

程序设计实习大作业

燕园地图信息管理系统

没有BUG全都队
陈力川 刘长笑 孙铭泽

一、程序功能介绍

1. 登录注册

用户可按照自定义的用户名与密码进行注册，并用已经注册过的用户名和密码登录程序。

2. 主页面地图缩放

主页面展示燕园地图，并随页面等比例缩放。

3. 主页面地图景点搜索显示

在主页面搜索边框通过输入关键字查找对应景点，并框选显示景点按钮。

4. 景点介绍

在主页面地图上各个景点位置标有对应景点按钮，点击后选择进入详情页面查看景点介绍。

5. 景点评论展示与上传

点击主界面景点按钮后选择进入评论区，可编辑评论并发送，显示所有用户对该景点的评论。

6. 景点收藏

点击主界面景点按钮后选择进入评论区，点击右上角星号收藏该景点。

7. 路线推荐与介绍

主界面设有游览燕园的推荐路线，点击路线后显示对应介绍。

二、主要模块和类设计

1. 主要模块

用户认证模块 (MainWindow, Signup)
地图展示模块 (Map)
景点交互模块 (ClickableRectItem, ClickablePathItem)
详情展示模块 (DetailWindow, detail_or_review)
数据存储模块 (SQLite数据库)
评论和收藏模块 (Review)

2. 核心类详细设计

2.1 用户认证模块

2.1.1 MainWindow类

用于处理用户登录逻辑和界面展示

详细设计:

UI组件:

label_image: 背景图片展示

lineEdit_username: 用户名输入框

lineEdit_password: 密码输入框

btn_signin: 登录按钮

btn_signup: 注册按钮

核心方法:

on_btn_signin_clicked():

执行SQL查询验证用户凭据

使用自定义样式的QMessageBox显示登录结果

成功登录后创建Map窗口并关闭当前窗口

on_btn_signup_clicked():

创建Signup窗口并关闭当前窗口

2.1.2 Signup 类

处理用户注册逻辑和界面展示

详细设计:

UI组件:

label_image: 背景图片展示

lineEdit_username: 用户名输入框

lineEdit_passwd: 密码输入框

lineEdit_surepasswd: 密码确认框

pushButton: 确认注册按钮

btn_return: 返回登录按钮

核心方法:

on_pushButton_clicked():

检查两次密码输入是否一致, 执行SQL插入操作, 使用自定义样式的
QMessageBox显示注册结果

on_btn_return_clicked():

返回登录窗口

2.2 地图展示模块

Map 类

校园地图展示、景点标记和交互

详细设计:

核心组件:

m_scene: QGraphicsScene对象, 管理所有图形项

m_view: QGraphicsView对象, 显示场景内容

m_mapItem: 地图图片项

m_spotItems: 存储所有景点标记

m_routeItems: 存储所有路线

关键方法:

loadMap():

加载地图图片资源, 设置场景矩形与图片大小匹配

addSpot():

创建ClickableRectItem实例, 设置位置和详情信息, 连接点击信号到槽函

数

addRoute():

使用贝塞尔曲线创建平滑路径, 创建ClickablePathItem实例, 存储原始路

径数 据用于缩放

resizeEvent():

动态调整地图缩放比例, 更新景点和路线位置, 处理不同窗口大小的显示

策略

setupSearchUI():

创建搜索面板和输入框, 连接搜索信号到槽函数

performSearch():

遍历景点名称进行匹配, 显示搜索结果标记, 自动居中显示找到的景点

2.3 景点交互模块

2.3.1 ClickableRectItem 类

可点击的圆形景点标记

详细设计:

继承关系: QObject + QGraphicsEllipseItem (支持信号槽)

成员变量:

m_name: 景点名称

m_detail: 景点详情

核心方法:

构造函数: 设置初始大小、颜色和悬停事件

mousePressEvent(): 发射clicked信号

hoverEnterEvent(): 改变颜色和光标形状

hoverLeaveEvent(): 恢复原始颜色

信号:

clicked(QString, QString): 传递景点名称和详情

2.3.2 ClickablePathItem 类

可点击的路线

详细设计:

继承关系: QObject + QGraphicsPathItem (支持信号槽)

核心方法:

构造函数: 设置路径和详情

mousePressEvent(): 发射clicked信号

信号:

clicked(QString): 传递路线详情

2.4 详情展示模块

2.4.1 DetailWindow 类

显示景点或路线的详细信息

关闭按钮

2.4.2 detail_or_review 类

扩展的详情展示界面(包含评论功能)

2.5 数据存储模块

SQLite数据库设计

表结构:

user表:

username (主键): 用户名

Password: 密码

favorites表:

id (自增主键)

username (外键)

spot_name: 景点名称

collected: 收藏状态

comments表:

id (自增主键)

username (外键)

content: 评论内容

spot_name

2.6 评论和收藏模块

Review类

进行景点评论发送和展示，以及经典收藏

详细设计：

成员变量：

commentModel: 评论数据模型

currentSpot: 当前景点名称

starButton: 收藏按钮

Stylesheet: 按钮样式表

Ui组件：

listView: 评论展示框

textEdit: 评论编辑框

pushButton: 发送按钮

核心方法：

explicit review():

构造函数：初始化ui界面组件，进行窗口页面布局，初始化收藏按钮，创建收藏按钮样式表，创建并初始化评论列表模型并与listView绑定用于展示评论，链接信号槽；

void on_pushButton_clicked():

发送按钮响应函数，将评论内容输入数据库，更新评论列表展示和数据模型；

void if_collected():

收藏按钮响应函数，从数据库查询当前用户和景点的收藏状态，更改ui界面显示以及数据库中的对应收藏状态；

void update_collect():

用于每次打开review界面后从数据库查询当前收藏状态，设置收藏按钮样式；

void loadComments(const QString &spotName):

从数据库中加载指定景点的评论，加入commentModel中；

三、小组分工情况

陈力川：数据库设计，评论与收藏界面设计与功能实现

刘长笑：登录注册和登录后的界面及缩放功能，景点按钮的设计与功能实现，以及项目框架的搭建

孙铭泽：景点路线的位置与后续部分交互功能以及搜索功能的实现

四、总结与反思

在这次程序设计与实现过程中很多设想中的功能，例如通过动画形象进行景点介绍和景点的评分功能，没能实现，还能更进一步完善功能使得程序功能更加多样，实用性更广；登陆注册功能使用的是SQLite数据库，无法同步多设备的账号信息；在景点路线描述及搜索功能这类用户可以交互的功能上应该加强引导，便于直观的使用和理解。