

基于数据孪生的立体仓库设备健康管理系统

THREE-DIMENSIONAL HEALTH MANAGEMENT SYSTEM OF WAREHOUSE EQUIPMENT BASED ON DIGITAL TWIN

学 院： 计算机科学与技术学院
专 业： 计算机科学与技术
姓 名： 唐浩凯
学 号： 181100436
指 导 教 师： 王璿



东 华 大 学

课 题 名 称 : 基于数字孪生的立体仓库设备健康管理系统
学 院 : 计算机科学与技术学院
专 业 : 计算机科学与技术
姓 名 : 唐浩凯
学 号 : 181100436
指 导 教 师 : 王璿

2021 年 11 月 8 日

基于数字孪生的立体仓库设备健康管理系统

摘要

近年来,互联网技术的发展与应用使得许多传统的工业设备管理方法已经不再适应企业的发展需要,市场上急需更先进、更高效的设备管理方法。而随着中国制造 2025 发展战略、工业 4.0、工业互联网的推进,涌现出了一批新的信息技术,其中数字孪生技术作为智能制造关键使能技术之一,受到了国内外的广泛关注。

本文在基于数字孪生模拟出的立体仓库车间环境下分别实现了 PC 端与移动端的设备健康管理系统。首先,分析了数字孪生和设备健康管理系统的现状,根据实际应用需求,给出了基于数字孪生技术实现的立体仓库管理系统的总体框架。着重分析了设备健康管理方面的功能需求。

其次,基于数字孪生 3D 仓库场景,设计和实现了 PC 端的设备健康管理模块,包含设备信息实时查看与更改、设备故障实时告警、故障信息实时上报等功能。再次,根据用户需求,设计并实现了移动端的设备健康管理系统,增加了方便设备管理员使用的部分功能模块。该系统的应用能有效改善传统工业设备管理方法实时性差、直观性不强等缺点。

关键词: 数字孪生, 设备健康管理, 故障告警, Unity 3D

THREE-DIMENSIONAL HEALTH MANAGEMENT SYSTEM OF WAREHOUSE EQUIPMENT BASED ON DIGITAL TWIN

ABSTRACT

In recent years, the development and application of Internet technology makes many traditional industrial equipment management methods no longer adapt to the development needs of enterprises, the market is in urgent need of more advanced and more efficient equipment management methods. With the development strategy of Made in China 2025, industry 4.0 and industrial Internet, a number of new information technologies have emerged, among which Digital Twin technology, as one of the key enabling technologies of intelligent manufacturing, has received extensive attention at home and abroad.

In this paper, based on Digital Twin simulation of the three-dimensional warehouse workshop environment to achieve the equipment health management system including PC terminal and mobile terminal. Firstly, we analyzed the present situation of Digital Twin and equipment health management system. According to the actual application requirements, we gave the overall framework of stereo warehouse management system based on Digital Twin technology. The functional requirements of equipment health management are analyzed emphatically.

Secondly, based on the three-dimensional warehouse workshop environment, the PC equipment health management module is designed and implemented, including the real-time view and change of device information, real-time alarm of device failure, real-time reporting of fault information and other functions. Thirdly, according to the user demand, we made the design and implementation of mobile terminal equipment health management system. The application of the system effectively improves the

shortcomings of traditional industrial equipment management methods such as poor real-time performance and poor intuitiveness.

Key words: Digital Twin, equipment health management(EHM), fault warning, Unity

3D

目录

1	绪论	1
1.1	研究背景与意义	1
1.2	研究现状与问题	1
1.3	研究内容	2
1.4	论文章节安排	3
2	设备健康管理系统总体设计	4
2.1	需求分析	4
2.2	系统设计原则	4
2.3	系统技术架构	5
2.4	系统功能架构	6
2.5	系统开发环境与工具简介	7
2.6	数据库设计	8
2.7	本章小结	10
3	PC 端设备健康管理系统的设计与实现	11
3.1	PC 端系统详细设计	11
3.2	系统开发和运行环境	13
3.3	PC 端设备健康管理系统实现	13
3.3.1	设备运行状态查看功能	13
3.3.2	设备查看视角转换功能	14
3.3.3	设备故障实时告警功能	17
3.4	系统功能测试	18
3.5	本章小结	19
4	移动端设备健康管理系统的设计与实现	20
4.1	移动端设备健康管理的详细设计	20
4.2	系统开发和运行环境	22
4.3	移动端设备健康管理的实现	22



4.3.1 查看管理员个人信息功能	22
4.3.2 设备详细信息查看功能	24
4.3.3 设备故障实时告警功能	26
4.3.4 设备故障问题上报功能	30
4.3.5 管理员密码修改功能	32
4.4 系统功能测试	33
4.5 本章小结	34
5 总结与展望	35
参考文献	36
致谢	38
译文及原文	39

1 绪论

1.1 研究背景与意义

近年来,互联网技术的发展与应用使得许多传统的工业设备管理方法已经不再适应企业的发展需要,市场上急需更先进、更高效的设备管理方法。而随着中国制造 2025 发展战略、工业 4.0、工业互联网的推进,涌现出了一批新的信息技术,其中数字孪生技术作为智能制造关键使能技术之一,受到了国内外的广泛关注^[1]。

数字孪生的概念最初于 2003 年在美国密歇根大学提出,并被定义为立体模型,同时指代实际模型、虚拟模型以及二者之间的结合关系^[2]。数字孪生能够模拟物理实体在现实环境中的行为,为虚拟模型与现实实体中架起一道数据的桥梁^[3]。数字孪生的核心是模型和数据,强调仿真、建模、分析和辅助决策,侧重于物理世界对象在数据世界的重现,通过集成物理反馈数据,并辅以深度学习、数据挖掘和软件分析,在信息化平台内建立一个数字化模拟。这个模拟会根据实际反馈,随着物理实体的变化而自动做出相应的变化^[4]。

而基于数字孪生的立体设备健康管理系统即是将数字孪生技术与工业设备健康管理相结合^[5],利用数字孪生技术所产生的虚拟模型对工业设备的健康状态进行实时监控与数据分析,并根据分析结果提供相应建议,从而实现辅助管理者决策、提高生产效率与安全性的目标^[6]。

1.2 研究现状与问题

目前国内外对于数字孪生技术的应用还在不断地探索与发展中,目前应用较为广泛的领域有工业制造领域、能源领域、城市规划领域、教育领域以及航空航天领域^[4-8]。工业领域中多将数字孪生技术用于优化价值链与生产智能化^[9],如将数字孪生技术与设备管理技术相结合;能源领域中则多用于指导油田的实时挖掘与相应的数据分析^[10];城市规划领域多用于灾害预警与道路规划^[11];教育领域多用于智能教育与全息课堂^{[12][13]};航空航天领域则多用于航空发动机与光电

探测系统^{[14][15]}。

在工业制造领域的设备管理方面，传统的设备管理方法多以纯文本为载体，需要较多的人力进行设备记录的分类与管理，并且在实际的管理过程中往往存在设备信息查找与更改困难、对设备故障响应不及时、需要管理人员长时间驻守等问题^[16]。而数字孪生技术为解决上述问题提供了一个新的思路。目前国内外已有很多将数字孪生技术与工业生产相结合的例子，例如国网上海市电力公司浦东供电公司已经尝试将数字孪生技术与变电设备的运行维护相结合，提出了如何构建变电设备数字孪生系统的基本框架，解决了设备运维领域存在运维资源不足，设备因周期性检修导致过修、欠修等痛点问题^[17]。又例如武汉宏数信息技术有限责任公司提出的基于数字孪生的大坝施工智慧管理平台，将数字孪生技术与大坝施工管理系统相结合，使得该平台相较于传统平台具有了实时性、可视化及协同性等优势^[18]。

本文的目的便是数字孪生技术与设备的运行维护相结合，利用虚拟模型来更为全面、准确、及时地了解设备的健康状态，便于设备管理人员实施差异化、精细化的检修策略，从而降低管理成本，提高管理效率^[19]。

1.3 研究内容

本文的研究内容主要包括三个部分，首先研究了数字孪生技术在工业生产环境下的实现方法，本文针对某公司立体仓库的仓储需求，利用 Unity 3D 技术对立库中设备的运行状态进行实时监控。设备运行状态可以实时反馈到 Unity 模型中，存在故障的设备可以高亮显示其故障位置与可能的故障名称，并以文字形式提醒，从而为仓库管理员作出相应对策提供便利。

其次，设计并实现了 PC 端的设备健康管理系统，本文提出了基于 Vue.js 框架编写的设备健康管理系统，实现一条从 PLC 采集数据到服务器端存储与处理数据再到管理系统呈现数据的数据通路，从而将虚拟模型与 Web 系统结合起来，实现简化设备信息的查看与删改的效果，并在传统设备管理功能集里增加了设备故障实时告警功能，能够适应对需要对设备故障迅速响应的场合。

最后，为进一步响应企业对设备健康管理系统的的需求，本文提出了能与 PC 端设备健康管理系统互通的移动端设备健康管理系统。该系统旨在将设备管理人员从设备一线解放出来，在移动端可随时随地查看设备信息与告警信息，并可以

将管理员发现的问题上报到服务器端，使管理更加精细化与人性化。

1.4 论文章节安排

本文主要研究基于数字孪生技术实现设备的健康管理，一共分为五章，除去第一章绪论外，各章节内容安排如下：

第二章主要介绍设备管理系统的总体设计，从系统的业务逻辑出发，介绍系统的相应功能，并对服务器端的数据存储格式作出说明。

第三章主要介绍 PC 端设备健康管理系统的详细设计与实现，具体介绍了对应的设计背景、设计思路以及重要功能实现方式与效果。

第四章主要介绍移动端设备健康管理系统的详细设计与实现，对移动端系统的结构与功能进行了详细的设计，并展示了多个功能模块的实现方式与效果。

第五章是对全文工作的总结，以及对未来的展望。

2 设备健康管理系统总体设计

2.1 需求分析

本文通过对立体仓库设备健康管理系统的调查研究,再结合企业、一线管理人员的要求,对系统进行设计与开发。

经过研究,首先了解到市场上传统的设备管理系统大多使用纯文档的方式对设备的各式信息进行分类记录与管理,但这种管理方式对于故障处理速度要求较高的环境来说效果并不理想。其次,当前设备系统对设备的管理往往要求管理员长时间驻守在一线生产现场,因此耗费的时间与人力都较多。

在听取了一线管理人员的建议后,本文提出设计一个基于数字孪生的 PC 端与移动端通用的立体仓库设备健康管理系统来满足上述需求。通过利用数字孪生技术实现的仓库可视化监控可以实时查看设备运行的轨迹与状态,大大增强了设备状态信息更新的实时性与直观性,从而改善传统设备管理方法对故障响应不及时的缺陷。而分别设计 PC 端管理系统与移动端管理系统则是为了帮助管理人员随时随地获取最新的状态信息,从而打破对管理人员时间、空间上的制约,从而提升管理效率。

经过分析不难得出,管理员与设备是两个独立的模块,所以将系统的功能模块分为设备管理模块与管理员信息管理模块。设备管理模块可以提供包括查看设备状态、转换设备查看视角、设备故障实时告警、设备故障实时上报等功能;管理员管理模块则可以提供包括管理员账号密码管理、管理员个人信息修改、管理员工作数量计数等功能。

2.2 系统设计原则

设备健康管理系统的的主要设计原则如下:

(1) 软件工程设计思想

软件工程是一门研究如何更好地进行软件开发与架构的科学,强调系统化、规范化。本系统主要使用变量封装法,将设备管理模块与管理员个人管理模块的

各个功能单独编写并封装，使得被封装的部分可以重复利用，从而减少系统开发的工作量，提高开发效率。

(2) 模块化设计思想

模块化设计是指将一个大的设备健康管理系统切分成设备信息查看模块、查看设备视角转换模块、设备故障实时告警模块、设备故障实时上报模块、管理员个人信息管理模块、管理员账号密码管理模块、管理员工作计数模块等数个模块，在完成这些功能模块后再将其拼装成一个完整的系统。这样的设计思想可以使系统保持较好的封装性，同时使系统开发者更好地理解系统。

(3) 浏览器-服务器开发思想

B/S 的开发模式比较灵活，在开发设备健康管理系统时考虑到企业的实际需要，选用 B/S 模式能够更好地满足系统使用与维护方面的需求。

2.3 系统技术架构

本系统采用 B/S 架构，基于 Unity 3D、Vue.js、Node.js 等框架，以 Java 为主要开发语言实现了包含表现层、业务逻辑层、数据访问层的三层架构体系。三层架构的具体作用如下：

表示层：表示层是三层架构体系中离用户最近的一层，主要包含 Web 页面与 Unity 虚拟模型。页面通过 Vue.js 框架编写，使用了 HTML、CSS、JavaScript 等开发语言；Unity 虚拟模型则通过 Unity 3D 开发工具编写，使用 Java 作为主要开发语言。该层主要负责系统与用户的交互，将系统按需求呈现在用户面前。

业务逻辑层：业务逻辑层采用 JavaScript 语言编写，主要包含 Web 服务器与数据库服务器，负责将表示层与数据访问层联系起来。

数据访问层：数据访问层主要负责 Web 系统对数据库数据的访问与调用，底层采集到的数据经过数据访问层梳理后，反馈到客户端上。

当 PLC 读取到实时设备与告警数据后，通过 3001 端口服务器分别传输给 Web 系统、Unity 系统以及 MySQL 数据库系统。Web 系统将通过 http 协议收到时设备数据、告警数据以及库存业务数据，Unity 系统将这些数据以 WebView 插件的形式呈现在 Web 系统里，而 MySQL 数据库系统则会将收到的数据分门别类地储存在事先设计好的数据表里。

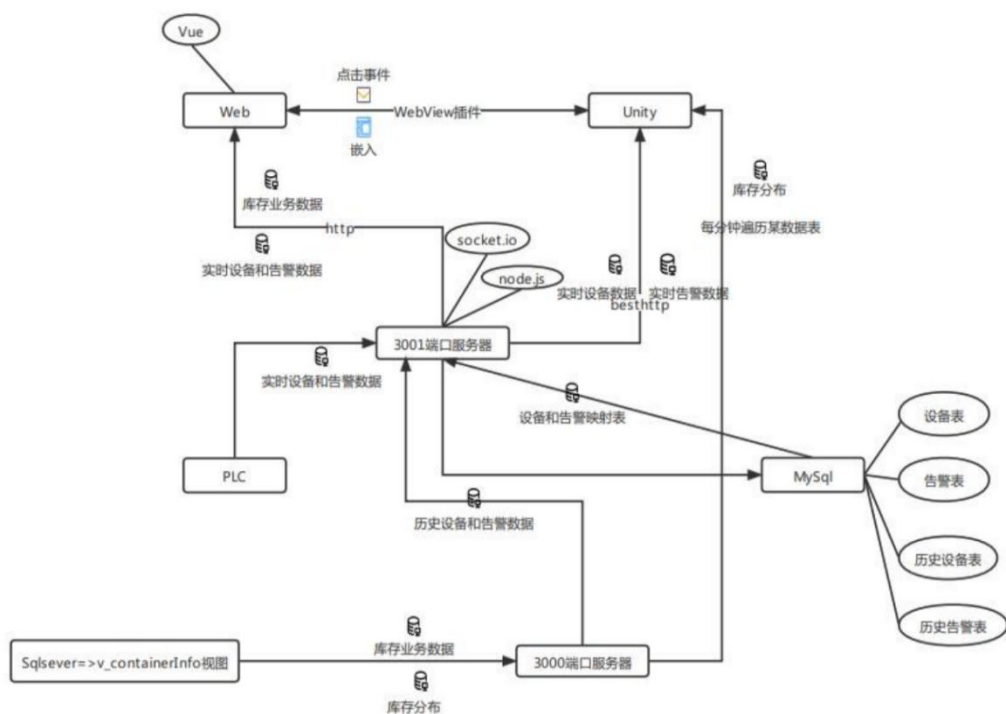


图 2-1 系统总体设计框架

2.4 系统功能架构

功能结构图以树状图的形式对系统的功能进行描述,可以直观地反应各个部分模块之间的联系。通过用户的业务需求,图 2-2 给出了立体仓库设备健康管理系统的功能结构图。

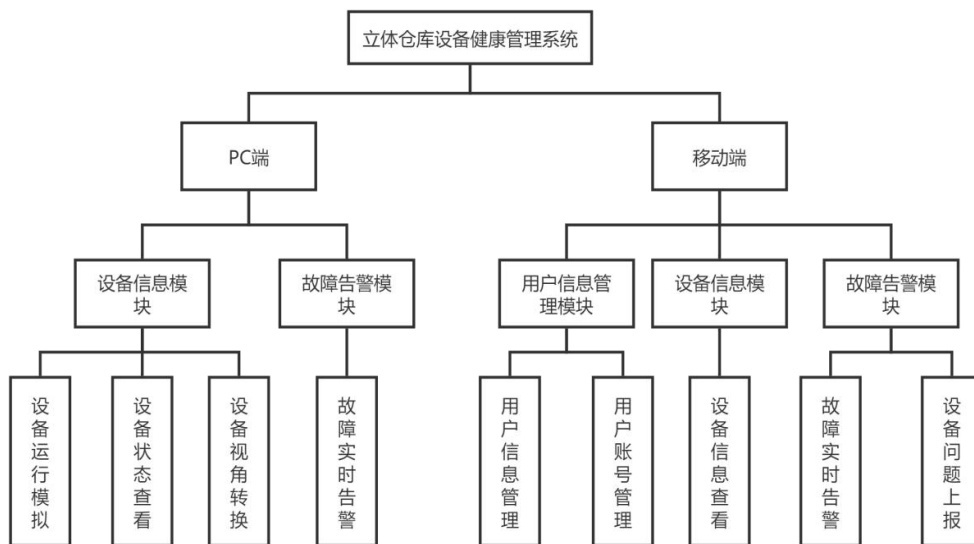


图 2-2 立体仓库设备健康管理系统功能架构

PC 端设备健康管理系统主要分为设备信息模块和故障告警模块两大模块。

设备信息模块主要负责查询并显示设备的各种状态信息，包括设备运行模拟、设备状态实时查看、设备查看视角转换等功能；故障告警模块负责实时显示设备传来的故障信息，主要包括故障的实时告警功能。

移动端设备健康管理系统主要分为用户信息管理模块、设备信息模块和故障告警模块三大模块。用户信息管理模块主要负责管理用户的相关信息，包含用户个人信息管理、用户账号管理两大功能；设备信息模块主要负责查询设备的详细信息与运行状态，包括设备状态实时查看功能；故障告警模块主要负责同步 PC 端设备健康管理系统的设备故障信息并将其显示在移动端设备健康管理系統上，同时将需要上报的故障信息上传至服务器储存，包括故障的实时告警与故障信息上报功能。

2.5 系统开发环境与工具简介

仓库立体设备健康管理系统是在 Windows10 系统平台下，基于 Vue.js 与 Node.js 框架并使用 Visual Studio Code 进行开发。数据存储方面则使用了 MySQL 数据库与 Navicat 数据库管理工具。

(1)VUE Vue 是一套用于设计并实现前端页面的 JavaScript 框架。Vue 框架的特点主要体现在可以实现自底向上的逐层应用。Vue 的库支持也十分丰富，虽然 Vue 自身携带的库重点在视图层的开发，但 Vue 框架也支持与其他库进行合并开发。此外，Vue 框架也是开发单页面富应用的重要手段之一。作为设备健康管理系统的载体，Vue 框架能在开发过程中省去许多重复性的劳动，提高开发效率，因此是很好的选择。

(2)Node.js Node.js 的发布距今已有十余年历史，是一个较为成熟的 JavaScript 运行环境。Node.js 的主要特点是能够将本来较为繁琐的服务器开发语言替换成较为简洁的 JavaScript 语言，从而简化服务器端的开发。Node.js 编写了一些特殊的 API 用以优化自身在各种环境下的运行速度，同时 Node.js 能够很好地支撑在如 Chrome 等主流浏览器下的开发与运行，这也意味着 Node.js 框架对主要使用 Chrome 内核开发的本系统来说具有很好的适配性。用 Node.js 编写服务器端代码可以使代码更加简洁的同时提高服务器的响应速度，因此成为本文的首选。

(3)Visual Studio Code Microsoft 在 2015 年 4 月 30 日 Build 开发者大会上正

式宣布一个运行于 Mac OS X、Windows 和 Linux 之上的，针对于编写现代 Web 和云应用的跨平台源代码编辑器，可在桌面上运行，并且可用于 Windows, macOS 和 Linux。它具有对 JavaScript, TypeScript 和 Node.js 的内置支持，并具有丰富的其他语言（例如 C++, C#, Java, Python, PHP, Go）和运行时（例如 .NET 和 Unity）扩展的生态系统。因为该编辑器完美地支持了本次开发所需要的语言与环境，因此成为本文开发编辑器的首选。

(4)MySQL MySQL 是最流行的关系型数据库管理系统之一，MySQL 对数据实施分类式管理，即根据用户需要将数据分门别类后储存在不同的数据表中，而在体积较小的单个数据表中作查询将会比包含所有数据的大数据表更快，这样的设计有利于使数据库响应速度更快。

MySQL 使用的是数据库系统中较为流行的 SQL 语言。MySQL 的“小、快、灵”特点对于数据量不大的网站开发来说，能够在实现基本功能的同时节省许多开发成本，所以 Mysql 在网站开发领域得到了广泛的应用。由于本文提出了设备健康管理系统的体量并不大，因此选择了相对小巧且速度快的 MySQL 作为数据库系统。

(5)Navicat 一套可创建多个连接的数据库管理工具，用以方便管理 MySQL、Oracle、PostgreSQL、SQLite、SQL Server、MariaDB 和 MongoDB 等不同类型的数据库，并支持管理某些云数据库，例如阿里云、腾讯云。Navicat 作为数据库管理工具来说体量较小、功能齐全且操作简单，使用者的学习成本并不高。Navicat 的用户界面 (GUI)设计良好，能够为设备管理系统在数据管理方便简化许多操作，同时使数据更加直观，提升了开发效率，因此选用为本系统的数据库管理工具。

2.6 数据库设计

本系统使用 MySQL 作为数据库系统，来完成有关数据的相应操作。具体的数据库设计将以数据表的形式说明，详情如下：

(1) 用户个人信息表

用户个人信息表包含的是设备管理员的登录信息，包含信息序号、管理员账号、密码、昵称、职位、工号、个人简介、年龄、性别、累计处理故障次数等，具体如表 2-1 所示。

表 2-1 用户个人信息表

列名	中文名称	数据类型	长度	允许空
id	信息序号	int	0	Not null
name	管理员账号	varchar	8	Not null
password	密码	varchar	255	Not null
number	工号	varchar	20	Null
title	职位	varchar	10	Null
description	个人简介	varchar	255	Null
age	年龄	varchar	8	Null
sex	性别	varchar	8	Null
handleWarningNumber	累计处理故障次数	varchar	100	Null

(2) 设备问题上报表

设备问题上报表包含的是当设备出现故障时，设备管理员认为设备出现的需要上报的问题，包括问题序号、维修工单号、设备编号、设备名称、报修时间，具体如表 2-2 所示。

表 2-2 设备问题上报表

列名	中文名称	数据类型	长度	允许空
id	问题序号	int	0	Not null
number	维修工单号	varchar	20	Null
deviceNumber	设备编号	varchar	40	Null
deviceName	设备名称	varchar	40	Null
time	报修时间	varchar	20	Null

(3) 设备详情信息表

设备详情信息表包含设备的主要基本信息，立库中有升降机、子母车托盘线等设备，每种设备的详细信息包括信息序号、设备编号、设备名称、规格型号、设备类型、所属车间、安装地点、负责人、联系方式、生产厂家、供应商、购买时间、使用时间，这些信息各自设计了不同的数据长度以及是否允许为空值，具体如表 2-3 所示。

表 2-3 设备详情信息表

列名	中文名称	数据类型	长度	允许空
id	信息序号	int	0	Not null
number	设备编号	varchar	40	Null
name	设备名称	varchar	40	Null
size	规格型号	varchar	40	Null
type	设备类型	varchar	40	Null
warehouse	所属车间	varchar	40	Null
address	安装地点	varchar	40	Null
people	负责人	varchar	8	Null
contract	联系方式	varchar	11	Null
producer	生产厂家	varchar	20	Null
provider	供应商	varchar	20	Null
buyTime	购买时间	varchar	20	Null
useTime	使用时间	varchar	20	Null

(4) 设备故障告警表

设备故障告警表包含设备发生故障时传送给服务器进行存储的主要信息，包括信息序号、设备编号、设备名称、规格型号、发生时间、设备状态，具体如表 2-4 所示。

表 2-4 设备故障告警表

列名	中文名称	数据类型	长度	允许空
id	信息序号	int	0	Not null
number	设备编号	varchar	40	Null
deviceNumber	设备名称	varchar	40	Null
type	规格型号	varchar	40	Null
time	发生时间	varchar	20	Null
state	设备状态	varchar	20	Null

2.7 本章小结

本章从树形图和组织结构图的形式出发，主要介绍系统的总体设计框架，分别介绍了系统的设计原则、技术架构、功能架构和数据库设计。



3 PC 端设备健康管理系统的设计与实现

3.1 PC 端系统详细设计

PC 端设备健康管理系统的的主要目的是方便一线管理人员实时查看仓库设备的运行状态和健康情况，同时将发生的故障信息及时、准确地传递给管理人员，因此 PC 端系统的主要模块有设备信息查看模块和故障信息告警模块，包含设备状态查看、设备查看视角转换、设备故障实时告警等功能。

PC 端设备健康管理系统主要设计了两种类，一种是业务逻辑类，包括 index 页面路由逻辑类和 socket 通信类，另一种是实体类，包括用于登录的 login 类、展示实时告警页面的 header 类、用于视角转换的 navigate 类、用于查看设备信息的 stata 类、用于展示故障信息的 warn 类等，该部分的类图如图 3-1 所示。

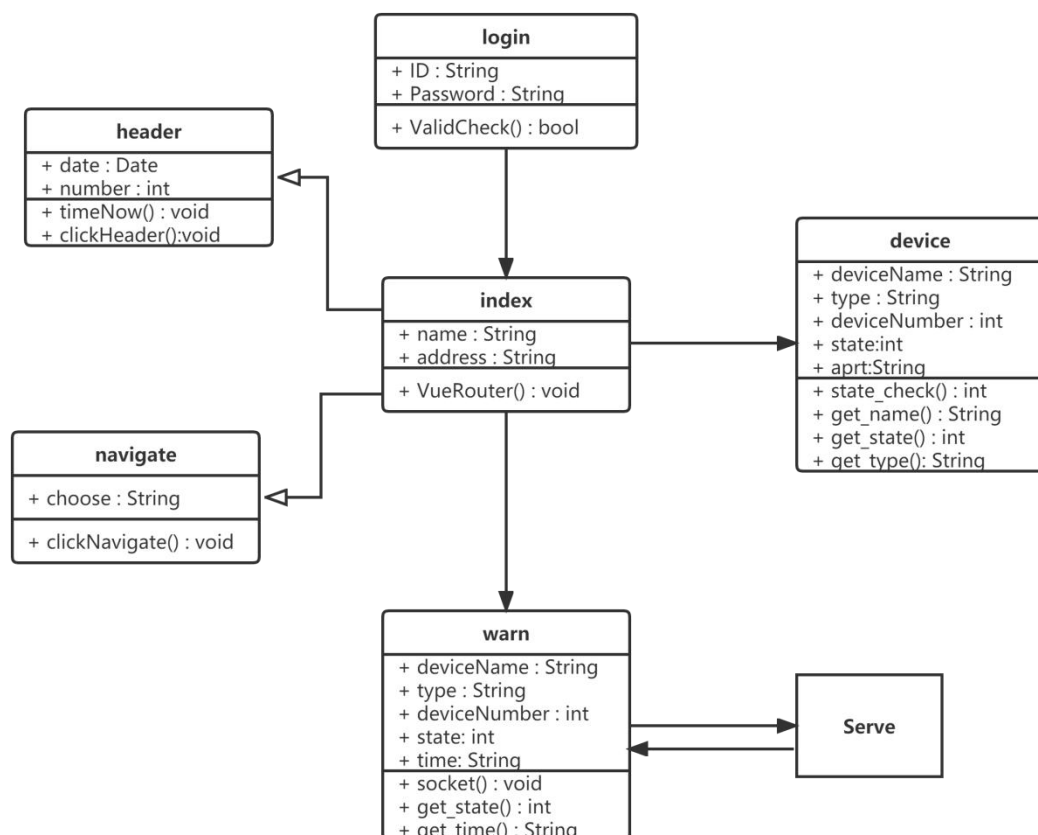


图 3-1 PC 端设备管理系统类图

Index 路由类收到请求后，通过路由类里定义的路由路径，跳转到对应的页面，与各种底层方法进行交互，以实现对各部分功能的完成，同时由 socket 类负责对服务器和数据库的通信，时序图如图 3-2、图 3-3 所示。

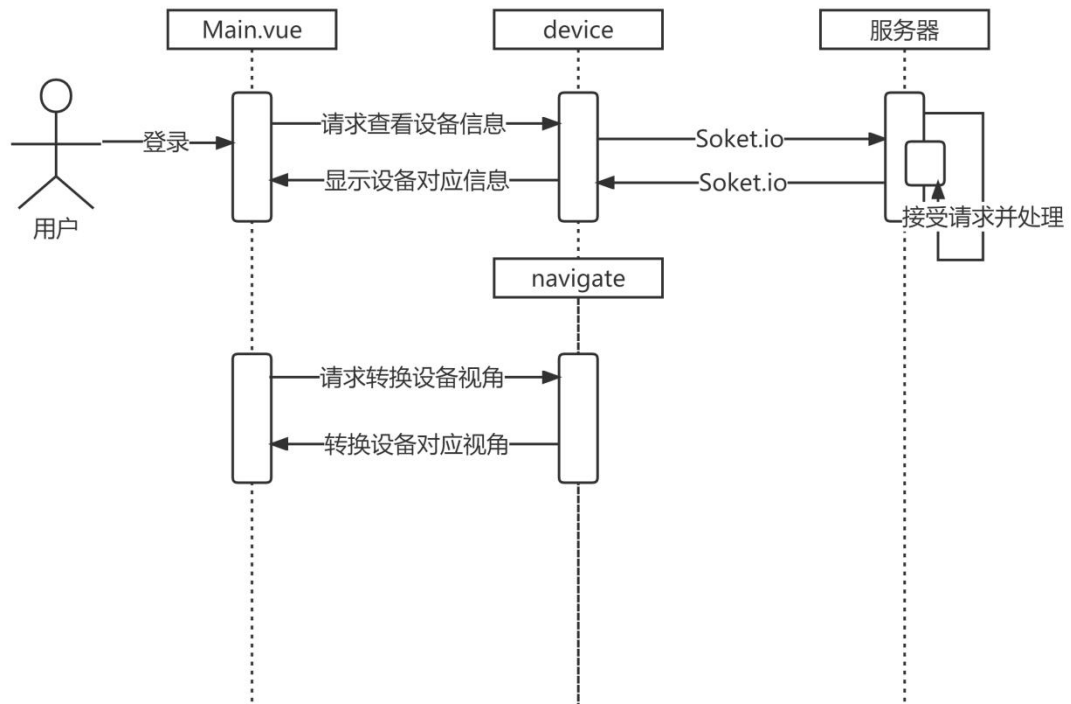


图 3-2 PC 端设备管理系统查看设备与转换视角功能时序图

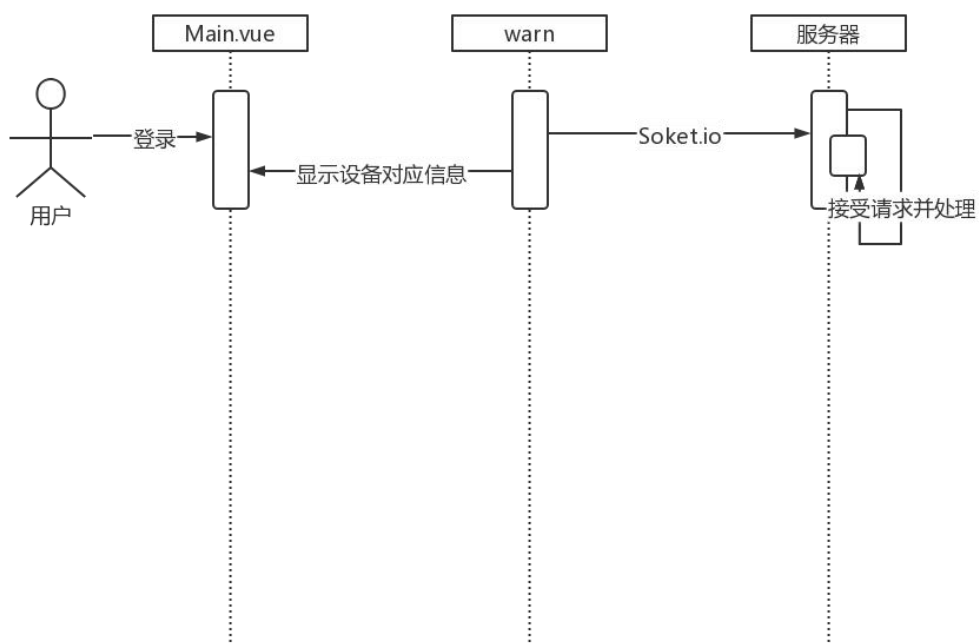


图 3-3 PC 端设备管理系统故障告警功能时序图

管理员请求查看设备信息的步骤为：管理员进入系统登录界面，输入正确的账号密码后，进入到系统主页面 `Main.vue`，管理员根据自身需要点击对应设备，系统根据点击的设备向 `device` 类发出请求，收到请求后，系统调用 `socket` 类向服务器发出请求，服务器返回对应的数值给 `device` 类，再由 `device` 类反馈给主页面 `Main.vue`，从而显示出对应信息。

管理员请求转换查看设备视角的步骤为：管理员进入到系统主页面 `Main.vue`，管理员根据自身需要点击所需视角，系统根据点击的视角向 `navigate` 类发出请求，收到请求后，系统调用 `navigate` 类中的方法将视角转换，并反馈给主页面 `Main.vue`，从而显示出对应视角。

管理员收到故障实时告警的步骤为：管理员进入到系统主页面 `Main.vue`。此时，`warn` 类如果发现设备运行故障，便向服务器上报数据，服务器收到数据后调用系统方法将故障数据录入数据库，同时 `warn` 类将故障数据反馈给主页面 `Main.vue`，从而实现故障的实时告警功能。

3.2 系统开发和运行环境

(1) 系统开发环境

PC 端仓库立体设备健康管理系统是在 Windows10 系统平台下，基于 Vue.js 框架、Node.js 框架以及 Unity 3D 框架使用 Visual Studio Code 进行开发。

(2) 系统运行环境

PC 端仓库立体设备健康管理系统的服务器端运行环境如下：

- 1) 硬件要求：固态硬盘 256G 以上，10 代 i5 或同级别 AMD 及以上；
- 2) 软件要求：Mysql V5.7 版本，Nginx V1.18.0 版本，Node.js V14.16.1 版本。

仓库立体设备健康管理系统的 PC 端运行环境如下：

- 1) 硬件要求：一般 PC；
- 2) 软件要求：Windows10 操作系统，Chrome 9+/Firefox 4/IE 6.0 及以上版本。

3.3 PC 端设备健康管理系统实现

仓库立体设备健康管理 PC 端系统的主要功能包括设备运行状态查看、设备查看视角转换、故障实时告警。

3.3.1 设备运行状态查看功能

仓库管理员成功登录进入管理系统主界面，管理系统主界面如图 3-4 所示。

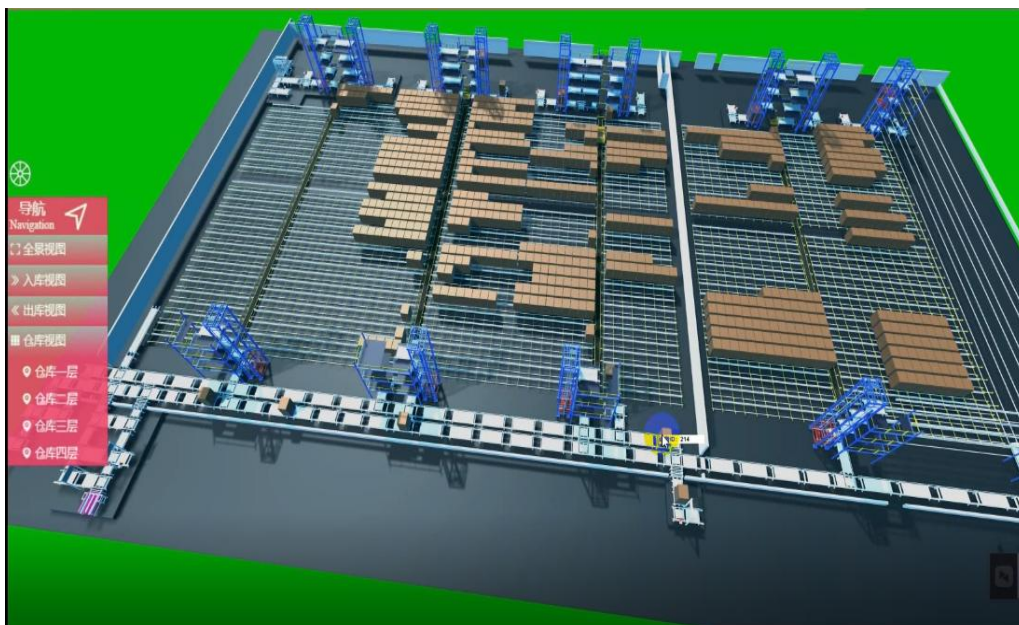


图 3-4 PC 端设备管理系统主界面

仓库管理员成功进入管理系统主界面后，双击需要查看的设备的虚拟模型，即可进入到设备实时运行状态界面，其中状态图标为红色代表运行异常，状态图标为黄色代表运行正常，如图 3-5 所示。该功能的具体实现方式是通过 Unity 中的 WebView 插件获取对应设备状态信息，从而将其显示在 Web 系统中。



图 3-5 PC 端设备管理系统设备运行状态界面

3.3.2 设备查看视角转换功能

仓库管理员成功登入系统主界面后默认以全景视角查看所有设备，如图 3-4

所示，如果管理员需要以不同的视角来查看设备，只需点击主界面左侧的导航界面，根据需要选择对应的视角即可，视角切换导航菜单如图 3-6 所示。



图 3-6 PC 端设备管理系统设备视角转换导航菜单界面

仓库管理员可以自由选择入库或出库视图，如图 3-7、3-8 所示。



图 3-7 PC 端设备管理系统设备入库视角界面

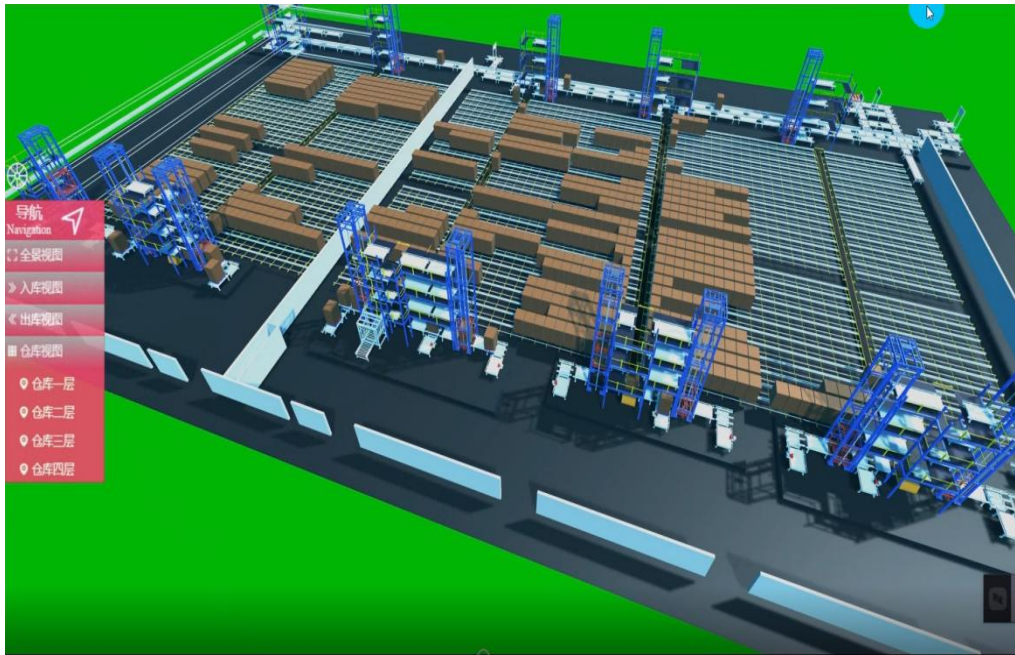


图 3-8 PC 端设备管理系统设备出库视角界面

该功能的具体实现方式如表 3-1 所示。

表 3-1 PC 端设备管理系统设备视角转换关键代码表

```
clickNavigate(e) {  
    let s = e.target.innerText;  
    if (s === "全景视图") this.handleView("view0");  
    else if (s === "入库视图") this.handleView("view1");  
    else if (s === "出库视图") this.handleView("view2");  
    else if (s === "仓库一层") this.handleView("layer1");  
    else if (s === "仓库二层") this.handleView("layer2");  
    else if (s === "仓库三层") this.handleView("layer3");  
    else if (s === "仓库四层") this.handleView("layer4");  
},
```

首先编写用于和 Unity 界面交互的 `handleView()` 函数，该函数可以根据传入参数的不同调用系统函数 `sendMessageToCSharp()` 函数来改变 Unity 模型视角。

其次编写 Web 前端中的点击事件响应函数 `clickNavigate()`，当用户点击导航栏中不同选项时，`clickNavigate()` 函数将给 `handleView()` 函数传入对应的参数来实现视角转换效果。

3.3.3 设备故障实时告警功能

仓库管理员可以通过点击系统主页面右上方的实时告警版块来检查是否有设备发出了告警，告警版块如图 3-9 所示。

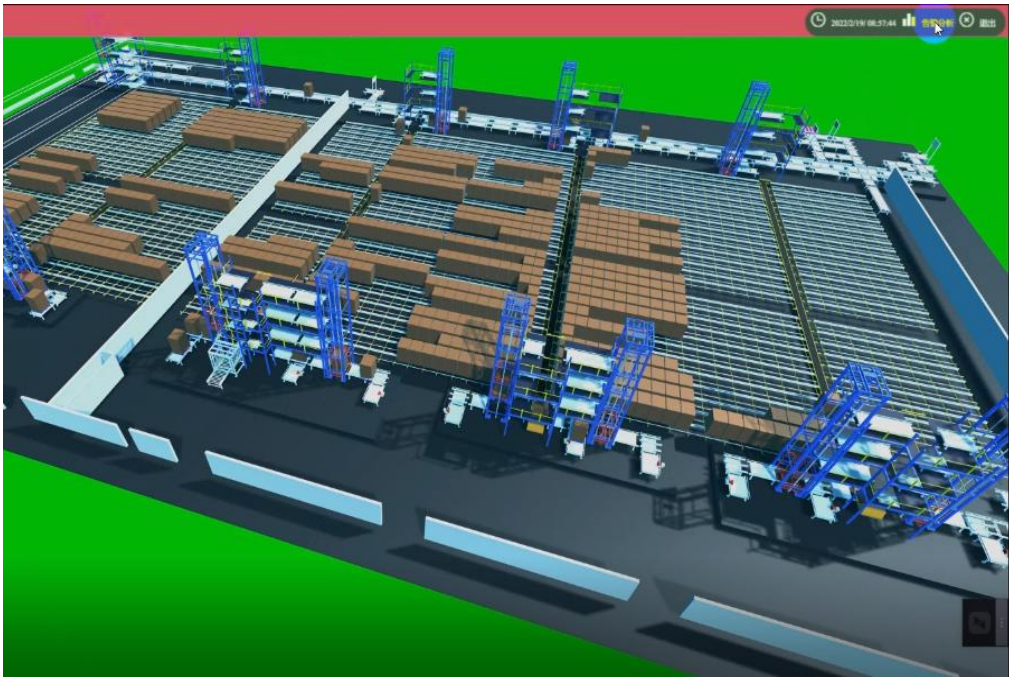


图 3-9 PC 端设备管理系统设备告警版块界面

仓库管理员单击“告警分析”后，会进入告警消息列表界面，如图 3-10 所示。



图 3-10 PC 端设备管理系统设备告警消息列表界面

在该界面可以查看故障设备的告警序号、设备类型、告警设备编号、告警类型等信息，方便仓库管理员作出针对性的措施，具体实现方法如表 3-2 所示。

表 3-2 PC 端设备管理系统设备告警关键代码表

```
if (window.vuplex) this.addMessageListener();  
  
else window.addEventListener("vuplexready", this.addMessageListener);  
  
this.sockets.subscribe("WarningInfo", (data) => {  
  
    this.warnData = data.data;  
  
    this.time = this.timeNow();  
  
}); //获取时间  
  
this.sockets.subscribe("states", (data) => {  
  
    this.states = data;    //获取状态信息  
  
});  
  
this.sockets.subscribe("goods", (data) => {  
  
    this.goods = data;    //获取货物信息  
  
});  
  
},
```

例如当 PLC 采集到某提升机产生异常数据并将其发送至 3001 端口时，一直监听 3001 端口上信息的 addMessageListener() 函数将通过 sockets 类将请求获取对应信息并转存到变量中，最终被 Web 端显示出来。

3.4 系统功能测试

本次系统功能测试方式选用了黑盒测试^[20]。黑盒测试的特点就是无需了解系统内部代码结构，只需观察对应操作得到的结果是否与预期结果符合，从而判断软件功能是否正常。黑盒测试的专业要求不如白盒测试高，对于本系统的实际应用状况来说，黑盒测试的方式更为适合，同时也能为企业的测试人员减少负担。

基于管理员的角色，对 PC 端设备健康管理系统的主要功能的测试，具体如表 3-3 所示。

表 3-3 PC 端设备管理系统主要功能测试用例表

测试标题	PC 端设备管理系统主要功能测试用例表
参与角色	系统管理员

续表 3-3

测试对象 设备信息查看功能、设备查看视角转换功能、设备故障实时告警功能			
测试项	测试内容	测试步骤	测试结果
1	对升降机的状态进行查看	找到升降机虚拟模型并双击	系统成功显示升降机的各零件状态
2	转换系统查看设备的视角	找到系统主页面左侧的导航栏，单击“全景视图”按钮	系统视角成功切换到全景鸟瞰图
3	转换系统查看设备的视角	找到系统主页面左侧的导航栏，单击“入库视图”按钮	系统视角成功切换到设备入库视角
4	转换系统查看设备的视角	找到系统主页面左侧的导航栏，单击“出库视图”按钮	系统视角成功切换到设备出库视角
5	转换系统查看设备的视角	找到系统主页面左侧的导航栏，单击“仓库一层”按钮	系统视角成功隐藏其它层仓库设备，只显示仓库一层设备
6	接收升降机故障告警信息	关闭升降机升停止检测传感器	系统成功接收到升降机故障告警信息

3.5 本章小结

本章从 PC 端设备健康管理系统出发，分别介绍了系统详细设计思路、具体实现效果以及系统包含的各功能详细介绍。因为 PC 的性能相较移动端更为优秀，所以 PC 端系统更加注重设备的实时运行模拟效果，功能方面有许多与 Unity 模型相结合的地方，这也使得该系统能够更快地响应故障以及更直观地查看设备信息。



4 移动端设备健康管理系统的设计与实现

4.1 移动端设备健康管理的详细设计

在设计 PC 端系统的过程中，了解到企业在实际管理过程中存在管理场地受限、人力成本较高、设备故障处理不及时等问题，为了改善上述问题，本文追加提出了移动端设备健康管理系统。移动端设备健康管理系统的的主要目的是方便管理人员随时随地查看仓库设备的运行状态和健康情况，并且能够将 PC 端收到的故障信息同步传递给移动端系统，从而尽可能减少故障信息传递给管理人员所需的时间、空间限制，从而提升管理效率，降低人力成本。

移动端设备健康管理系统设计了两种类，一种是业务逻辑类，包括 router 页面路由逻辑类和 API 接口类，另一种是实体类，包括用于登录的 login 类、构成主页面框架的 home 类、负责底部菜单显示的 layouts 类、用于登录错误显示的 public_404 类、用于保存个人信息的 account 类、用于展示设备信息的 detail 类和 list 类、用于修改密码的 password 类、用于故障问题上报的 question 类等，该部分的类图如图 4-1 所示。

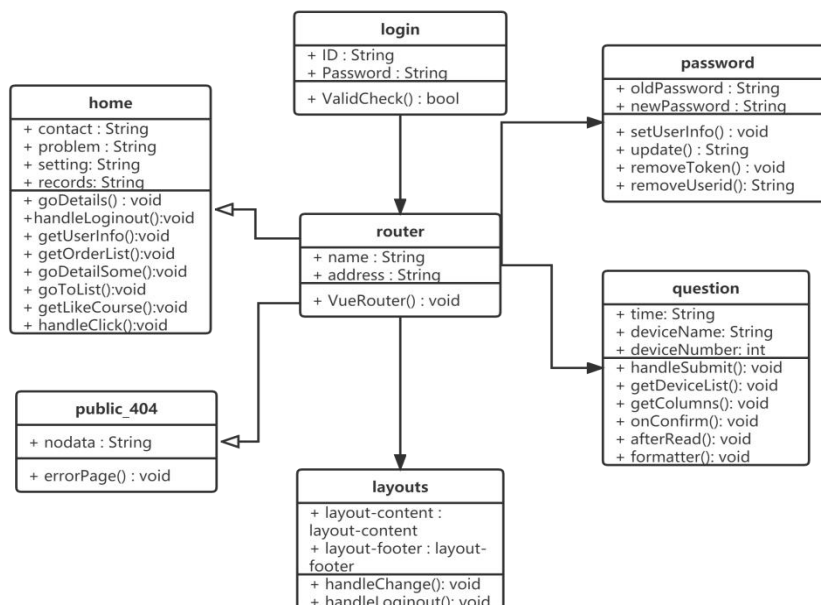


图 4-1 移动端设备管理系统类图

router 路由类收到请求后，通过路由类里定义的路由路径，跳转到对应的页面，与各种底层方法进行交互，以实现设备信息显示和故障信息的告警，同时由 api 类负责对服务器和数据库的通信，时序图如图 4-2、图 4-3 所示。

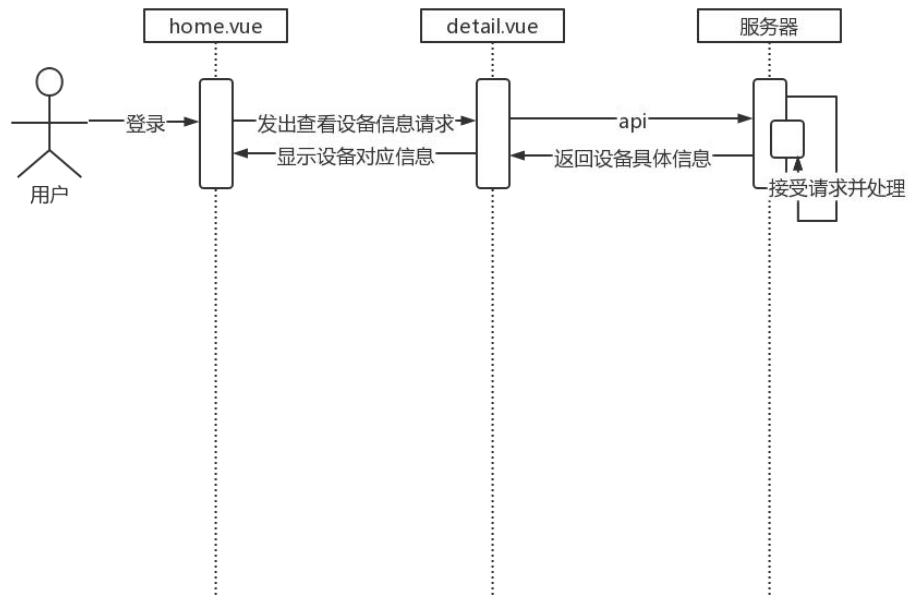


图 4-2 移动端设备管理系统查看设备信息功能时序图

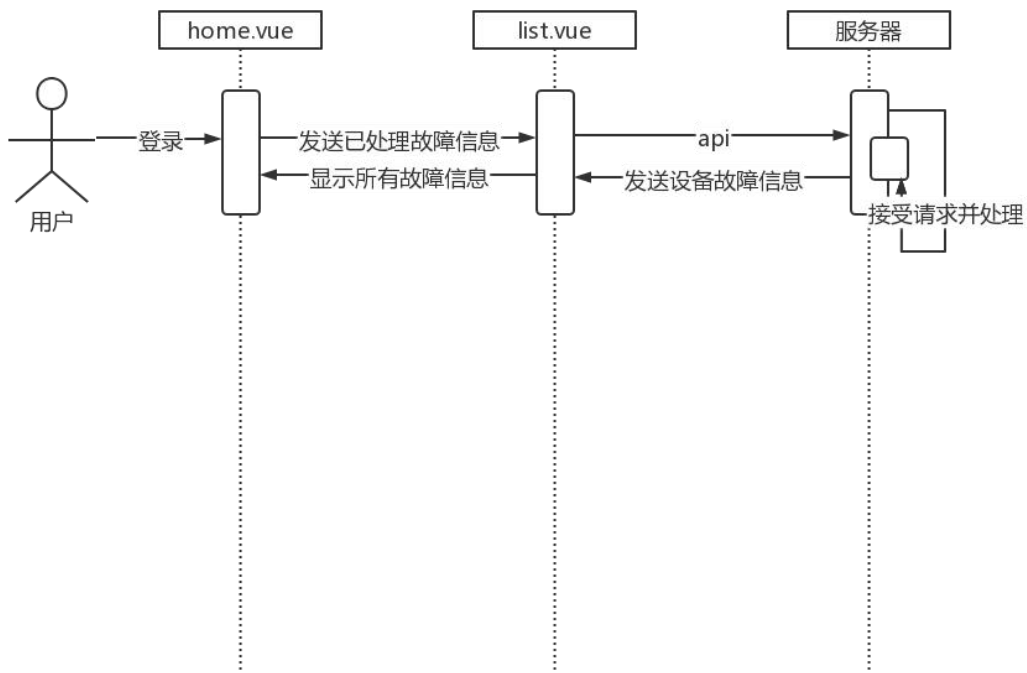


图 4-3 移动端设备管理系统故障告警功能时序图

管理员请求查看设备信息的步骤为：管理员进入系统登录界面，输入正确的账号密码后，进入到系统主页面 `home` 的下属界面 `about.vue`，管理员单击“设备列表”按钮，系统根据点击的按钮跳转到 `detail.vue` 页面并向服务器发出请求，收到请求后，服务器返回对应的数值给 `detail` 类，再由 `detail` 类反馈给主页面 `device.vue`，从而显示出对应信息。

管理员处理设备故障告警信息的步骤为：管理员进入系统登录界面，输入正确的账号密码后，进入到系统主页面 `home` 的下属界面 `about.vue`，管理员单击“未维修工单”按钮，系统根据点击的按钮跳转到 `list.vue` 页面并向服务器发出请求。收到请求后，服务器返回对应的数值给 `list` 类，再由 `list` 类反馈给主页面 `device.vue`，从而显示出对应信息，同时管理员可以点击 `device.vue` 页面中的“已处理”按钮，向 `list.vue` 页面发出已处理故障的信息，`list.vue` 页面再向服务器发送对应信息，服务器收到信息后更新数据库并发送更新是否成功信息。`list.vue` 根据服务器反馈的信息，将故障信息处理后的结果反馈到 `device.vue` 页面上，从而显示出对应信息。

4.2 系统开发和运行环境

(1) 系统开发环境

移动端仓库立体设备健康管理系统是在 Windows10 系统平台下，基于 Vue.js 框架、Node.js 框架并使用 Visual Studio Code 进行开发。

(2) 系统运行环境

仓库立体设备健康管理系统的移动端运行环境如下：

- 1) 硬件要求：普通智能机；
- 2) 软件要求：Android/iOS 操作系统，Safari V14.0.3 版本及以上。

4.3 移动端设备健康管理的实现

仓库立体设备健康管理移动端系统的主要功能包括查看管理员个人信息、设备详细信息查看、故障实时告警、故障问题上报、修改管理员密码。

4.3.1 查看管理员个人信息功能

管理员首先需要经过如图 4-4 的登录界面成功登入管理系统后，进入到系统主界面。



图 4-4 移动端设备管理系统登录界面

单击下方“我的”按钮，进入个人页面，再单击“个人信息”按钮，即可查看管理员个人信息，如图 4-5、4-6 所示。



图 4-5 移动端设备管理系统个人主页界面



图 4-6 移动端设备管理系统个人信息界面

个人信息页面中包含了管理员的编号、管理员账号、职位、工号、个人简介、年龄、性别、累计处理故障次数等个人信息。这些信息对应储存在表 2-1 的用户个人信息表中，移动端设备健康管理系统可以将前端对应的组件与数据库信息进行数据绑定，通过连接数据库后对各信息进行查询或更改，然后将其显示在前端页面中。

4.3.2 设备详细信息查看功能

进入个人页面后，单击“设备列表”按钮，即可查看现有设备信息，如图 4-7 所示。



图 4-7 移动端设备管理系统设备列表界面

设备列表界面包含了设备的一些基本信息，包括设备编号、工单编号、设备名称、设备规格、设备信息更新时间以及设备负责人等信息。单击设备列表中的任意设备即可进入设备详情页面，如图 4-8 所示。



图 4-8 移动端设备管理系统设备详情界面

设备详情界面包含了设备的详细信息，包括设备编号、设备名称、设备状态、设备规格、设备类型、所属车间、安装地点、负责人、联系方式、生产厂家、供应商设备等信息。该页面可以帮助管理人员快速确认设备运行状况，提高管理效率。具体获取信息方式如表 4-1 所示。

表 4-1 移动端设备管理系统获取设备详情关键代码表

```
goDetailsSome(item){  
    console.log('item', item);  
    this.$router.push(`/detail/${item.id}`)  
}
```



```
async getLikeCourse(){
    this.likeService.listQuery.pageNum ++
    await getAll({}).then(res => {
        if (res.code === 100) {
            //获取设备信息表
            this.likeService.list = this.likeService.list.concat(res.data)
            this.likeService.loading= false
            if(res.data.count > this.likeService.list.length){
                this.likeService.finished = true//设备信息传输完毕
            }else{
                this.likeService.finished = true
            }
        } else {
            this.$notify(res.message)//显示获取失败信息
        }
    })
}
```

首先编写 likeService 类用于获取服务器存储的各种信息以及确认数据传输是否完毕。其中 likeService.list 获取服务器设备信息表，likeService.finished 用于确认数据是否传输完毕。

然后编写 goDetailsSome()函数用于从服务器获取所需设备的详细状态表，并保存在 res 变量中输出。

4.3.3 设备故障实时告警功能

管理员登录成功后，单击下方“告警”按钮可以进入设备告警页面，如图 4-9 所示。设备故障实时告警页面会统计本月所有告警数量，并将已处理的告警信息与未处理的告警信息分开显示，单击“所有告警数量”可查看所有告警信息，如图 4-10 所示。



图 4-9 移动端设备管理系统设备告警信息界面



图 4-10 移动端设备管理系统设备所有告警信息界面

该页面会显示设备告警编号、设备类型、设备名称以及告警类型，同时配有处理确认按钮，方便管理员管理告警信息。在图 4-9 界面中点击“未处理告警数量”即可查看还未处理的设备故障信息，如图 4-11 所示。



图 4-11 移动端设备管理系统设备未处理告警信息界面

在此页面点击“确认处理”按钮后，已处理告警统计将对应增加，同时管理员的累计处理故障数也会同时增加，如图 4-12、4-13 所示。



图 4-12 变化后的告警信息界面



图 4-13 变化后的个人信息界面

告警功能具体的实现方式如表 4-4 所示。

表 4-4 设备告警功能关键代码表

```
goDetailsSome(item){  
    console.log();  
}  
  
async getLikeCourse(){  
    this.likeService.listQuery.pageNum ++  
    await getAll({}).then(res => {  
        if (res.code === 100) {  
            if (this.filterState) { //过滤信息，只将异常数据存入 list 表中  
                this.likeService.list = res.data.filter(d => d.state === this.filterState)  
            } else {  
                this.likeService.list = res.data  
            }  
        }  
    })  
}
```

```

this.likeService.loading= false

    if(res.data.count > this.likeService.list.length){
        this.likeService.finished = true//信息传输完毕
    }else{
        this.likeService.finished = true}
    } else {
        this.$notify(res.message)//显示获取失败信息
    }
    })
}

```

首先编写 likeService 类用于获取服务器存储的各种信息以及确认数据传输是否完毕，其中 likeService.list 获取服务器设备信息表，likeService.finished 用于确认数据是否传输完毕。当设备数据出现异常时，首先通过 filterState 方法过滤设备信息，只将需要的故障信息存入 likeService.list 表中，等待后续方法的请求。然后编写 goDetailsSome()函数用于从服务器获取过滤后的设备故障状态表，并保存在 res 变量中输出。

4.3.4 设备故障问题上报功能



图 4-18 问题上报界面

在实际的设备管理过程中可能会出现不能短时间内解决的设备故障，需要向有关部门上报后再商定处理措施。本文为此增设设备故障问题上报功能。管理员可在图 4-9 界面中点击问题上报，进入如图 4-18 所示的问题上报界面。

该界面包含需上报设备的一些基本信息，包括维修工单号、设备编号、设备名称、报修时间等，点击“提交问题”按钮后，服务器将会将其存入数据库中，如图 4-19 所示。

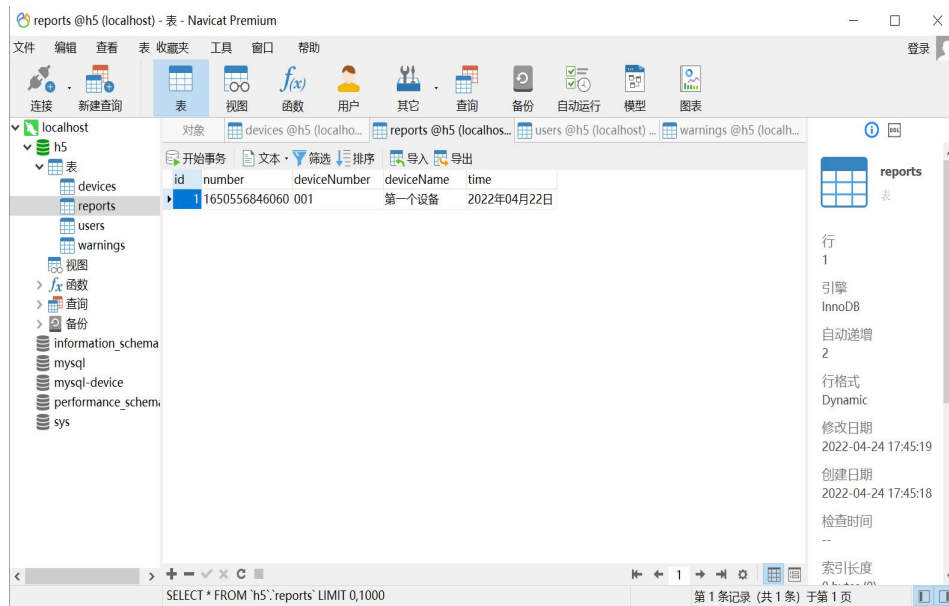


图 4-19 问题上报后数据库存储界面

该功能的具体实现如表 4-5 所示。

表 4-5 问题上报功能关键代码表

```
async handleSubmit() {
    await add({
        ...this.data,
        time: this.time,//上传时间
        deviceNumber: this.deviceNumber,//上传设备编号
    }).then(res => {
        if (res.code === 100) {//判断传输是否成功
            this.$notify('上报成功！！')
            this.handleLoginOut();
        } else {
```

```
this.$notify(res.message)//输出错误信息
```

```
}
```

```
})
```

```
},
```

首先移动端设备健康管理系统将前端组件与后端数据表 2-2 对应数据项绑定。系统与数据库连接成功后，当用户点击“提交问题”按钮时，触发响应函数 `handleSubmit()`，获取用户在各个组件里的输入，并将其对应保存在数据表 2-2 中。

4.3.5 管理员密码修改功能

出于管理员账号安全考虑，系统增设了管理员密码修改功能。管理员可在图 4-9 界面点击“修改密码”按钮，即可进入密码修改页面，如图 4-20 所示。



图 4-20 管理员密码修改界面

界面主要包含管理员账号名称、旧的密码、新的密码、确认密码等信息，用户可以在此更改自身管理员账号与密码，系统在对照新旧密码不同并确认两次输入的新密码相同后将修改后的账号密码传输至表 2-1 中，同时注销账户，等待用户重新登录。

4.4 系统功能测试

基于管理员的角色，对移动端设备健康管理系统的主要功能的测试，具体如表 4-7 所示。

表 4-7 移动端设备健康管理系统主要功能测试用例表

测试标题 移动端设备管理系统主要功能测试用例表			
参与角色 系统管理员			
测试对象 设备信息查看功能、设备故障实时告警功能、设备问题上报功能、管理员个人信息查看功能、管理员密码修改功能			
测试项	测试内容	测试步骤	测试结果
1	对 1 号设备的信息进行查看	单击底部导航栏“我的”按钮， 点击页面中“设备列表”按钮	系统成功显示设备的详细信息
2	查看设备所有故障告警信息	单击底部导航栏“告警”按钮， 点击“所有告警数量”按钮	系统成功显示所有设备故障告警信息
3	查看设备未处理告警信息	单击底部导航栏“告警”按钮， 点击“未处理告警数量”按钮	系统成功显示未处理故障告警信息
4	确认处理设备未处理故障告警信息	单击底部导航栏“我的”按钮， 点击页面中“待办问题”按钮， 点击 1 号告警信息	系统成功显示已处理，并对应更改设备故障信息表
5	对暂时不能处理的故障信息进行上报	单击底部导航栏“我的”按钮， 点击页面中“问题上报”按钮， 填写对应故障信息	系统成功更新故障信息上报表
6	查看管理员个人信息	单击底部导航栏“我的”按钮， 点击页面中“个人信息”按钮	系统成功显示管理员各项个人信息
7	修改管理员登录密码	单击底部导航栏“我的”按钮， 点击页面中“修改密码”按钮， 填写旧密码 123123，填写新密码 1234 后单击“确认修改”按钮	系统成功返回登录页面，只有以 1234 为密码时才能成功登录

4.5 本章小结

本章从移动端系统出发,分别介绍了系统的具体实现效果以及系统包含的各功能详细介绍。移动端较 PC 端来说性能有限,因此系统设计得较为轻量化,更加注重设备信息的传递与管理,所以设计了许多方便管理员进行实际管理的功能,也是对 PC 端系统功能的一种补充。

5 总结与展望

本文详细地描述了立体仓库设备健康管理系统从分析到实现的过程。下面是对本文的工作的一个总结：

立体仓库设备健康管理系统运用了模块化的设计思想，通过将这个大的系统划分成 PC 端系统、移动端系统这两个较小的系统，实现了分模块的开发设计。其中 PC 端系统分成三个模块：设备信息查看模块、设备视角转换模块、设备故障告警模块，分别实现管理员对设备信息的管理，对查看设备视角的管理以及对设备故障信息的管理；移动端系统分为五个模块：管理员信息管理模块、设备详细信息管理模块、设备故障告警模块、设备问题上报模块和管理员密码修改模块，分别实现管理员对自身信息的管理，对设备信息的管理，对设备故障信息的管理、对设备维修信息的管理以及对管理员账号安全的管理。通过这样的细化，实现对系统功能的开发与实现。

由于时间与水平所限，系统仍有许多能够改善的地方，例如 PC 端系统功能可以更为丰富，多设计一些方便用户使用的界面与功能，又例如移动端的界面可以更为美化，代码可以更为高效，从而提升系统的相应速度等。这些问题都是日后系统可以改善的重点。

参考文献

- [1] 周帅昌,刘丽兰,高增桂.数字孪生车间的模型与数据采集优化[J].工业控制计算机,2021,34(12):14-16.
- [2] 龙绪明,黄昊,闫明,顾晓青,龙震,李魏俊.数字孪生电子智造工厂的实现[A].四川省电子学会、四川省电子学会 SMT/MPT 专业委员会.2021 中国高端 SMT 学术会议论文集[C],2021:210-216.
- [3] Corallo Angelo et al. Shop Floor Digital Twin in Smart Manufacturing: A Systematic Literature Review[J]. Sustainability, 2021, 13(23) : 12987-12987.
- [4] 赵亮,许娜,张维.我国数字孪生研究的进展、热点和前沿——基于中国知网核心期刊数据库的知识图谱分析[J].实验技术与管理,2021,38(11):96-104.
- [5] 陶飞,张贺,戚庆林,张萌,刘蔚然,程江峰,马昕,张连超,薛瑞娟.数字孪生十问:分析与思考[J].计算机集成制造系统,2020,26(01):1-17.
- [6] 苏新瑞,徐晓飞,卫诗嘉,等.数字孪生技术关键应用及方法研究[J].中国仪器仪表,2019(7):47-53.
- [7] Jiang Yuchen et al. Industrial applications of digital twins[J]. Philosophical Transactions of the Royal Society A, 2021, 379(2207) : 20200360-20200360.
- [8] Eric J Tuegel et al. REENGINEERING AIRCRAFT STRUCTURAL LIFE PREDICTION USING A DIGITAL TWIN[J]. Aerospace Engineering, 2011, DOI:10.1155/2011/154798.
- [9] 胡秀琨,张连新.数字孪生车间在复杂产品装配过程中的应用探索[J].航空制造技术,2021,64(03):87-96.
- [10] Thumeera R. Wanasinghe et al. Digital Twin for the Oil and Gas Industry: Overview, Research Trends, Opportunities, and Challenges[J]. IEEE Access, 2020, 8 : 104175-104197.
- [11] Al Sehrawy Ramy and Kumar Bimal and Watson Richard. A DIGITAL TWIN USES CLASSIFICATION SYSTEM FOR URBAN PLANNING & CITY INFRASTRUCTURE MANAGEMENT[J]. JOURNAL OF INFORMATION TECHNOLOGY IN CONSTRUCTION, 2021, 26 : 832-862.
- [12] 朱珂,张莹,李瑞丽.全息课堂: 基于数字孪生的可视化三维学习空间新探[J].远程教育杂志,2020,38(04):38-47.
- [13] 王璐,张兴旺.面向全周期管理的数字孪生图书馆理论模型、运行机理与体系构建研究[J].图书与情报,2020,(05):86-95.
- [14] 孙惠斌,颜建兴,魏小红,常智勇.数字孪生驱动的航空发动机装配技术[J].中国机械工程,2020,31(07):833-841.
- [15] 任涛,于劲松,唐荻音,时祎瑜.基于数字孪生的机载光电探测系统性能退化建模研究[J].航空兵器,2019,26(02):75-80.
- [16] 本刊编辑部.数字孪生: 数字化转型新方向[J].信息技术与标准化,2021,(11):1.
- [17] 黄鑫,汤蕾,朱涛,张弛,顾力,万轶伦,张毅洲.数字孪生在变电设备运行维护中

- 的应用探索[J].电力信息与通信技术,2021,19(12):102-108.
- [18] 邓院林,陈敏,王伟.基于数字孪生的大坝施工智慧管理平台[J].人民长江,2021,52(S2):302-304+311.
- [19] 庄重,李宇舟,李阳.基于数字孪生的设备大数据智能运维平台构建[J].四川建筑,2021,41(04):211-213.
- [20] 张泽平. 本科毕业论文管理系统的设计与实现[D]. 山东:山东师范大学,2018.

致谢

当落笔至此，代表着我的大学剩余即将告一段落。以后的生活仍然充满着未知和挑战，但不论如何，我都十分感谢陪我一起走过这一段人生路的人们。

首先我要向我的导师王璿老师表示感谢。在本科期间我曾选修过王老师的Linux 系统课程，在学习的过程中多次遇到问题，每次向王老师请教，王老师总是和蔼可亲、不厌其烦地向我伸出援手。后来在毕业设计时又一次有幸得到王老师的指导，毕业设计的内容对我来说还是颇有挑战性，很多知识是本科时未曾深究的，这也导致在进行毕业设计的过程中我多次遇到了难以解决的问题，而王老师仍然是我在代码的汪洋里迷惘时的一盏明灯，无论我何时向老师请教，老师总会向我提出宝贵的意见。王老师事务繁多却对我的设计定期探问，这使我非常感动，也更加敬重王老师。

然后，我要向我的辅导员张政老师致谢。在本科期间，张老师无数次地为同学们费心指导，无论在生活上还是学习上我认为张老师都是一位负责又认真的好导师。毕业设计恰逢疫情期间，许多以往的经验都已不再适用，张老师在这个时候为我提供了许多学业与择业方面的指导，我十分感谢。

同时我要向李纪奇学长以及我的室友们致谢。四年时光转瞬即逝，感谢这一路的陪伴，使我校园中的日日夜夜都不孤单。虽然专业不同，但在一起度过的快乐回忆不会更改，祝愿 4027 的各位和学长都有美好的前程！

最后，我要向我的家人致以最深沉的谢意。感谢你们在我人生的二十余年中为我遮风挡雨，才使我能健康快乐地成长至今。大学毕业在即，希望我能早些为你们减轻负担，让你们更加快乐地生活。

在此，祝愿大家一生和乐安康，幸福美满！