

设计报告

需求分析

数据要求

- 至少 200 个校园和景点，校园内建筑物数目不少于 20 个，其他服务设施不少于 10 种，数量不少于 50 个
- 建立景点和校园的内部道路图，包括建筑物和服务设施等信息，边数不少于 200 条
- 系统用户数不少于 10 人

功能要求

游学推荐

- 按照游学热度，评价和个人兴趣进行推荐
- 学生可以输入景点的名字、类别、关键字等进行查询，查询结果不止一项时还可以按照推荐算法进行排序

路线规划

- 学生可以输入一个或多个目标景点或场所信息，系统给出从当前位置到目标位置的最佳路线（根据用户需求提供多条路线，如用时最短，距离最短，途径景点最多）

场所查询

- 选定一个学校或景区内部的场所，可以找出附近的服务设施如超市，卫生间等，并根据距离进行排序和类别检索
- 可以通过选定类别来超出某个地点附近的服务设施，根据距离排序

游学日记管理

- 学生可以通过文字的方式撰写游学日记，由系统统一管理
- 学生可以根据浏览量，评论量，评分高低等来对日记排序，浏览完日记后可以发表评论或者评分
- 学生可以通过日记名，景点名，作者名等查询日记
- 可以对日记内容全文检索
- 可以对日记压缩存储

图形化界面（选做）

实现系统和导航的图形化界面

室内导航（选做）

模拟教学楼和景点建筑物内部的结构，实现室内导航，如大门到电梯，楼层间电梯和某一房间等

考虑交通工具的最短时间策略（选做）

- 校园内可以选择自行车或步行，自行车只能通过自行车道路
- 景区内可以选择电瓶车或步行，电瓶车有固定路线和固定的发车时间
- 在考虑不同拥挤度的情况下实现用时最短（可以多种交通工具混合）

更多推荐算法（选做）

采用基于内容推荐、协同过滤推荐等算法对景点、学校、游学日记进行推荐

动画生成（选做）

使用 AIGC 算法根据拍摄的景点和学校照片进行动画生成

人员分工

前端网页和图形化开发：宋健

后端服务开发：李宇星，蔡逸文

时间安排

- 第四周：各模块接口和数据定义
- 第五周：推荐和搜索模块实现
- 第六周：导航和日记模块实现
- 第七周：编辑测试用例，测试系统
- 第八周：用户管理模块和前端接口实现
- 第九周及以后：前端网页开发和图形化实现

系统设计

Server

用户管理模块

存储用户信息（账号密码，使用历史）

实现登录注册操作

数据管理模块

景区的数据结构

- 建筑群
- 步行道路群
- 公交（动车）道路群
- 人气
- 评论
- 评分

建筑的数据结构

- 位置坐标
- 名称
- 功能

日记的数据结构

- 日记名
- 日期
- 内容
- 评论
- 评分

业务模块

推荐模块

使用排序算法根据待排序对象的某个属性进行排序，默认推荐使用协同过滤为用户个性化推荐

排序算法：使用快速排序算法，可以对任意对象的某个属性为关键字进行排序，定义如下：

Obj quickSort(objArray 排序对象数组,Key 排序关键字,Mode 排序模式(升序降序))

返回一个新的排序好的对象数组

导航模块

根据用户不同需求找出最短路径（费用为时间，价格，建筑距离的函数）

拥挤度计算：**根据不同道路的人数计算拥挤度**

查询模块

使用查找算法、协同过滤、基于内容的推荐算法等，根据用户需求按关键词查找

日记模块

采用推荐和查找模块给用户推荐日记，查找指定日记

日记的评论与评分：每位用户的评论都会保存在对应的日记里，评分会显示平均评分

前端接口模块

```
api/getPositon {
  req:{
    建筑名:string
  }
  return:{
    经纬度:double[]
  }
}
```

```
api/getBuildings {
  req:{
    经纬度:double[]
  }
  return:{
    建筑:object[]
  }
}
```

```
api/getComment{
  req:{
    建筑名:string
  },
  return:{
    评论:object[]
  }
}
```

```
api/getUserInfo{
  req:{
    id:string
  },
  return:{
    info:object
  }
}
```

```
api/getPath{
  req:{
    from:string,
    to:string
  },
  return:{
    edges:string[]
  }
}
```

Browser/Client

前端导航模块

通过server/api获取最佳路径,并显示

地图显示模块

使用leaflet+OSM显示地图