

**项目计划书**

|  |  |
| --- | --- |
| 项目名称： | 交大实验室危险气体  实时监测与预警系统 |
| 项目计划书撰写人： | 何恩杰 |

目录

[**项目开发计划** 3](#_Toc198132044)

[**1.引言** 3](#_Toc198132045)

[**1.1编写目的** 3](#_Toc198132046)

[**1.2 项目背景** 4](#_Toc198132047)

[**1.3 定义** 4](#_Toc198132048)

[**1.4 参考文献** 4](#_Toc198132049)

[**2 项目概述** 4](#_Toc198132050)

[**2.1 工作内容** 4](#_Toc198132051)

[**2.2 主要参加人员** 4](#_Toc198132052)

[**2.3 产品** 4](#_Toc198132053)

[**2.4 验收标准** 5](#_Toc198132054)

[**2.5 本计划的批准者和批准日期** 5](#_Toc198132055)

[**3 实施计划** 5](#_Toc198132056)

[**3.1 工作任务的分解与人员分工** 5](#_Toc198132057)

[**3.2 联系人** 5](#_Toc198132058)

[**3.3 进度** 5](#_Toc198132059)

[**3.4 预算** 6](#_Toc198132060)

[**3.5 关键问题** 6](#_Toc198132061)

[**4 支持条件** 7](#_Toc198132062)

[**4.1 计算机系统支持** 7](#_Toc198132063)

[4.2 需由用户承担的工作 7](#_Toc198132064)

[4.3 由外单位提供的条件 7](#_Toc198132065)

[**5 专题计划要点** 7](#_Toc198132066)

[5.1 开发人员培训计划 7](#_Toc198132067)

[5.2测试计划 8](#_Toc198132068)

[5.3安全保密计划 8](#_Toc198132069)

[5.4质量保证计划 8](#_Toc198132070)

[5.5配置管理计划 8](#_Toc198132071)

[5.6用户培训计划 8](#_Toc198132072)

[5.7系统安装计划 9](#_Toc198132073)

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **项目开发计划**  **1.引言**  **1.1编写目的**  本《项目开发计划》旨在明确"高校实验室危险气体实时监测与预警系统"的开发流程、任务分工、进度安排及质量控制标准。本计划基于《可行性分析报告》的核心需求，结合软件工程方法论与UML（统一建模语言）工具，指导开发团队完成需求分析、系统设计、编码实现及测试验证的全过程。通过规范化文档编写和模型构建，确保项目在技术可行性、时间管理、团队协作等方面符合预期目标，并为后续开发提供可追溯的参考依据。预期读者为项目指导教师、开发团队成员及系统最终用户。  **1.2 项目背景**  随着高校实验室安全管理要求的提升，化学、生物等学科实验室的危险气体泄漏风险亟需智能化管控手段。北京交通大学计算机学院提出由大二学生团队开发"实验室危险气体实时监测与预警系统"。本项目聚焦实验室气体安全监测的痛点，通过物联网传感器网络实时采集气体浓度数据，结合B/S架构实现多层级用户权限管理、动态预警推送及历史数据分析功能。依托Java EE技术栈与UML建模工具完成系统开发，项目周期为2024年3月至2024年12月。  **1.3 定义**  LGS：Laboratory Gas Monitoring System 实验室危险气体实时监测与预警系统。  SQL Server：所用的数据库管理系统。  Eclipse：所用的开发工具。  UML：统一建模语言，用于系统分析与设计的标准化建模工具  B/S架构：浏览器/服务器架构，用户通过Web浏览器访问系统功能  传感器节点：部署于实验室的气体浓度采集终端（含硫化氢、甲烷等传感器）  预警阈值：触发系统警报的气体浓度临界值（分一级/二级警报）  **1.4 参考文献**  （1） 陈明.软件工程实用教程.北京：电子工业出版社，2006年1月  （2） 张海藩.《软件工程导论》.人民邮电出版社.2006年1月  （3） 潘孝铭.《软件文档编写》.高等教育出版社.2004年8月  （4） 罗先文.《软件工程实物》.重庆大学出版社.2005年3月  （5） 《图书管理系统可行性分析报告》。  **2 项目概述**  **2.1 工作内容**  该系统应该具有对实验室多种危险气体的实时监测和管理功能，并能够存储气体浓度数据、报警记录、实验室信息、用户信息，并具备权限管理的功能。该系统能显著降低实验室安全管理人员的日常工作负担，并提供气体安全报表，为实验室的安全管理提供辅助决策支持。  在本项目开发过程中需要进行可行性分析、制定项目开发计划、软件需求、软件分析设计、软件实现、软件测试以及相应文档的编写工作。  **2.2 主要参加人员**  杨敏龙、何恩杰  **2.3 产品**  **2.3.1 程序**  所用的编程语言为JAVA。  本项目将采用Java语言进行全栈开发，技术栈组成如下：  后端框架：Spring Boot 3.1.5（集成Spring Security实现权限控制）  前端框架：Vue 3.3 + Element Plus（实现响应式管理界面）  UML建模工具：StarUML 5.0（生成符合OMG标准的类图、用例图、时序图）  持续集成：Jenkins Pipeline（实现每日构建与自动化测试）  **2.3.2 文件**  向用户提交的文件名称LMS.WAR，将该包放在tomcat的webapps中可自动解压部署。  **2.3.3 服务**  向用户提供的服务为需求分析文档和用户手册，用户可从中得到关于软件使用方面的信息。  **2.3.4 非移交的产品**  所有文件都应上交项目委托单位某大学。  **2.4 验收标准**  对于上述这些产品和服务，按照企业产品要求进行验收。  **2.5 本计划的批准者和批准日期**  本计划的批准者为北京交通大学，批准日期为2025年4月20日。  **3 实施计划**  **3.1 工作任务的分解与人员分工**  可行性分析：杨敏龙  项目开发计划：何恩杰  软件需求：杨敏龙  软件分析设计：何恩杰  编码：杨敏龙  测试与维护：何恩杰  **3.2 联系人**  本小组共有2个人，何恩杰作为本项目的联系人，负责本项目和委托单位的信息沟通。  **3.3 进度**  可行性分析：4月25日－5月1日 标志：提交可行性分析报告  项目开发计划：5月2日－5月6日 标志：提交项目开发计划  需求分析：5月7日－5月21日 标志：完成需求分析报告  软件设计：5月22日－5月27日 标志：完成软件分析与设计文档  软件实现：5月28日－6月10日 标志：代码编写全部完成  测试与实施：6月11日－7月25日 标志：完成软件测试，可以投入使用  项目名称 交大实验室危险气体实时监测与预警系统  表2.1 项目进度表   |  |  |  |  | | --- | --- | --- | --- | | 时间（月）  任务 | 5 6 7 | 所需工作量  参加人数 | 所需资源 | | 可行性分析 |  | 0.23人-月  1人 | 实验室安全标准文档 | | 项目开发计划 | 0.17人-月  1人 | 项目管理模板 | | 需求分析 | 1.00人-月  2人 | 用户访谈记录 | | 软件分析与设计 | 0.40人-月  2人 | UML工具 | | 软件实现 | 0.93人-月  2人 | 开发环境、版本控制 | | 测试与实施 | 3.00人-月  2人 | 测试设备、用户培训手册 | | 总计工作量 | 5.73人-月 | 2人 |  |   **3.4 预算**  人员成本：500元/人-月，共计：500\*5.73=2865元  项目所需要的工作量（人-月）如下表所示：  表3.2 项目工作量   |  |  | | --- | --- | | 任务 | 需要工作量/（人-月） | | 可行性分析 | 0.23 | | 项目开发计划 | 0.17 | | 需求分析 | 1.00 | | 软件设计 | 0.40 | | 软件实现 | 0.93 | | 测试 | 3.00 | | 共计 | 5.73 |   其它经费：  办公费用：700元 差旅费：无  机时费：无 资料费：1000元  设备费：（学校实验室提供） 专用设备租金：无  总计费用支出：4565元  **3.5 关键问题**  最主要的是技术方面的问题，即如何通过分析设计、软件实现完成系统需要的功能。其它如空间数据与属性数据的关系、数据库设计、数据结构设计等问题，也起着关键性的作用。  （1）技术架构设计与实现问题   * 问题描述：如何通过合理的分析设计实现系统核心功能（如实时监测、多级报警、可视化展示），并确保高并发、低延迟的技术指标 * 关键挑战：①复杂功能集成：需同时处理传感器实时数据流（20秒/次）、多级报警逻辑（房间→楼栋→学院→校级）、地图可视化（空间数据与属性数据关联）   ②性能与稳定性：需满足200+用户并发操作（如物业负责人查看楼栋数据）及10秒内报警响应  （2）数据关系与数据库设计问题   * 问题描述：如何高效管理多源异构数据（传感器数据、用户权限、报警记录、实验室空间信息） * 关键挑战：①数据规模：每天产生数万条传感器数据（需长期存储）   ②查询性能：需支持多维度组合查询（如“某楼栋过去一周甲烷浓度>100ppm的记录”）  **4 支持条件**  **4.1 计算机系统支持**  （1）服务器端软件选择的具体说明：   * 操作系统：Ubuntu Server 22.04 LTS（长期支持版本） * 数据库管理系统：MySQL 8.0集群（主从复制+ShardingSphere分库分表） * 开发工具：Eclipse。   （2）客户端软件选择的具体说明：   * 操作系统： Windows 2000 Profession 或 Windows 98。 * 浏览器：Chrome 120+ / Edge 110+ * 移动端：Android 12+/iOS 15+（配套预警APP）   4.2 需由用户承担的工作  （1）需求细化与验证：   * 提供各实验室平面图、气体类型清单及历史事故报告 * 参与用户验收测试，对预警阈值、报警推送策略（短信/APP/声光）进行确认   （2）运维支持：   * 指定专职系统管理员（2名），负责日常监控看板维护、权限分配及日志审计 * 定期巡检传感器设备（每月1次），提交设备状态报告   （3）培训计划   * 组织三级培训：基础操作培训（面向实验室成员，2学时）；应急处理培训（面向负责人，4学时）；系统管理培训（面向IT部门，8学时）   4.3 由外单位提供的条件  （1）学院内部资源支持   * 硬件设备：高性能计算集群、物联网实验平台、网络与信息安全实验室 * 软件设备：学院正版软件库，包括Microsoft Imagine订阅（免费获取Windows Server、SQL Server授权）、JetBrains教育许可证、MATLAB/Simulink（用于气体扩散模型仿真）   （2）学校合作单位支持   * 信息中心：提供校园5G专网切片服务，保障实验室楼宇间数据传输低延迟（≤50ms） * 保卫处：共享校园安防系统接口（如摄像头位置信息），实现报警联动   提供应急处理预案模板，指导系统报警升级流程设计   * 资产管理处：协调实验室楼宇弱电改造   **5 专题计划要点**  5.1 开发人员培训计划  （1）目标：确保开发团队掌握系统核心技术栈及安全规范  （2）要点：   * 技术培训：Java微服务架构（Spring Cloud）与物联网协议（MQTT/CoAP）实战、实时数据处理技术、传感器硬件接口开发（STM32嵌入式编程） * 安全规范：代码安全审计、敏感数据加密（AES-256）与脱敏策略 * 培训周期：集中培训：3天、持续学习：每周技术分享会（1小时）   5.2测试计划  （1）**目标**：覆盖功能、性能、安全全维度验证  （2）要点：   * 测试阶段：单元测试：JUnit 5（代码覆盖率≥85%）、   集成测试：Postman模拟多终端并发（200+用户）  压力测试：JMeter模拟10,000传感器节点持续写入  安全测试：渗透测试（Burp Suite）与SQL注入防御验证   * 测试环境：仿真实验室环境（模拟气体泄漏场景）   独立测试服务器（与生产环境隔离）  5.3安全保密计划  （1）**目标**：保障系统数据安全与隐私合规。  （2）要点：   * 数据安全：传输加密：TLS 1.3协议全链路加密   存储加密：MySQL TDE（透明数据加密）  访问控制：RBAC模型（四级角色：平台负责人→学院→物业→房间）   * 物理安全：服务器机房双因素认证（指纹+IC卡）   传感器终端防拆报警（GPS定位+蜂鸣器）  5.4质量保证计划  （1）目标：确保交付系统符合功能需求与性能指标  （2）要点：   * 代码质量：每日代码审查（Pull Request评审）   静态代码分析（Checkstyle + PMD规则集）   * 文档规范：需求文档、设计文档、API文档的版本一致性验证   用户手册通过“小白用户”可用性测试   * 性能基线：数据采集延迟≤20秒   报警响应时间≤10秒  5.5配置管理计划  （1）目标：实现代码、文档、环境版本可控  （2）要点：   * 版本控制：GitLab分支：main（生产）、dev（开发）、feature/\*（功能分支） * 变更管理：需求变更需通过CCB（变更控制委员会）评审   紧急Hotfix流程：从main分支直接修复并打Tag   * 基线管理：里程碑基线：需求确认、设计完成、代码冻结   环境基线：开发、测试、生产环境镜像存档  5.6用户培训计划  （1）目标：确保各级用户熟练操作系统并处理应急事件  （2）要点：   * 分角色培训：   ①实验室成员（2学时），实时数据查看、报警确认操作，移动端APP推送处理（模拟演练）  ②负责人（4学时），多级报警升级流程（从房间→学院→平台），历史数据导出与安全报表生成  ③IT运维人员（8学时），系统监控看板，故障诊断手册   * 培训材料：交互式操作视频、应急场景沙盘演练（如传感器故障模拟）   5.7系统安装计划  （1）目标：实现平滑部署与最小化停机时间  （2）要点：   * 部署步骤：硬件部署：传感器安装（位置校准）、边缘网关配置（IP地址绑定）   软件部署：Docker容器化微服务（Kubernetes集群滚动更新）  数据迁移：历史报警记录导入（MySQL批量脚本）   * 回退方案：出现严重故障时，切换至备用服务器   版本回滚：通过GitLab Tag快速回退至上一稳定版本   * 验收标准：72小时无故障试运行（报警触发成功率≥99.5%）   用户签署《系统上线确认书》  备注：  所有计划需通过学院项目管理办公室（PMO）审批，并与《可行性分析报告》中2.3条款（资源限制）及3.4（预算）对齐。 |