**智慧交通系统设计**

**需求分析**

**1 概述**

本课题的目标是基于云端平台和前端多智能设备架构，提供基于云端的**智能数据处理与决策服务**，并重点设计与开发面临多种数据输入的智慧交通大脑原型，实现类似滴滴或京东物流派送的智慧交通应用。未来将进一步扩展机器之间的通信，实现机器与机器之间的交互（含交易）与机器与人之间的交互。

**2 功能性需求**

|  |  |  |  |
| --- | --- | --- | --- |
| 序 | 需求 | 子需求 | 说明 |
| 1 | 基础数据管理 | 采用数据库进行存储。在系统关机期间还能存储整个系统的基本资料和状态便于重启动后恢复系统状态。 |  |
| 1.1必做 |  | 车辆数据的CRUD操作 | 数据库 |
| 1.2必做 |  | 提供地图存储。这里的地图采用Graph模型进行描述，建议采用稀疏矩阵进行存储。 | 数据库 |
|  |  |  |  |
| 2 | 数据缓存 | 主要在内存中实现。 |  |
|  |  | 采用队列数据接口或队列产品实现实时数据流输入接口。与物联网平台提供的接口配接，接收来自物联网的车辆状态数据，并尽可能减小丢失概率。 | 假定物联网平台能够提供相应的接口支持实时获取数据。开发阶段用车辆模拟器程序产生模拟数据。 |
|  |  | 客户请求记录，记录客户发出的模拟叫车请求序列，包含出发地和目的地。 |  |
| 必做 |  | 记录和保存车辆属性，包括车辆是否空闲、车辆当前位置、终点目标等。 | 提供高效的数据结构，实现内存中的高效数据存储与访问。 |
|  |  | 支持系统外部车辆状态数据的实时输入，并动态更新系统内的车辆状态数据。这里的状态数据主要是车辆是否空闲和车辆当前位置数据{x,y} |  |
|  |  | 支持车辆历史数据的持久化存储（即支持保存车辆轨迹到数据库中） | 目前只考虑存储过去一小段时间的轨迹数据，这样就可以完全利用内存实现，无需考虑持久化存储。 |
|  |  | 数据查询接口，能够为AI计算服务和人机交互服务提供与之匹配的便捷的数据访问接口 |  |
|  |  |  |  |
| 3 | 车辆模拟器 | 同物联网组的智能硬件模拟器 |  |
| 3.1  必做 |  | 模拟车辆的行为和能力与服务端进行通信，使得整个工作在智能小车不完备或未完工的情况下依然可以顺利开发下去。包括通信能力和上报位置相关数据的能力。 |  |
|  |  | 一个独立程序能单独运行。 |  |
|  |  |  |  |
| 4 | 基于AI的智慧决策服务 | 主要是用到优化算法。优化是基石。 |  |
| 4.1  必做 |  | 能够给出每一辆车的行进路线建议。相应算法的数据输入来自基础数据管理模块和数据缓存模块。 | 算法不限。常用最短路法、动态规划、虚拟力场法等。 |
|  |  | 在选出路径的时候，遵循优化原则。目前的优化原则是路径最短。 | 假定车辆恒速 |
|  |  | 位置服务1。在系统不能准确提供车辆位置数据但是可以提供部分相关数据的情况下（例如一些车与车的距离以及一些车与锚点设备的距离），通过优化方法求解车辆位置。 | 视前端能够提供的数据而定 |
|  |  | 位置服务2：基于顶置摄像头实现车辆位置获取。 | 视现场能够提供的数据而定 |
|  |  | 通过物联网平台发送指令给智能车（包括模拟车程序），指挥车辆行进。 |  |
| 5  必做 | 日志模块。 | 记录每辆车的行为，包括何时启动，合适停止，从何地到何地。这些时间和里程数据用于衡量决策的性能指标。 |  |
|  |  |  |  |
| 6 | 人机交互 |  | 基于Web开发 |
| 必做 |  | 提供车辆数据的CRUD页面 |  |
| 必做 |  | 提供地图数据的CRUD页面 |  |
|  |  | 提供简易地图的显示 |  |
| 必做 |  | 能够动态显示每辆车的位置。 |  |
| 必做 |  | 能够接受客户的车辆请求输入。输入信息包括出发位置和目的位置。（模拟客户叫车请求） |  |
| 7  必作 | 系统性能评价 | 能够从日志数据中统计出平均车辆行驶时间和里程。 | 作为不同算法PK的标准。 |
| 8 | 系统集成需求 | 要求所有需求的实现代码应集成在一起协同工作，要求能够按照指令要求自主的从源A走到目标B |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**3 技术指标需求**

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | 能够对不少于5辆车进行调度，调度时间不大于10秒钟。 | 允许全部模拟车，这样整个系统可以在计算机上模拟，便于开发测试评价。  但推荐结合真实智能小车。 |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**4 技术路线和选型要求**

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | 采用Java开发 |  |  |
|  | 服务端采用微服务架构， | 服务端内部的接口采用RESTful方式。 |  |
| 2 | 日志数据统计用Python | 服务端采用Java开发 |  |
| 3 | 智能决策部分可以用Java或Python |  |  |
| 4 | 人机交互部分采用前后端分离方案 | UI呈现部分采用vue.js。 |  |
| 5 | 服务端数据库 | 采用开源数据库Postgres，包含一些地理位置相关的操作，可能能简化一些开发。 |  |