**智慧交通系统设计**

**概要设计**

1. **数据结构**

数据结构设计主要针对数据库。本系统有三部分数据需要在数据库中存储，分别是车辆数据、地图数据和日志数据

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 表名 | Vehicle | | | | |
| 功能描述 | 存储车辆属性和状态 | | | | |
| 列名 | 数据类型 | 允许空 | 默认值 | 约束 | 说明 |
| ID | serial | N |  | PK,ID,UQ | 车辆编号 |
| License | varchar | N |  | UQ | 车牌号 |
| isEmpty | boolean | N | TRUE |  | 车辆是否空闲 |
| Location | point | N |  |  | 车辆当前位置 |
| Destination | point | Y |  |  | 车辆目标位置 |
| Path | path | Y |  |  | 车辆短时行驶轨迹 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 表名 | Map\_Loc | | | | |
| 功能描述 | 存储地图上的地点 | | | | |
| 列名 | 数据类型 | 允许空 | 默认值 | 约束 | 说明 |
| LID | serial | N |  | PK,ID,UQ | 地点编号 |
| Location | point | N |  | UQ | 地点坐标 |
| State | int | N |  |  | 地点状态 |
|  |  |  |  |  |  |
| 表名 | Map\_Road | | | | |
| 功能描述 | 存储地图上的道路 | | | | |
| 列名 | 数据类型 | 允许空 | 默认值 | 约束 | 说明 |
| RID | serial | N |  | PK,ID,UQ | 道路编号 |
| Begin | int | N |  |  | 道路起始地点编号 |
| End |  | N |  |  | 道路终止地点编号 |
| Length | int | N |  |  | 道路长度 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 表名 | Log | | | | |
| 功能描述 | 日志，记录车辆历史行为 | | | | |
| 列名 | 数据类型 | 允许空 | 默认值 | 约束 | 说明 |
| LogID | serial | N |  | PK,ID,UQ | 日志记录编号 |
| VehID | int | N |  |  | 记录对应车辆编号 |
| StartTime | timestamp | N |  |  | 发车时间 |
| StopTime | timestamp | N |  |  | 停车时间 |
| StartLoc | point | N |  |  | 发车地点 |
| StopLoc | point | N |  |  | 停车地点 |
| Mileage | int | Y |  |  | 行驶里程 |

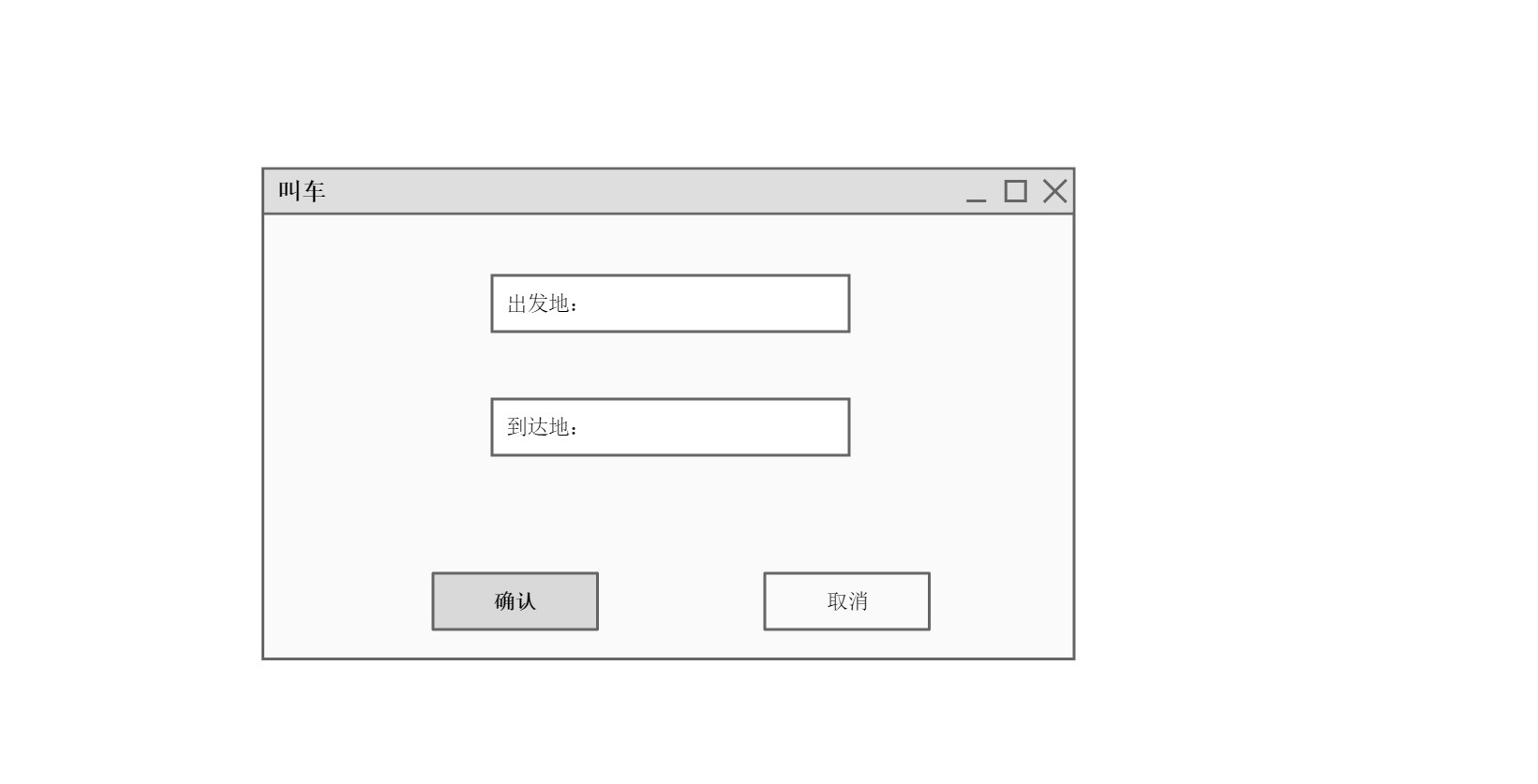
1. **UI设计**

本系统有两部分人机交互界面，一部分提供给乘客用于叫车，一部分提供给开发人员用于维护数据库

这是乘客界面主页，地图与车辆数据由系统给出



当乘客希望用车时，点击“叫车按钮”，弹出输入对话框。乘客可以输入出发地和目的地，可以点击“取消”按钮退回主页



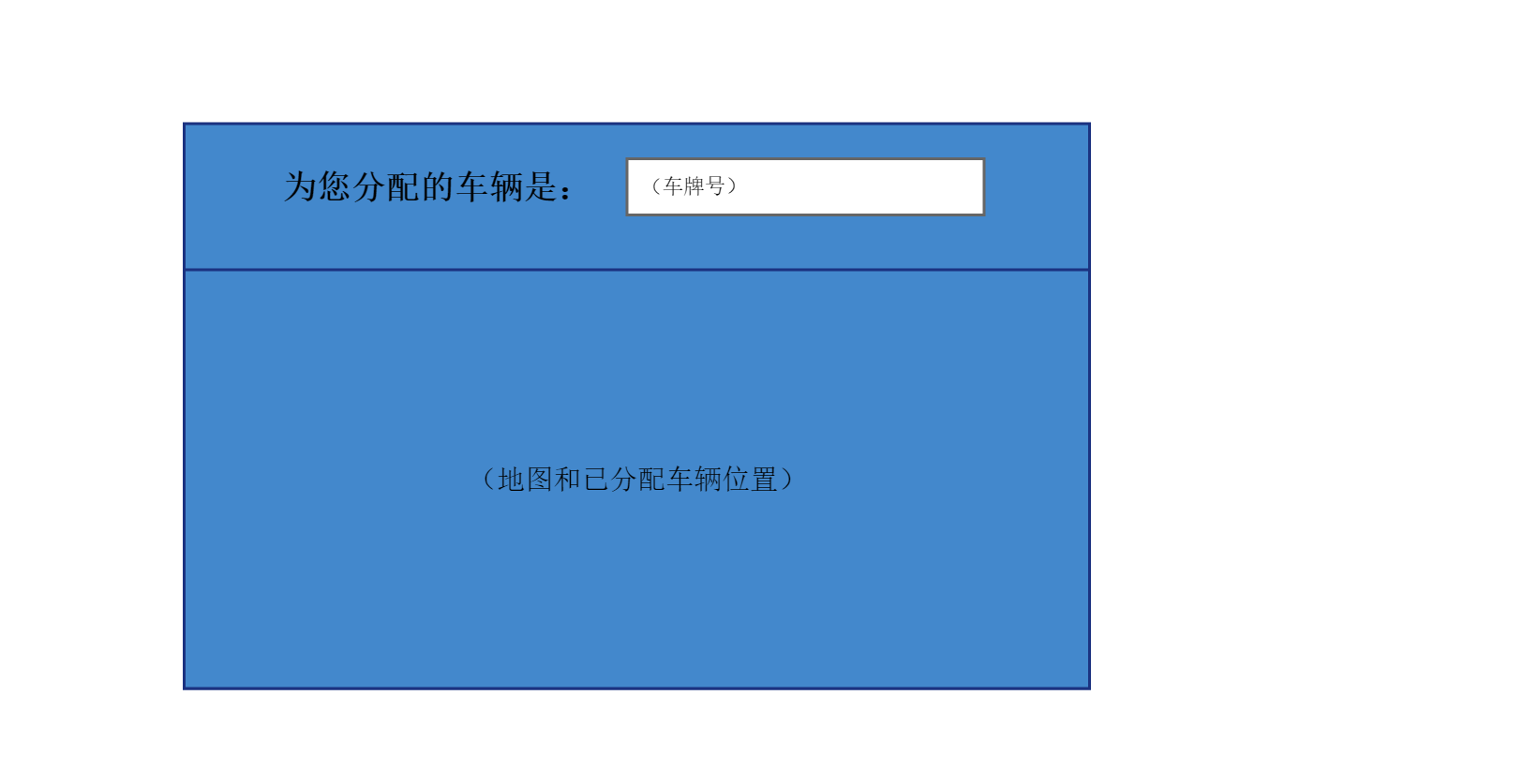
乘客输入完毕，点击“确认”按钮，决策引擎开始工作，弹出等待处理对话框



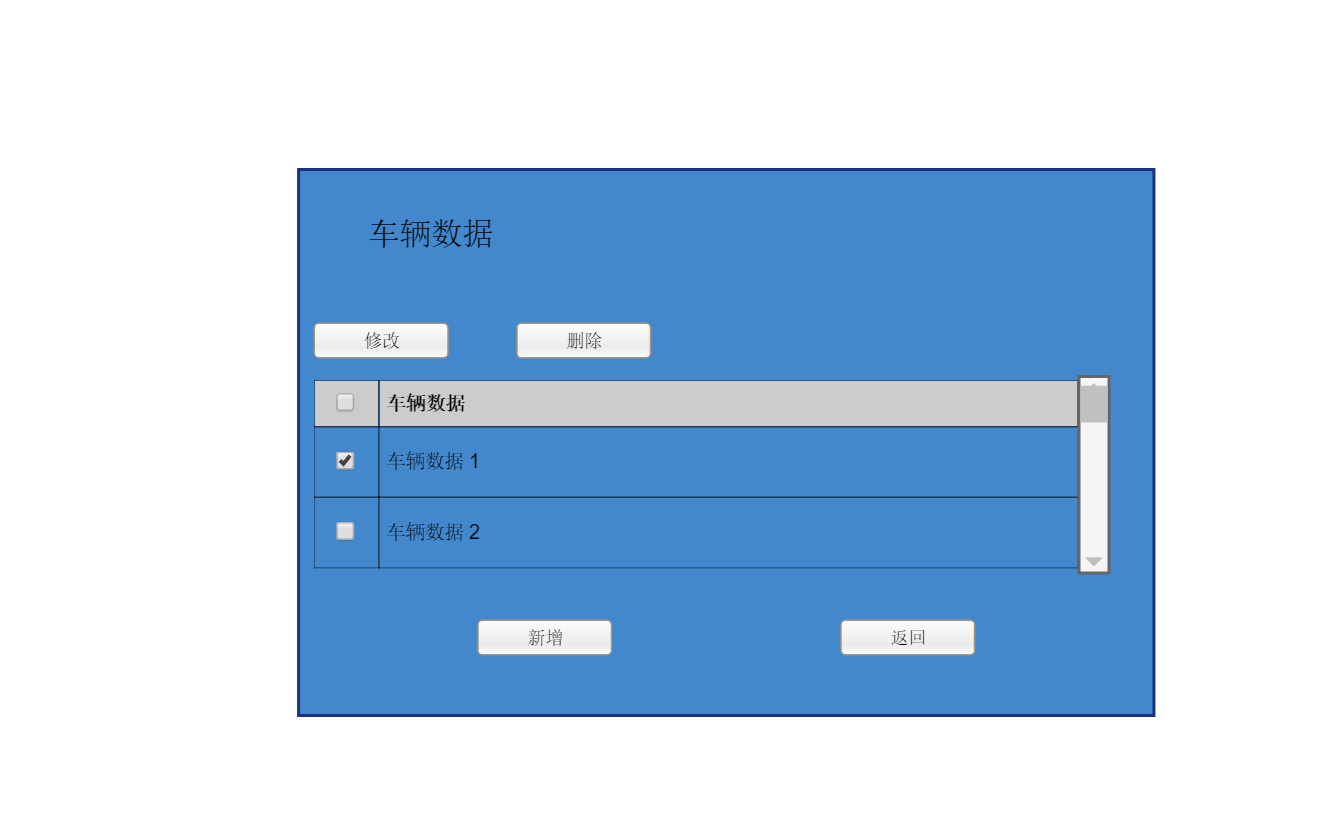
车辆调度成功，弹出成功对话框



乘客点击“OK”按钮或成功对话框显示一秒后，自动跳转至等车页，车牌号等数据由系统给出



以下是开发人员界面





开发人员可以审阅地图和车辆数据；用“新增”按钮新增数据；可以选中一条或多条数据后，用“修改”和“删除”按钮修改和删除数据

1. **模块分解及其接口**

|  |  |  |  |
| --- | --- | --- | --- |
| 模块名称 | 模块功能 | 模块接口 | 接口说明 |
| DecisionEngine | 智慧决策模块，基于最短路算法，为车辆提供行进路线建议 | resultForUser() | 在乘客界面输出为乘客分配车辆的车牌号 |
| resultForSystem() | 为DataCache模块提供车辆调度结果数据，包括分配的车辆编号和其行进路线。 |
| DataCache | 数据缓存模块，缓存乘客请求数据和车辆状态数据，供DecisionEngine模块和UI层使用 | pasRequestWrite() | 供乘客界面写入乘客请求数据，包括出发地、目的地、请求发出时间和请求处理状态 |
| pasRequestRead() | 供DecisionEngine模块读取乘客请求数据 |
| vehicleDataWrite() | 供物联网平台和UI层实时写入任一车辆数据，包括当前位置、是否空闲、目的地和短时轨迹等 |
| vehicleDataRead() | 供UI层和DecisionEngine模块读取任一车辆数据 |
| mapRead() | 供DecisionEngine模块和UI层读取地图数据 |
| Log | 日志模块，记录所有车辆历史行为，以便评估优化算法性能 | behaviorWrite() | 供DecisionEngine和DataCache模块写入日志记录，包括每段行程的起止时间、起止点和里程 |
| BasicData | 基础数据管理模块，存储地图和车辆数据 | mapRead() | 供DataCache模块读取地图数据 |
| vehicleDataRead() | 供DataCache模块读取车辆数据 |
| vehicleDataWrite() | 供DataCache模块写入车辆数据 |