

web api项目报告

设计和实现

1. 引言

城市信息查询系统旨在为用户提供关于全球城市的多维度信息，包括基本信息、经济状况、天气预报、旅游景点和新闻等。用户通过输入城市名称，即可获取相关数据，并在地图上查看城市位置。此系统使用现代前后端分离架构，确保高效的数据处理和友好的用户体验。

2. 系统架构

2.1 前端

- 技术栈：**前端使用 **Vue.js** 作为框架，使用 **Axios** 进行 HTTP 请求。
- 功能：**前端负责展示用户界面，包括城市信息的搜索、显示和地图的展示。

2.2 后端

- 技术栈：**后端采用 **Node.js** 环境和 **Express** 框架，使用 **Axios** 与外部 API 交互。
- 功能：**后端处理来自前端的请求，整合各类城市信息，包括基本信息、经济、天气、旅游景点和新闻。

2.3 数据源

系统集成了多个外部 API 以获取所需数据，包括：

- 城市基本信息：**通过某个百度百科爬取城市描述。
- 经济信息：**从worldbank提取数据。
- 天气信息：**使用和风天气API 获取实时天气预报。
- 旅游景点：**从百度地图 API 获取城市旅游景点信息。
- 新闻信息：**从新闻 API 获取城市的最新新闻。
- 地图：**从高德地图获取地图

3. 功能实现

3.1 用户界面

用户界面设计友好，包含搜索框、信息展示区和地图展示区。用户可以输入城市名称并获取相关信息。

3.2 信息查询

- 用户输入城市名称后，前端通过 Axios 向后端发送请求。
- 后端根据请求并发调用多个 API 获取城市信息，并将数据返回给前端。
- 前端处理并展示返回的数据，包括基本信息、经济信息、天气预报、旅游景点和新闻。

3.3 地图展示

系统使用高德地图 API 显示城市的位置，地图在用户搜索时动态更新中心点。

3.4 响应式设计

采用 CSS 进行响应式布局，确保在不同设备上均能提供良好的用户体验。

3.5 天气查询

先使用查找城市经纬度的api，再通过经纬度，调用查询天气的api。

4. 数据结构

在实现过程中，数据结构设计考虑了各类信息的清晰呈现，包括：

- **基本信息**：城市的描述和链接。
- **经济信息**：经济数据列表，每个条目包含标题和链接。
- **天气信息**：包括三天的天气预报，包含最高和最低气温等。
- **旅游景点**：景点名称、地址、区域和电话。
- **新闻信息**：新闻标题和链接。

5. 总结

城市信息查询系统成功实现了用户对城市信息的查询需求，通过整合多种数据源，提供全面的信息展示。系统的前后端分离设计确保了良好的扩展性和维护性。未来，可以考虑加入用户注册、搜索历史等功能，以进一步提升用户体验；如果项目规模继续扩大，之后会考虑使用微服务架构。

配置和部署说明：

安装npm并安装一些Node.js包

npm版本10.8.2

```
npm install
```

```
npm init -y
```

```
npm install express axios cheerio dotenv
```

node版本v20.17.0

vue版本@vue/cli 5.0.8

前端端口8080，后端端口5000

使用的web api文档

使用了6个不同提供商的7个api

worldbank: [API 文档\(worldbank.org\)](https://worldbank.org)

百度地图: [地点检索](#) | [百度地图API SDK\(baidu.com\)](#)

和风天气: api1: 城市经纬度: [城市搜索 for API](#) | [和风天气开发服务\(qweather.com\)](#)

api2: 天气预报: [每日天气预报 for API](#) | [和风天气开发服务 \(qweather.com\)](#).

网易新闻: [网易新闻接口-CSDN博客](#)

百度百科: 爬取基本信息: <https://baike.baidu.com/>

高德地图: 前端获取城市地图: [JS API 的加载-基础-进阶教程-地图 JS API 2.0](#) | [高德地图API \(amap.com\)](#).