|  |  |
| --- | --- |
| 学院标识组合（方） | **校名** |

**本科生毕业论文（设计）**

基于J2EE的尾矿库坝体安全性能检测系统

**Safety performance testing system of tailings dam based on J2EE**

专 业 **电子信息工程**

姓 名 **李兴成**

学 号 **15033113**

指 导 教 师 **王 博**

完 成 时 间 **2019.6**

独创性声明

本人声明所呈交的论文是我个人在导师指导下进行的研究工作及取得的研究成果。尽我所知，除了文中特别加以标注和致谢的地方外，论文中不包括其他人已经发表或撰写过的研究成果，也不包含为获得商洛学院或其他教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示了谢意。

学位论文作者签名：\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 日期：\_\_\_\_\_\_\_\_\_\_\_\_\_\_

关于论文使用授权的说明

本人完全了解商洛学院有关保留、使用学位论文的规定，即：学校有权保留送交论文的复印件，允许论文被查阅和借阅；学校可以公布论文的全部或部分内容，可以采用影印、缩印或其他复制手段保存论文。

□公开 □保密（\_\_\_\_年\_\_\_\_月） (保密的学位论文在解密后应遵守此协议)

签名： 导师签名： 日期：

基于J2EE的尾矿库坝体安全性能检测系统

**摘 要：**随着科技的不断发展，信息化进程不断加快，互联网传输能力不断增强，现如今已逐步发展为将数据和各类软件聚合存储于“网络云”（大型处理中心）中统一处理，用户只需要一个上网终端即可完成一系列的需求。

设计了一套web尾矿库安全云管理系统，为矿区管理者进行尾矿安全性能检测、日常维护等提供决策支持。系统主要实现的功能有：尾矿库温湿度，浸润线，位移，水位，有毒有害气体，重金属元素数据监测，异常报警，视频监控，企业用户管理，系统信息系统异常记录处理，生成报表等功能。平台使用，前后端分离的开发方式，后端主要采用Java语言开发，主要应用SpringBoot框架，使用mysql + mybatis + jpa的方式进行存储数据，使用POI技术开发报表功能，前端使用了Vue 、Element-ui、 iView、 axios 、 Echarts等前端框架和技术开发用户交互页面。通过这些功能和技术实现一个高效的、稳定的、实用的云管理系统。

**关键字:** J2EE；尾矿库；数据可视化；安全检测；

**Safety performance testing system of tailings dam based on J2EE**

**Abstract:**with the continuous development of science and technology, the information process is accelerating, and the Internet transmission capacity is increasing. Nowadays, it has gradually developed to store data and various kinds of software in the "network cloud" (large processing center) for unified processing. Users only need an Internet terminal to complete a series of requirements.

A safety cloud management system of web tailings pond is designed to provide decision support for mine managers to conduct safety performance testing and daily maintenance of tailings.The main functions of the system are: tailing pond temperature and humidity, infiltration line, displacement, water level, toxic and harmful gas, heavy metal element data monitoring, abnormal alarm, video monitoring, enterprise user management, system information system abnormal record processing, report generation and other functions.The platform USES the development mode of separation of front and rear ends. The back-end is mainly developed by Java language, mainly using SpringBoot framework, using mysql + mybatis + jpa to store data, using POI technology to develop report function, and front-end using Vue, element-ui, iView, axios, Echarts and other front-end frameworks and technologies to develop user interactive pages.Through these functions and technologies to achieve an efficient, stable, practical cloud management system.

**Key words:** J2EE；Tailings；Data visualization；[security](javascript:;) [detection](javascript:;)；

目 录

[1 绪论 1](#_Toc11010825)

[1.1 选题背景及意义 1](#_Toc11010826)

[1.2 尾矿库管理系统概述 1](#_Toc11010827)

[1.3.国内外研究现状 1](#_Toc11010828)

[1.4 基于J2EE的尾矿库安全云管理系统的特点 2](#_Toc11010829)

[1.5 本课题要解决的问题 2](#_Toc11010830)

[2 相关技术与开发、运行环境 2](#_Toc11010831)

[2.1 Vue框架 2](#_Toc11010832)

[2.2 Element-UI组件 3](#_Toc11010833)

[2.2 MyBatis框架 3](#_Toc11010834)

[2.2 spring data jpa框架 4](#_Toc11010835)

[2.4 Druid技术 4](#_Toc11010836)

[2.5 SpringBoot框架 5](#_Toc11010837)

[2.6 ECharts开源技术 6](#_Toc11010838)

[2.6.1 提供丰富的图表类型 6](#_Toc11010839)

[2.6.1 数据驱动特点 6](#_Toc11010840)

[2.7 WebSocket技术 6](#_Toc11010841)

[2.8 POI技术 7](#_Toc11010842)

[2.9基于JWT的Token认证机制 8](#_Toc11010843)

[2.10 core 技术 8](#_Toc11010844)

[2.11 开发工具 9](#_Toc11010845)

[2.11.1 IntelliJ IDEA开发工具 9](#_Toc11010846)

[2.11.2 WebStorm开发工具 10](#_Toc11010847)

[3项目需求分析 11](#_Toc11010848)

[3.1系统的需求分析 11](#_Toc11010849)

[3.3 系统的性能需求 11](#_Toc11010850)

[4 系统的设计 12](#_Toc11010851)

[4.1系统设计的原则 12](#_Toc11010852)

[4.2 数据库的设计 12](#_Toc11010853)

[5 系统的实现 17](#_Toc11010854)

[5.2.1登录模块 17](#_Toc11010855)

[5.2.2 测量中心模块 19](#_Toc11010856)

[5.2.3系统管理模块 23](#_Toc11010857)

[5.2.4视频监控模块 24](#_Toc11010858)

[5.2.5 监控地图模块 24](#_Toc11010859)

[5.2.6用户管理模块 25](#_Toc11010860)

[6 系统的测试 25](#_Toc11010861)

[6.1单元测试 26](#_Toc11010862)

[6.2 OneAPM CPT云压测 26](#_Toc11010863)

[7总结和展望 27](#_Toc11010864)

[参考文献 29](#_Toc11010865)

[致 谢 30](#_Toc11010866)

# 1 绪论

## 1.1 选题背景及意义

尾矿库是指筑坝阻挡谷口或围地组成的，用以堆存金属或非金属矿山举行矿石选别后排挤尾矿或其他工业废渣的场所[1]。尾矿库是一个具有高势能的人造泥石流伤害源，存在溃坝伤害，一旦出事，极易造成重特大事故[2]。美国克拉克大学公害评定小组的研究表明，尾矿库事故造成的危害，在世界93种事故、公害隐患中，名列第18位。1962年9月26日，云南个旧云锡火谷都尾矿库事故，造成171人死亡、92人受伤，11个村寨及1座农场被毁。1985年8月25日，牛角垅尾矿库事故，造成49人死亡，直接经济损失达1300多万元[3]。2008年9月8日，襄汾尾矿溃坝造成277人死亡、4人失踪、33人受伤，直接经济损失达9619万元[4]。鉴于安全事故频发，迫切需要采用现代化手段对尾矿库进行监控、管理、和事故的预警处理。

本次研究与开发的这个基于J2EE的尾矿库安全云管理系统，使得尾矿库的管理从之前本地、固定的管理模式转化为云端、灵活的管理模式，用户不在再受地域和终端的限制，只需要任意一款浏览器即可完成一系列的管理、监控、异常处理和生成报表等工作。

## 1.2 尾矿库坝体安全性能检测系统概述

本系是前后端全分离的开发形式开发，页面用Vue、echarts等框架开发，后台用Java语言为基础，利用SpringBoot、Vue、echarts、jpa、Mybatis、POI等最新的技术进行开发，主要是以尾矿库的安全数据监控，异常预警，异常处理，人员管理功能为主，实现最大程度的保障人民的生命和财产安全。

## 1.3.国内外研究现状

就当前现状而言，我国家大部分企业和实验室，对于尾矿库监控管理技术发展的比较缓慢，多采用人工或部分设备的实地检测、记录，技术落后。国外的企业和实验室在这一方面却做得相对成熟一些，规模比较大，也有专业的团队提供技术和服务支持。由于整体信息化水平相对落后，资金短缺，理论体系不完善，国内尾矿库监控管理技术，服务范围、服务质量和人员的要求，与国外相比有一定差距。

## 1.4 基于J2EE的尾矿库坝体安全性能检测系统的特点

健壮性：系统在输入部分做了过滤，针对用户名、密码电子邮箱等有规则的输入进行了正则判定，过滤非法输入，对错误输入进行容错提示。

可扩展性：尾矿库安全管理系统采用前后端分离的方式开发，前端后台解耦，对前端文件也进行了模块化封装，增强了代码的可扩展性，新增功能只需添加新的模块后引用即可。

兼容性：系统是基于云平台的，只存在前端页面的兼容性问题，本系统前端页面可兼容大部分浏览器，已适配谷歌、火狐、360等主流浏览器基本无兼容性问题。

可移植性：系统的前端部分采用的是Vue框架实现，对电脑环境没有任何要求，有浏览器即可。

易用性：系统再设计之初就进行了多方调研，力争真正解决尾矿库安全管理的及时性和人力难以控制等问题。在代码实现前期进行了充分的原形和交互逻辑的设计，保证系统的易用性。

## 1.5 本课题要解决的问题

1. 解决尾矿库温度、湿度、浸润线等安全数据的实时监测上报。
2. 帮助管理人员进行全方位管理，系统集成实时视频监控和监控地图显示等。
3. 解决用人单位对人员进行管理的问题，并可实现权限控制等功能。

# 2 相关技术与开发、运行环境

## 2.1 Vue框架

vue是渐进式JavaScript框架，“渐进式框架”和“自底向上增量开发的设计”是Vue开发的两个概念。Vue可以在任意其他类型的项目中使用，使用成本较低，更灵活，主张较弱，在Vue的项目中也可以轻松融汇其他的技术来开发，并且因为Vue的生态系统特别庞大，可以找到基本所有类型的工具在vue项目中使用。

在前端开发中，如何高效的操作dom、渲染数据是一个前端工程师需要考虑的问题，而且当数据量大，流向较乱的时候，如何正确使用数据，操作数据也是一个问题，而js框架对上述的几个问题都有自己趋于完美的解决方案，开发成本降低。高性能高效率。唯一的缺点就是需要使用一定的成本来学习。

特点：易用（使用成本低），灵活（生态系统完善，适用于任何规模的项目），高效（体积小，优化好，性能好）。Vue是一个MVVM的js框架，但是，Vue 的核心库只关注视图层，开发者关注的只是m-v的映射关系。

## 2.2 Element-UI组件

Element-UI时基于Vue.js的高质量页面组件，帮助开发人员快速，简单的实现页面功能，其主要特征如下：

一致性 Consistency：与现实生活一致：与现实生活的流程、逻辑保持一致，遵循用户习惯的语言和概念；在界面中一致：所有的元素和结构需保持一致，比如：设计样式、图标和文本、元素的位置等。

反馈 Feedback：控制反馈：通过界面样式和交互动效让用户可以清晰的感知自己的操作；页面反馈：操作后，通过页面元素的变化清晰地展现当前状态。

效率 Efficiency：简化流程：设计简洁直观的操作流程；清晰明确：语言表达清晰且表意明确，让用户快速理解进而作出决策；帮助用户识别：界面简单直白，让用户快速识别而非回忆，减少用户记忆负担。

可控 Controllability：用户决策：根据场景可给予用户操作建议或安全提示，但不能代替用户进行决策；结果可控：用户可以自由的进行操作，包括撤销、回退和终止当前操作等。

## 2.2 MyBatis框架

MyBatis是支持定制化SQL、存储过程以及高级映射的优秀的持久层框架，MyBatis避免了几乎所有的JDBC代码和手动设置参数以及获取结果集。MyBatis可以对配置和原生Map使用简单的XML或注解，将接口和Java的POJOs(Plain Old Java Objects,普通的Java对象)映射成数据库中的记录。

MyBatis的着力点，在于POJO与SQL之间的映射关系。然后通过映射配置文件，将SQL所需的参数，以及返回的结果字段映射到指定POJO。相对Hibernate"O/R"而言，iBATIS是一种“Sql Mapping”的ORM实现。

每个基于MyBatis的应用都是以一个SqlSessionFactory的实例为中心的，SqlSessionFactory的实例可以通过SqlSessionFactoryBuilder获取。而SqlSessionFactoryBuilder则可以从核心配置文件或者预先定制的Configuration的实例构建出该实例。

## 2.2 spring data jpa框架

简单方便：JPA的主要目标之一就是提供更加简单的编程模型：在JPA框架下创建实体和创建Java 类一样简单，没有任何的约束和限制，只需要使用 javax.persistence.Entity进行注释，JPA的框架和接口也都非常简单，没有太多特别的规则和设计模式的要求，开发者可以很容易地掌握。

非侵入式：JPA基于非侵入式原则设计，因此可以很容易地和其它框架或者容器集成。

查询能力： JPA的查询语言是[面向对象](https://baike.baidu.com/item/%E9%9D%A2%E5%90%91%E5%AF%B9%E8%B1%A1" \t "_blank)而非面向数据库的，它以面向对象的自然语法构造查询语句，可以看成是Hibernate HQL的等价物。JPA定义了独特的[JPQL](https://baike.baidu.com/item/JPQL" \t "_blank)（Java Persistence Query Language），JPQL是EJB QL的一种扩展，它是针对实体的一种查询语言，操作对象是实体，而不是关系数据库的表，而且能够支持批量更新和修改、JOIN、GROUP BY、HAVING 等通常只有 SQL 才能够提供的高级查询特性，甚至还能够支持[子查询](https://baike.baidu.com/item/%E5%AD%90%E6%9F%A5%E8%AF%A2" \t "_blank)。

## 2.4 Druid技术

数据库连接是一种关键的有限的昂贵的资源，这一点在多用户的网页应用程序中体现得尤为突出。对数据库连接的管理能显著影响到整个应用程序的伸缩性和健壮性，影响到程序的性能指标。数据库连接池正是针对这个问题提出来的。数据库连接池负责分配、管理和释放数据库连接，它允许应用程序重复使用一个现有的数据库连接，而不是再重新建立一个;释放空闲时间超过最大空闲时间的数据库连接来避免因为没有释放数据库连接而引起的数据库连接遗漏。这项技术能明显提高对数据库操作的性能。Druid是阿里巴巴技术团队开发的一款优秀的数据库连接池，并且已经在GitHub上开源，它与其他国外优秀的数据库连接池用法类似，性能更好，同时加入了日志监控，可以让开发者明确的看到自己的sql语句的执行情况。可以说 Druid是目前国内互联网企业中使用范围最广的数据库连接池。

Druid首先是一个数据库连接池。Druid是目前最好的数据库连接池，在功能、性能、扩展性方面，都超过其他数据库连接池，包括DBCP、C3P0、BoneCP、Proxool、JBoss DataSource。Druid已经在阿里巴巴部署了超过600个应用，经过一年多生产环境大规模部署的严苛考验。Druid是一个JDBC组件，它包括三个部分： 基于Filter－Chain模式的插件体系。 DruidDataSource 高效可管理的数据库连接池。SQLParser。Druid的功能如下：

1、替换DBCP和C3P0。Druid提供了一个高效、功能强大、可扩展性好的数据库连接池。

2、可以监控数据库访问性能，Druid内置提供了一个功能强大的StatFilter插件，能够详细统计SQL的执行性能，这对于线上分析数据库访问性能有帮助。

3、数据库密码加密。直接把数据库密码写在配置文件中，这是不好的行为，容易导致安全问题。DruidDruiver和DruidDataSource都支持PasswordCallback。

4、SQL执行日志，Druid提供了不同的LogFilter，能够支持Common-Logging、Log4j和JdkLog，你可以按需要选择相应的LogFilter，监控你应用的数据库访问情况。

5、扩展JDBC，如果你要对JDBC层有编程的需求，可以通过Druid提供的Filter机制，很方便编写JDBC层的扩展插件。

## 2.5 SpringBoot框架

Spring Boot是由Pivotal团队提供的全新框架，其设计目的是用来简化新Spring应用的初始搭建以及开发过程。该框架使用了特定的方式来进行配置，从而使开发人员不再需要定义样板化的配置。通过这种方式，Spring Boot致力于在蓬勃发展的快速应用开发领域(rapid application development)成为领导者。Springboot包含spring和SpringMVC的所有功能， Spring是一个开放源代码的设计层面框架，他解决的是业务逻辑层和其他各层的松耦合问题，因此它将面向接口的编程思想贯穿整个系统应用。Spring是于2003 年兴起的一个轻量级的Java 开发框架，由Rod Johnson创建。简单来说，Spring是一个分层的JavaSE/EE full-stack(一站式) [轻量级](https://baike.so.com/doc/585215-619452.html)开源框架。Spring 框架提供了构建 Web 应用程序的全功能 MVC 模块。 使用 Spring 可插入的 MVC 架构，可以选择是使用内置的 Spring Web 框架还是 Struts 这样的 Web 框架。通过策略接口，Spring 框架是高度可配置的，而且包含多种视图技术，例如 JavaServer Pages（JSP）技术、Velocity、Tiles、iText 和 POI。Spring MVC 框架并不知道使用的视图，所以不会强迫您只使用 JSP 技术。Spring MVC 分离了[控制器](https://baike.so.com/doc/422704-447609.html" \t "_blank)、模型对象、分派器以及处理程序对象的角色，这种分离让它们更容易进行[定制](https://baike.so.com/doc/6757325-6971924.html)。其他特点如下：

1. 创建独立的Spring应用程序。
2. 嵌入的Tomcat，无需部署WAR文件。
3. 简化Maven配置。
4. 自动配置Spring。
5. 提供生产就绪型功能，如指标，健康检查和外部配置。
6. 绝对没有代码生成并且对XML也没有配置要求。

## 2.6 ECharts开源技术

Echarts 基于H5的Canvas类库 ZRender,来源于百度的开源项目，Echar提供丰富的js图表库，并且支持高度定制，Echar主要的三大特点是支持拖拽后重算，数据驱动视图和值域的漫游，提高了使用者的体验性和良好的交互。

### 2.6.1 提供丰富的图表类型

ECharts 提供折线/柱状/饼/散点/雷达/k线/热力等多种图表类型，并且不同类型的图可以混搭使用。

### 2.6.1 数据驱动特点

ECharts图表得益于数据驱动改变，后台提供的数据改变，图表便轻松发生更新。开发者只需要关注输入图表数据的变换，图表会自动检查最新数据数据与当前数据的差异，并更新图表。

## 2.7 WebSocket技术

WebSocket协议是基于TCP的一种新的网络协议。它实现了浏览器与服务器全双工(full-duplex)通信--允许服务器主动发送信息给客户端。WebSocket通信协议于2011年被IETF定为标准RFC 6455，并被RFC7936所补充规范。长久以来, 创建实现客户端和用户端之间双工通讯的web app都会造成HTTP轮询的滥用: 客户端向主机不断发送不同的HTTP呼叫来进行询问。这会导致一系列的问题:

1.服务器被迫为每个客户端使用许多不同的底层TCP连接:一个用于向客户端发送信息，其它用于接收每个传入消息。

2.有些协议有很高的开销，每一个客户端和服务器之间都有HTTP头。

3.客户端脚本被迫维护从传出连接到传入连接的映射来追踪回复。

一个更简单的解决方案是使用单个TCP连接双向通信。 这就是WebSocket协议所提供的功能。 结合WebSocket API ，WebSocket协议提供了一个用来替代HTTP轮询实现网页到远程主机的双向通信的方法。

WebSocket协议被设计来取代用HTTP作为传输层的双向通讯技术,这些技术只能牺牲效率和可依赖性其中一方来提高另一方，因为HTTP最初的目的不是为了双向通讯。[编辑本段](https://baike.so.com/create/edit/?eid=6840040&sid=7057314&secid=3)实现原理：

在实现websocket连线过程中，需要通过浏览器发出websocket连线请求，然后服务器发出回应，这个过程通常称为"握手" 。在 WebSocket API，浏览器和服务器只需要做一个握手的动作，然后，浏览器和服务器之间就形成了一条快速通道。两者之间就直接可以数据互相传送。在此WebSocket 协议中，为我们实现即时服务带来了两大好处:

1. Header：互相沟通的Header是很小的-大概只有 2 Bytes

2. Server Push：服务器的推送，服务器不再被动的接收到浏览器的请求之后才返回数据，而是在有新数据时就主动推送给浏览器。

## 2.8 POI技术

Apache POI是一种流行的API，它允许程序员使用Java程序创建，修改和显示MS Office文件。这由Apache软件基金会开发使用Java分布式设计或修改Microsoft Office文件的开源库。它包含类和方法对用户输入数据或文件到MS Office文档进行解码。

在我们实际的开发中，表现层的解决方案虽然有多样，但是IE浏览器已成为最多人使用的浏览器，因为大家都用Windows。在企业办公系统中，常常有客户这样子要求：你要把我们的报表直接用Excel打开(电信系统、银行系统)。或者是：我们已经习惯用Excel打印。

Apache的Jakata项目的POI子项目，目前比较成熟的是HSSF、XSSF等接口，在处理MSExcel对象。它不象我们仅仅是用csv生成的没有格式的可以由Excel转换的东西，而是真正的Excel对象，你可以控制一些属性如sheet,cell等。

## 2.9基于JWT的Token认证机制

JSON Web Token（JWT）是一个非常轻巧的规范。这个规范允许我们使用JWT在用户和服务器之间传递安全可靠的信息。它由三部分组成，头部、载荷与签名。优点如下：

支持跨域访问: Cookie是不允许垮域访问的，这一点对Token机制是不存在的，前提是传输的用户认证信息通过HTTP头传输。

无状态(也称：服务端可扩展行):Token机制在服务端不需要存储session信息，因为Token 自身包含了所有登录用户的信息，只需要在客户端的cookie或本地介质存储状态信息.

更适用CDN: 可以通过内容分发网络请求你服务端的所有资料（如：javascript，HTML,图片等），而你的服务端只要提供API即可。

去耦: 不需要绑定到一个特定的身份验证方案。Token可以在任何地方生成，只要在你的API被调用的时候，你可以进行Token生成调用即可。

更适用于移动应用: 当你的客户端是一个原生平台（iOS, Android，Windows 8等）时，Cookie是不被支持的（你需要通过Cookie容器进行处理），这时采用Token认证机制就会简单得多。

CSRF:因为不再依赖于Cookie，所以你就不需要考虑对CSRF（跨站请求伪造）的防范。

性能: 一次网络往返时间（通过数据库查询session信息）总比做一次HMACSHA256计算 的Token验证和解析要费时得多。

不需要为登录页面做特殊处理:如果你使用Protractor 做功能测试的时候，不再需要为登录页面做特殊处理。

基于标准化:你的API可以采用标准化的 JSON Web Token (JWT). 这个标准已经存在多个后端库（.NET, Ruby, Java,Python, PHP）和多家公司的支持（如：Firebase,Google, Microsoft）。

## 2.10 core 技术

跨域资源共享([CORS](https://developer.mozilla.org/en-US/docs/Glossary/CORS)) 是一种机制，它使用额外的 [HTTP](https://developer.mozilla.org/en-US/docs/Glossary/HTTP) 头来告诉浏览器  让运行在一个 origin (domain) 上的Web应用被准许访问来自不同源服务器上的指定的资源。当一个资源从与该资源本身所在的服务器不同的域、协议或端口请求一个资源时，资源会发起一个跨域 HTTP 请求。

比如，站点 http://domain-a.com 的某 HTML 页面通过 [<img> 的 src](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/Img#Attributes)请求 http://domain-b.com/image.jpg。网络上的许多页面都会加载来自不同域的CSS样式表，图像和脚本等资源。

出于安全原因，浏览器限制从脚本内发起的跨源HTTP请求。 例如，XMLHttpRequest和Fetch API遵循同源策略。 这意味着使用这些API的Web应用程序只能从加载应用程序的同一个域请求HTTP资源，除非响应报文包含了正确CORS响应头。

跨域资源共享（ [CORS](https://developer.mozilla.org/en-US/docs/Glossary/CORS) ）机制允许 Web 应用服务器进行跨域访问控制，从而使跨域数据传输得以安全进行。现代浏览器支持在 API 容器中（例如 [XMLHttpRequest](https://developer.mozilla.org/zh-CN/docs/Web/API/XMLHttpRequest" \o "使用XMLHttpRequest (XHR)对象可以与服务器交互。您可以从URL获取数据，而无需让整个的页面刷新。这使得Web页面可以只更新页面的局部，而不影响用户的操作。XMLHttpRequest在 Ajax 编程中被大量使用。) 或 [Fetch](https://developer.mozilla.org/en-US/docs/Web/API/Fetch_API) ）使用 CORS，以降低跨域 HTTP 请求所带来的风险。

跨域资源共享标准新增了一组 HTTP 首部字段，允许服务器声明哪些源站通过浏览器有权限访问哪些资源。另外，规范要求，对那些可能对服务器数据产生副作用的 HTTP 请求方法（特别是 [GET](https://developer.mozilla.org/zh-CN/docs/Web/HTTP/Methods/GET) 以外的 HTTP 请求，或者搭配某些 MIME 类型的 [POST](https://developer.mozilla.org/zh-CN/docs/Web/HTTP/Methods/POST) 请求），浏览器必须首先使用 [OPTIONS](https://developer.mozilla.org/zh-CN/docs/Web/HTTP/Methods/OPTIONS) 方法发起一个预检请求（preflight request），从而获知服务端是否允许该跨域请求。服务器确认允许之后，才发起实际的 HTTP 请求。在预检请求的返回中，服务器端也可以通知客户端，是否需要携带身份凭证（包括 [Cookies](https://developer.mozilla.org/zh-CN/docs/Web/HTTP/Cookies)和 HTTP 认证相关数据）。

CORS请求失败会产生错误，但是为了安全，在JavaScript代码层面是无法获知到底具体是哪里出了问题。你只能查看浏览器的控制台以得知具体是哪里出现了错误。

## 2.11 开发工具

### 2.11.1 IntelliJ IDEA开发工具

IntelliJ IDEA，是用于java语言开发的集成环境(也可用于其他语言)，IntelliJ在业界被公认为最好的java开发工具之一，尤其在智能代码助手、代码自动提示、重构、[J2EE](https://baike.so.com/doc/901087-952427.html" \t "_blank)支持、[Ant](https://baike.so.com/doc/5418621-10440907.html" \t "_blank)、[JUnit](https://baike.so.com/doc/1858643-1965718.html" \t "_blank)、[CVS](https://baike.so.com/doc/1774040-1876053.html" \t "_blank)整合、代码审查、 创新的[GUI设计](https://baike.so.com/doc/7065585-7288495.html)等方面的功能可以说是超常的。



图2-3 IntelliJ IDEA工作界面图

### 2.11.2 WebStorm开发工具

WebStorm 是[jetbrains](https://baike.baidu.com/item/jetbrains" \t "_blank)公司旗下一款JavaScript 开发工具。目前已经被广大中国JS开发者誉为“Web前端开发神器”、“最强大的HTML5编辑器”、“最智能的JavaScript IDE”等。与[IntelliJ IDEA](https://baike.baidu.com/item/IntelliJ%20IDEA" \t "_blank)同源，继承了IntelliJ IDEA强大的JS部分的功能。

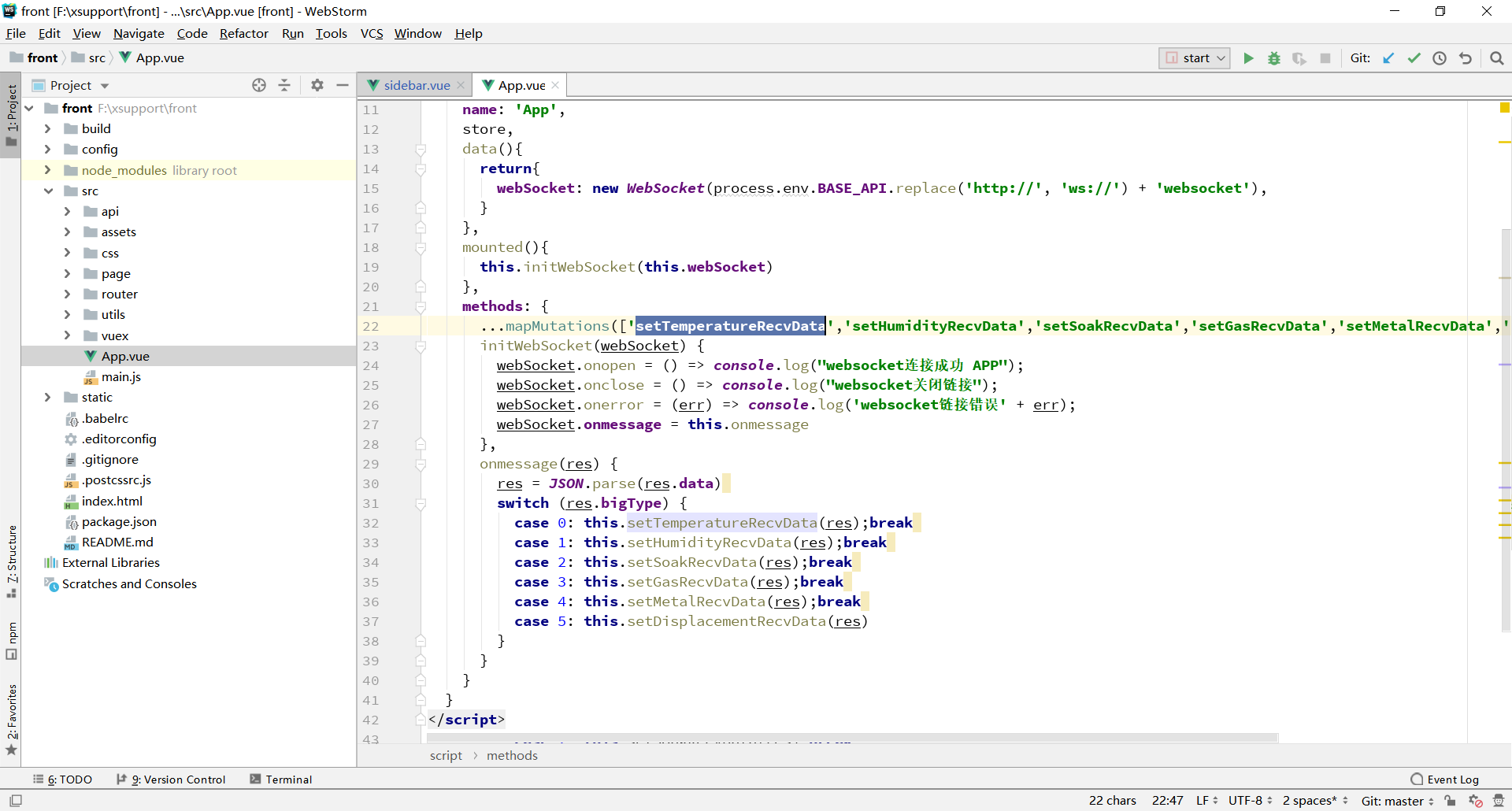


图2-3 WebStorm工作界面图

# 3项目需求分析

## 3.1系统的需求分析

该云管理系统主要是以尾矿库的安全数据监控，异常预警，异常处理，人员管理功能，以及导出数据报表等功能为主。该系统的需求如下：

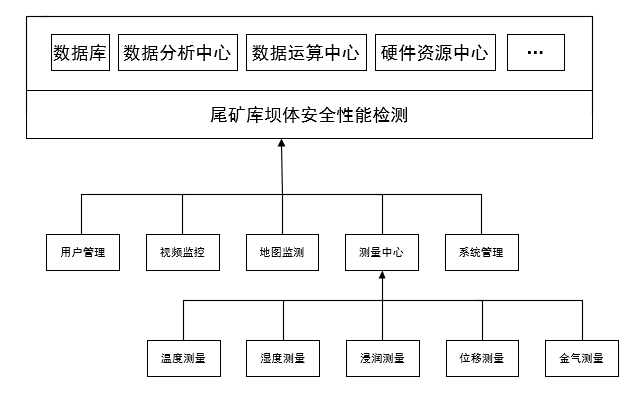


图3-1项目总体需求图

测量中心：下设温度测量，湿度测量，水位测量，浸润测量，位移测量，金气测量等模块，实时获取后台数据，并在前台做可视化展示，如果数据异常则及时预警，并可报表导出等功能。具体结构如图3-2测量中心图所示。

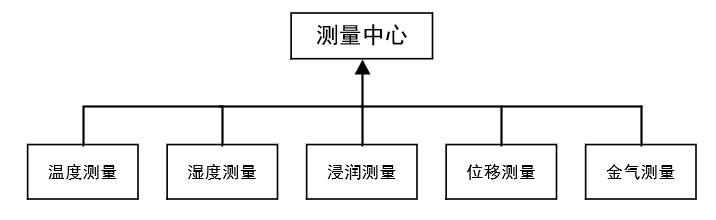


图3-2测量中心

视频监控，实时采集摄像头可见范围，帮助管理人员更好的掌握尾矿库的实际环境。

监测地图，对周围地形地貌进行全方位展示，方便在突发情况来临前做战略部署或在特殊时期做应急指挥。

用户管理：主要由人员的增加，删除，修改，查询，禁用等功能构成，系统配置了管理员和普通用户的权限，实现不同的用户拥有不同的权限和可操作控制。

系统管理：记录了系统运行时测量值超过阈值线的数据，进行统一管理，可一键导出所有异常信息等。

## 3.3 系统的性能需求

根据现在互联网的开发模式下，一个能够上线的项目需要满足以下几个方面：

1. 按需求分析完成系统设计，编码和实现，没有影响系统正常运行的致命bug。
2. 确保设计功能能正常使用，确定上线的功能没有功能缺失，在功能上可以形成一个闭环。
3. 提供完整的项目文档和接口文档，让新加入的开发人员能够快速加入开发。
4. 提供并建立完备的问题处理机制和响应体系，拥有完整的运维体系，可以随时修复线上问题和进行回滚。

# 4 系统的设计

## 4.1系统设计的原则

良好的系统设计原则与项目的完美实现密切相关，在程序设计之中要遵循以下总结的设计原则：

1. 避免重复性（Avoid repeatability），避免重复性是程序设计的最基本的原则。在程序中必然拥有很多数据、存储结构，例如：函数、循环体、对象等。一旦重复就会造成代码冗杂，影响效率。
2. 简单原则（Simple principle），程序的简单是一个程序设计中不可缺少的，使用最少的代码，实现一个很重要的功能，并且简单代码的空间大小和执行时间更少，发现漏洞也会更加及时，对于维护更加便利。
3. 低耦合原则（Low coupling principle），这是所有的框架和程序设计都在遵循的原则和追求的目标，需要在代码的任何一个地方都减少对其他代码的依赖，底耦合往往是一个优秀系统设计的标准。

## 4.2 数据库的设计

无论哪一个系统都离不开数据库，它是所有数据的家，没有数据库任何系统就无法使用，在软件设计中，数据库设计的好坏一定程度决定程序设计的好坏。所以在程序设计中数据库的设计尤为的重要。数据库设计中需要设计好数据存储结构，好的数据存储结构对系统的功能和性能都会产生非常大的影响[8]。同样，优秀的数据存储结构可以使代码拥有更好的扩展性，对后期需求增加提供一个良好基础。

1. 测量类型表（big\_type），该表用于记录不同测量类型的各自的信息如单位等信息，具体结构如表4-1测量类型表所示。子类型详细信息表（type），该表用于记录不同类型下多个子类型的详细信息，具体结构如表4-2子类型详细信息表所示。测量记录表（measure），该表用于记录测量的温度、湿度、浸润线、有毒气体和重金属的相关信息。具体结构如表4-3测量记录表所示。

表4-1测量类型表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段含义 | 字段名称 | 类型 | 长度 | 主键 |
| id | id | varchar | 32 | Yes |
| 类型code | type | int | 2 | No |
| 单位 | unit | varchar | 10 | No |
| 异常定位名称 | exceptionLocationName | varchar | 20 | No |
| 备注 | remark | varchar | 255 | No |

表4-2子类型详细信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段含义 | 字段名称 | 类型 | 长度 | 主键 |
| ID | id | varchar | 32 | Yes |
| 创建时间 | createTime | datetime | 0 | No |
| 更新时间 | updateTime | datetime | 0 | No |
| 大类型 | bigType | int | 1 | No |
| 子序号 | subIndex | int | 2 | No |
| 阈值 | limitValue | float | 5 | No |
| 名称 | name | varchar | 10 | No |
| 备注 | remark | varchar | 255 | No |

表4-3测量记录表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段含义 | 字段名称 | 类型 | 长度 | 主键 |
| id | id | varchar | 32 | Yes |
| 采集时间 | createTime | datetime | 0 | No |
| 测量类型 | bigType | int | 2 | No |
| 子序号 | subIndex | int | 2 | No |
| 采集值 | value | float | 5 | No |
| 阈值 | limitValue | float | 5 | No |
| 湿度状态 | state | int | 11 | No |

测量记录各实体关系如图4-1 测量记录关系图所示：



图4-1 测量记录关系图

1. 位移采集记录表，有四张表构成，测量点记录表（point\_enum），该表用于记录采集位移信息的点的相关信息，具体结构如表4-4测量点记录表所示。位移采集汇总记录表（displacement），该表用于记录单次采集到的所有测量点的公共信息，具体结构如表4-5位移采集汇总记录表所示。移采集单条记录表（displacement\_every），该表用于记录单个测量点采集到的数据，具体结构如表4-6位移采集单条记录表所示。位移采集值表（displacement\_value），该表用于记录每个点采集到的x、y、z轴上采集详细数据，具体结构如表4-7位移采集值表所示。

表4-4测量点记录表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段含义 | 字段名称 | 类型 | 长度 | 主键 |
| id | id | varchar | 32 | Yes |
| code | code | int | 2 | No |
| 测量线 | line | int | 1 | No |
| 测量点 | point | int | 1 | No |
| 测点名称 | name | varchar | 10 | No |

表4-5位移采集汇总记录表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段含义 | 字段名称 | 类型 | 长度 | 主键 |
| 湿度ID | id | varchar | 32 | Yes |
| 采集时间 | createTime | datetime | 0 | No |
| 备注 | remark | text | 255 | No |

表4-6 移采集单条记录表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段含义 | 字段名称 | 类型 | 长度 | 主键 |
| id | id | varchar | 32 | Yes |
| 组ID | groupId | varchar | 32 | No |
| 测量点 | point | int | 3 | No |
| 本次测值 | thisValue | varchar | 32 | No |
| 上次测值 | beforeValue | varchar | 32 | No |

表4-7 位移采集值表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段含义 | 字段名称 | 类型 | 长度 | 主键 |
| id | id | varchar | 32 | Yes |
| x值 | x | float | 20 | No |
| y值 | y | float | 20 | No |
| z值 | z | float | 20 | No |

位移测量各实体关系如图4-2 位移测量关系图所示：



图4-2 位移测量关系图

1. 系统测量异常记录（sys\_warn），该表用于记录测量模块的异常记录，具体的结构如表4-8系统测量异常记录表所示：

表4-8系统测量异常记录表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段含义 | 字段名称 | 类型 | 长度 | 主键 |
| id | id | varchar | 32 | Yes |
| 创建时间 | createTime | datetime | 0 | No |
| 修改时间 | updateTime | datetime | 0 | No |
| 操作人 | optionUser | varchar | 10 | No |
| 异常类型 | exceptionLocation | int | 1 | No |
| 异常描述 | exceptionDescription | int | 2 | No |
| 异常状态 | exceptionState | int | 1 | No |
| 异常值 | exceptionValue | float | 5 | No |
| 阈值 | limitValue | float | 5 | No |
| 备注 | remark | varchar | 255 | No |

测量异常属性关系如图4-3测量异常属性图所示：



图4-3测量异常属性图

1. 用户表（user），用于存储系统注册用户的相关信息，具体的结构如用户表4-9用户表所示：

表4-9用户表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段含义 | 字段名称 | 类型 | 长度 | 主键 |
| id | id | Varchar | 32 | Yes |
| 创建时间 | createTime | datetime | 0 | No |
| 账号 | username | varchar | 20 | No |
| 密码 | password | varchar | 20 | No |
| 是否管理员 | isAdmin | tinyint | 10 | No |
| 是否禁用 | isForbidden | tinyint | 1 | No |
| 姓名 | name | Varchar | 10 | No |
| 性别 | sex | tinyint | 1 | No |
| 年龄 | age | int | 3 | No |
| 家庭住址 | address | varchar | 255 | No |
| 联系电话 | phone | varchar | 11 | No |
| 备注 | remark | varchar | 255 | No |

用户具体属性如图4-4用户属性图所示：



图4-4用户属性图

# 5 系统的实现

依据需求分析，本系统实现了全部功能，并且保证了每一个功能点可用性和实用性，主要完成了测量中心模块、视频监控模块、人员管理模块、系统管理模块等。

各个角色登录时，输入用户账号、用户密码，点击登录，系统自动识别用户类型，根据不同的权限提供不同的操作。

## 5.2.1登录模块

登录模块是本平台的入口模块，本平台使用了基于token的访问验证，客户端在在访问平台内所有路由和接口调用时，前都会验证客户端localstore的token值，如果没有或token已过期，则会拒绝访问，并跳转登录页面，在提交登录信息时，后端会验证登录信息，如有效则会生成新的token，并为此token设置其过期时间（7天），在之后的所有前后端交互时时，都必须携带此token令牌。否则将会被拦截请求。

在进入登录页面和登录成功后时页面的右上角会有不同的提示信息，方便不熟悉此平台的人员使用此系统。页面如图5-1所示：



图5-1 尾矿库安全云管理平台登录页

登录成功后将进入系统内，系统页面主要由四部分构成,分别为：

Header（头部）：显示系统平台名称和登录的用户姓名，最左侧按钮为折叠（展开）侧边栏按钮，单击侧换侧边栏不同状态。最右侧下拉菜单主要为修改自己密码，退出登录的功能。侧边栏展开如图5-2 页面区域说明图所示.

Tag（标签列表区）：在Header下方，记录在平台内访问过的页面信息，并可作为页面路由跳转功能，单击可跳转至不同页面，最右侧按钮可关闭当前除当前页面所在标签外的其他标签。

Sidbar（侧边栏）：显示页面所有的路由和页面跳转信息。可通过Header的最左侧按钮改变显示样式。

Content（内容区）：根据页面的不同路由显示不同的内容。



图5-2 页面区域说明

## 5.2.2 测量中心模块

测量中心主要有温度测量、湿度测量，以水位测量，有毒气体检测，重金属含量测量，位移测量等模块。

WebSocket链接：在客户端第一次访问该服务器时，会建立一个长连接，服务器保存所有客户端的会话，当测量值发生变化时，服务器将变化数据推送给所有保持链接的客户端，不需要客户端主动向服务器发送请求。客户端会通过onmessage监听数据变化，并根据不同类别信息，分发到不同页面，动态更新图表信息。在传统的客户端轮询请求服务器时，一般会携带较多的无用信息，如header等认证信息等，当客户端数量较多时，服务器会在短时间内接收到大量请求，极大的消耗服务器和数据库的资源。在使用webtoken传输数据时，只传递数据，无需其他冗余信息，极大地减轻服务器的压力。



图5-3 服务器向所有客户端推送数据

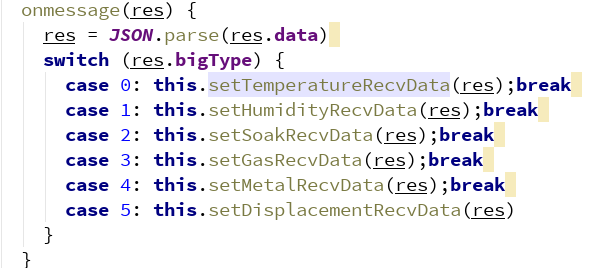


图5-4 客户端接收消息并分发给不同组件

报表导出：使用了POI的Java的API进行操作Excel，通过读取后台模板文件，将通过反射获取每个表格的特定数据，将数据写入指定单元格，并一一对应。具体步骤如下：

1. 在字段上加特定注解，使其value属性与导出的模板的表头数据相同。
2. 通过反射得到该特定注解的值将数据与表头信息关联。
3. 读取Excel模板文件。导出excel文件。



图5-5 在导出字段上加@ApiModelProperty注解，并设置value属性



图5-6 通过反射读取注解的value属性值



图5-7 读取Excel模板

权限操作：管理员账号可以对各个类别设置不同测量阈值，页面显示阈值相关信息，超过阈值时在系统管理里添加相关异常数据，统一管理。普通用户不支持修改阈值信息。

（1）温度测量模块负责展示实时采集来的温度数据。



图5-8 温度测量

（2）湿度测量模块负责展示实时采集来的湿度数据。



图5-9 湿度测量

（3）浸润测量模块负责展示坝体水位线高度和尾矿所在地区的近期降雨量和水流量信息，方便管理人员及时开展防洪防汛工作。

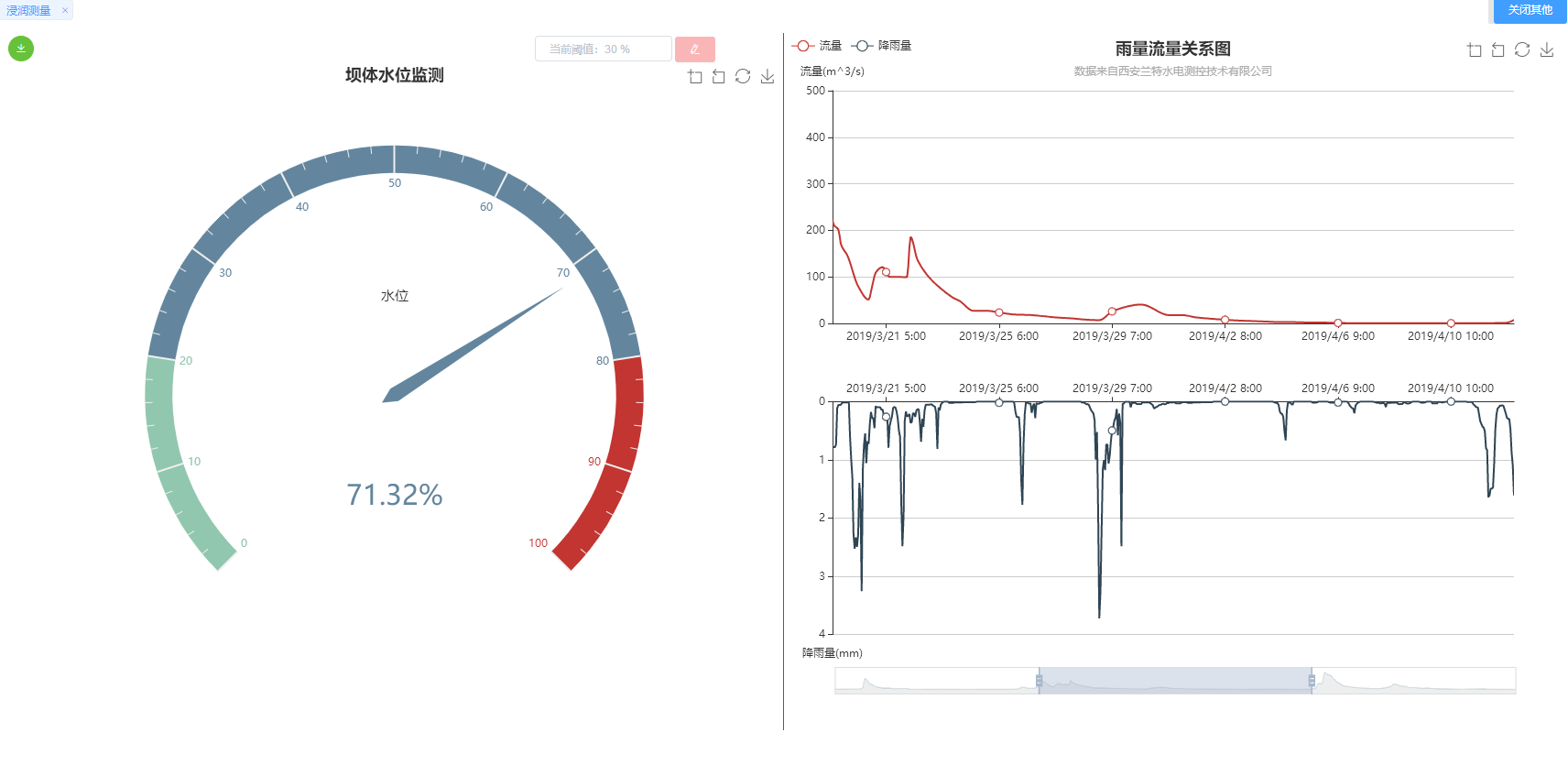


图5-10 浸润测量

（4）金气测量模块负责展示坝体有毒有害气体占比和重金属元素含量，帮助工作人员及时排除危险，避免不必要的人员伤害，可对各个子项设置不同的阈值。

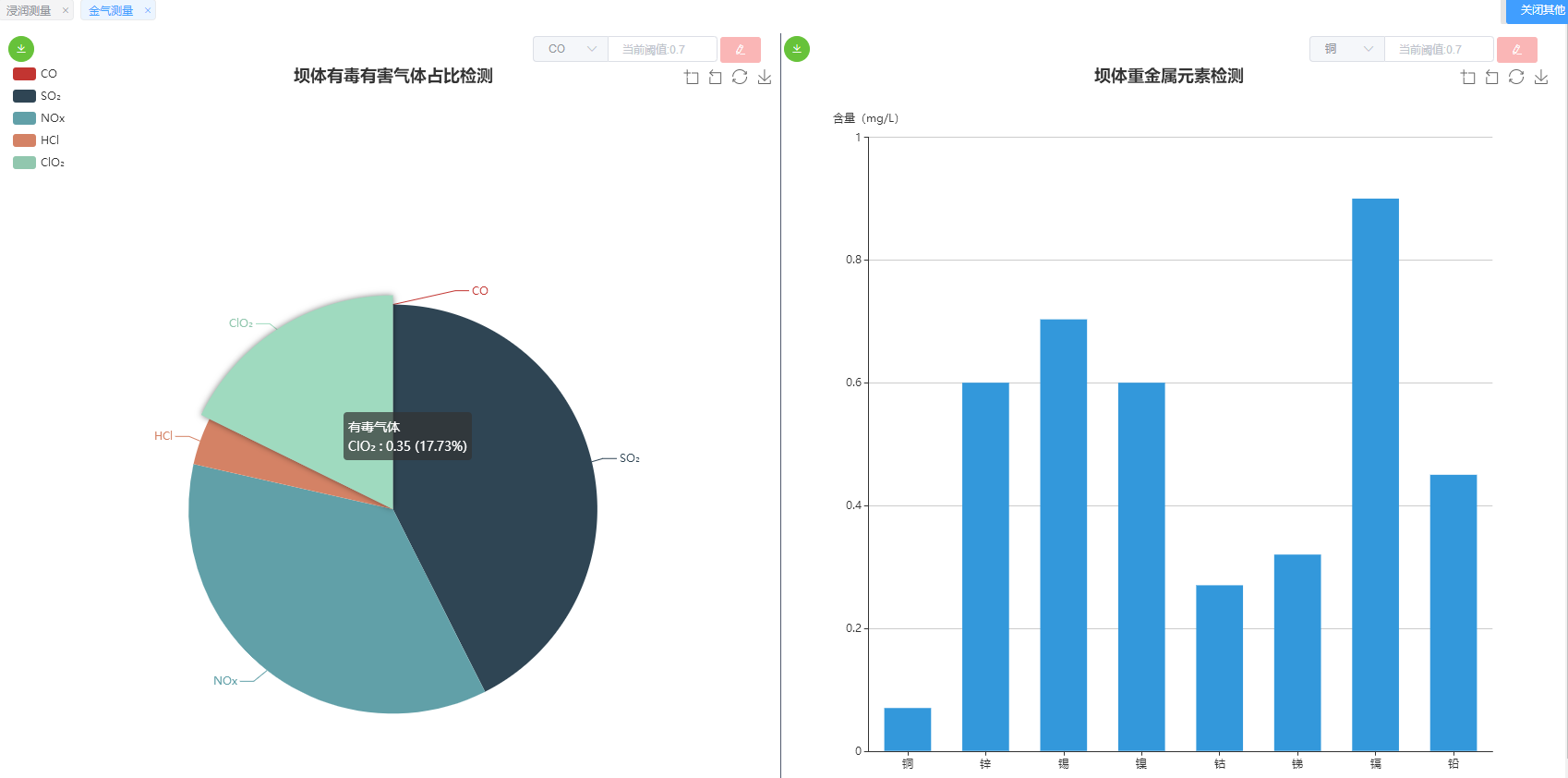


图5-11 金气测量

（5）位移测量模块负责展示坝体多个点的坝体位移x，y，z各个轴线上位移的变化情况。数据实时更新，可同时显示本次测量和上次测量的值。

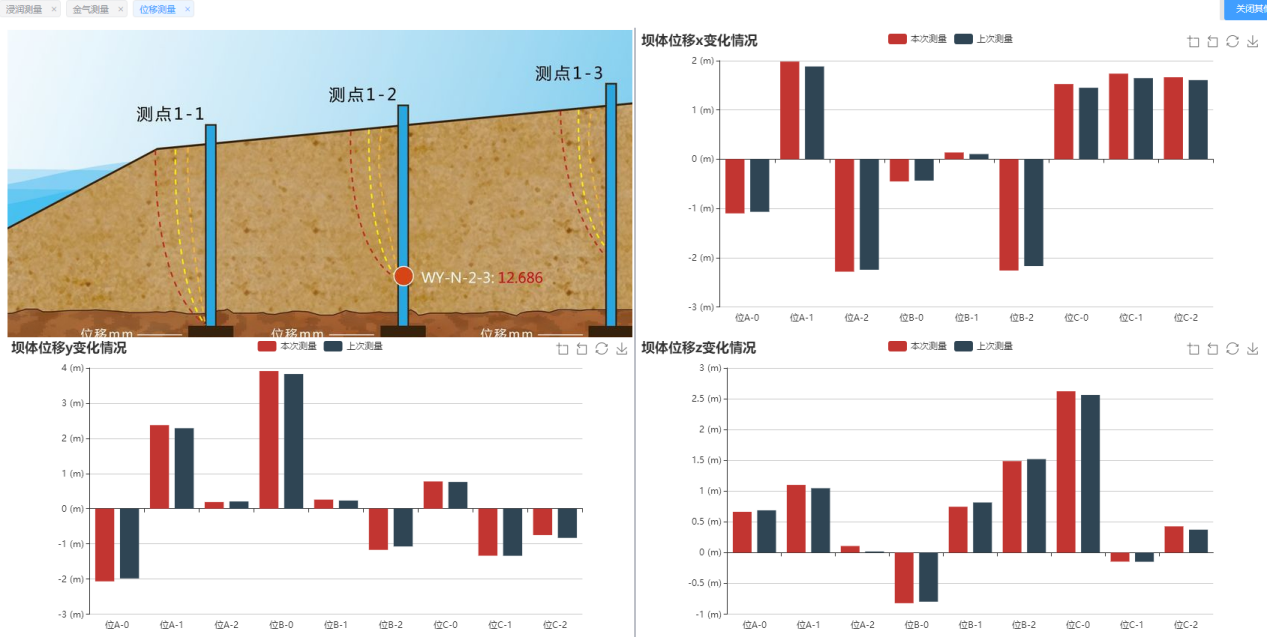


图5-12 位移测量

## 5.2.3系统管理模块

系统管理模块主要记录了系统运行时测量值超过阈值线的数据，进行统一管理，可一建导出所有异常信息，可以进行状态修改，添加备注，条件筛选等，并记录创建、修改时间，修改人等信息。



图5-13 系统管理页面图

## 5.2.4视频监控模块

视频监控模块的功能主要提供监控功能。帮助企业管理人员实时查看和监控尾矿库的各种信息。



图5-14视频监控页面图

## 5.2.5 监控地图模块

监控地图模块，根据需求通过调用百度地图API实现周围环境的地图监控功能，通过百度地图进行实时定位，可事实展示路况地图和，卫星地图，实时将GPS采集到的数据可视化展示给用户，方便在突发情况来临前做战略部署或在特殊时期做应急指挥，如图5-6所示。



图5-15监控地图页面

## 5.2.6用户管理模块

用户管理模块提供日常的人员变动时的所有操作，为了方便管理人员变动提供模糊搜索，批量删除，临时禁用等功能，提供了权限管理的不同操作以下提供了不同权限的角色的不同操作，主要权限区别如下：

管理员：查看所有用户信息，可对所有用户执行信息修改，删除禁用等操作。

普通用户：查看所有用户信息，修改自己的信息，不能修改其他人信息。



图5-16用户管理页面图

# 6 系统的测试

本项目的代码编写历时一个多月时间，为了能使系统达到可以线上部署的要求，就必须对系统进行各种各样的测试和BUG修改工作，所以很大一部分时间都用在了测试和完善bug中。

常用的测试方法有：单元测试、黑白盒测试、压力测试、动态测试等。本项目中使用了单元测试和Jmeter压力测试。

## 6.1单元测试

单元测试（Unit Testing），是最微小的测试方法，最大的特征是单元测试是有程序员来编写的而不是由专业的测试人员编写。

## 6.2 OneAPM CPT云压测

压力测试又称负载测试，是一个应用程序能够承担最大的的负荷量。压力测试与服务器的硬件设备密不可分，所以不同的代码在不同的硬件机器上可能会出现不能的结果。



图6-1 OneAPM CPT压测节点

在压测过程中，可对事务耗时深入到各应用组件, 自定义事务深入到执行线, 数据库监控下钻至 SQL 执行计划, 误定位到代码行及错误堆栈信息，文章的篇幅有限，这里只展示了菜品列表的页面，该页面的数据量最大，能够更好的体现系统的性能，测试结果图如图6-2所示。OneAPM CPT 压测节点遍布全球,节点分部图如6-2所示



图6-2 OneAPM CPT 压测结果

压力测试结果说明：

本次测试每秒启动1193个用户线程访问，相当于每秒的访问量。在此环境下所有线程错误率0.03%，90%的线程在2秒内完成，95%在5秒内完成，平均在1秒内完成，平均每秒内8221.5次点击。测试结果通过，本尾矿库安全管理系统完全可以满足一般企业正常业务使用。

# 7总结和展望

本文遵守程序设计的原则和规范，运用了当前互联网建站的主流技术和资源，最终完成了基于J2EE的尾矿库安全云管理系统。

尾矿库安全云管理系统每一个尾矿企业相关企业都应该配备，每一个网站设计都要有自身的特色，本项目中也体现出来自己的特色，本项目在如下的两个方面有新颖的地方。

1. 采用数据可视化展示，管理人员不用面对大量的数据，只需要关注可视化变化和系统自动预警即可完成日常的管理工作。
2. 本系统与传统的管理网站不一样，传统的网站只拥有web系统，但是本项目中添加连接硬件设备的接口，可以从任何可以采集数据的硬件端导入相应数据。

当然，在上面两个之外，本项目还有许多创新之处，由于文章有限，这里就不一一举例说明了。

通过一个多月的编码实现功能，最终完成了这个管理项目，自己从需求分析、系统设计、代码实现到测试运行最到达上线部署，自己了解了一整套软件项目开发流程。但是每一件事情都肯定不是完美的，在这一个多月中我也发现了很多不足的地方，比如：

1. 网站的某些页面交互不够好看，自己前端技术不够完善。
2. 由于时间的问题，里面很多安全性的问题没有十分完善，比如数据传输中的加密问题等。
3. 测试工作太少，只进行了单元测试和OneAPM CPT云压测，没有进行完整测试。

总结完本次项目，有收获也有不足的地方，对今后的工作有了明确的目标，就是完善自己提出的项目中还存在问题，完善自己的学习体系，争取将本项目能够完成的更好。

# 参考文献

1. 于广明,宋传旺,潘永战,李亮,李冉,路世豹.尾矿坝安全研究的国外新进展及我国的现状和发展态势[J].岩石力学与工程学报,2014,33(S1):3238-3248.
2. 唐利红.基于Spring MVC及MyBatis的Web应用框架解析[J].课程教育研究,2017(48):230.
3. [基于J2EE的WEB框架的研究和实现](http://kns.cnki.net/kcms/detail/detail.aspx?filename=2006056868.nh&dbcode=CMFD&dbname=CMFD2006&v=)[D]. 邹盼霞.东华大学 2006
4. 秦飞.基于MVC模式的尾矿库信息化监控系统的分析与设计[J].山东工业技术,2017(24):114.
5. 康文杰,王勇,俸皓.云平台中MySQL数据库高可用性的设计与实现[J].计算机工程与设计,2018,39(01):296-301
6. 梁晓弘,黄兴荣.J2EE架构下基于Web的信息管理系统框架研究[J].数字技术与应用,2017(12):86-88.
7. 李万锋. 基于Java EE的数据可视化平台设计及主要功能实现[D].中国科学院大学(中国科学院工程管理与信息技术学院),2017.
8. 荣艳冬.关于Mybatis持久层框架的应用研究[J].信息安全与技术,2015,6(12):86-88.
9. 张宝玉.浅析HTTPS协议的原理及应用[J].网络安全技术与应用,2016(07):36-37+39.
10. James Goodwill. Pure Java Server Pages .Indianapolis Ind: Sams, 2015, 54(3)，61-63
11. Zoya Ali. Designing Object Oriented Software Applications within the Context of Software Frameworks[D]. Ohio State University,2011.
12. [基于Vue.js的Web前端应用研究](http://kns.cnki.net/kcms/detail/detail.aspx?filename=KJYX201720061&dbcode=CJFQ&dbname=CJFD2017&v=)[J]. 朱二华.  科技与创新. 2017(20)

# 致 谢

大学四年学习时光已经接近尾声，在此我想对我的母校，我的父母、亲人们，我的老师和同学们表达我由衷的谢意。感谢我的家人对我大学四年学习的默默支持;感谢我的母校商洛学院给了我在大学四年深造的机会，让我能继续学习和提高；感谢商洛学院的老师和同学们四年来的关心和鼓励。老师们课堂上的激情洋溢，课堂下的谆谆教诲；同学们在学习中的认真热情，生活上的热心主动，所有这些都让我的四年充满了感动。

我做毕业设计的每个阶段，从选题到查阅资料，论文提纲的确定，中期论文的修改，后期论文格式调整等各个环节中王博老师都给予了我悉心的指导。在此谨向王博老师致以诚挚的谢意和崇高的敬意。

四年的大学时光匆匆走过，在此，我再次感谢包括此次论文指导老师王博在内的所有商洛学院的教师，感谢你们四年孜孜不倦的教诲。