

南农智慧地图系统设计报告

1. 系统需求

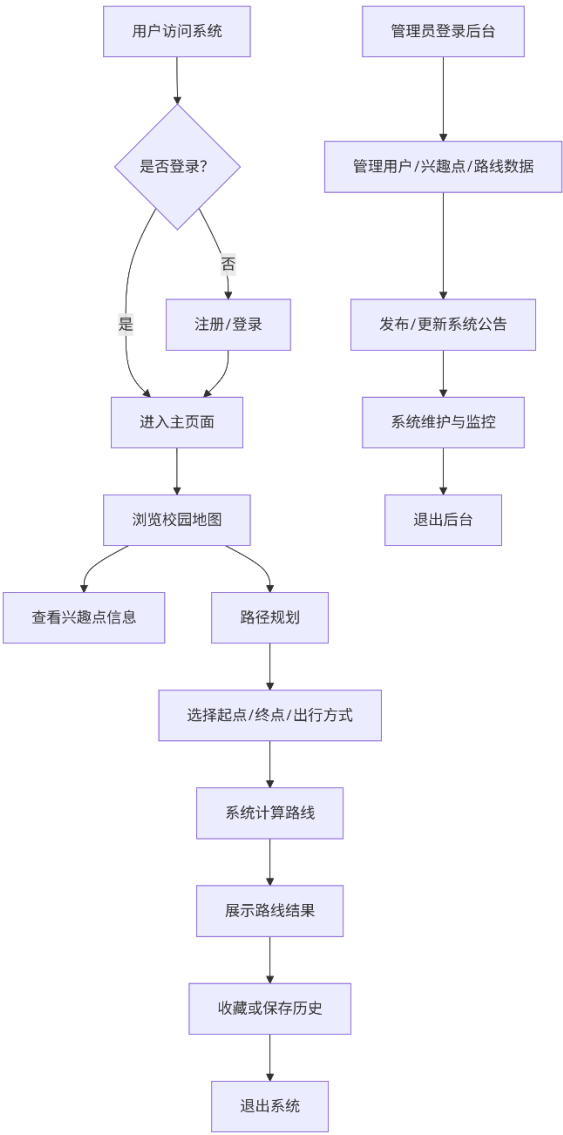
1.1 系统设计目标

构建一个面向南京农业大学校园的智慧地图系统，提供地图浏览、路径规划、兴趣点查询、用户个性化收藏与历史记录等功能，提升校园导航效率与用户体验，推动智慧校园建设。

1.2 可行性分析

系统采用成熟的 Flask + Python 后端架构，结合高德地图 API 与 GeoPandas 处理地理数据，前端使用 HTML/CSS/JavaScript，数据库采用 MySQL，技术栈清晰、稳定，具备较高的技术可行性。

1.3 系统流程分析



2. 系统架构设计

南农智慧地图系统采用标准的三层 B/S 架构设计：表示层基于 HTML/CSS/JavaScript 与高德地图 API 构建，提供跨平台、响应式的前端交互界面；业务逻辑层依托 Python 与 Flask 框架实现，负责用户认证、路径规划、地

图服务集成等核心功能处理；数据访问层采用 MySQL 数据库进行结构化与地理空间数据的持久化存储。通过三层 B/S 架构，实现前后端分离、模块解耦，支持负载均衡与扩展部署，在保障系统性能与安全性的同时，具备良好的可维护性与可扩展性，完全适配校园智慧地图服务的应用场景与技术要求。

3. 数据库设计

系统主要数据表包括：

用户表（user）

字段名	数据类型	约束	说明
user_id	INT	PRIMARY KEY, AUTO_INCREMENT	用户 ID（主键）
username	VARCHAR(50)	UNIQUE, NOT NULL	用户名（唯一）
password_hash	VARCHAR(255)	NOT NULL	密码哈希值
email	VARCHAR(100)	UNIQUE	邮箱（唯一）
role	VARCHAR(20)	DEFAULT 'user'	角色（user/admin）
created_at	DATETIME	DEFAULT CURRENT_TIMESTAMP	创建时间
last_login	DATETIME		最后登录时间

兴趣点表（poi）

字段名	数据类型	约束	说明
poi_id	INT	PRIMARY KEY, AUTO_INCREMENT	兴趣点 ID（主键）
name	VARCHAR(100)	NOT NULL	兴趣点名称（如教学楼、图书馆）

字段名	数据类型	约束	说明
description	TEXT		详细描述
latitude	DECIMAL(10,8)	NOT NULL	纬度
longitude	DECIMAL(11,8)	NOT NULL	经度
type	VARCHAR(30)	NOT NULL	类型 (building/facility/landmark)
created_at	DATETIME	DEFAULT CURRENT_TIMESTAMP	创建时间
updated_at	DATETIME	ON UPDATE CURRENT_TIMESTAMP	更新时间

路线历史表 (route_history)

字段名	数据类型	约束	说明
history_id	INT	PRIMARY KEY, AUTO_INCREMENT	历史记录 ID (主键)
user_id	INT	FOREIGN KEY (user_id) REFERENCES user(user_id)	用户 ID (外键)
start_poi_id	INT	FOREIGN KEY (start_poi_id)	起点兴趣点 ID

字段名	数据类型	约束	说明
		REFERENCES poi(poi_id)	
end_poi_id	INT	FOREIGN KEY (end_poi_id) REFERENCES poi(poi_id)	终点兴趣点 ID
travel_mode	VARCHAR(20)	NOT NULL	出行方式 (walking/driving/cycling)
route_data	JSON		路线详情 (JSON 格式存储路径点)
created_at	DATETIME	DEFAULT CURRENT_TIMESTAMP	查询时间

收藏表 (favorite)

字段名	数据类型	约束	说明
favorite_id	INT	PRIMARY KEY, AUTO_INCREMENT	收藏 ID (主键)
user_id	INT	FOREIGN KEY (user_id) REFERENCES user(user_id)	用户 ID (外键)
poi_id	INT	FOREIGN KEY (poi_id) REFERENCES poi(poi_id)	收藏的兴趣点 ID

字段名	数据类型	约束	说明
favorite_type	VARCHAR(20)	DEFAULT 'poi'	收藏类型 (poi/route)
created_at	DATETIME	DEFAULT CURRENT_TIMESTAMP	收藏时间

管理员表 (admin)

字段名	数据类型	约束	说明
admin_id	INT	PRIMARY KEY, AUTO_INCREMENT	管理员 ID (主键)
user_id	INT	FOREIGN KEY (user_id) REFERENCES user(user_id)	关联用户 ID (外键)
permissions	JSON		权限列表 (JSON 格式存储)
created_at	DATETIME	DEFAULT CURRENT_TIMESTAMP	创建时间

4. 系统环境与实现

4.1 系统设计原则

系统性：虽然系统有不同的模块，但把它认为是统一的整体存在。所以，我们在系统设计中，需要考虑整个系统的各个方面，实现统一的系统代码，传输语言尽可能一致，将设计规范化到标准，使系统设计思路保持一致 一个开发可以使用多个，以方便随后的修改和改进，当设计人员可以更好地理解程序时。

灵活性：那就是适应环境的能力，其目的是使系统保持长久的活力。在系统开发和设计过程中，我们应该尝试使用模块化结构，从而能够提高模块之间的独立性，减少模块间的相互影响，减少子模块之间的链接。这有利于对系统功能的修改，并添加新的内容，以提高系统的适应性，而不影响其他模块。

可靠性：可靠性是指恢复系统的抗干扰能力和干扰。一个标准的社团管理系统应具有一定的安全性和稳定性，可以实现对数据的修改和保存。

经济性：尽量降低系统的开发成本。不能盲目追求硬件配置，高科技，以用

户的实际需求为先决条件；另外，系统设计应尽量避免太复杂，模块应该简单，以缩短时间，减少耗电量。

4.2 开发环境

前端：HTML/CSS/JavaScript，高德地图 API

后端：Python 3.x + Flask

数据库：MySQL 5.7+

部署环境：Windows/Linux + Waitress 服务器

4.3 硬件环境

服务器需具备稳定网络与中等硬件配置。客户端需兼容主流操作系统（Windows 10+/iOS 13+/Android 9+）及现代浏览器（Chrome 90+/Safari 等），并保障网络连接畅通。确保系统在校园典型使用场景下稳定运行，支持不少于 100 名并发用户的同时访问，满足路径规划响应时间小于 3 秒的性能要求，并具备良好的可扩展性与容灾能力。。

5. 系统功能实现

5.1 前端功能模块

地图展示：基于高德地图 API，支持缩放、拖拽、标记点展示

路径规划：支持多种出行方式，提供最优路径推荐

用户交互：登录注册、收藏管理、历史记录查看

自然语言交互：支持语音或文本输入路线规划

5.2 后端功能模块

用户认证：基于 Session 或 Token 的登录机制

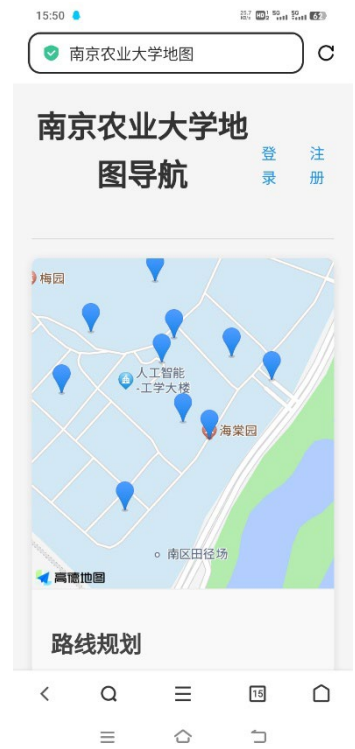
路径计算：调用高德 API，结合本地数据处理

数据管理：兴趣点、用户数据、路线历史的增删改查

管理员功能：用户管理、数据维护、系统监控

5.3 系统界面展示

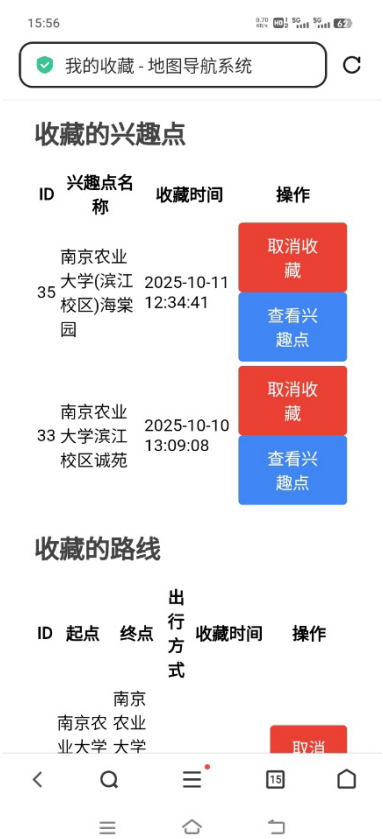
主地图页面



登录/注册页面



收藏管理页面



路线规划结果页面



访问地址: <https://map.njau.homes>