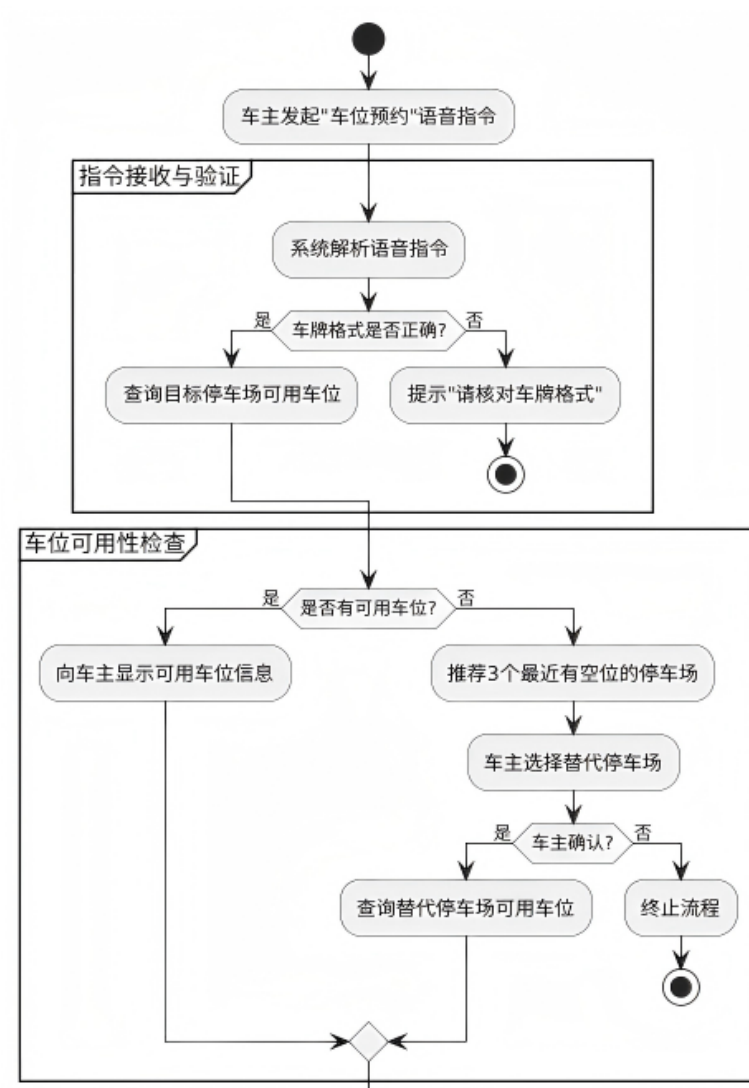


图 5 车位预约流程 - 指令接收与验证阶段

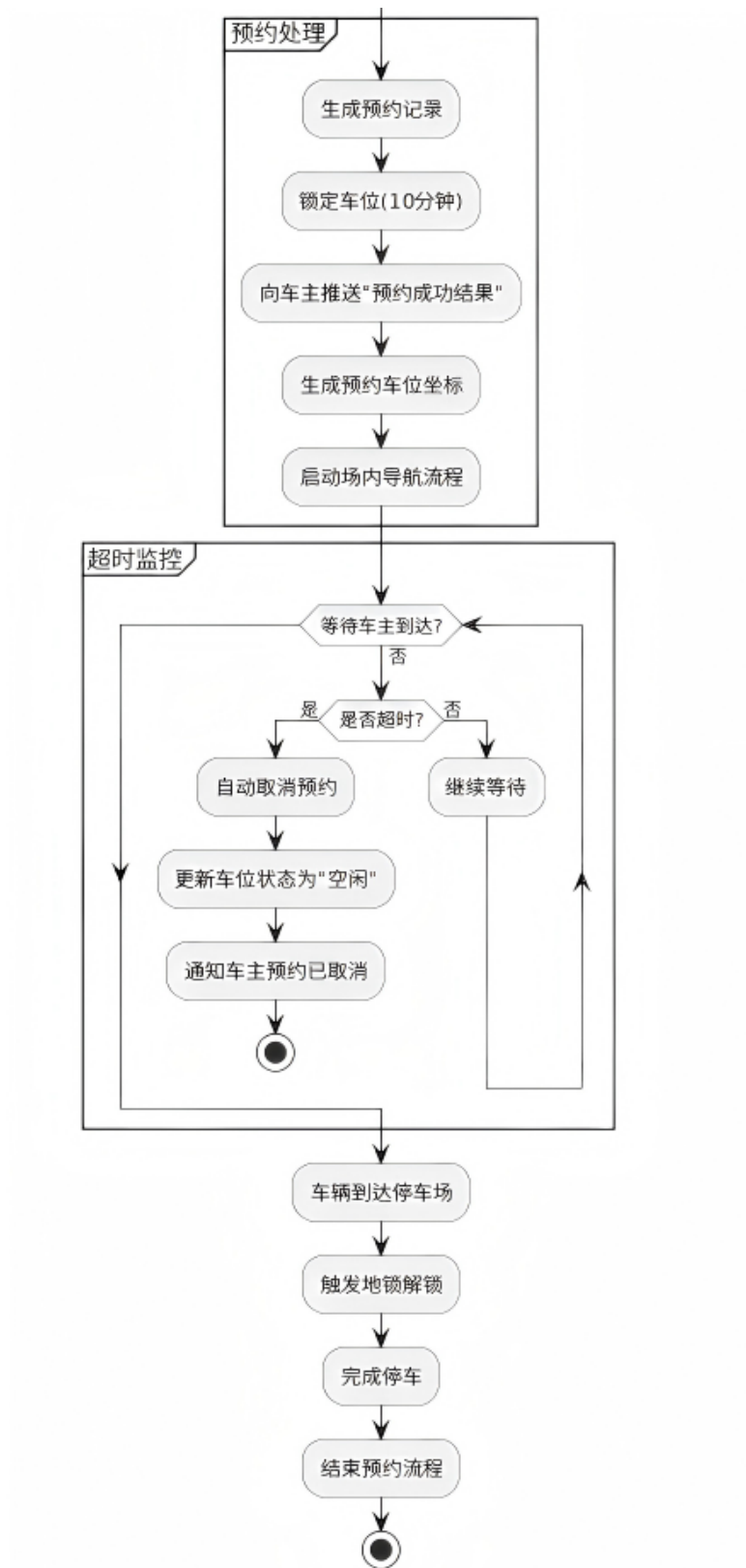


图 6 车位预约流程 - 预约处理与结果阶段

3.2 非功能性需求

3.2.1 性能需求 (响应时间、并发用户数)

系统应保证关键业务路径的端到端响应与吞吐在可接受范围内，对外接口的延迟目标为：核心接口的 p95 时延小于 800ms，整体端到端成功率（从发起预约到后台落库并返回结果）99%，业务错误率 < 0.5% 以便保证演示与评审过程中的体验可控与指标可比。性能验证以“典型 MVP 规模”文档中用于测算的 200 个车位、若干停车场与日均若干百次服务量）为基准，要求在中等压力下（模拟峰值到达与并发预约场景）核心接口仍能满足上面延迟/成功率目标；对于极端并发的竞价/抢占情形，后端必须采用事务与行级锁或分布式锁策略来保证“预约—更新车位—地锁控制”的原子性，且在演示阶段可通过限制并发提交速率（例如对同一车位临时限流）作为保护措施以避免状态错乱。为确保观测性与回放可重复，需提供压力脚本与一键回放工具，以及在压力测试中记录的 p50/p95/p99、错误率、并发会话数等指标供验收使用。

3.2.2 安全性需求 (认证、授权、数据加密)

身份与权限、数据保护与第三方接入风险是本系统的三大安全关注点。

认证层面要求用户账号以手机/短信或第三方 OAuth（如微信/支付宝）完成双要素或强绑定，车牌须在预约流程中完成绑定并作为发起地锁解锁的关键核验条件；任何车牌与预约不一致须拒绝解锁并记录审计日志以便事后追溯。管理端采用基于角色的最小权限模型（管理员、运营、客服、审计只赋予必要权限），所有敏感操作（手动解锁、退款、策略变更）必须有操作人、时间与理由的审计链。传输层务必启用 TLS（建议 TLS 1.2+），支付与第三方 SDK 的接入使用厂商推荐的安全 token 机制；静态存储对车牌、手机号等敏感字段至少做字段脱敏或加密存储（演示/公开资料中需脱敏展示，如隐藏中间 3 位），并对加密密钥实行集中管理与访问控制。考虑到外部地图/语音/支付 API 的配额与可用性风险，必须设计降级与兜底策略（例如本地 JSON 静态数据、手动输入/二维码支付等），并对第三方回调与请求采用时间戳/签名校验以防重放或伪造。系统还应提供完整的安全日志与入侵检测告警，关键安全事件（例如异常解锁、支付异常、频繁失败的车牌比对）需实时推送至运维/管理员。

3.2.3 可用性需求 (用户界面友好性)

面向不同年龄与技能层次的用户群，UI/交互必须做到“直观、容错、可降级”。默认提供响应式网页/移动端界面（兼容主流 iOS/Android 与现代浏览器），关键业务不得超过五步完成且关键操作使用大按钮与明确的确认/取消提示。针对中青年用户保留语音优先交互以支持驾驶场景的免手动操作，同时为老年/识别失败用户提供“图形简化模式”与一键求助/人工辅助通道，确保系统在识别不佳或网络差时仍能完成核心任务。可用性还包括可见的状态反馈（锁定剩余时间提醒、预约即将失效通知通过推送 + 语音播报双通道），以及离线/弱网兜底（预加载停车场静态地图、生成离线二维码支付并在网络恢复后补传），以保证在地下或信号差场景下用户不会被完全阻断。为降低学习成本与提升自助率，发布时应附带一段 1 分钟演示短片与内置的交互引导，管理员端提供简明操作手册与“常见故障速查”入口，配合

演示账号与脚本来保障课堂/评审演示的一致性。

3.2.4 可靠性需求 (系统稳定性、故障恢复)

为保证演示与后续生产化迁移的稳定性，系统需在架构与运维上满足可观测、可恢复与可修复的要求。数据库与关键状态变更必须使用事务保障一致性，外部设备（地锁/感应）与后台状态要有定期校验机制（示例：周期性脚本每分钟比对硬件与数据库并自动修正不一致），并对定时任务并发冲突制定保护策略（例如对同一车位操作串行化或短时限流）以避免“超售”或重复解锁。运维角度要求建立基础的健康检查与告警（接口错误率、响应时延、主流程失败率等），每晚生成简单的运行摘要（能否跑通主流程、是否有卡顿或新风险）以便快速判断演示可用性；在发生故障时应有分级响应流程（演示级快速兜底如切换到本地静态数据或一键重启脚本，运维级恢复如回滚发布/恢复数据库备份），且对严重线上缺陷制定修复时限（演示/教学环境下的 bug 修复响应 24 小时作为目标）。数据备份与恢复策略需明确：至少保留最近 N 日的全量/增量备份并验证恢复可行性；对支付与对账相关数据建议更严格的 RPO/RTO（在生产化阶段应细化为具体分钟级或小时级目标）。系统还应在设计时考虑水平扩展能力（服务无状态化、使用负载均衡与可扩容实例）以便在后续真实部署时能提升可用性与吞吐。

3.3 接口需求

3.3.1 用户接口

本系统为典型的“前端交互 + 后端服务 + 仿真设备”结构，用户接口以移动端 Web 页面与模拟 App 端为主要交互渠道，设计目标是实现从预约到入场的无缝操作体验。系统在界面设计上遵循简洁直观、操作路径短、状态反馈及时的原则。用户入口主要包括首页（车场查询与地图展示）、预约页（车位选择与确认）、导航页（第三方地图跳转或 ETA 展示）、到场页（扫码/语音解锁）、离场结算页（自动支付或手动确认）等模块。UI 风格采用深浅双主题自适应设计，核心操作按钮保持一致颜色与位置，以降低学习成本。界面交互支持点击、滑动与语音输入三种方式；语音输入模块基于集成的语音识别服务，可解析自然语义指令（如“帮我预约靠近南门的车位”）并给出结果确认提示。系统针对不同用户群提供差异化界面：普通用户默认进入标准模式（地图 + 预约入口），老年用户可启用简洁模式（大字号、语音播报、关键按钮放大），管理员与运营端则进入管理界面，提供预约统计、地锁监控、车位状态一览、异常告警与手动干预功能。所有界面均要求具备明显的状态提示与错误处理反馈，例如网络异常、预约超时、地锁异常时提供可理解的提示与重试选项。界面交互应支持在主流浏览器（Chrome/Edge/Safari）和主流移动系统（iOS/Android）上稳定运行，并兼容弱网或离线场景（本地缓存与静态地图预载）。此外，用户接口需配合后台提供统一的身份登录与信息同步能力，即同一用户在不同设备上登录后可保持预约历史、车牌信息与支付状态一致。

3.3.2 外部接口 (与其他系统或 AI 服务的交互)

系统的外部接口设计遵循“松耦合、可替换、可降级”的原则，支持与第三方 AI 服务、地图服务、支付网关及设备模拟系统的交互，保证系统在真实部署和仿真演示两种环境下均可正常运行。

地图与路径规划接口：通过高德地图 Web 服务 API 获取停车场地理位置、实时路况和 ETA（预计到达时间）数据，接口形式为 HTTPS RESTful 请求，核心数据字段包括目的地坐标、行驶路线、时间与距离估算。若外部 API 不可用，系统自动调用本地静态地图数据进行降级显示。

语音识别与自然语言解析接口：调用科大讯飞或百度智能语音识别 API，实现语音唤醒、语义解析及语音播报功能。交互流程为“前端录音 → API 上传音频流 → 返回识别文本 → 后端语义解析并执行相应指令”，识别结果采用 UTF-8 编码的 JSON 结构传输。为保护隐私，系统在传输过程中启用 TLS 加密，并在日志中脱敏处理语音内容。

支付与结算接口：系统集成支付宝/微信支付沙箱接口，实现预约支付、地锁使用费结算及退款流程。接口采用官方 SDK 的 HTTPS 通道，回调采用签名校验机制并记录交易流水号与时间戳。支付成功后，系统自动更新订单状态并触发地锁解锁逻辑；若第三方接口异常，则切换到模拟支付流程用于演示或测试。

设备与物联网接口：停车地锁与车位感应设备通过模拟控制器或 MQTT 协议虚拟接入。接口定义包括状态上报（地锁状态、车位占用状态）与控制指令（升锁/降锁、重置），采用 JSON 格式报文；后端通过 HTTP/MQTT 中间件转发并记录日志，用于演示物理交互过程。

数据与运维接口：提供对外的数据导出接口（CSV/JSON 格式），便于运营人员进行预约记录统计与后续分析；同时预留内部 API 供后续接入 AI 预测模块（如利用机器学习算法预测车位空闲率、优化调度策略）。该接口需提供统一认证 token，并定义调用频率与数据格式规范。

4 运行环境规定

参考文献

- [1] 佚名. 智能停车系统: 解决城市停车难题的创新方案 [EB/OL]. CSDN 博客, 2024. <https://blog.csdn.net/universssky2015/article/details/143247320>
- [2] 佚名. 智能停车系统.docx[EB/OL]. 人人文库, 2024. <https://m.renrendoc.com/paper/401577998.html>
- [3] 佚名. 智能停车系统研究.docx[EB/OL]. 人人文库, 2024. <https://m.renrendoc.com/paper/305288101.html>
- [4] 佚名. 智能停车系统资料.pptx[EB/OL]. 金锄头文库, 2024. <https://m.jinchutou.com/shtml/view-595430979.html>

- [5] 佚名. 智能停车系统设计毕业论文.docx[EB/OL]. 原创力文档, 2025. <https://m.book118.com/html/2025/0125/5343140223012041.shtm>
- [6] 佚名. 智能停车系统.docx[EB/OL]. 原创力文档, 2025. <https://m.book118.com/html/2025/0228/8132114012007036.shtm>