项目计划

**一、需求分析：**

我们需要设计学生游学系统，该系统具备游学推荐、游学路线规划、场所查询、游学日记管理等功能，这些功能分别在游学前、中、后实现。

**功能分析：**

**游学前**：

对于游学推荐：

需先按照游学热度、评价和个人兴趣选择游学目的地；系统采用排序算法和查找算法向学生推荐游学景点和学校，可以按照游学热度、评价和个人兴趣进行推荐；

学生可以输入景点和学校的名称、类别、关键字等进行查询，查询结果有多项时，可以对查询结果按照热度和评价进行排序。（核心算法为查找算法和排序算法）

**游学中**：

对于游学路线规划：

在学校和景点内部也需要根据游览的目标规划最优的参观线路，在游览过程中还可以给出相应的景点介绍和场所查询；当进入景区或者学校后，学生可以输入多个目标景点或者场所信息，系统会为学生规划从当前位置出发，参观多个景点或者场所的最优游学线路。

对于场所查询：

在景区或者学校内部时，选中某个景点或者场所，会找出附近一定范围内的超市、卫生间等设施，并根据距离进行排序；（核心算法为排序）

可以通过选择类别对结果进行过滤；

可以由用户输入类别名称查找某个地点附近的服务设施，并根据距离进行排序；

**游学后**：

对于游学日记管理：

学生游学过程中或者游学结束时可以撰写游学日记，通过文字的方式记录游学内容；

需要对所有学生的游学日记进行统一的管理；

学生可以根据浏览和查询所有学生的游学日记，游学日记的浏览量即为该日记的热度，每位同学浏览完可以对游学日记进行评分；

学生在浏览所有游学日记时，可以按照日记热度、评价和个人兴趣进行推荐，推荐算法基础要求为排序算法，可以根据热度和评分进行排序；

学生可以输入游学目的地，对目的地相关的游学日记根据热度和评分进行排序；学生可以输入游学日记的名称进行精确查询；

可以按日记内容进行全文检索

可以对游学日记进行压缩存储

**选做部分**：

（选做1）设计导航功能的图形界面，包括地图展示和输出路径展示；

（选做2）室内导航策略：模拟教学楼的结构和景区内博物馆等建筑物的内部结构，进行室内导航，包括大门到电梯的导航、楼层间的电梯导航和楼层内到房间的导航；

（选做4）美食推荐，在选中游览景点和学校后，可以按照用户选择的热度、评价和距离进行排序，并根据菜系进行过滤；（核心算法为排序算法，考虑到用户通常只看前10个美食，要求不经过完全排序可以排好前10的美食）；可以输入美食名称、菜系、饭店或窗口名称等进行查询，查询结果有多项时，可以对查询结果按照热度、评价和距离进行排序。

可以根据所拍照片和游览经历生成游学日记；文本搜索，高效查询

**数据要求：**

景区和校园数量至少200个，景区和校园内部可以一致；

景区和校园内建筑物（景点、教学楼、办公楼、宿舍楼）数不少于20个；其它服务设施不少于10种（商店、饭店、洗手间、图书馆、食堂、超市、咖啡馆等），数量不少于50个；

建立景区和校园内部道路图

包括各种建筑物、服务设施等信息；

边数不能少于200条（尽量接近真实景区和校园）；

系统用户数不少于10人；

多用户单并发：系统基础要求为多用户，单并发系统；系统需要能够支持多个用户登录使用，但是同时只会有一位用户使用；也就是单机版系统，多位用户可以在同一台PC上依次登录使用；

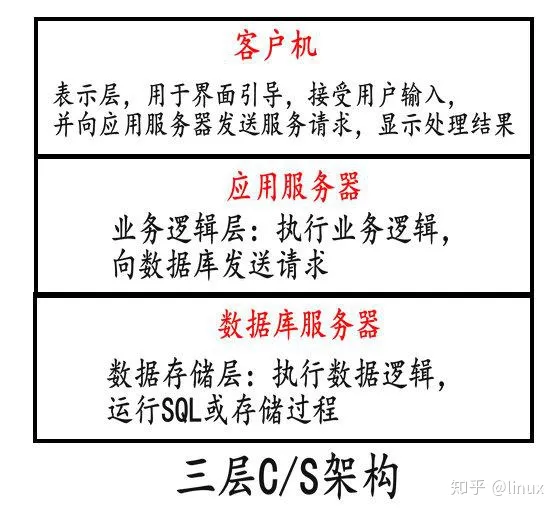
**二、项目进度表：**

|  |  |
| --- | --- |
| 时间 | 项目内容 |
| 1-3周 | 明确项目内容及计划 |
| 4周 | 初步学习Springboot和vue相关技术 |
| 5-8周 | 边做边学，基本实现所有基本功能 |
| 9-12周 | 完善所有基本功能，并实现选做功能 |
| 13-14周 | 并实现调试 |

期中复习前完成大部分核心内容，不妨碍期中复习和期末复习，第三周确定项目计划，第四五，六，周完成

**三、架构设计**

整体架构上，选择 B/S 架构，采用前后端分离的方式进行合作开发，前端部分采用目前较为流行的 Vue 框架进行开发，并调用后端提供的 API 进行数据交互；后端采用了Spring boot+Spring Cloud Alibaba微服务架构



**模块划分：**

路由（Router）：使用 Vue Router 进行页面导航和路由管理，将路由配置拆分为多个文件，可以使用路由守卫进行权限控制和登录验证。

UI 组件库（Vuetify）: 基于 Vuetify 2 组件库构建用户界面，使用 Vuetify 提供的组件、样式和布局来设计和构建应用程序的用户界面。

Ajax 请求封装（Axios）: 使用 Axios 库来处理与后端的 AJAX 请求，可以创建通用的请求封装，设置统一的请求拦截器和响应拦截器，处理请求错误和统一的返回数据格式。

静态资源管理: 将图像、字体等静态资源放置在 assets/目录下，可以使用相对路径引用，或在构建过程中进行打包和处理。

**开发平台-开发语言：**

语言java和vue

1. **项目分工**
2. 肖麒：负责后端开发
3. 陈柳杨：负责后端开发
4. 张一瑞：负责前端开发