**需**

**求**

**分**

**析**

**文**

**档**

**校园导游系统**

专 业 名 称 ：计算机科学与技术

课 程 名 称 ：软件系统实践

学 生 1姓名 ：鲍明颉

学 生 2姓名 ：王博

指导教师 ：李文海老师

2024.7

**0.问题描述**

设计一个校园导游系统，为来访的客人提供导游咨询服务，如景点查询、路径查询等。

【基本要求】

（1）设计所在学校的校园平面图，所含景点不少于10个。

（2）为来访客人提供图中任意景点相关信息的查询。

（3）为来访客人提供图中任意景点的问路查询，即查询任意两个景点之间的一条通路，包括任意两个景点之间的所有路径，任意两个景点之间的最短路径等。

（4）扩展内容：提供景点和道路的扩充及撤销功能。

**1.引言**

**1.1编写目标**

本需求分析文档旨在详细描述校园导游系统的功能性和非功能性需求，以指导开发团队进行设计和实现。文档将涵盖系统的各个方面，包括用户需求、系统功能、界面需求和实现约束等。

**1.2读者对象**

本需求分析文档的读者对象涵盖了项目开发和管理过程中涉及的各个角色和利益相关者，具体包括但不限于以下群体：

系统开发团队：包含前端开发人员、后端开发人员。负责用户界面和系统后台的设计。文档中关于界面需求和算法需求的部分将为他们的工作提供指导。

技术文档编写者：根据本文档编写用户手册和技术文档。

测试团队：负责系统的功能测试、性能测试、质量保证工作，确保系统符合需求文档中的描述和用户期望。需求文档将为他们编写测试用例和进行测试提供依据。

维护和运营团队：负责系统的部署、维护和运营。需求文档将帮助他们理解系统的结构和功能，以便更好地进行系统的维护和管理。

学校管理人员：负责系统的内容更新和管理。需求文档将帮助他们了解系统的功能，以便进行有效的管理和使用。

来访客人：虽然来访客人不是文档的直接读者，但他们的需求和使用体验是系统设计的核心，文档中的需求描述将确保系统满足他们的需求。

验收教师：负责对文档进行验收和评价。

**1.3文档描述**

本文档是校园导游系统的需求分析文档，它详细阐述了系统的功能需求、性能需求、可靠性需求、可用性需求、安全需求、界面需求。同时本文档还阐述了进度要求和交付要求，方便设计者把握整体进度。通过详细描述以上各部分内容，本需求分析文档将为校园导游系统的设计、开发、测试和维护提供全面、清晰的指导，确保系统能够满足用户需求并成功实施。

本文档的编写基于对校园导游系统的深入研究和分析，所有需求都是基于实际用户场景和业务目标而提出的，旨在确保系统的实用性和有效性。

**1.4术语定义**

在本需求分析文档中，涉及到一些专业术语和缩写。为了确保所有读者能够准确理解文档内容，现对相关术语进行定义：

景点：校园内具有特定功能或特色的地点，如图书馆、教学楼、操场、宿舍等。

邻接矩阵：一种用于存储图结构的方法，采用矩阵形式存储图中顶点及其边的信息。在邻接矩阵中，矩阵的行和列分别代表图的顶点，矩阵元素表示对应顶点之间是否有边及边的权重。

弗洛伊德算法（Floyd Algorithm）：一种用于计算图中所有顶点对之间的最短路径的算法，即求任意两个顶点之间的最短路径。

导游：在校园导游系统中，导游指的是系统提供的自动化导览服务，通过查询和路径规划帮助来访客人了解校园景点及其之间的路径。

用户：使用校园导游系统的人员，包括来访客人和学校管理人员。来访客人主要使用系统查询景点和路径信息，学校管理人员则负责更新和维护系统中的景点和路径数据。

顶点：图结构中的一个节点，表示校园内的一个景点。

边：图结构中连接两个顶点的线段，表示两个景点之间的路径。边可以有权重，表示路径的长度或其他度量。

权重：在图结构中，边的属性之一，表示两个顶点之间路径的长度或其他度量。

图：由顶点和边组成的数据结构，用于表示景点及其之间的可达关系。

通过对上述术语的定义，确保读者在阅读本需求分析文档时能够准确理解相关概念，避免因术语不清导致的误解。

**1.5参考文献**

* 《软件需求工程》
* 《用户界面设计指南》
* 《软件性能测试与优化》

**2.软件系统概述**

**2.1软件产品概述**

校园导游系统是一款为来访客人和校园管理人员提供综合导览和信息查询服务的软件系统。其主要目的是提升来访者对校园的认知和体验，同时提高校园管理效率。系统通过图形界面展示校园平面图，并通过强大的路径规划和信息查询功能，为用户提供便捷的导览服务。系统具备以下核心功能：

交互式地图：系统以图形化方式展示校园平面图，包括各个主要景点和连接它们的道路。用户可以在地图上点击放，方便地浏览和查找景点信息。

景点信息查询：用户可以查询任意景点的基础信息，如景点名称、代号、简介、坐标等。此外用户可以查询景点内以及附近的具体设施，如卫生间、食堂等，了解其功能和使用情况。

路径查询：用户可以查询任意两个景点之间的一条通路，包括路径的长度和经过的节点。系统提供多个路径选项，供用户选择最合适的路线。

最短路径查询：通过弗洛伊德算法（Floyd），为用户计算并推荐两个景点之间的最短路径。

景点和道路的扩充及撤销：软件管理员可以在系统中添加、修改和删除景点信息，确保数据的准确性和及时性；可以添加、修改和删除景点之间的道路信息，维护校园平面图的完整性。

用户界面：系统设计了直观、简洁的用户界面，方便用户快速上手和操作。

校园导游系统不仅面向来访客人，帮助他们轻松了解校园布局和找到所需景点；也为校园管理人员提供了高效的管理工具，确保校园信息的准确性和及时更新。通过提供便捷的导览服务和信息管理功能，系统将显著提升校园的服务质量和管理水平。

**2.2用户特征**

校园导游系统的用户主要分为来访客人和学校管理人员。不同用户群体有不同的需求和特征。以下是对这两类用户特征的详细描述：

**来访客人**

多样化背景：来访客人包括家长、潜在学生、校友、学术交流人员、访客等，具有不同的背景和需求。

短期停留：大部分来访客人在校园的停留时间较短，他们希望在有限的时间内尽可能多地了解校园。

导航需求强烈：由于对校园环境不熟悉，来访客人需要明确的路线指引和景点介绍。

信息获取便捷：希望能够快速、便捷地获取所需的景点信息、设施信息和路线信息。

使用便捷性：对于系统的操作要求简单易用，即使是第一次使用系统，也能快速上手。

**学校管理人员**

熟悉校园环境：学校管理人员对校园的布局和各景点的功能非常熟悉，负责维护和更新系统中的信息。

信息管理需求：管理人员需要及时更新系统中的景点和路径信息，确保数据的准确性和时效性。

操作频繁：由于需要经常进行信息维护，管理人员需要一个高效、稳定的管理界面。­

通过描述用户特征，校园导游系统将更好地满足不同用户群体的需求，提供优质的导览和信息管理服务。

**2.3设计和实现约束**

在设计和实现校园导游系统的过程中，需要考虑多方面的约束条件，以确保系统能够按预期功能、性能和安全要求实现。以下是详细的设计和实现约束：

技术约束：系统的开发主要使用C++语言。前端开发使用EasyX框架。选择txt文本文件存储景点和路径数据。使用简易地图作为地图展示并实现交互功能。系统需实现Floyd算法，以满足路径查询的功能需求。开发过程中使用Git进行版本控制，确保代码管理和协同开发的高效性。

性能约束：系统需能够高效处理和存储大量景点和路径数据，确保数据操作的实时性和可靠性。

可扩展性约束：系统设计需采用模块化架构，确保各功能模块的独立性和可扩展性，便于后期功能扩展和维护。数据设计需考虑数据量增长和业务扩展的需求。

可维护性约束：系统开发需遵循统一的代码规范和最佳实践，确保代码的可读性和可维护性。系统需提供完整的开发文档、用户手册和维护手册，便于开发团队和用户的理解和使用。系统需进行全面的功能测试、性能测试和安全测试，确保系统的可靠性和稳定性。

时间约束：根据项目计划，系统需要在特定的截止日期前完成开发和验收。

通过遵循上述设计和实现约束，校园导游系统将能够高效、稳定、安全地实现其预期功能，满足用户需求并提供优质的导览服务。

**2.4假设和依赖**

在设计和实现校园导游系统时，我们做出了一些假设，并且系统的某些功能和性能依赖于特定的条件和外部因素。这些假设和依赖有助于明确系统的边界和前提条件，确保系统的开发和运行顺利进行。以下是详细的假设与依赖描述：

**假设**

环境稳定：假设系统的用户（包括来访客人和学校管理人员）能够在稳定的环境下访问系统，确保系统的正常运行和数据传输的顺畅。

硬件设备支持：假设用户的硬件设备能够正常启动软件，并且具有足够的计算和存储能力，以便运行系统的前端和后端功能。

用户具备基本的计算机操作能力：假设系统的用户具备基本的计算机操作能力和互联网使用经验，能够轻松地进行景点查询、路径查询和其他交互操作。

校园环境相对稳定：假设校园的景点和道路布局在短期内不会发生重大变化，系统中的景点和路径信息在大多数情况下是准确和可靠的。

**依赖**

地图图片提供方：系统依赖于第三方地图提供的地图展示。如果地图图片出现问题，将影响实际的使用。

第三方库和框架：系统依赖于多个第三方库和框架（如EasyX）进行前端和后端开发。这些库和框架的更新和兼容性将影响系统的维护和升级。

操作系统：系统依赖于用户的操作系统和浏览器的兼容性，确保在不同平台上的一致性和功能正常。系统需兼容主流操作系统（如Windows）

用户数据输入：系统依赖于学校管理人员准确、及时地输入和维护景点和路径数据，确保系统提供的信息准确、可靠。

通过明确这些假设与依赖，可以更好地理解校园导游系统的运行环境和前提条件，并在开发和维护过程中采取相应的措施，确保系统的稳定性和可靠性。

**3.功能性需求描述**

**3.1软件功能概述**

校园导游系统旨在为来访客人和校园管理人员提供全面的导览和信息管理服务。系统主要功能涵盖了校园景点一览、校园景点速查、、景点信息查询、景点附近设施查询、路径查询、景点和道路管理等多个方面。以下是软件功能的详细概述：

校园景点一览/景点速查：提供景点速查功能，用户输入景点全名，即可显示景点图片。

景点信息查询：系统以图形化方式展示校园平面图，包括各个主要。用户可以在地图上点击景点对应的按钮，方便地浏览和查找景点信息。点击按钮后，将显示景点的基础信息，包括景点名称、代号、坐标，介绍等。

景点附近设施查询：用户查询景点信息后，即可查询景点内部或附近的食堂，卫生间等设施。

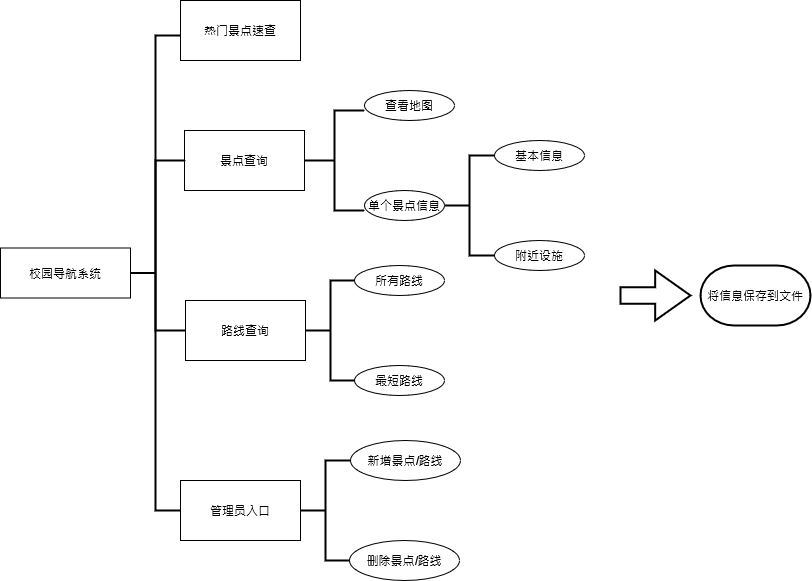
路径查询：用户可以查询任意两个景点之间的一条通路，包括路径的长度和经过的节点。系统提供多个路径选项，供用户选择最合适的路线。通过弗洛伊德算法（Floyd），为用户计算并推荐两个景点之间的最短路径。

景点和道路管理：学校管理人员可以在系统中添加、修改和删除景点信息和道路信息，确保数据的准确性和及时性。

通过详细描述上述各项功能，校园导游系统将能够全面满足来访客人和学校管理人员的需求，提供高效、便捷的导览和信息管理服务。

**3.2 产品结构图**

产品结构图如下图



**4.非功能性需求**

**非功能性需求定义了系统必须具备的质量特性，以确保系统在功能实现之外，还能在性能、可靠性、可用性、安全性等方面满足用户的期望和需求。以下是校园导游系统的非功能性需求的详细描述：**

**性能需求：系统需在较快时间响应用户的查询请求，以确保用户体验的流畅性。系统需能够在较快时间处理景点信息查询和路径计算等操作，以保证数据处理的实时性。**

**可扩展性需求：系统需采用模块化架构，确保各功能模块的独立性和可扩展性，便于后期功能扩展和维护。数据文件设计需考虑数据量增长和业务扩展的需求，以应对未来的数据增长。**

**可维护性需求：系统开发需遵循统一的代码规范和最佳实践，确保代码的可读性和可维护性。系统需提供完整的开发文档、用户手册和维护手册，便于开发团队和用户的理解和使用。**

**通过满足上述非功能性需求，校园导游系统将能够在性能、可靠性、安全性、可用性等方面达到用户的期望，提供优质的导览和信息管理服务。**

**5.界面需求**

主界面：提供便捷的导航菜单，包括首页、景点查询、路径查询、管理功能等主要模块的入口。在显著位置提供搜索栏，支持用户通过关键词快速查找景点和设施。主界面中央展示校园标志性建筑，美观大气。

景点查询界面：提供校园简洁版地图。点击地图中的任意景点，显示该景点的详细信息，包括名称、序号、坐标、简介、图片等。

路径查询界面：提供地图及景点按钮供用户选择起点和终点， 选择完成后显示两点间的所有路径，并高亮最短路径，以供用户选择最适合的路径。

管理功能界面：提供添加、编辑、删除景点和道路的功能界面。

视觉设计；界面设计风格一致，使用统一的配色方案、字体和按钮样式，确保界面美观统一。界面布局简洁，信息层次分明，避免过多的文字和复杂的操作步骤。

**6.接口定义**

**查看地图接口**

名称：ShowMap

输入参数：无

输出结果：校园地图数据

**规划路线接口**

名称：FindShort

输入参数：起点（Point）、终点（Point）

输出结果：路线规划结果

**查看景点内容接口**

名称：ShowInfo

输入参数：无

输出结果：景点内容

**7.进度要求**

7月5日前完成需求分析与概要设计；

7月14日前完成详细设计和程序编码；

7月17日进行迭代与验收。

**8.交付要求**

提供完整可运行的校园导游系统运行程序，提供完整的源代码，包括所有开发过程中的代码和注释。

源代码应具有良好的可读性和可维护性。

提供系统的测试报告，包括单元测试、集成测试和系统测试的结果。测试报告应详细记录测试用例、测试结果和发现的问题。

**9.何种形式来交付**

通过答辩、现场演示程序文件和各种文档并随即通过样例来验证程序正确与否，并根据需求分析来检查需求是否全部满足。

**10.验收要求**

功能性验收：系统的所有功能必须按照需求分析文档中描述的规格正常工作。用户界面应直观易用，符合用户体验设计原则。

性能验收：系统的响应时间等性能指标应符合需求分析文档中规定的标准。

文档验收：所有交付的文档应完整、准确、易于理解。源代码应包含足够的注释，方便未来的维护和扩展。

验收测试：教师应对系统进行测试，确认系统满足文档需求。