目录

[摘 要 2](#_Toc68164182)

[第1章 绪论 3](#_Toc68164183)

[1.1研究背景 3](#_Toc68164184)

[1.2国内外研究现状 4](#_Toc68164185)

[1.3本文研究内容 5](#_Toc68164186)

[1.4论文结构 6](#_Toc68164187)

[第2章 相关理论与技术 6](#_Toc68164188)

[2.1 Web前端相关技术 6](#_Toc68164189)

[2.2 服务器端相关技术 8](#_Toc68164190)

[2.3 手机端相关技术 10](#_Toc68164191)

[2.4 流媒体相关技术 10](#_Toc68164192)

[2.5 相关工具 11](#_Toc68164193)

[2.6 本章小结 12](#_Toc68164194)

[第3章 系统需求分析 12](#_Toc68164195)

[3.1 功能需求分析 12](#_Toc68164196)

[3.1.1 用户角色分析 12](#_Toc68164197)

[3.1.2 系统功能 15](#_Toc68164198)

[3.2系统流程分析 16](#_Toc68164199)

[3.3 可行性分析 17](#_Toc68164200)

[3.4 系统的性能需求分析 18](#_Toc68164201)

[3.5 本章小结 19](#_Toc68164202)

[第4章 系统总体设计 19](#_Toc68164203)

[4.1 功能模块设计 19](#_Toc68164204)

[4.1.1 应急演练直播模块 20](#_Toc68164205)

[4.1.2 应急演练录播模块 21](#_Toc68164206)

[4.1.3 应急演练交互模块 22](#_Toc68164207)

[4.1.4 应急演练配置模块 23](#_Toc68164208)

[4.1.5 应急演练评价模块 23](#_Toc68164209)

[4.1.6 用户管理模块 24](#_Toc68164210)

[4.2 数据库设计 25](#_Toc68164211)

[4.3 界面设计 28](#_Toc68164212)

[4.3.1 手机端界面设计 28](#_Toc68164213)

[4.3.2 Web前端界面设计 31](#_Toc68164214)

[4.4 系统架构设计 33](#_Toc68164215)

[4.4.1 系统物理架构设计 33](#_Toc68164216)

[4.4.2 系统架构设计 34](#_Toc68164217)

[4.5 本章小结 35](#_Toc68164218)

[第5章 系统实现与测试 35](#_Toc68164219)

[5.1 系统功能实现 36](#_Toc68164220)

[5.1.1 应急演练直播模块 36](#_Toc68164221)

[5.1.2 应急演练录播模块 38](#_Toc68164222)

[5.1.3 应急演练交互模块 40](#_Toc68164223)

[5.1.4 应急演练配置模块 42](#_Toc68164224)

[5.1.5 应急演练评价模块 43](#_Toc68164225)

[5.1.6 用户管理模块 47](#_Toc68164226)

[5.2 系统测试 48](#_Toc68164227)

[5.2.1 测试环境 49](#_Toc68164228)

[5.2.2 测试方案 49](#_Toc68164229)

[5.2.3 测试报告 50](#_Toc68164230)

[5.3 系统运行环境与系统部署 51](#_Toc68164231)

[5.4 本章小结 57](#_Toc68164232)

[第6章 总结与展望 57](#_Toc68164233)

[6.1 工作总结 57](#_Toc68164234)

[6.2 研究展望 58](#_Toc68164235)

# 摘 要

随着国家不断地发展，社会不断地进步，突发事件日益成为公众关注的焦点。在应对突发事件的过程中，各类应急处置人才发挥着不可替代的作用。随着国家对应急管理的重视，我国对应急管理人才队伍建设也越来越重视，在国家公共安全体系建设和应急管理能力方面，急需培养更多具有应急管理知识和能力的高校人才。在人才的培养模式中，应该整合传统教育教学方式、强化实践能力、增强学生风险危机意识、突发公共事件的应对及管理能力。

因此，高校迫切需要建设应急演练辅助教学系统，该系统具备自定义应急演练、自定义评价指标、记录应急演练情况、多窗口观看应急演练直播、实时交互指挥和应急演练事后评价等功能，可以有效地为应急管理教学实践提供支撑。

本文通过对应急演练辅助教学系统进行系统需求分析，确定了系统的功能需求，将系统划分为应急演练直播模块、应急演练录播模块、应急演练交互模块、应急演练配置模块、应急演练评价模块和用户管理模块等六个模块。然后对系统的整体流程进行分析，介绍用户角色和系统之间的协作关系，随后是可行性分析，从市场可行性和经济可行性对系统进行分析，确保了系统的可行性，接下来根据系统的性能需求明确了系统的技术选型和框架搭建，

然后对系统进行总体设计，

最后，在以上的基础上，并分模块对系统功能需求进行设计与实现，最终完成系统的开发和测试。

从整体角度来说，应急演练辅助教学系统可以分为Web前端、手机端、服务器端三个部分。在开发技术方面，该系统采用了B/S架构模式，前服务器端分离进行开发，使得Web前端和服务器端只进行数据的交互，提高开发效率。Web前端使用简单易用的Vue框架，服务器端使用Java语言结合Spring Boot框架和Mybatis框架进行编程实现，手机端使用Uni-app框架。流媒体服务器采用Nginx结合RTMP模块进行实现，具有简单易用、高性能、支持高并发等特点， 本系统界面简洁大方，经过一系列的测试表明系统功能完善，运行平稳，支持多用户并发推流直播，可以有效的锻炼学生应对灾害的处置能力，为应急演练教学方面提供支持。

**关键字：**应急演练；教学系统；流媒体；直播；Spring Boot；

# 第1章 绪论

## 1.1研究背景

我国作为世界上地震灾害最严重的国家之一，公众对政府特别是地震部门在应对地震灾害时提供相应的震前防御预报、地震监测和震后应急指挥救援有着很高的要求。十五期间，中国建设了国务院抗震救灾总部技术体系和31个省级抗震救灾总部技术体系，“十二五”期间完成了国家地震社会服务工程项目，实现了多部门联动协调机制，国家和省抗震救灾指挥部之间的信息沟通和数据交换渠道。地震作为一种自然灾害，是不可避免的，我们所能做的就是尽量减少地震造成的损失。在过去的几十年里，中国发生过多次地震。据统计，唐山地震造成24万多人死亡，16.4万多人受伤。大多数人都曾在2008年体验过汶川地震。汶川地震是近年来中国破坏性最大的地震，造成17万多人失踪，6.9万多人死亡，37.5万多人受伤。2010年青海和玉树的地震玉树造成2600多人死亡，270人死亡失踪了。那个2011年云南营口地震造成25人死亡，314人受伤。虽然这两个数字不大，但有近10万间房屋倒塌。从这些数据中，我们可以看出地震是无情的。我们只能面对地震，挑战地震。为了最大限度地减少地震造成的损失，需要采取方便有效的应对措施。

在应对突发事件的过程中，各类应急人员发挥着不可替代的作用。随着国家对应急管理的重视，我国越来越重视应急管理人才队伍的建设。在国家公共安全体系建设和应急管理能力建设方面，迫切需要培养更多具有应急管理知识和能力的高校人才。在人才培养模式上，要整合传统的教育教学方法，强化实践能力，增强学生的风险危机意识，提高学生的应对和管理能力。

近年来，手机的普及率逐渐提高。到2013年，我国用户超过11亿，手机普及率高达84.9%，50%的城镇居民拥有智能手机，其中 69%用户每天会使用智能手机访问互联网，移动手机成为人们日常生活中必不可少的工具。美国皮尤研究中心发表的数据报告显示，中国的智能手机普及率达到 58%。智能手机的发展，移动媒体和5G网络发展使得存在可能性。

在高校传统的应急管理教学中，以课堂教学为主，主要教授应急管理的理念，应急演练实践内容较少，学生很难将课堂中学到的知识灵活运用到实践中，以及深刻体会其重要性，并且传统的应急演练成本较高，组织执行流程复杂，同时很难记录应急演练现场执行情况，对于教师事后评价应急演练的执行情况更是难以执行。因此，高校迫切需要建设应急演练辅助教学系统，使得教师可以自定义应急演练，学生根据应急演练相关内容进行实地演练，使用方便快捷的手机实时传输应急演练执行画面，教师通过浏览器实时观看多窗口直播内容，掌控全局应急演练执行情况，同时可在应急演练的执行过程中，进行交互指挥，并且可以事后进行复盘评价等功能。本文拟采用手机结合流媒体技术，利用B/S架构实现应急演练辅助教学系统，帮助高校应急教学实践开展，为应急演练教学提供系统支持，这对于应急管理人才培养是非常有意义的。

## 1.2国内外研究现状

1966年邢台地震是我国现代地震工作新时代的开始。随着四十多年的发展，我国的地震应急措施也是经历了翻天覆地的变化，取得了实质性的进展。应急指挥系统的起步阶段，成小平（1998）提出以全国地震计算机网络系统为支撑环境，对地震速报信息快速响应[1]。王孝铣（2000）介绍了福建省地震局采用100Base-TX快速以太网、TCP/IP通信协议等技术建成的福建地震应急指挥系统，在1999年地震紧急状态期间，也得到应用和考验[2]。王公学（2002）指出震后应急的时间尺度以及震后应急与震后救灾的区别与联系。从科学决策的意义上讲[3]，聂高众（2002）则提出地震应急基础数据库是各级地震部门开展应急工作的基础，是各级政府履行地震应急指挥职能的基本保障[4]。范灵春（2003）利用便携式电脑和h。323标准以及ISDN等技术设计的视频会议系统，为地震救灾指挥辅助决策提供了有效的依据[5]。魏艳华（2005）利用地理信息系统（GIS）技术应用于地震应急工作，为政府人员作出程式化的响应提供了科学依据[6]。侯建盛（2005）根据日本新潟县7.0级城市直下型地震震后救灾行动，提出统一协调的指挥机构、快速便捷的信息系统是防震减灾的关键[7]。管友海（2006）基于空间决策支持技术，设计与开发了地震应急对策软件，将地震灾害信息生动的可视化，有效提高了地震应急的效率和响应速度，对全国大中城市地震应急指挥技术系统的建设具有示范意义[8]。陈强（2008）则提出利用Google Earth的平台，发布公众地震信息，进行救援路线导航与跟踪，开展地震应急工作是高效率、低成本的[9]。宋立军（2009）利用VSAT卫星系统、海事卫星（128K）终端、GPRS/CDMA路由器等技术车载地震现场应急指挥技术系统，在汶川8.0级特大地震灾害发生后，为地震现场应急指挥技术发挥了重要作用[10]。赵晓林（2010）提出利用GNSS（卫星导航定位）技术和北斗卫星开展地震应急工作[11]。陈琳（2010）开发了基于手机短信的地震灾情速报系统，以震级为基础，以震中为圆心，划分对应半径的人员进行地震信息的收集，在Arcmap平台上生成灾情速报图，得到了广泛地应用[12]。范灵春（2011）则首次采用GIS平台——Super Map GIS，开发了西安市地震应急指挥技术系统，全面提升了地震应急综合能力[13]。张晖（2012）采用Nagios监控系统，应用于天津地震应急系统，全面提升了地震应急指挥系统的工作效率[14]。陈飞（2016）利用VOIP技术结合地震应急指挥车，极大地加强前后方联系并有效节省通信费用，并将其应用于江苏省地震局[15]。郁璟贻（2019）结合MCU高清视频会议系统，提高了应急联动协调能力，在四川省、云南省等地震多发省份通过三方级联模式在地震应急指挥中取得了较好的减灾实效[16]。张方浩（2019）提出地震现场情况复杂，传统的视频会议系统会出现音视频传输卡顿、不同步、声音飘忽等问题，云计算、移动互联网技术在视频会议中能有效缓解这题，同时提出地震应急信息服务系统云化是未来发展的趋势[17]。

Hisashi Matsumoto（2013）针对东日本大地震后航空医疗救灾活动的经验指出，医生直升机机队对日本的灾难医疗援助和救援活动至关重要[18]。Kazuhiko KINOSHITA（2012）则指出政府和地方社区合作实现抗灾通信和网络的重要性[19]。Alex Greer M.S。（2012）在对比日本和美国的地震应急体系后提出在很好的缓解地震带来的损失之后，建立一个弹性的社会，一个人性的社会，能够吸收灾难的影响并迅速恢复是至关重要的[20]。Symposium提出分布式无线网状网络（WMN）应用于医疗紧急响应的可行性[21]，Jennifer　Wilson等人认为，一个完整的应急管理组织结构应该涵盖五个层次的应急管理机构，包括联邦政府、州政府、县政府、市政府和社区。该机构能够连接政府首脑、联邦调查局、中央情报局、国防部等部门和工作人员。Corriveau等人通过各级应急管理机构实现跨层次、跨部门的合作[22]，从危机管理信息流的角度，系统地研究了政府跨部门应急合作的决策与信息。他们指出，有效的信息沟通和信息共享在政府跨部门应急协调中发挥着重要作用。Gloria的Simo和Bies认为，跨部门协作可以弥补单一部门应对措施的不足，在应急方面具有明显优势。同时，他们也强调非政府组织的重要作用[24]。Louise K.comfort和Naim kapucu从复杂适应系统的角度研究了信息交换和设备共享过程中各部门的协同作用，指出应对突发事件依赖于多个部门的合作[25]。艾辛格在研究应急管理政府的跨部门协调时，提出要建立联邦政府与州政府、地方各级政府及其部门之间的密切合作关系[26]。钱虎和纳伊姆·卡普库梳理了美国应急管理从联邦集权到跨部门协作的发展历程，总结了“9·11”事件后美国政府构建的应急管理协作框架，并通过该框架测试了波士顿马拉松爆炸事件中跨部门协作的有效性[27]。佩里·希克斯认为，造成政府部门间协调障碍的原因，不是专业化、差异化等内部问题，而是不同政府部门之间发展的碎片化。由于政府部门各自为政，难以协同工作。为了解决政府部门分散的问题，他提议建立一个完整的政府[28]。EVA Sorenson等人分析了公共部门合作的五大潜在障碍，即家长式工作方式、零错误管理、政政分离、群体思维和边界争端[29]。迈克尔·麦奎尔（Michael McGuire）和罗伯特·阿格拉诺夫（Robert Agranoff）在借鉴美国应急管理实践的基础上，提出了六种政府部门协同管理的假设模式：自上而下的管理模式、基于管辖权的管理模式、适度管理模式、接受捐赠管理模式，保守管理模式和满意管理模式[30]。

综上所述，本文针对应急演练教学的系统功能需求，基于成熟的Vue、Uni-app、Spring Boot等框架，设计与实现了应急演练辅助教学系统，以满足应急演练教学需求，可以锻炼学生的应对灾害的处置能力，为高校应急演练教学方面提供支持。

## 1.3本文研究内容

本文通过对高校教师和学生进行需求调研，详细研究了高校教师在应急演练实践课程开展过程中的业务需求，从市场可行性和技术可行性进行分析之后，确定了应急演练辅助教学系统的可行性，然后对应急演练辅助教学系统进行了功能需求分析，包括用户角色分析、系统用例分析、系统流程分析。接着进行了性能需求分析，在此基础上，对系统进行了整体的架构设计和功能模块设计，最后对系统进行了设计与实现、部署及测试，测试结果表明本文所设计与实现的应急演练辅助教学系统能够满足高校教师和学生在应急演练实践教学方面的需求。

本文的主要研究内容如下：

1. 需求调研。通过对高校教师和学生进行需求调研，结合相关文档，对系统进行了系统需求分析，详细分析了在应急演练过程中所需要完成的功能，形成明确的需求文档。
2. 确定技术框架。在系统需求分析的基础上，确定了系统的整体架构，采用前后端分离的思想，使得Web前端和服务器端只有数据的交互。服务器端采用面向对象的Java语言为基础，结合成熟的Spring Boot框架和持久层框架Mybatis，使我们只需专注于业务代码的实现，极大地提高了开发效率。Web前端采用Vue框架，双向绑定的特点使得业务代码简洁高效。手机端采用uni-app框架，一套代码，多端运行，极大地减少了学习成本和开发成本。流媒体服务器采用Nginx结合RTMP模块实现，支持高并发和稳定低延迟，使系统更加健壮。
3. 系统设计与实现。在确定了系统整体架构和技术选型之后，详细确定了各个模块的功能实现，包括Web前端、服务器端、手机端和流媒体服务器的设计方案。
4. 模块设计。对应急演练辅助教学系统的核心模块进行设计与实现，详细阐述核心模块的实现过程和细节，同时展示了系统中部分界面设计实现和的数据库设计。
5. 功能测试。主要是通过现有的测试方法和工具对系统进程测试，保证了应急演练辅助教学系统在功能和性能方面都能有效地满足应急演练教学方面的需求。

本文主要介绍了应急演练辅助教学系统地设计与实现过程，包括系统需求分析、系统架构设计、主要功能模块的实现、数据库设计、界面设计、系统实现部署和测试等过程。最后对本文的工作进行了总结，对系统的不足和未来发展做出展望。

## 1.4论文结构

第1章：绪论。主要描述了应急演练辅助教学系统的背景和意义、国内外研究现状和本文的主要工作内容。

第2章：相关理论与技术。本章主要介绍了本文系统开发所用到的技术、框架和相关开发工具。主要包括Java语言、Vue框架、Spring Boot框架、Uni-app框架、Nginx-RTMP流媒体服务器。

第3章：系统需求分析。在本章对系统进行了系统需求分析，可行性分析，包含市场可行性、技术可行性进行分析论证，并结合功能需求进行分析，分别从用户角色分析，系统用例分析，系统流程分析等，通过系统用例图进行分析论证，最后是系统的性能需求分析。

第4章：系统总体设计。本章中在上文的系统需求分析的基础上进行系统总体设计，包括系统架构设计和功能模块设计，最后是数据库设计和界面设计。

第5章：系统实现与测试。在本章中介绍了系统的运行环境和部署方案，并使用成熟的测试方案对系统的核心模块进行测试。

第6章：总结与展望。在本章中对全文的工作内容进行总结，并对系统中存在的不足，进行了总结和展望。

# 第2章 相关理论与技术

为了提高系统的稳定性和可维护性，本文应急演练辅助教学系统Web前端采用了流行的Vue框架进行编程实现，具有体积小、更高的运行效率、生态丰富、学习成本低等特点。服务器端采用Java语言， Spring Boot和Mybatis框架，使程序更容易维护和扩展。手机端采用Uni-app框架实现，一套代码，多端运行，降低了代码的开发成本和学习成本。流媒体服务器采用的Nginx结合RTMP模块进行实现，支持高并发和稳定低延迟，使系统更加健壮。本章将对上述技术和相关框架进行详细介绍。

## 2.1 Web前端相关技术

（1）Vue。本文的应急演练辅助系统Web前端是基于Vue框架实现。Vue是一款多用途、高性能、渐进式的框架，它允许开发者慢慢的将框架引入到项目中，可以很好的结合原代码进行工作，其中数据双向绑定的特定，使得开发者无须手动操作DOM获取数据和更新数据，就可以很方便的操作数据和渲染刷新页面。拥有组件化的思想，让原本臃肿的代码抽离出来，达到复用和维护的目的，使得代码更加优雅，

基于以上Vue框架的特点，本文的应急演练辅助教学系统遵循组件化的思想，将Web前端多个页面抽离成组件，以达到复用和维护的目的。双向绑定的特性，更容易获取数据和更新数据。基于以上优点，本文的应急演练辅助教学系统的Web前端页面基于Vue框架进行编程实现。

（2）Vuex。Vuex 是一个专为 Vue.js 应用程序开发的状态管理模式。它采用集中式存储管理应用的所有组件的状态，并以相应的规则保证状态以一种可预测的方式发生变化。Vuex 也集成到 Vue 的官方调试工具 devtools extension （opens new window），提供了诸如零配置的 time-travel 调试、状态快照导入导出等高级调试功能。它就像一个全局的单例存储容器，让组件之间的通信更为简单。

（3）Axios。Axios是一个基于promise的HTTP库，可以用在浏览器和node.js中。axios主要是用于向后台发起请求的，还有在请求中做更多是可控功能。简单地讲就是可以发送get、post请求。也正是Vue框架的出现，促使了Axios轻量级库的出现。其特性有：

1. 可以在浏览器中发送 XMLHttpRequests。
2. 可以在 node.js 发送 http 请求。
3. 支持 Promise API。
4. 拦截请求和响应。
5. 转换请求数据和响应数据。
6. 能够取消请求。
7. 自动转换 JSON 数据。
8. 客户端支持保护安全免受 XSRF 攻击。

（4）Element UI。Element-Ul是国内饿了么前端团队推出的一款基于Vue.js 2.0 的桌面端UI框架，Element UI 是一套采用 Vue 2.0 作为基础框架实现的组件库，一套为开发者、设计师和产品经理准备的基于 Vue 2.0 的组件库，提供了配套设计资源，帮助网站快速成型。是简洁、直观、强悍的前端UI组件开发框架，能更快速、简单的开发Web应用程序。

（5）Flv.js。Flv.js 是 HTML5 Flash 视频（FLV）播放器，纯原生 JavaScript 开发，没有用到 Flash，由 bilibili 网站开源。该项目依托于 Media Source Extensions。Flv.js 实现了在 HTML5上播放FLV格式视频，使Bilibili网页端平滑过渡到HTML5播放器。对于视频直播，在HTML5上支持了延迟极低HTTP FLV播放，解开网页端直播对Flash的依赖。

（6）Vue-Router。Vue Router 是 Vue.js 官方的路由管理器。它和 Vue.js 的核心深度集成，让构建单页面应用变得易如反掌。包含的功能有：

1. 嵌套的路由/视图表。
2. 模块化的、基于组件的路由配置。
3. 路由参数、查询、通配符。
4. 基于Vue.js过渡系统的视图过渡效果。
5. 细粒度的导航控制。
6. 带有自动激活的 CSS class 的链接。
7. HTML5 历史模式或 hash 模式，在 IE9 中自动降级。
8. 自定义的滚动条行为。

## 2.2 服务器端相关技术

（1）Spring Boot。Spring的Boot是Pivotal团队在2013年开发的一个新的开源轻量级框架，2014年4月发布了它的第一个版。它基于Spring4．0设计，既继承了Spring框架原有的优秀特色，又通过简化配置简化了Spring应用程序建筑和开发的全过程。此外，Boot整合了大量的框架来解决版冲突和引用不稳的问题。Spring，Boot所有以下特征：

1能创建独立的Spring应用程序，也可以基于Maven或Gradle插件创建可执行的JARs和WARs。

2嵌入式Servlet集装箱，如Tomcat或Jetty。

3提供自动配置的“starter”项目对象模型（POMS），简化Maven配置。

4尽可能自动配置Spring集装箱。

5提供准备好的特性，如指示器，运行状况检查和外部配置。

6绝对没有代码生成，不需要配置。

在SpringBoot的框架中，还有两个非常重要的策略：开箱即用和约定优于配置。这意味着在开发过程中使用pom的配置文件而不是相关的注释。这一特性使开发人员摆脱了复杂的配置工作和依赖的管理工作，更加关注业务逻辑。约定优于配置，这是一种软件设计范式，Boot配置目标结构，开发人员向结构中添加信息。虽然这个特性降低了部分灵活性，增加了BUG定位的复杂性，但它减少了开发人员需要做出的决策数量，同时减少了大量的配置，并且可以自动进行代码编译，测试和打包。

（2）MySQL。MySQL是开放源代码的数据库管理系统（RDBMS），它使用结构化查询语言（SQL），最常用的数据库管理语言，进行数据库管理。MySQL是一个安全、跨平台、高效的数据库系统，它与PHP、Java等主流编程语言紧密结合。数据库系统是由瑞典的MySQL　AB公司开发、发布和支持的。MySQL的标志是一家名为Sakila的海豚，它代表了MySQL数据库的速度、能力、准确性和卓越性。目前，MySQL广泛应用于Internet的中小型网站。由于其体积小，速度快，总拥有成本低，尤其是开源，许多公司使用MySQL数据库来降低成本。MySQL数据库是最快的SQL语言数据库之一，除了其他数据库没有的许多功能之外，MySQL数据库是一个完全免费的产品，用户可以通过网络直接下载MySQL的数据库，而无需支付任何费用。MySQL提供多种数据库存储引擎。每种存储引擎都有自己的优点，适合不同的应用。用户可以选择最合适的引擎来获得最高的性能，并且可以处理高强度的搜索Web网站。由于本文应急演练辅助教学系统需要使用事务来实现业务逻辑，所以使用InnoDB引擎作为数据库底层引擎。

（3）Mybatis。MyBatis 是一款优秀的持久层框架，它支持自定义 SQL、存储过程以及高级映射。MyBatis 免除了几乎所有的 JDBC 代码以及设置参数和获取结果集的工作。MyBatis 可以通过简单的 XML 或注解来配置和映射原始类型、接口和 Java POJO（Plain Old Java Objects，普通老式 Java 对象）为数据库中的记录。

MyBatis 本是apache的一个开源项目iBatis， 2010年这个项目由apache software foundation 迁移到了google code，并且改名为MyBatis 。2013年11月迁移到Github。iBATIS一词来源于“internet”和“abatis”的组合，是一个基于Java的持久层框架。iBATIS提供的持久层框架包括SQL Maps和Data Access Objects（DAOs）

1. 简单易学：本身就很小且简单。没有任何第三方依赖，最简单安装只要两个jar文件+配置几个sql映射文件易于学习，易于使用，通过文档和源代码，可以比较完全的掌握它的设计思路和实现。
2. 灵活：mybatis不会对应用程序或者数据库的现有设计强加任何影响。sql写在xml里，便于统一管理和优化。通过sql语句可以满足操作数据库的所有需求。解除sql与程序代码的耦合：通过提供DAO层，将业务逻辑和数据访问逻辑分离，使系统地设计更清晰，更易维护，更易单元测试。sql和代码的分离，提高了可维护性。提供映射标签，支持对象与数据库的orm字段关系映射，提供对象关系映射标签，支持对象关系组建维护提供xml标签，支持编写动态sql。

（4）WebSocket。Websocket是一种通过单个TCP连接进行全双工通信的协议。Websocket通信协议在2011年被IETF定义为标准RFC 6455，并由rfc7936补充。websocketapi也被W3C标准化了，Websocket使得客户端和服务器之间的数据交换更加容易，并且允许服务器主动地将数据推送到客户端。在websocketapi中，浏览器和服务器只需完成一次握手，就可以直接建立持久连接并进行双向数据传输。为了实现推送技术，很多网站都采用了轮询技术。轮询是在特定的时间间隔（例如每1秒）内，浏览器向服务器发送HTTP请求，然后服务器向客户端浏览器返回最新的数据。Websocket是一种通过单个TCP连接进行全双工通信的协议。Websocket通信协议在2011年被IETF定义为标准RFC 6455，并由rfc7936补充。websocketapi已经被W3C标准化，Websocket使得客户端和服务器之间的数据交换更加容易，并且允许服务器主动地将数据推送到客户端。在websocketapi中，浏览器和服务器只需完成一次握手，就可以直接建立持久连接并进行双向数据传输。为了实现推送技术，许多网站采用了轮询技术。轮询是指浏览器在特定的时间间隔内（例如，每秒）向服务器发送HTTP请求，然后服务器将最新数据返回给客户端浏览器。在没有扩展的情况下，对于从服务器到客户端的内容，头大小仅为2到10字节（与数据包长度有关）；对于从客户端到服务器的内容，头需要额外屏蔽4字节。与每次都有完整标头的HTTP请求相比，开销显著减少。更实时。因为协议是全双工的，所以服务器可以随时向客户机发送数据。与HTTP请求相比，服务器需要等待客户端发起请求，且延迟明显减少；即使与comet等类似的长轮询相比，也能在短时间内多次传递数据。保持联系。与HTTP不同，websocket需要先创建一个连接，使其成为有状态协议，然后在通信时省略部分状态信息。HTTP请求可能需要在每个请求中携带状态信息（例如身份验证）。更好的二进制支持。Websocket定义了二进制框架，它比http更容易处理二进制内容。它可以支持扩展。Websocket定义了扩展，用户可以扩展协议并实现一些自定义的子协议。例如，一些浏览器支持压缩。更好的压缩效果。与HTTP压缩相比，websocket可以在适当的扩展支持下使用以前内容的上下文，在传递相似数据时可以显著提高压缩比。

## 2.3 手机端相关技术

（1）uni-app。uni-app 是一个使用 Vue.js 开发所有前端应用的框架，开发者编写一套代码，可发布到iOS、Android、Web以及各种小程序等多个平台。uni-app属于Dcloud公司，DCloud公司拥有800万开发者、数百万应用、12亿手机端月活用户、数千款uni-app插件、70+微信/qq群。阿里小程序工具官方内置uni-app，腾讯课堂官方为uni-app录制培训课程等。即使不跨端，uni-app也是很好的小程序开发框架、更好的App跨平台框架、更方便的H5开发框架。使用uni-app框架时，不需要转换开发思维、不需要更改开发习惯，可以有效地降低学习成本，提升开发效率。

首先，uni-app兼容Vue的写法，解决了开发者的难题，具备很好的兼容性。其次，uni-app使用HbuilderX进行开发，HbuilderX很好的支持Vue语法，智能代码提示，提升开发速度。再次，uni-app拓展能力强，封装了H5+，支持nvue，也支持原生Android，ios开发。再次，uni-app强大的跨平台能力，uni-app能够跨多个终端，H5，安卓，Ios，微信小程序，百度小程序，头条小程序，支付宝小程序，真正实现了一套代码，多端运行，很好适应了市场，满足市场需求。最后，uni-app拥有日益丰富的插件市场，uni的插件市场也在日益强大，能够基本上满足我们平时的开发需求。综合以上优势，应急演练辅助教学系统手机端基于uni-app框架进行开发是非常高效和稳定的。

## 2.4 流媒体相关技术

流媒体是指通过流媒体在网络中传输音频、视频和多媒体文件的媒体形式。与下载后观看的网络播放形式相比，流媒体的典型特点是将连续的音视频信息压缩后放到网络服务器上，用户可以边下载边观看，而无需等待整个文件被下载。由于流媒体技术的优势，它被广泛应用于视频点播、视频会议、远程教育、远程医疗和在线直播系统。流媒体技术作为新一代互联网应用的标志，近年来发展迅速。常见的流协议如下

（1）RTMP协议。RTMP是实时消息协议的缩写。RTMP（real-time messaging protocol）是Adobe系统开发的一种开放协议，用于flash播放器和服务器之间的音频、视频和数据传输。该协议是基于TCP的一系列协议，包括RTMP基本协议、rtmpt/rtmps/rtmpe等多种变体。RTMP是一种为实时数据通信而设计的网络协议。主要用于flash/air平台与支持RTMP协议的流媒体/交互服务器之间的音视频和数据通信。支持该协议的软件包括Adobe Media Server/Ultrant Media Server/red5等。

RTMP 是一个古老的协议。RMTP 最初由 Macromedia 开发，后被 Adobe 收购，至今仍被使用。由于 RTMP 播放视频需要依赖 Flash 插件。而 Flash 插件多年来一直受安全问题困扰，正在被迅速淘汰。因此，目前 RTMP 主要用于提取 stream。也就是，当设置解编码器将视频发送到托管平台时，视频将使用 RTMP 协议发送到 CDN，随后使用另一种协议（通常是HLS）传递给播放器。RTMP 协议延迟非常低，但由于需要 Flash 插件，不建议使用该协议，但流提取是例外。在流提取方便，RTMP 非常强大，且几乎得到了普遍支持。

（2）HLS协议。HTTPLiveStreaming（简称HLS）是苹果公司提出来的基于HTTP的流媒体网传送协议。它是苹果公司QuickTime，X和苹果软件系统的一部分。它的工作原理是将整个流划分为基于HTTP的小文件进行下载，一次只能下载几个。在观看媒体流时，客服端能选择以区别的速率从许多区别的备份源下载相同的资源，从而允许流会话适应不同的数据速率。在流会话开始时，客户端将下载包含元数据extended，M3U（m3u8）的playlist文件，以查找可用的媒体流。

HLS只请求基本的HTTP报文，与实时传输协议（RTP）不同，HLS可以穿过任何允许HTTP数据通过的防火墙或者代理服务器。它也很容易使用内容分发网络来传输媒体流。HLS协议规定：视频的封装格式是TS，视频的编码格式为H264，音频编码格式为MP3、AAC或者AC-3。除了TS视频文件本身，还定义了用来控制播放的m3u8文件（文本文件）。具体参数对比如表2-1所示。

表2-1流媒体协议对比表

|  |  |  |
| --- | --- | --- |
| 协议名称 | RTMP | HLS |
| 传输方式 | TCP流 | HTTP |
| 视频封装格式 | Flv | Ts切片 |
| 延时 | 1-3s | 大于10s |
| 数据分段 | 连续流 | 切片文件 |
| 是否支持HTML5播放 | 不支持（依赖flash） | 支持 |

基于上述描述和表格对比，本文应急演练辅助教学系统采用RTMP协议进行视频流的提取，采用HLS协议进行视频流的播放。

## 2.6 本章小结

本章主要介绍了应急演练辅助教学系统设计与实现过程中所使用到的关键技术，包括Web前端用到的Vue框架、Vuex、Axios、Element UI、flv.js和Vue Router等技术；服务器端所用到的Java语言、Spring Boot框架、Mybatis框架、WebSocket技术、Mysql数据库等；手机端所用到的Uni-app框架；流媒体服务器所使用的RTMP、HLS流媒体协议。

# 第3章 系统需求分析

系统需求分析是软件开发之前的准备工作，是软件计划阶段极为重要的一步，也是决定着日后软件开发效率的重要活动，好的系统需求分析能够为软件开发打下坚实的基础。需求分析是系统分析和软件设计阶段之间的桥梁。 需求分析将系统规范和项目计划作为分析活动的基本起点，并从软件角度检查和调整它们。 另一方面，需求规范是软件设计，实施，测试和维护。 良好的分析活动有助于尽快避免或消除早期错误，从而提高软件生产率，降低开发成本并提高软件质量。

## 3.1 系统功能需求分析

### 3.1.1 用户角色分析

应急演练辅助教学系统以应急演练直播为核心，以为目标，能够实现 相关业务。本文所设计与实现的应急演练辅助教学系统主要面向三类用户：摄影用户、管理员用户和评价用户。下面分别对这三类用户进行介绍。

1. 摄影用户

摄影用户是手机端的基本用户，是应急演练直播的核心用户。摄影用户可以注册登录到应急演练辅助教学系统手机端，主要功能为根据管理员用户发布的应急演练进行现场直播，以及查看当前用户的录播。摄影用户用例图如图3-1所示。

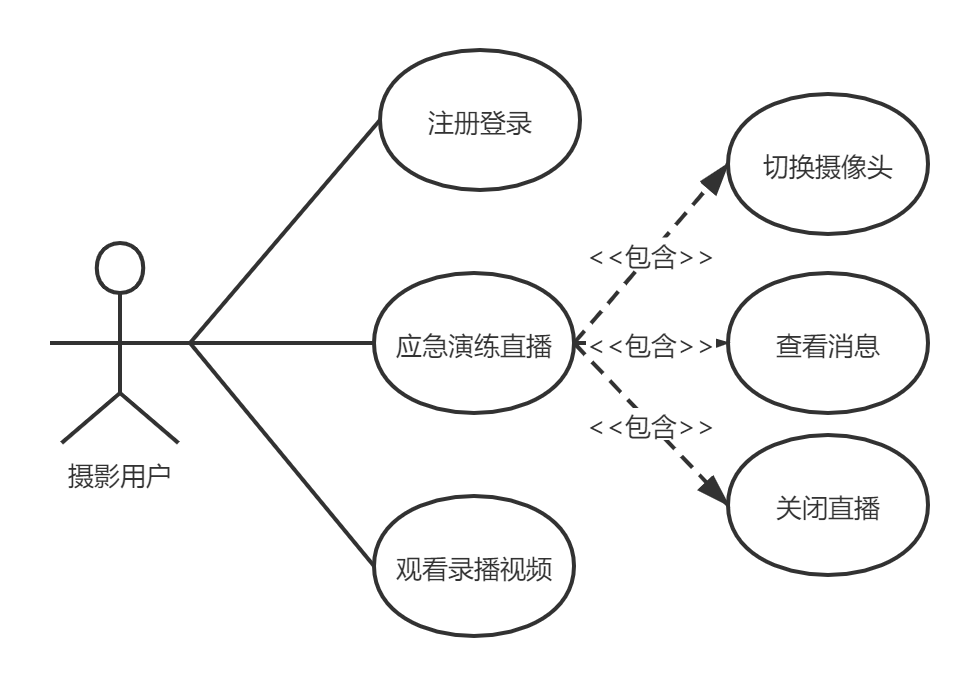


图3-1 摄影用户用例图

1. 管理员用户

管理员用户是Web前端的主要用户，可以对应急演练进行配置，包括应急演练开放时间管理、评价人员管理、评价指标管理、附件管理等。也可以对普通用户和管理员用户进行新增、删除、修改、查看等操作。此外，管理员用户还可以多窗口观看应急演练现场直播情况，和摄影用户实时文字交互，掌控应急演练全局执行情况。还可以观看录播视频，对多个录播视频进行同步播放，复盘分析应急演练整体执行情况。管理员用户用例图如图3-2所示。

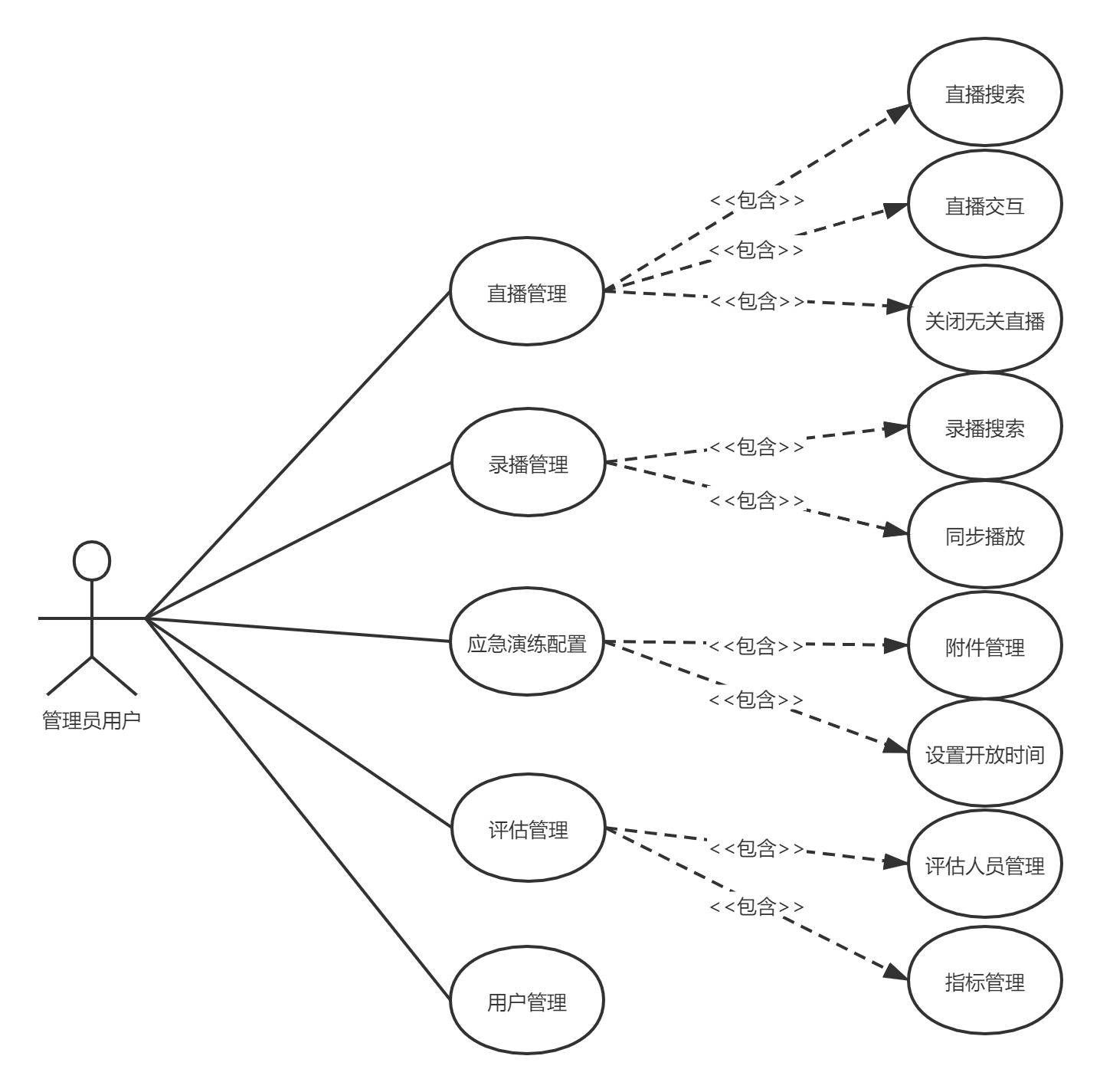


图3-2 管理员用户用例图

1. 评价用户

评价用户是Web前端的用户，评价用户可以多窗口观看应急演练执行状况，然后根据管理员用户配置的应急演练中的评价指标，对应急演练执行情况做出具体的评价。评价用户用例图如图3-3所示。

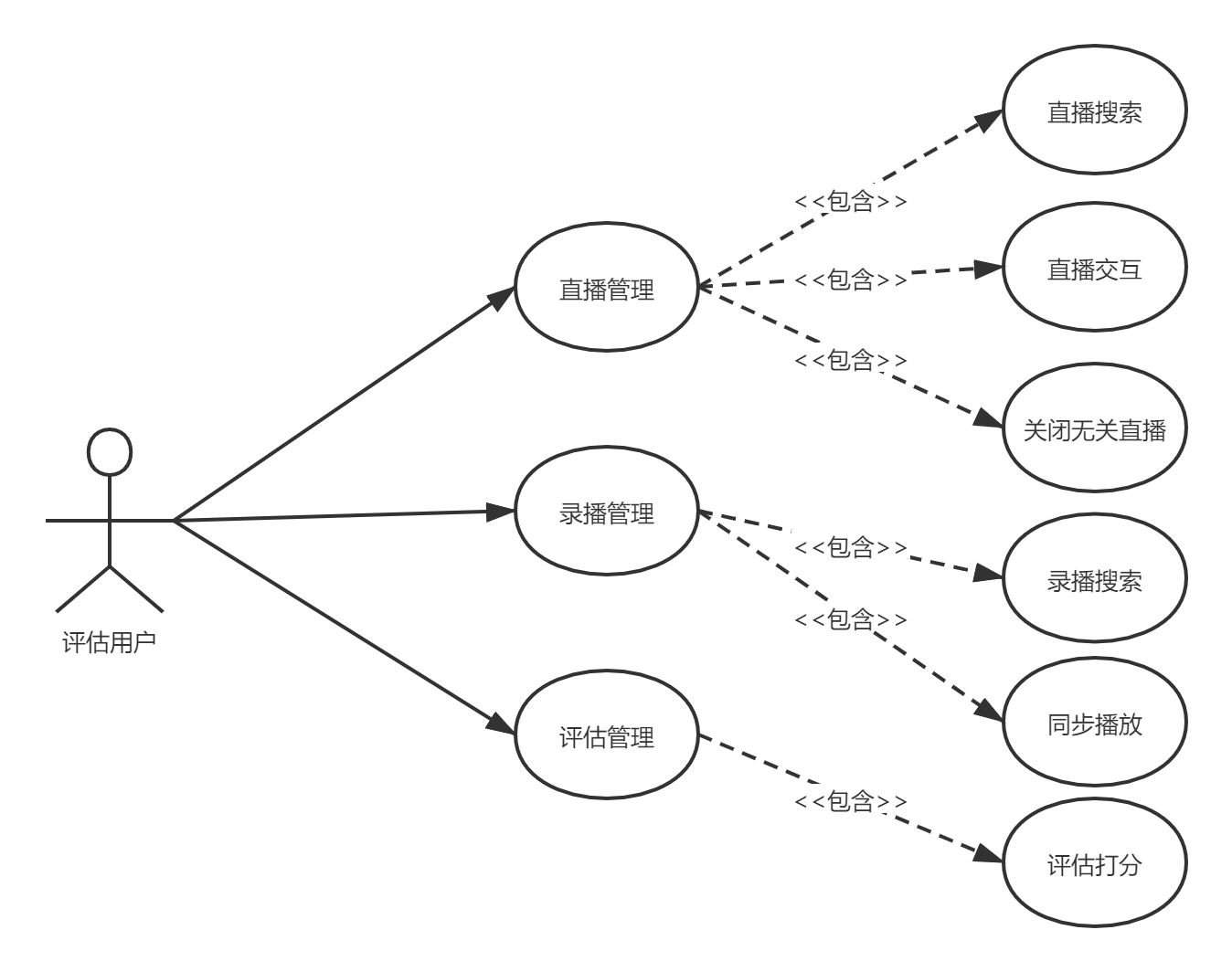


图3-3 评价用户用例图

### 3.1.2 系统功能

通过前期对需求地调研可以得出，本文所要设计和实现的应急演练辅助教学系统主要包括应急演练的配置、自定义评价指标、应急演练直播、录播、事后复盘评价等业务功能。管理员用户通过该系统对应急演练设置执行时间、指标、评价人员等。摄影用户可以登录手机端系统，在应急演练的开放时间内选择对应的应急演练，对应急演练现场执行情况进行直播。

评价人员可以通过该系统实时观看多窗口应急演练的执行状况，结合管理员用户设置的指标，对应急演练执行情况进行具体评价。

基于以上需求分析，将本文所要设计和实现的应急演练辅助教学系统按模块分为应急演练直播模块、应急演练录播模块、应急演练交互模块、应急演练配置模块、应急演练评价模块和用户管理模块六个模块。

1. 应急演练直播模块

在应急演练辅助教学系统中，应急演练直播模块是系统的核心模块，摄影用户通过选择应急演练和填写直播标题，对应急演练现场进行直播，管理员用户可通过用户名或直播标题进行筛选直播，详细功能如下：

* 直播搜索：用户可以在直播管理页面中通过搜索框对应急演练直播画面进行筛选，搜索条件包括用户手机号和应急演练直播标题。
* 关闭无关直播：用户可以在上述直播搜索的基础上，进一步对应急演练直播画面列表进行筛选过滤，通过点击特定直播画面右上方的“X”号，会将当前直播画面从直播列表中移除。

1. 应急演练录播模块

应急演练录播是对应急演练历史直播视频资源进行存储和播放，用户可以通过时间、标题和用户名等条件搜索录播视频资源，也可以选择多个录播视频进行同步播放，对应急演练全局状况进行复盘评价，详细功能如下：

* 录播搜索：用户可以在录播管理页面中通过文字搜索框和时间搜索框来筛选录播视频资源，文字搜索框可以输入用户名和直播标题，时间搜索框可以输入开始时间和结束时间的范围。
* 同步播放：用户可以通过选择两个或两个以上的录播视频资源，点击同步播放按钮，系统会根据数据库中的数据信息，计算出所选录播资源列表中开始时间最晚的时间，然后将最晚时间和所选的所有录播视频资源的开始时间进行计算，得到每个录播视频资源的时间偏移量，进而设置此偏移量就可以实现同步播放功能。

1. 应急演练交互模块

具体来说，直播交互是管理员用户和评价用户在观看应急演练现场直播时，可以通过选择一个或多个应急演练直播窗口，发送文字信息进行交互指挥。

1. 应急演练配置模块

应急演练配置模块是面向管理员用户的，管理员用户可以对应急演练进行管理，包括新增应急演练，以及对应急演练添加附件，可以传达更详细的信息，详细功能如下：

* 应急演练管理：管理员用户可以对应急演练进行新增、修改、查看和删除，其中具体包含的信息有应急演练的标题、开始时间和结束时间。
* 附件管理：管理员用户对特定的应急演练上传附件，其他用户可以直接点击附件地址对附件内容进行预览。

1. 应急演练评价模块

应急演练评价模块是在应急演练配置模块的基础上，对应急演练添加评价人员和自定义指标等操作，最后评价人员根据应急演练执行状况作出具体的评价，详细功能如下：

* 评价人员管理：管理员用户在上述应急演练管理的基础上，选择特定的应急演练，指定多个评价人员，为后期评价打分做准备。
* 指标管理：管理员用户选择具体的应急演练设置自定义指标，包括单选形式、打分形式和评语形式等。
* 评价打分：评价人员在上述指标管理的基础上，针对特定应急演练的执行情况，进行具体的评价打分。

1. 用户管理模块

用户管理模块的主要功能是对应急演练辅助教学系统中的所有用户进行管理，包括查看、新增、删除和编辑等操作，详细功能如下：

* 新增用户：管理员用户可以通过用户管理页面新增普通用户和管理员用户。新增用户时选择用户角色，填写用户手机号和姓名即可，用户密码默认为“123456”。
* 删除用户：管理员用户可以通过点击用户列表中每一行的特定删除按钮来删除用户，也可以通过多选框选择多个用户进行批量删除，服务器端通过用户手机号对用户进行删除。
* 编辑用户：管理员用户可以对每个用户的信息进行编辑修改，能够修改的信息只有用户名，用户手机号和用户角色不允许修改。

## 3.2系统流程分析

应急演练辅助教学系统是面向摄像人员、管理员和评价人员的，同时具备手机端、Web前端和服务器端系统实现。摄影人员对应手机端系统，管理员和评价人员对应Web前端系统，用户和系统之间存在一定的关系，同时系统与系统之间、各个功能之间存在前后顺序和协作关系，这样系统才能够满足业务需求，系统安全稳定地运行。

在系统整体运行流程中，首先管理员登录Web前端系统，自定义配置应急演练的相关信息，其中包括演练名称、开始时间和结束时间，还可以配置评价人员和评价指标等信息用于评价人员对应急演练进行评价。然后摄影人员登录手机端系统，根据管理员用户配置的应急演练列表，在开放时间之内选择具体的应急演练开始直播，最后评价人员登录Web前端系统，观看摄影人员的直播画面，在摄影人员直播结束之后，根据管理员用户配置的评价指标，对应急演练执行情况进行定性或定量评价。应急演练辅助教学系统的整体系统流程图如图3-4所示。

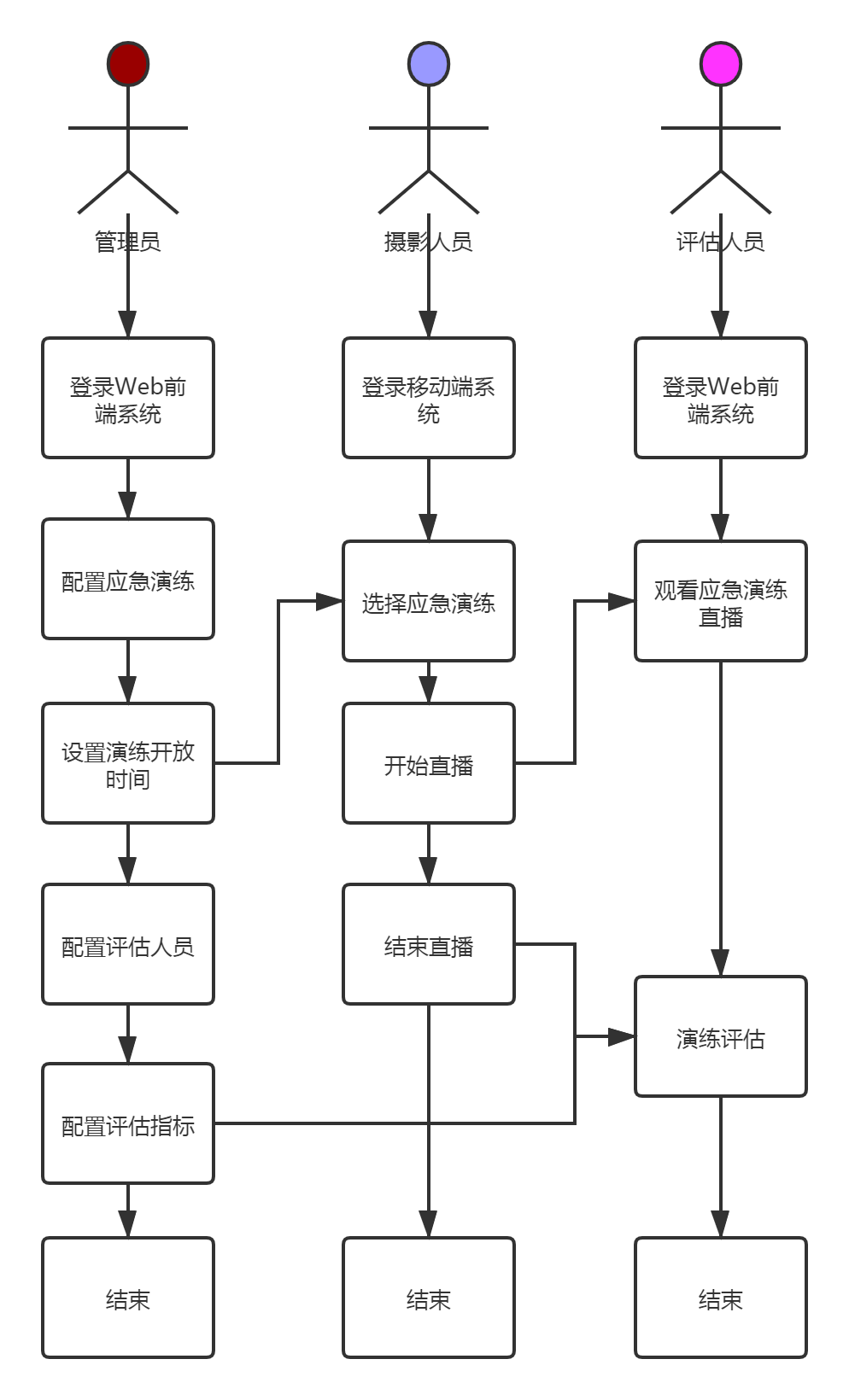


图3-4 系统整体系统流程图

## 3.3 可行性分析

通过对高校教师和学生进行需求调研，明确了本文所要设计和开发的应急演练辅助教学系统的目标，系统所面向的用户人群和主要功能。下面从市场可行性和技术可行性对应急演练辅助教学系统进行可行性分析。

（1）市场可行性。在当代经济条件下，随着经济和技术的发展，灾害所带来的经济损失和人员伤亡逐渐扩大，但是普遍人员在应对灾害的过程中存在不规范的情况，为了避免这一情况的发生，应急演练变得越来越重要。在传统的应急演练过程中，指挥人员需要在现场观看应急演练，很难统筹全局，同时应急演练事后复盘评价较为困难。因此，从市场前景来看，这样一个应急演练辅助教学系统是非常有必要的。

（2）技术可行性。随着网络技术的发展，特别随着Web技术的不断成熟，B/S 这种软件体系结构功能越来越强大。这种结构可以有效降低资源成本，提高系统的性能。本文拟采用前服务器端分离的模式对应急演练辅助教学系统进行设计与实现，能够极大地提高开发效率，降低软件的耦合度，提高了处理复杂业务的能力，使得服务器端只需专注业务逻辑代码处理，Web前端专注于页面渲染，分工明确，简单高效。因此，本文应急演练辅助教学系统技术上是可行的。

本文应急演练辅助教学系统服务器端拟采用面向对象编程语言Java为基础，Spring Boot框架处理系统业务，Mybatis框架对数据进行持久化，结合MySQL关系型数据库对数据进行存储。Web前端基于Vue框架进行编程实现，搭配Element UI框架对页面进行美化和渲染，Axios框架进行网络请求，能够安全有效地进行数据交互。手机端采用Uni-app框架开发，编写一套代码多端运行的特点，提升开发效率，减小学习成本。流媒体服务器采用Nginx结合RTMP模块进行实现，在此基础上使用RTMP协议和HLS协议对视频流进行提取和播放。以上都是成熟的技术和框架，因此，本文应急演练辅助教学系统在技术上可行的。

综上所述，本文应急演练辅助教学系统具有较高的技术可行性。

## 3.4 系统的性能需求分析

系统的功能需求分析是完成系统功能的重要环节，但是一个优秀的系统只有功能需求分析是远远不够的，还需要进行系统性能需求分析。

包括可靠性、容错性、可维护性、并发性。以下对上述指标进行详细描述。

系统的可靠性。系统可靠性一般是指系统在规定的时间和工作条件下完成规定功能的能力/概率。由于科学技术的进步，系统的组成越来越复杂，系统的可靠性问题也日益突出。系统越复杂，承载的信息量越大，重要性越高，功能越强，适用范围越广。一旦系统失灵，损失将是巨大的，甚至是灾难性的。如何快速、有效、准确地评价和分析系统的可靠性，正确估计系统的实际性能，降低系统的风险具有重要的现实意义。本文应急演练辅助教学系统采用的技术和框架都是比较成熟的，所以具有较高的可靠性。

系统的容错性。容错性是产品进行错误操作的能力，即产品操作中发生错误的概率和解决错误的概率和效率。容错最初应用于计算机领域，它的存在可以保证系统在发生故障的情况下不发生故障，仍能正常工作。产品容错设计可以使产品与人或借助产品与人之间的交流更加流畅。因此，系统容错性非常重要。

系统的可维护性。主动式可维护性和反应式可维护性。主动式可维护性包括构建易于理解和更改的代码库。随着开发的进行，会出现与现有代码不兼容的问题。如果工程师写的是面条式代码，而不是优先考虑可维护性，就容易出问题，并且很难发现和解决问题。主动维护还包括质量保证和测试等程序。反应式可维护性描述了服务在事故发生后被修复的能力。这受服务的事故响应过程的影响。大型事故的反应和防范是必要的，如果事故响应程序可靠，团队将迅速解决事件。适当的事故反应也有助于减少复发。本文是在了解系统需求之后进行的开发实现，对页面进行组件化开发，接口也使用的是规范的rest类型，所有系统具有较好的可维护性和扩展性。

系统的并发性。在计算机操作系统中，并发实际上是一个物理cpu在若干给程序之间的多路复用，它与并行性不同。并行性是指两个和两个以上的事件或活动在同一时刻发生。可见，并行的事件或活动一定是并发的，但是，并发的事件或活动未必是并行的。并行性是并发性的特例，而并发性是并行性的扩展。实现并发性的关键技术之一是对系统内的多个程序进行切换，这将涉及进程调度的问题。在本文系统中，采用Nginx为流媒体服务器，Nginx是一个高性能的服务器，支持高并发，所以本系统拥有较好的并发性。

## 3.5 本章小结

本章对应急演练辅助教学系统进行了深入的系统需求分析，首先对系统所面向的用户角色进行分析，包括摄影用户、管理员用户及评价用户，结合用例图详细分析了每个角色对应的功能，然后对系统进行了功能需求分析，将系统分为应急演练直播模块、应急演练录播模块、应急演练交互模块、应急演练配置模块、应急演练评价模块和用户管理模块等六个模块，分析了各个模块中的功能和细节，接着通过三种用户角色对整体的系统运行进行了流程分析，又从系统的可行性出发，分析了系统的市场可行性和技术可行性，最后分析了系统的性能需求。

# 第4章 系统总体设计

## 4.1 功能模块设计

本节将在系统需求分析和系统架构设计的基础上，对本文所设计与实现的应急演练辅助教学系统进行功能模块的详细设计。主要包括应急演练直播模块、应急演练录播模块、应急演练交互模块、应急演练配置模块、应急演练评价模块和用户管理模块等核心模块。本节将使用功能模块类图进行设计与实现。系统功能模块总体结构如图4-1所示。

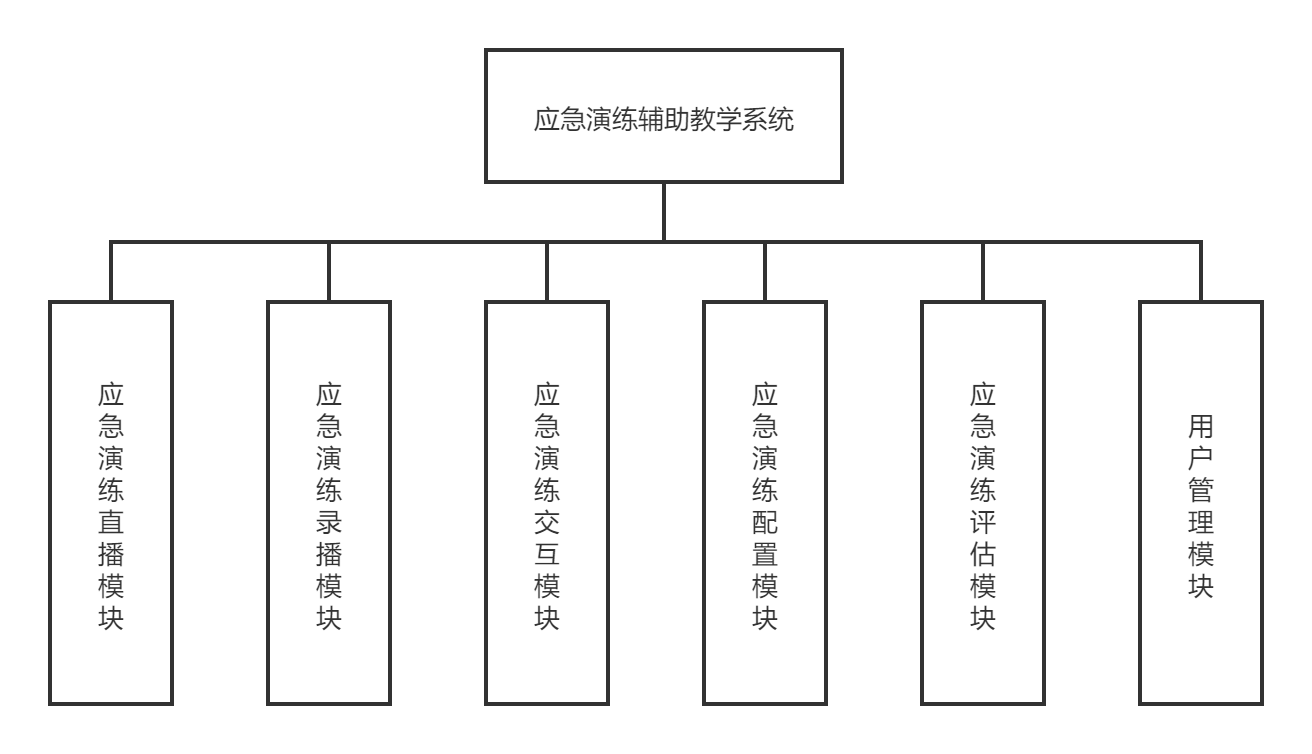


图4-1 系统功能模块图

### 4.1.1 应急演练直播模块

应急演练直播模块是应急演练辅助教学系统的核心模块，其工作流程为管理员用户配置应急演练信息并设置开放时间，摄影用户在开放时间内选择对应的应急演练，填写相应的直播标题，开启直播实时推送应急演练现场执行情况，期间可自由切换手机前后摄像头进行画面采集，管理员用户和评价用户观看多窗口应急演练现场画面，并可以根据自身情况关闭无关的直播窗口。其类图如图4-2所示。

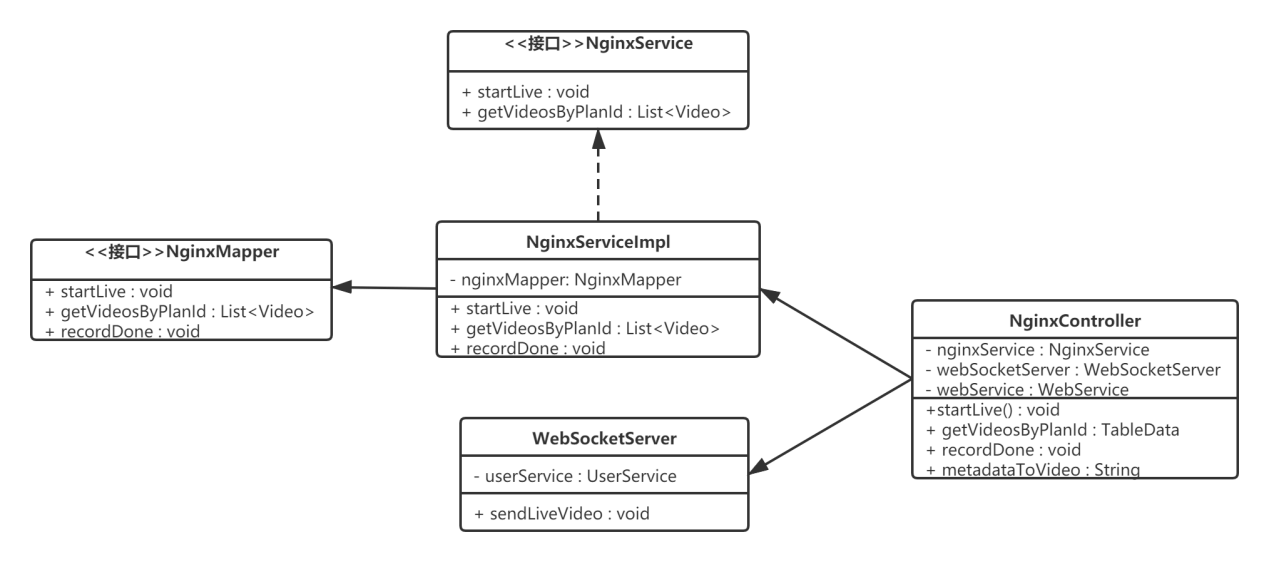


图4-2 应急演练直播功能模块类图

实现应急演练直播模块的核心类为NginxController，类内部包含了NginxService、WebSocketServer和WebService等私有成员变量，其中部分成员变量属于Service层，而Service层中又通过Mapper层的对象调用Mapper层的代码实现，这种分层的代码结构是基于Spring MVC的架构思想实现的，通过将系统分为控制层、业务逻辑层、数据访问层等层级，上一层应该通过接口来调用下一层提供的服务。使得代码更加清晰，易于维护和扩展。

Web前端通过网络请求调用NginxController类中的getVideosByPlanId（）方法来返回当前正在进行直播的视频列表，这里返回的TableData类型是对List<Video>视频列表的进一步封装，其中额外增加了total、rows字段用来Web前端分页实现，还有code、msg等字段用来更加清楚地表示网络请求状态和具体信息。接着通过NginxServiceImpl类中的getVideosByPlanId（）方法对业务逻辑进行处理，然后调用NgxinMapper接口中的getVideosByPlanId（）方法，此时Mybatis框架会对Mapper接口中的每个方法调用进行拦截，进而执行对应的SQL语句，得到相应的List<Video>视频列表，返回给Mapper层、Service层、Controler层，最后返回给Web前端，渲染到浏览器中。

当有新的摄影用户进行直播时，会通过流媒体服务器回调NginxController类中的startLive（）方法，此方法通过HttpServletRequest来获取必要的信息，包括摄影用户的经纬度、直播标题、直播URL、开始时间等信息，将这些数据封装成Video对象用于分层传递，首先调用NginxService类中的startLive（）方法，并将Video对象传递过去，通过业务逻辑处理之后调用NginxMapper接口中的startLive（）方法将Video对象保存到MySQL数据库中。然后调用WebSocketServer类中的sendLiveVideo（）方法和传递Video对象，sendLiveVideo（）方法通过WebSocket技术将Video对象数据发送给所有已连接WebSocket的用户，这时Web前端通过WebSocket接受新的直播消息，动态的渲染到浏览器中即可。

当摄影用户结束直播时，流媒体服务器将会回调NginxController类中的recordDone（）方法，此方法通过HttpServletRequest对象获取摄影用户的手机号、结束时间、录播URL等信息，依次传递给Service层和Mapper层来更新数据库。此外NginxControler类中的metadataToVideo（）方法会通过yamdi给录播视频资源添加元数据，达到播放录播视频时可以随意拖拽进度的功能。

### 4.1.2 应急演练录播模块

应急演练录播模块的主要功能包括对录播资源的筛选，以及多个录播资源同步到统一的时间点进行播放等，其类图如图4-3所示。

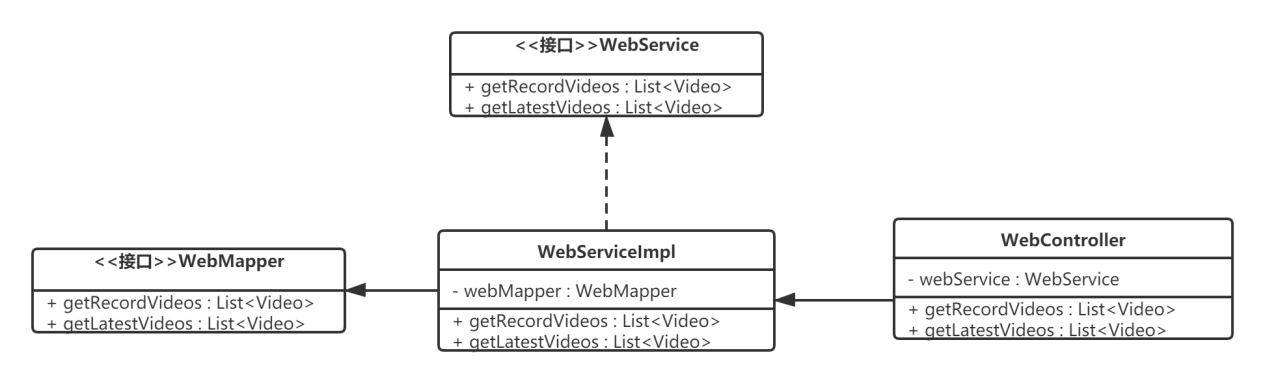


图4-3 应急演练录播功能模块类图

实现应急演练录播模块的核心类为WebServiceImpl类，实现了WebService接口。WebController类内部包含了WebService成员变量，通过调用getRecordVideos（）方法来获取录播资源列表，还可以通过标题、时间等条件来筛选录播资源，WebServiceImpl类对筛选条件做处理，传递给WebMapper，Mybatis拦截调用并执行对应的映射SQL语句返回录播视频资源信息。获取最新录播资源的操作时通过WebController类中的getLatestVideos（）方法实现的，默认返回当天的所有录播资源。

同步播放的功能逻辑在Web前端，主要是通过选择框选择两个或两个以上的录播视频窗口，点击同步播放时会根据所选的录播视频中包含的信息，比较其中的开始时间，然后选择一个最晚的开始时间，用这个最晚的开始时间与当前录播视频的开始时间做运算，分别算出所选录播视频的时间偏移量，最后设置录播视频的正确偏移量即可完成同步播放功能。

### 4.1.3 应急演练交互模块

应急演练交互模块的主要功能为管理员用户选择一个或多个直播用户，来进行实时文字交互，达到交流指挥的作用。其类图如图4-4所示。

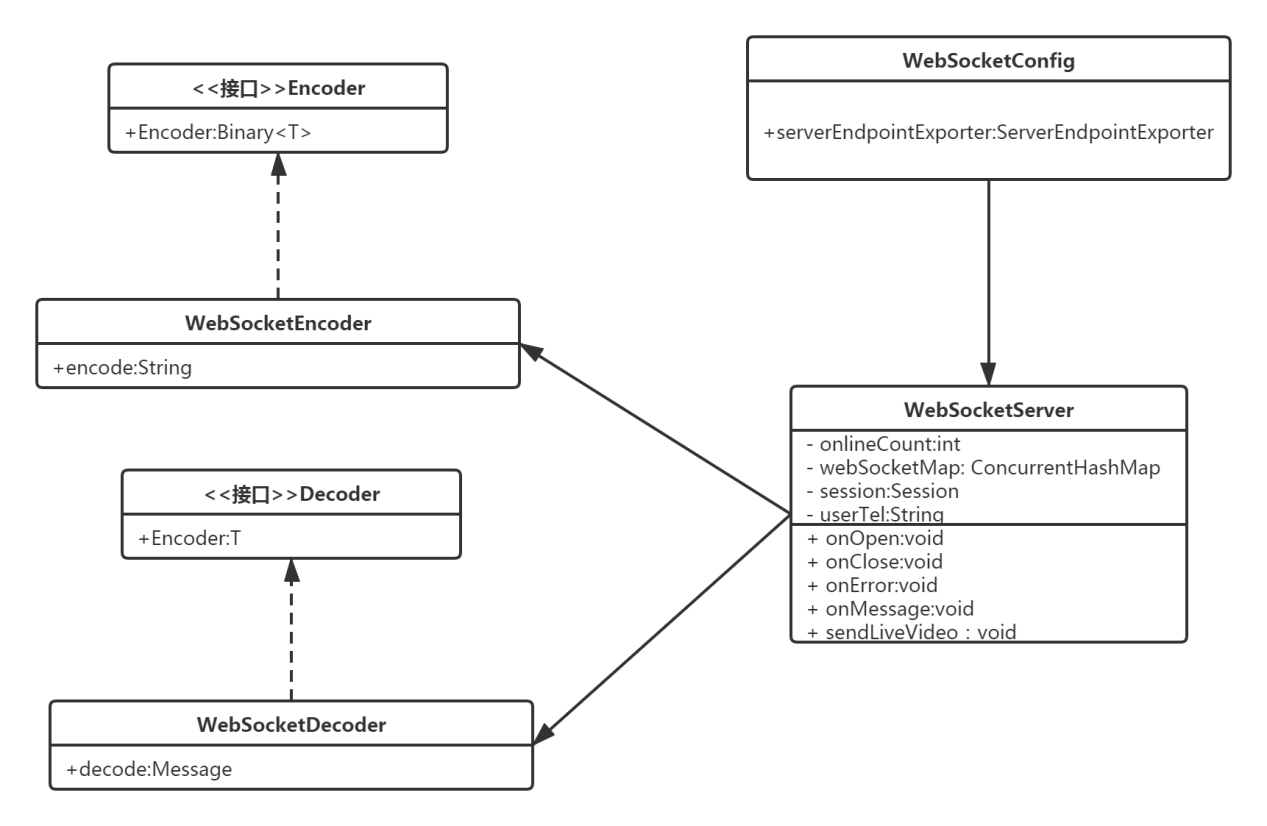


图4-4 应急演练交互功能模块类图

应急管理交互模块的核心实现类为WebSocketServer类，其中WebSocketConfig类是配置类，负责创建WebSocket服务端点。WebSocketEncoder类实现了Encoder接口，其中的encode（）方法的主要功能是将WebSocket要发送的消息进行编码，再进行网络传输，WebSocketDecoder实现了Decoder接口，其中的decode方法的主要功能是对发送到服务器端的消息进行自定义解码。在WebSocketServer类的私有属性中，onlineCount用来记录当前在线连接数，webSocketMap用来存放每个客户端对应的WebSocketServer对象，这里使用ConcurrentHashMap用来保证线程安全，session是与某个客户端的连接会话，需要通过它来给客户端发送数据，userTel用来记录当前连接会话的手机号。onOpen（）方法是客户端连接建立成功调用的方法，onClose（）是连接关闭调用的方法，onError（）是连接出现异常调用的方法，onMessage（）是该类的核心方法，通过该方法可以实现客户端与服务器双向交互功能，具体业务逻辑是结合WebScoketEncoder类和WebSockerDecoder类进行实现。首先将客户端发送的信息通过WebScoketEncoder进行解码，对解码后的数据进行处理，获得要发送的手机号列表，对要发送消息手机号列表进行循环遍历，利用WebScoketEncoder将要发送的消息进行编码，最终实现将消息广发出去的功能。

### 4.1.4 应急演练配置模块

应急演练配置模块的主要功能为应急演练开放时间、附件管理，其类图如图4-5所示。

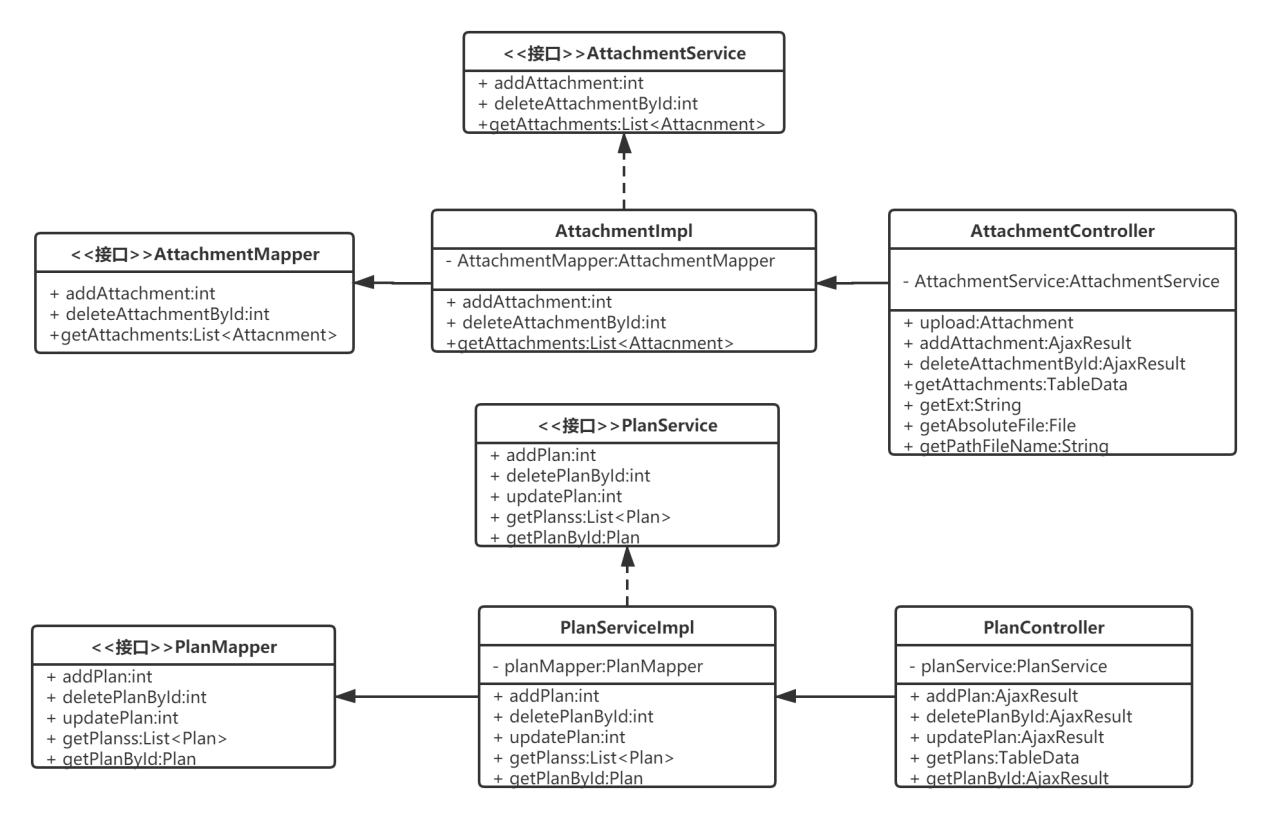


图4-5 应急演练配置功能模块类图

PlanServiceImpl是实现应急演练相关信息配置的核心类，PlanServiceImpl类调用addPlan（）方法来实现调用PlanMapper类中的addPlan（）方法，可以添加新的应急演练，其中内容包括演练名称、开始时间和结束时间等信息。PlanServiceImpl类中的updatePlan（）方法的功能是更新预案信息，deletePlanById（）方法是根据应急演练id删除应急演练，getPlanList（）方法是获取应急演练列表，由于篇幅限制，这里就不详细展开描述了。

AttachmentController类的功能是应急演练附件管理。AttachmentController类中的update（）方法为将上传到服务器的附件资源存储到服务器中并返回虚拟路径。其中getExt（）方法是获取文件的后缀名，通过UUID的随机数加上文件后缀名组成新的文件名称，这样可以防止服务器文件名冲突，然后调用getAbsoluteFile（）方法将文件存储到服务器中并返回路径，接着调用getPathFileName（）方法将文件真实路径映射为文件虚拟路径，可以有效地保护服务器数据安全，最后通过服务器端返回文件虚拟路径，前端在访问虚拟路径时，服务器端会将虚拟路径映射成文件真实路径，保证了数据的正确性。

### 4.1.5 应急演练评价模块

应急演练评价模块的主要功能包括两方面，一方面是指标管理，管理员用户对应急演练添加自定义评价指标，包含单选、评分和评语等指标类型。另一方面是评价打分，评价用户通过观看应急演练直播，结合自定义配置的评价指标，对应急演练整体执行状况做出具体评价，其类图如图4-6所示。

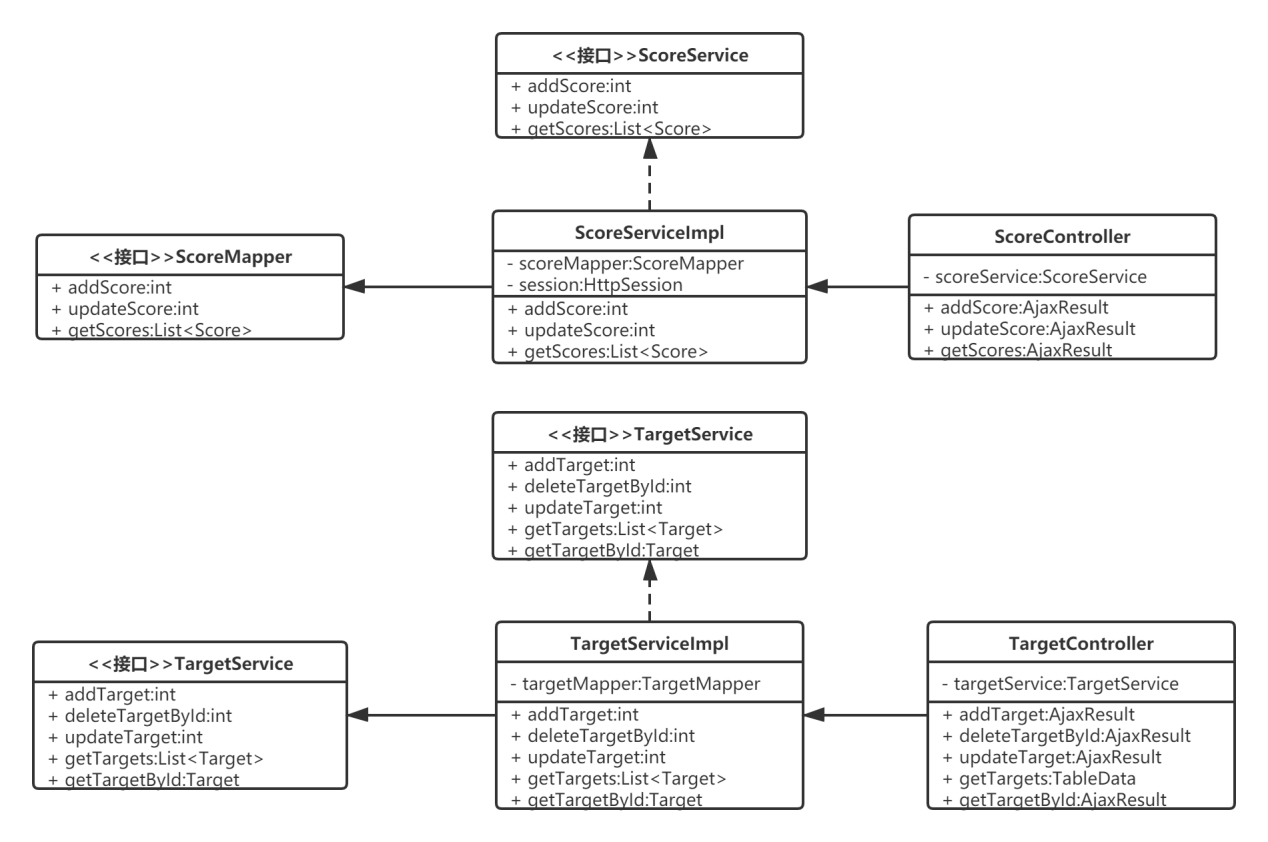


图4-6 应急演练评价功能模块类图

应急演练评价模块的核心实现为ScoreServiceImpl类和TargetServiceImpl类，分别实现了ScoreService接口和TargetService接口。ScoreServiceImpl类通过调用getScores（）方法来调用ScoreMapper类中的getScores（）方法来实现查看当前评价用户对选择的应急演练所作出的具体评价内容，其中评价用户的信息通过ScoreServiceImpl类中的成员变量session获取。评价用户对应急演练做出新的评价的实现是依赖ScoreServiceImpl类通过调用addScore（）方法来调用ScoreMapper类中的addScore（）方法。

指标管理的逻辑实现主要依赖TargetServiceImpl类，其中addTarget（）方法用来添加新的指标，deleteTargetById（）方法用来通过指标id来删除对应指标，更新指标内容的方法是调用updateTarget（）方法实现，getTartgetList（）方法为查看指标列表，getTartgetById（）方法是根据指标id查看指标，由于其中没有包含复杂逻辑，这里就不一一阐述了。

### 4.1.6 用户管理模块

用户管理模块的主要功能包括摄影用户和管理员用户的新增、删除、批量删除、修改信息、查看信息等，其类图如图4-7所示。

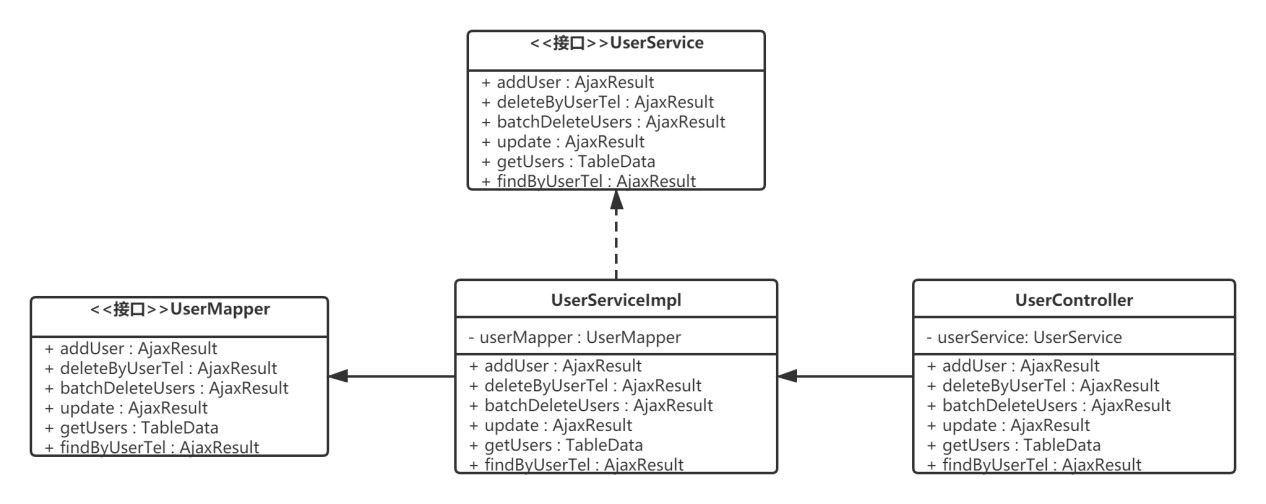


图4-7 用户管理功能模块类图

用户管理模块的核心类为UserServiceImpl类，它实现了UserService接口，类内部包含了UserMapper成员变量，来实现摄影用户和管理员用户的管理。

管理员用户可以通过用户管理模块来新增摄影用户和管理员用户。UserController类通过addUser（）方法来接收Web前端发送的用户信息，再通过UserService实例将用户信息传递给UserServiceImpl类的addUser（）方法，接着通过UserMapper实例将用户信息传递给UserMapper接口的addUser（）方法，最后通过SQL语句将用户信息保存到MySQL数据库中。

管理员用户对摄影用户和管理员用户的修改和删除操作和上述新增操作类似，这里就不再赘述了。批量删除的操作是UserController类的batchDeleteUsers（）方法实现，因为数据库中每一行数据的主键是用户的手机号，所以Web前端在批量删除操作中传递的是以逗号拼接的用户手机号字符串，batchDeleteUsers（）方法将接收到的字符串传递给UserServiceImpl类的batchDeleteUsers（）方法，它对接收到的字符串执行分割操作，然后以列表的形式对用户进行批量删除即可。

## 4.2 数据库设计

本文应急演练辅助教学系统采用MySQL关系型数据库，以应急演练直播为核心，面向摄影用户、管理员用户、评价用户，用到的数据库表主要包括：附件表、应急演练表、评价人员表、分数表、指标表、用户表、视频资源表。系统E-R图如图4-8所示。

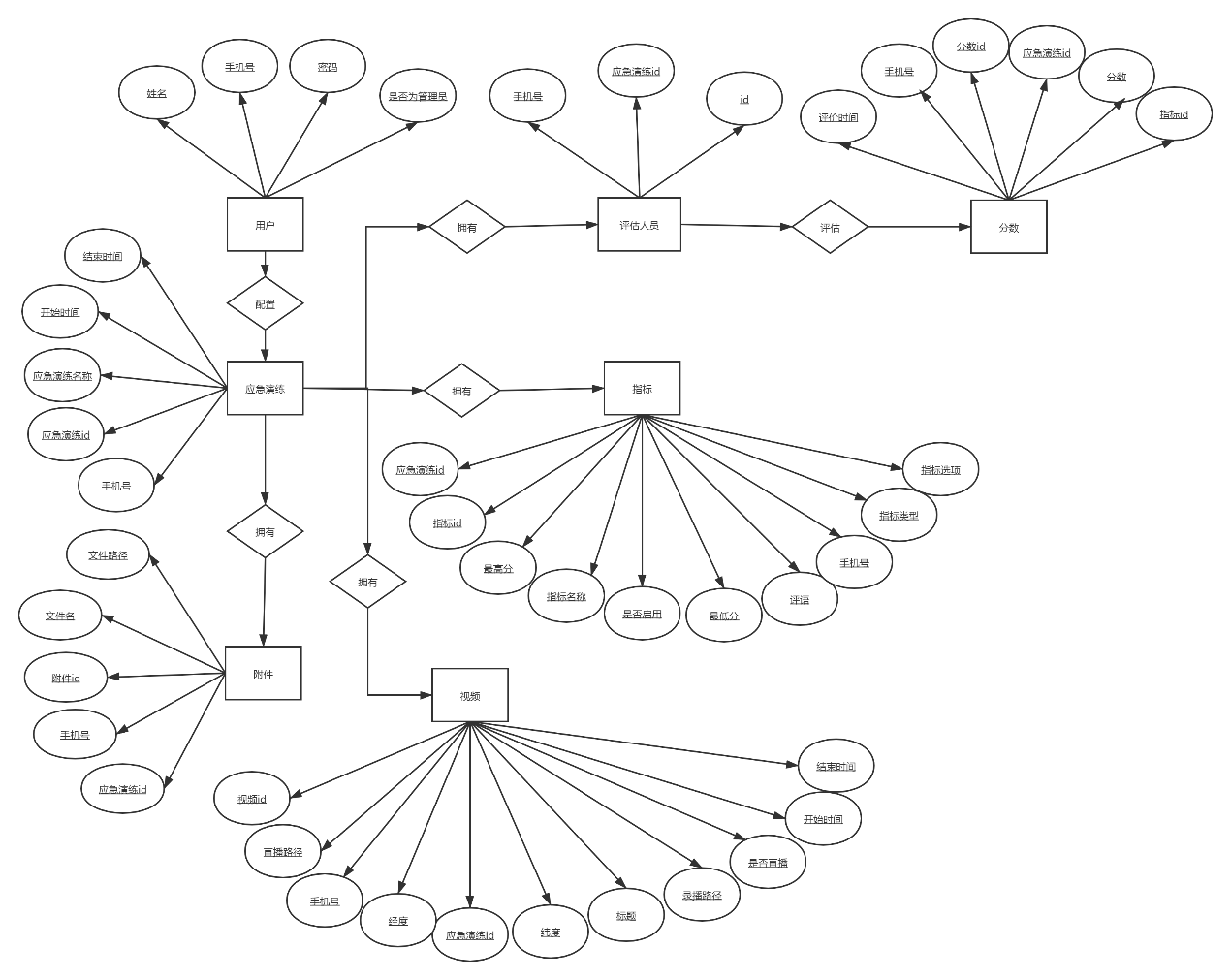


图4-8 应急演练辅助教学系统E-R图

下面分别介绍各个表结构设计。

1. 用户表

用户表用来保存应急演练辅助教学系统中手机端和Web前端用户的基本信息，其中用户角色字段用来表明是否为管理员用户，1表示为管理员用户，0表示非管理员用户。表结构设计如表4-9所示。

表4-9 用户表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 属性 | 字段名 | 类型 | 约束 | 长度 |
| 手机号 | userTel | varchar | 主键 | 20 |
| 密码 | userPwd | varchar | 非空 | 20 |
| 用户名 | userName | varchar | 非空 | 20 |
| 用户角色 | isAdmin | int | 非空 | 2 |

1. 应急演练表

应急演练表用来保存应急演练的名称、开始时间和结束时间等字段，表结构如表4-10所示。

表4-10 应急演练表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 属性 | 字段名 | 类型 | 约束 | 长度 |
| 演练id | planId | Int | 主键（自增1） | 100 |
| 演练名称 | planName | varchar | 非空 | 100 |
| 开始时间 | startTime | datetime | 非空 | 0 |
| 结束时间 | endTime | datetime | 非空 | 0 |
| 手机号 | userTel | varchar | 外键 | 20 |

1. 视频资源表

视频资源表是应急演练辅助教学系统的核心表，通过视频资源表来保存视频资源的相关信息和内容，包括视频资源的标题、手机号、经纬度、开始时间、结束时间、直播路径、录播路径等信息，视频资源表结构如表4-11所示。

表4-11 视频资源表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 属性 | 字段名 | 类型 | 约束 | 长度 |
| 视频id | videoId | int | 主键（自增1） | 100 |
| 录播路径 | videoUrl | varchar |  | 100 |
| 开始时间 | startTime | datetime |  | 0 |
| 结束时间 | endTime | datetime |  | 0 |
| 是否直播 | flag | Int | 非空 | 2 |
| 纬度 | lng | varchar |  | 20 |
| 经度 | lat | varchar |  | 20 |
| 直播路径 | url | varchar | 非空 | 100 |
| 视频标题 | title | varchar | 非空 | 100 |
| 手机号 | userTel | varchar | 非空 | 20 |
| 演练id | planId | Int | 外键 | 100 |

1. 分数表

本文应急演练辅助教学系统通过分数表来记录评价用户依据相应的指标对应急演练进行评价的信息，分数表结构设计如表4-12所示。

表4-12 分数表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 属性 | 字段名 | 类型 | 约束 | 长度 |
| 分数id | scoreId | int | 主键（自增1） | 100 |
| 演练id | planId | int | 外键 | 100 |
| 分数 | score | varchar | 非空 | 255 |
| 手机号 | userTel | varchar | 非空 | 20 |
| 评价时间 | reviewTime | datetime | 非空 | 0 |
| 指标id | targetId | int | 外键 | 100 |

1. 附件表

本文应急演练辅助教学系统通过附件表来记录应急演练配置中的附件信息，附件表结构设计如表4-13所示。

表4-13 附件表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 属性 | 字段名 | 类型 | 约束 | 长度 |
| 附件id | attachmentId | int | 主键（自增1） | 100 |
| 附件名称 | fileName | varchar | 非空 | 100 |
| 附件路径 | filePath | varchar | 非空 | 100 |
| 演练id | planId | int | 外键 | 100 |
| 手机号 | userTel | varchar | 外键 | 20 |

1. 指标表

本文应急演练辅助教学系统通过指标表记录管理员用户对应急演练自定义配置的指标

内容，指标表结构设计如表4-14所示。

表4-14 指标表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 属性 | 字段名 | 类型 | 约束 | 长度 |
| 指标id | indexId | int | 主键（自增1） | 100 |
| 指标名称 | IndexName | varchar | 非空 | 100 |
| 是否启用 | enabled | bit | 非空 | 1 |
| 最低分 | minScore | int |  | 10 |
| 最高分 | maxScore | int |  | 10 |
| 评语 | description | varchar |  | 255 |
| 指标选项 | options | varchar |  | 100 |
| 演练Id | planId | int | 外键 | 100 |
| 指标类型 | indexType | varchar | 非空 | 100 |
| 手机号 | userTel | varchar | 外键 | 20 |

1. 评价人员表

本文应急演练辅助教学系统通过评价人员表记录应急演练中的评价人员信息，指标表结构设计如表4-15所示。

表4-15 评价人员表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 属性 | 字段名 | 类型 | 约束 | 长度 |
| id | Id | int | 主键（自增1） | 100 |
| 演练id | planId | int | 外键 | 100 |
| 评价人员手机号 | userTel | varchar | 外键 | 20 |
| 操作人员手机号 | userTel | varchar | 外键 | 20 |

## 4.3 界面设计

### 4.3.1 手机端界面设计

本节主要介绍应急演练辅助教学系统的手机端界面设计与实现。手机端用户登录系统使用正确的手机号和密码即可，登录界面原型图如图4-16所示。



图4-16 手机端登录界面原型图

用户成功登录手机端系统之后，可以点击系统中下方导航栏中的“+”号，通过悬浮窗点击“开始直播”按钮，进入应急演练直播准备界面，首先需要在应急演练下拉列表中选择具体的应急演练，然后输入合适的应急演练直播标题就可开始直播。应急演练直播画面主要分为主体部分、交互部分和功能栏三部分，主体部分是显示应急演练直播画面的部分，占满屏幕大小，交互部分是在主体部分之上的一部分，位于屏幕的左下角，显示当前应急演练直播中所收到的文字消息，功能部分在屏幕的底部，主要功能有“开启美颜”、“结束直播”和“切换摄像头”，应急演练辅助教学系统手机端直播准备界面原型图和直播界面原型图分别如图4-17、4-18所示。



图4-17 手机端直播准备界面原型图



图4-18手机端直播界面原型图

### 4.3.2 Web前端界面设计

本节主要介绍应急演练辅助教学系统的Web前端界面设计与实现， 登录界面原型图如图4-19所示。

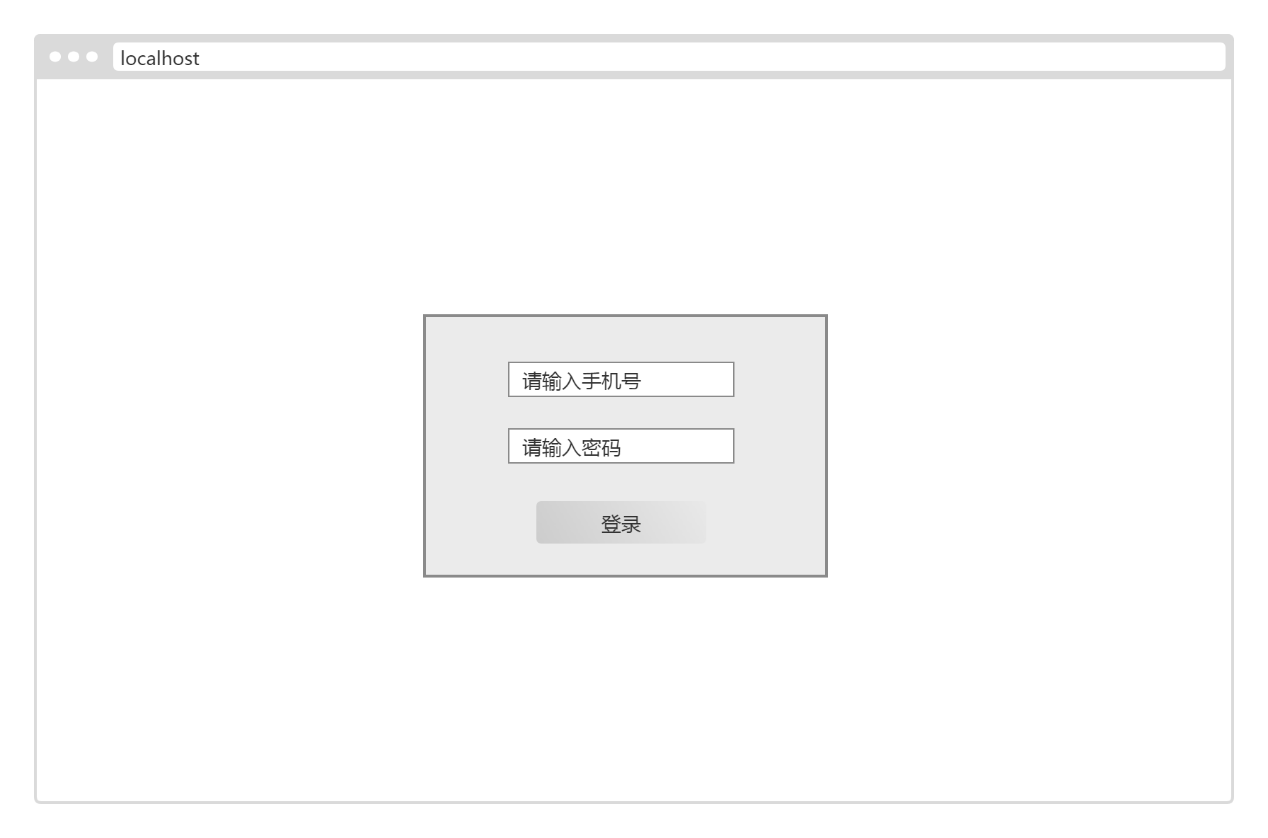


图4-19 Web前端登录界面原型图

用户登录成功的主页面由四部分组成，分别是页面头部、侧边栏、页面主体和页面底部。

页面头部包含系统名称和用户名，用户名右方的倒三角有退出登录的功能。侧边栏为应急演练辅助教学系统的导航栏，点击对应的功能模块，具体的页面会渲染到页面主题部分。页面底部是版权等信息。应急演练辅助教学系统主页面原型图如图4-20所示。

图4-20 Web前端主页面原型图

直播管理界面分为两部分，一部分为窗口的主界面，显示多个窗口的直播画面，可自定义配置每行的窗口数量，另一部分为右侧的信息交互区域，显示管理员用户或评价用户发送给所选择的直播窗口的文字信息，应急演练辅助教学系统直播管理页面原型图如图4-21所示。

图4-21 Web前端直播管理页面原型图

## 4.4 系统架构设计

### 4.4.1 系统物理架构设计

经过详细地了解，本文应急演练辅助教学系统所面向的部分高校内部已经有了非常完善的内部网络架构，学校内部的信息化水平也较为成熟，网络也较为稳定，因此，本文所设计与实现的应急演练辅助教学系统拥有较好的部署环境。

应急演练辅助教学系统是面向高校的，因此要考虑如下方面：一方面是系统的安全性，能够保证高校内部教师和学生安全访问，另一方面是系统需要具备稳定性和扩展性，基于以上考虑，本文所设计与实现的应急演练辅助教学系统的物理网络架构图4-22如下所示。



图4-22 应急演练辅助教学系统物理架构图

### 4.4.2 系统架构设计

本文应急演练辅助教学系统采用前后端分离的模式进行开发实现。这种开发模式将Web前端代码逻辑和服务器端业务逻辑进行解耦，两者之间只进行JSON数据的交互，同时减轻了动态服务器的压力，将静态内容请求分离出来，专心从事后台框架和业务的开发，进行更精准的单元测试，也使得Web前端代码更加规范化，模块化，让Web前端逻辑更清晰明了，极大地提高了开发效率。本文所设计与实现的应急演练辅助教学系统的架构图4-23如下所示。



图4-23 应急演练辅助教学系统架构图

## 4.5 本章小结

本章首先对应急演练辅助教学系统的系统架构进行设计，包括物理结构和逻辑结构，然后对系统的应急演练直播模块、应急演练录播模块、应急演练交互模块、应急演练配置模块、应急演练评价模块和用户管理模块六大主要功能模块进行详细的设计，阐述了模块中所包含的类与方法，以及其中执行的流程，并介绍了基于数据库表结构设计的数据库，最后展示了手机端和Web前端的界面设计方案。

# 第5章 系统实现与测试

本文应急演练辅助教学系统Web前端基于Vue实现。服务器端基于Java语言、Spring Boot框架实现。手机端基于Uni-app框架进行实现。本章将主要阐述系统实现和测试过程，包括系统运行环境的搭建、系统部署方案、系统主要功能模块的实现以及系统测试等内容。

## 5.1系统功能实现

下面介绍本文应急演练辅助教学系统的核心功能实现，主要包括应急演练直播模块、应急演练录播模块、应急演练交互模块、应急演练配置模块、应急演练评价模块和用户管理模块。

### 5.1.1应急演练直播模块

本文应急演练辅助教学系统Web前端是面向管理员用户和评价用户。管理员用户通过手机号和密码可成功登录系统主页面，点击侧边栏直播管理可以看到当前正在进行直播的窗口内容，如图5-1所示。

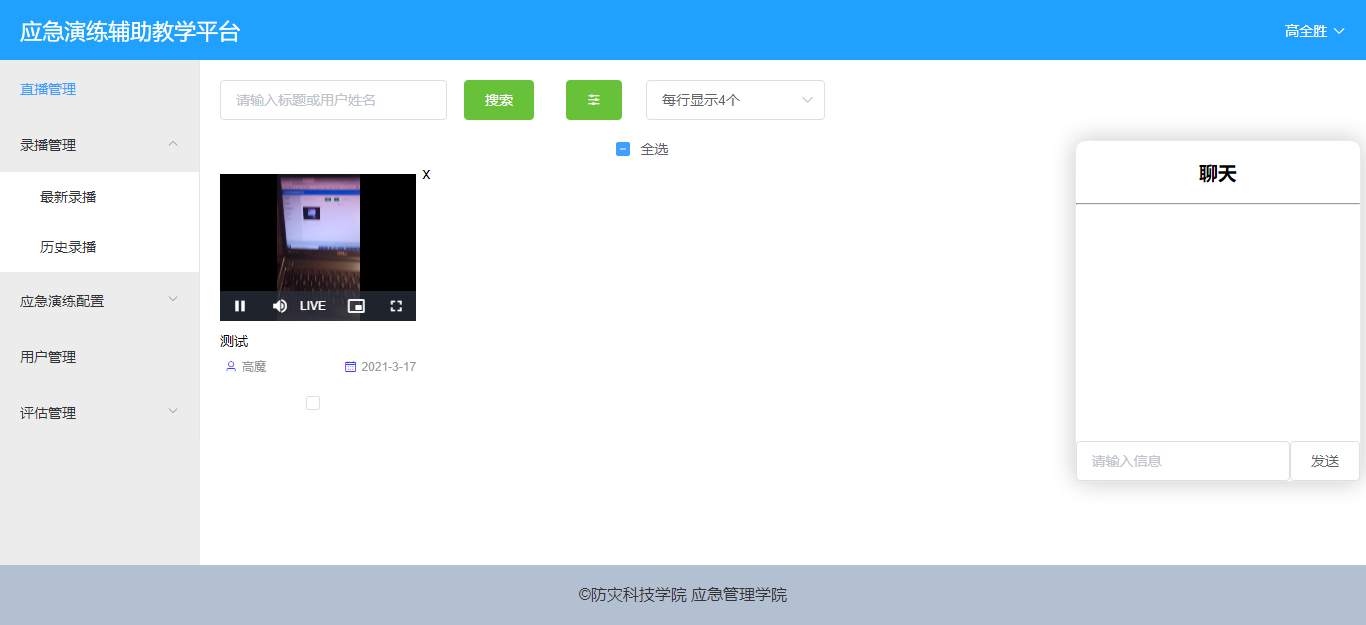


图5-1 Web前端直播界面图

下面分别应急演练辅助教学系统流媒体服务器配置文件和服务器端直播代码逻辑。

首先是流媒体服务器配置信息，流媒体服务器采用Nginx结合RTMP模块实现视频流的转发和存储，在nginx.conf文件中详细配置了信息，在rtmp模块中配置了rtmp协议的端口和频道的名称，在频道模块中配置了视频存储地址和回调函数，具体配置代码如下所示。

|  |
| --- |
| #user nobody;  worker\_processes 1;  events {  worker\_connections 1024;  }  rtmp {  server {  listen 1935;  application live {  live on;  record all;  record\_path /record;  record\_unique on;  on\_publish http://localhost:8080/video/publish;  on\_record\_done http://localhost:8080/video/record;  }  application vod {  play /record/;  }  application hls {  live on;  hls on;  hls\_path /tmp/hls;  record all;  record\_path /tmp/hls;  record\_unique on;  on\_publish http://localhost:8080/video/publish;  on\_record\_done http://localhost:8080/video/record;  }  }  }  http {  server {  listen 80;  server\_name localhost;  location / {  root html;  index index。html index。htm;  }  location /hls{  types{  application/vnd。apple。mpegurl m3u8;  video/mp2t ts;  }  root /tmp;  add\_header Cache-Control no-cache;  add\_header Access-Control-Allow-Origin \*;  add\_header Access-Control-Allow-Headers X-Requested-With;  add\_header Access-Control-Allow-Methods GET，POST，OPTIONS;  }  }  } |

服务器端在接收到流媒体服务器发出的直播回调之后，会通过HttpServletRequest参数获取必要的信息，包括流媒体服务器IP、频道名称、用户手机号、经纬度、直播标题、直播所属的应急演练等信息，将流媒体服务器IP、频道名称、用户手机号结合服务器端的配置文件组成完整的HLS协议的URL，将这些信息封装到Video对象之中，传递给业务逻辑层，进而传递给数据持久层存储到MySQL数据库中。之后通过websocket将直播信息传输给所以已成功连接websocket的用户，具体代码如下所示。

|  |
| --- |
| @RequestMapping（"/publish"）  public void startLive（HttpServletRequest request） throws Exception{  String tcurl = request.getParameter（"tcurl"）;  String nginxIP = tcurl.split（":"）[1];  String userTel = request.getParameter（"name"）;  String application = request.getParameter（"app"）;  String url = GraduationConfig.getHlsPre（） + ":" + nginxIP + ":" + GraduationConfig.getHlsPort（） + "/" + application  + "/" + userTel + GraduationConfig.getHlsExt（）;  Double lat =Double.parseDouble（request.getParameter（"lat"））;  Double lng = Double.parseDouble（request.getParameter（"lng"））;  String title = request.getParameter（"title"）;  String planId = request.getParameter（"planId"）;  Date startTime = new Date（）;  Video video = new Video（）;  video.setUserTel（userTel）;  video.setLat（lat）;  video.setLng（lng）;  video.setFlag（1）;  video.setStartTime（startTime）;  video.setUrl（url）;  video.setTitle（title）;  video.setPlanId（Integer.parseInt（planId））;  nginxService.startLive（video）;  webSocketServer.sendLiveVideo（webService.getLiveVideos（video）.get（0））;  } |

### 5.1.2应急演练录播模块

点击侧边栏录播管理会出现最新录播和历史录播菜单，最新录播默认播放当前时间的录播资源。页面主体中可以通过设置开始时间和结束时间来筛选符合条件的录播资源，同时可以选择多个录播资源进行同步播放，具体内容如图5-2所示。



图5-2 Web前端录播界面图

这里介绍一下多个录播资源同步播放的逻辑实现，首先获取所有已选择的录播视频资源列表，比较其中所有资源的开始时间，得到最大的开始时间，然后遍历录播资源列表，设置当前时间即可，具体代码如下所示。

|  |
| --- |
| synVideos（）{  let latestTime = this.getLatestTime（this.checkedVideos）;  let videos = this.$refs['myvideo'];  videos.forEach（video => {  video.seek（latestTime）;  }）  }，  getLatestTime（videos）{  let timeList = videos.map（video => {  return new Date（video.startTime）;  }）  let latestTime = timeList.reduce（（time1，time2） => {  return time1>time2?time1:time2;  }）  return latestTime;  }， |

服务器端在接收到流媒体服务器直播结束的回调事件之后，首先获取视频资源路径，通过yamdi给视频资源添加元数据，使得录播视频可以随意拖拽播放，接着通过HttpServletRquest获取流媒体服务器IP地址，结合服务器端配置文件中的配置信息，组成HLS协议的视频播放地址，然后封装成Video对象依次传递给业务逻辑层和数据持久层，最后保存到Mysql数据中，具体代码如下所示。

|  |
| --- |
| @RequestMapping（"/record"）  public void recordDone（HttpServletRequest request）throws Exception{  String path = request.getParameter（"path"）;  String fileName = metadataToVideo（path）;  String application = request.getParameter（"app"）;  String tcurl = request.getParameter（"tcurl"）;  String nginxIP = tcurl.split（":"）[1];  String videoUrl = GraduationConfig.getHlsPre（）+":"+nginxIP+":"+  GraduationConfig.getHlsPort（）+  "/"+application+"/"+fileName;  String userTel = request.getParameter（"name"）;  Date date = new Date（）;  Video video = new Video（）;  video.setVideoUrl（videoUrl）;  video.setUserTel（userTel）;  video.setEndTime（date）;  video.setFlag（0）;  nginxService.recordDone（video）;  }  public static String metadataToVideo（String path） throws Exception {  String osName = System.getProperty（"os.name"）;  StringBuilder stringBuffer = new StringBuilder（）;  if （osName.equals（"Linux"））{  stringBuffer.append（"/bin/sh -c yamdi -i "）;  }else {  stringBuffer.append（"cmd /c yamdi -i "）;  }  stringBuffer.append（path）;  stringBuffer.append（" -o "）;  String newPath = path.split（"\\."）[0]+"\_."+path.split（"\\."）[1];  stringBuffer.append（newPath）;  System.out.println（stringBuffer.toString（））;  Runtime.getRuntime（）.exec（stringBuffer.toString（））;  return newPath.split（"/"）[newPath.split（"/"）.length - 1];  } |

### 5.1.3应急演练交互模块

应急演练交互模块是管理员用户与摄影用户进行实时文字通信的操作，管理员用户可以选择一个或多个正在进行直播的窗口，在右方聊天窗口中数据要发送的文字信息，点击发送即可传输至已选择的摄影用户的直播画面中，具体显示如图5-3 5-4所示。

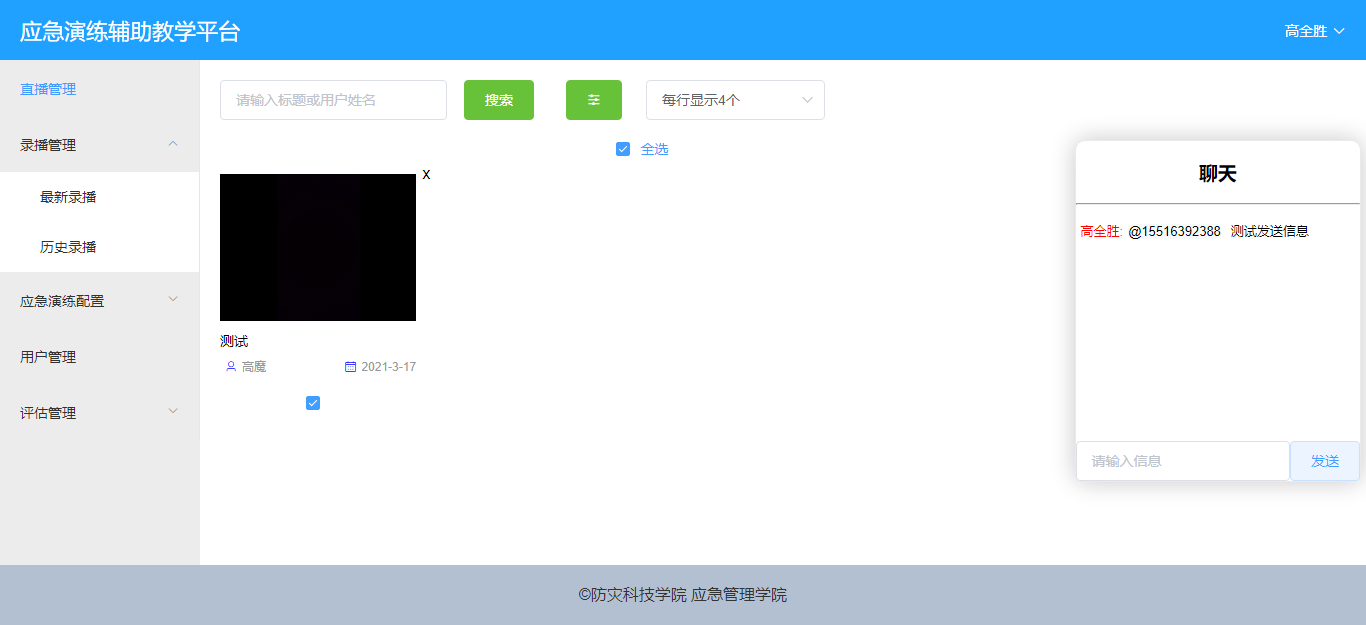


图5-3 应急演练交互界面图

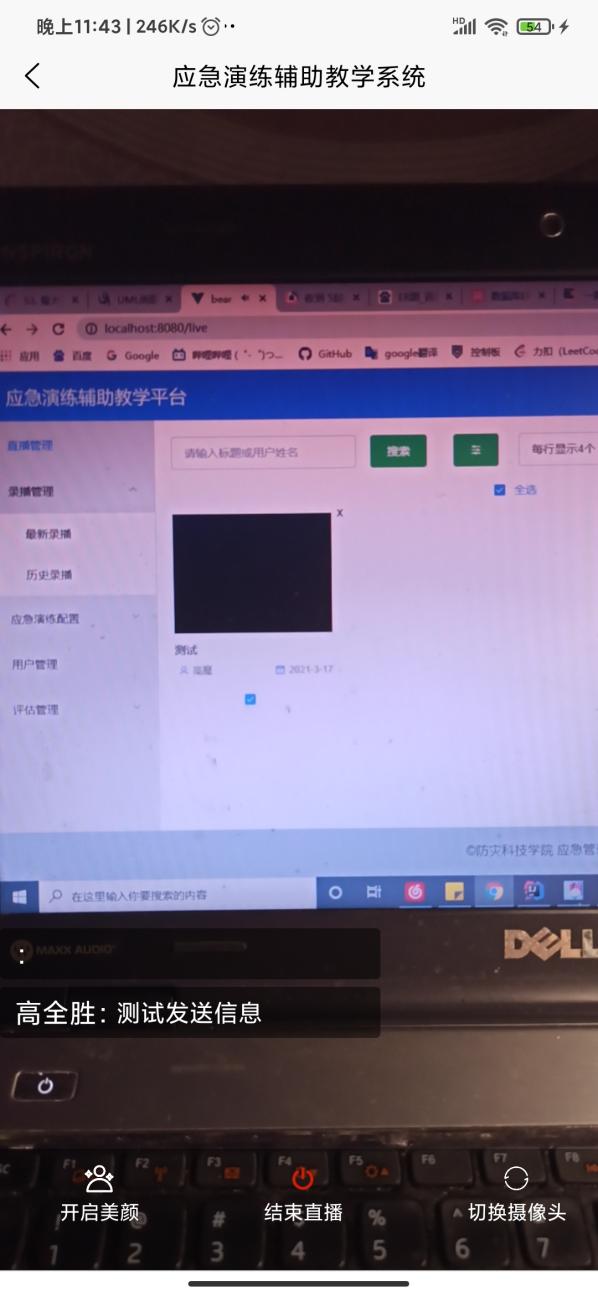


图5-4 手机端交互界面图

管理员用户登录至Web前端进入直播管理页面时，会将当前用户自动连接到服务器端的websocket，同时摄影用户登录到手机端开始直播时，也会连接到websocket，服务器端通过session的形式保存连接的用户信息。管理员用户发送消息时，会将要发送的消息和发送的用户传输至服务器端，服务器端通过处理消息，遍历要发送的用户列表，然后依次发送消息内容，实现文字交互的功能，服务器端交互模块具体代码如下所示。

|  |
| --- |
| @OnMessage  public void onMessage（Session session，Message message）throws Exception {  User user = userService.findByUserTel（message.getSendUserTel（））;  System.out.println（user）;  message.setSendUserName（user.getUserName（））;  for （String userTel:message.getUserTels（）.split（"，"））{  if（webSocketMap.containsKey（userTel））{  webSocketMap.get（userTel）.session.getBasicRemote（）.sendObject（message）;  }else{  System.out.println（"未连接"）;  }  }  } |

### 5.1.4应急演练配置模块

应急演练配置模块包括演练管理和附件管理。演练管理主要是配置演练的名称、开始时间和结束时间等信息，应急演练配置页面如图5-5所示。



图5-5 应急演练配置界面图

附件管理页面如下所示，点击页面中的链接可以实现跳转并展示，



图5-6 附件管理界面图

上传附件功能的具体实现是通过服务器端的upload（）方法实现的。首先通过MultipartFile接收Web前端发送的文件对象，然后通过MultipartFile获取文件的原始名称，通过原始名称得到文件的后缀名，接着使用UUID加上后缀名组成新的文件名，并把文件存储到新文件名称的路径中，最后进一步将真实路径转换为虚拟路径，包装成Attachment对象返回Web前端，具体代码如下所示。

|  |
| --- |
| @PostMapping("/upload")  public Attachment upload(MultipartFile file, HttpServletRequest request) throws IOException {  String filename = file.getOriginalFilename();  String ext = getExt(filename);  String newFileName = UUID.randomUUID().toString()+ext;  File absoluteFile = getAbsoluteFile(newFileName);  file.transferTo(absoluteFile);    String path = getPathFileName(newFileName,request);  Attachment attachment = new Attachment();  attachment.setPath(path);  attachment.setFileName(filename);  return attachment;  } |

### 5.1.5应急演练评价模块

应急演练评价模块包括指标管理、评价人员管理、评价管理等，指标管理页面如图5-7所示。



图5-7 指标管理界面图

指标管理的功能是对应急演练添加自定义评价指标，其中指标类型有单选、打分和评语三种类型，Web前端通过区分三种指标类型来添加指标，具体代码如下所示。

|  |
| --- |
| submit(){  //过滤指标选项的内容  if(this.indexForm.indexType == this.indexTypeList[1].value){  let tmp = []  this.dynamicOptionList.forEach(option => {  tmp.push(option.value);  });  this.indexForm.options = tmp.join("\_")  this.indexForm.minScore = "";  this.indexForm.maxScore = "";  }  if(this.indexForm.indexType == this.indexTypeList[2].value){  this.indexForm.options = "";  this.indexForm.minScore = "";  this.indexForm.maxScore = "";  }  if(!this.indexForm.planId){  this.indexForm.planId = this.queryParams.planId;  postRequest("/index/indexes",this.indexForm).then(res => {  this.flag = false;  this.getIndexes();  this.$message({  message: '新增成功',  type: 'success'  });  })  }else {  putRequest("/index/indexes",this.indexForm).then(res =>{  this.flag = false;  this.getIndexes();  this.$message({  message: '修改成功',  type: 'success'  });  })  }  }, |

评价人员管理页面如图5-8所示。



图5-8 评价人员管理界面图

评价人员管理是管理员用户对特定的应急演练添加评价用户的功能，其中为了防止添加重复的评价用户，首先根据应急演练id,返回已有的评价人员用户列表，将Web前端新添加的评价人员列表与已有评价人员列表作差集，然后将差集结果作为要新增的评价用户即可，服务器端具体代码如下所示。

|  |
| --- |
| @Override  @Transactional  public AjaxResult addReviewers(Reviewer reviewer) {  List<Reviewer> reviewers = reviewerMapper.getReviewersByPlanId(reviewer);  List<String> oldReviewers = new LinkedList<>();  for (Reviewer item: reviewers){  oldReviewers.add(item.getUserTel());  }  String userTels = reviewer.getUserTel();  String[] userTel = userTels.split("-");  ArrayList<String> newReviewers = new ArrayList<>(Arrays.asList(userTel));  newReviewers.removeAll(oldReviewers);  for (String item : newReviewers){  reviewerMapper.addReviewers(reviewer.getPlanId(),item);  }  return AjaxResult.success();  } |

评价管理页面如图5-9所示。



图5-9 评价管理界面图

评价管理是评价用户根据管理员用户设置的评价指标，对应急演练执行情况作出具体评价。详细代码如下所示。

|  |
| --- |
| handleUpdate(row){  if(!row.scoreId){  postRequest("/score/scores", row).then(res => {  this.$message({  type: "success",  message: res.msg  })  this.getTargets();  })  }else {  putRequest("/score/scores", row).then(res =>{  this.$message({  type: "success",  message: res.msg  })  this.getTargets();  })  }  } |

### 5.1.6用户管理模块

应急演练辅助教学系统的管理员用户可以通过此模块对整个系统中的所有用户进行管理，包含新增用户、删除用户、批量删除用户、修改用户信息、查看用户列表等操作，用户管理的界面如图5-10所示。



图5-10 用户管理界面图

点击添加用户按钮会弹出一个表单弹出框，其中包含新增用户的角色、姓名和手机号等信息，系统默认密码为123456。添加用户的界面如图5-11所示。

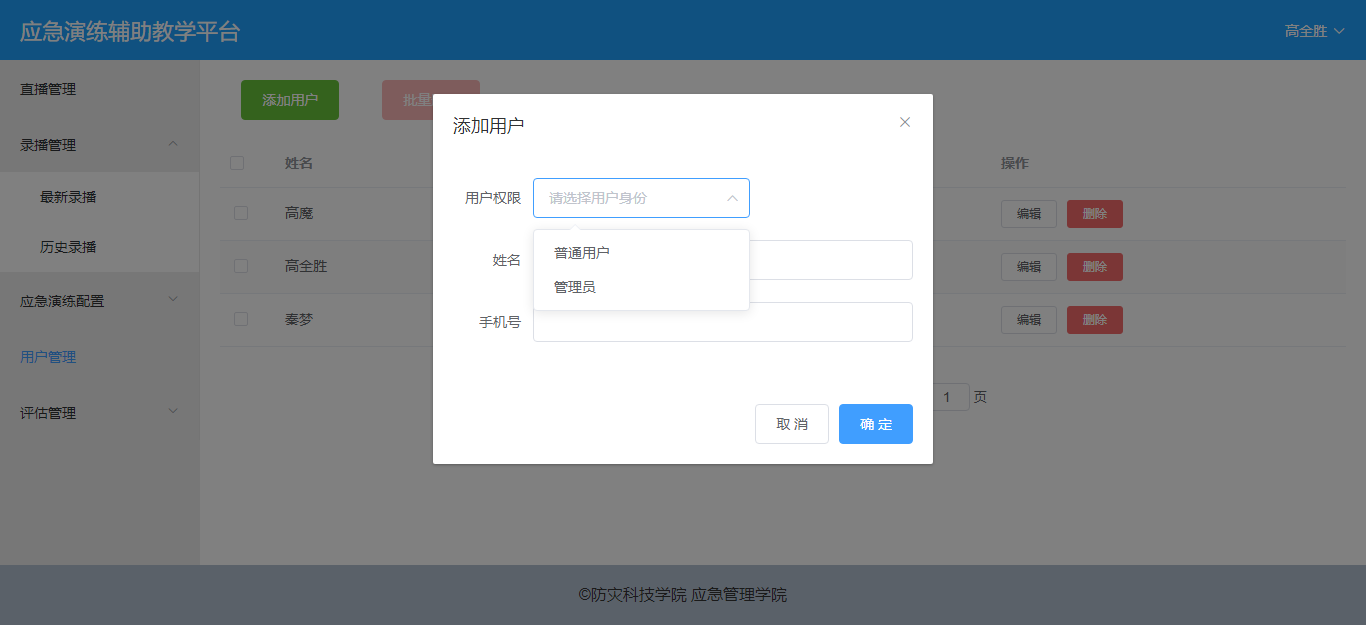


图5-11 添加用户界面图

应急演练辅助教学系统中用户删除功能可以通过表格中每一行数据对应的删除按钮进行逐个删除，也可以通过用户表格前方的选择框来多选用户进行批量删除，系统是删除功能是依赖用户的手机号实现的。

用户编辑操作也是依赖添加用户的表单实现，如图所示，系统会将对应的用户信息动态的添加到弹出框表单中，且只能修改用户姓名信息，用户角色和手机号不允许修改。其中Web前端代码如下所示。

|  |
| --- |
| addUser(){  this.clearForm();  this.flag = true;  this.title = "添加用户"  },  submit() {  if(this.title == "添加用户"){  postRequest("/user/users", this.userForm).then(res => {  this.flag = false;  this.getUsers();  console.log(res);  this.$message({  message: res.msg,  type: 'success'  });  });  }else if(this.title == "编辑用户"){  putRequest("/user/users",this.userForm).then(res => {  this.flag = false;  this.getUsers();  this.$message({  type: "success",  message: res.msg  })  })  }  } |

服务器端添加用户具体代码如下所示。

|  |
| --- |
| @Override  public int addUser(User user) {  List<User> users = userMapper.getUsers(user);  if (users.size() > 0){  return 0;  }  user.setUserPwd("123456");  return userMapper.addUser(user);  } |

## 5.2系统测试

系统测试是对整个系统的测试，将硬件、软件、操作人员看作一个整体，检验它是否有不符合系统需求的地方。主要目的是尽可能多地发现已编程序中存在的错误，以及通过与系统的需求定义作比较，发现软件与系统定义不匹配或与之矛盾的地方。

系统测试是对整个系统的测试，将硬件、软件和操作人员作为一个整体来检查是否满足系统的要求。其主要目的是通过与系统需求定义的比较，找出所编制软件中尽可能多的错误，找出软件与系统定义之间的不匹配或矛盾。

### 5.2.1测试环境

本文应急演练辅助教学系统在设计与实现完成之后，综合考虑了高校内部网络环境，在服务器端使用了一台服务器对本文系统进行了部署和测试，其中分别部署了Nginx流媒体服务、数据库服务和应用服务。在手机端使用了一台Android手机对本文手机端系统进行部署，相关配置如表5-12所示。

表5-12 设备环境表

|  |  |  |
| --- | --- | --- |
| 设备名称 | 硬件环境 | 软件环境 |
| 服务器 | CPU: Intel(R) Xeon(R) X3400 2.40GHz  内存:1T  运行内存:8GB | Windows 10 |
| 手机端 | 手机版本:Redmi Note 8 Pro  内存:128GB  运行内存:6GB  CPU:处理器Helio G90T 8核2.05GHZ  分辨率:2340\*1080  屏幕尺寸:6.53英寸 | Android 版本:10  MIUI版本:12.0.5 |
| 客户端 | CPU: Intel(R) Xeon(R) E5-2650 v2 2.60GHz  内存:3.5T  运行内存:32GB | Windows 7 |

### 5.2.2测试方案

黑盒测试，又称功能测试或数据驱动测试，是已知产品的功能测试。用于测试各功能是否能正常使用。在测试时，把程序看作一个不能打开的黑盒子，在完全不考虑程序内部结构和内部特性的情况下，测试人员进行直接测试，检查系统功能是否按照需求规格说明书的规定正常使用、是否能适当的接收输入数据而输出正确的结果等，检查相应的文档是否采用了正确的模板、是否满足规范需求。

本文所设计与实现的应急演练辅助教学系统，在进行系统测试时，拟采用黑盒测试方案，对系统的功能进行详细的测试。在系统测试过程中，依据系统需求说明书，对本文应急演练辅助教学系统进行功能测试，验证功能是否正常，以及是否安全执行，及时发现系统的缺点和漏洞，进行修改和更正。

### 5.2.3测试报告

本文所设计与实现的应急演练辅助教学系统，依据系统需求说明，对系统的Web前端和手机端分别进行了测试，包括界面测试和功能测试，以保证此系统能够符合系统需求，各个功能安全正常的运行。

由于本系统涉及的功能模块较多，本文篇幅有限，下面只列举出本系统核心功能测试报告，如表5-13所示。

表5-13 系统功能测试表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 测试项目 | 测试目的 | 测试方法 | 测试结果 | 改进意见 |
| 用户登录 | 用户合法登录 | 使用正确的用户名和密码以及使用错误的用户名和密码分别进行登录，尝试SQL注入 | 正确的用户名和密码可正常登录系统，错误的用户名和密码无法成功登录，尝试SQL注入失败 | 无改进意见 |
| 新增应急演练 | 验证应急演练配置的流程 | 管理员用户通过弹出框输入演练名称、开始时间和结束时间 | 对应急演练名称的限制只有长度限制，无其他限制 | 应急演练名称应该限制特殊字符 |
| 应急演练直播 | 验证直播流程 | 摄影用户选择应急演练和输入标题进行直播，直播时尝试切换摄像头 | 应急演练直播标题输入长度没有进行限制，输入标题过长时导致观看应急演练直播画面扭曲 | 对应急演练直播标题进行长度限制 |
| 应急演练录播 | 验证录播功能 | 管理员用户通过标题和时间筛选应急演练录播视频并播放 | 流程符合系统需求 | 无改进意见 |
| 应急演练交互 | 验证与多个直播进行文字交互 | 管理员用户选择一个或多个应急演练直播窗口发送文字消息 | 管理员用户选择应急演练直播发送消息时，显示的为用户手机号 | 应该将用户手机号改为用户姓名 |
| 录播同步 | 验证多个视频同步播放功能 | 管理员用户选择两个或两个以上的录播视频资源进行同步播放 | 流程符合系统需求 | 无改进意见 |
| 评价管理 | 验证评价用户对应急演练进行多指标评价 | 评价用户根据应急演练指标对应急演练进行评价 | 流程符合系统需求 | 无改进意见 |
| 用户管理 | 验证用户新增、删除和编辑功能 | 管理员用户新增、删除、批量删除、编辑和翻页查看用户 | 流程符合系统需求 | 无改进意见 |

在系统测试过程中，除功能测试外，还需要测试系统的并发性能，确保多个用户在同时直播时系统能够良好地运行，满足系统的业务需求。

本文应急演练辅助教学系统在并发测试过程中，使用FFmpeg多媒体处理工具模拟多用户直播功能，模拟了20个用户同时进行直播，并发的进行直播推流测试。测试结果表明，在直播用户并发访问量为20的情况下，系统的直播功能能够良好地运行，直播延迟时间与单用户直播延迟时间相同，能够满足高校对并发性能方面的要求。

## 5.3系统运行环境与系统部署

本文应急演练辅助教学系统采用前服务器端分离的模式进行实现，Web前端采用Vue框架进行开发实现，结合Visual Studio Code编辑器，Visual Studio Code是一款轻量级、免费、丰富的插件系统的编辑器。服务器采用Java语言进行编程实现，结合Spring Boot框架，编辑器采用Intellij IDEA集成开发环境，IDEA提供智能的代码提示功能，极大地提高了开发效率。手机端采用Uni-app框架进行编程实现，使用HbuilderX编辑器。本节将介绍在系统开发实现的过程中开发环境的搭建。

1. Web前端

下面介绍文本系统搭建Web前端开发环境的基本步骤。在项目的开发过程中，主要通过npm来管理项目中所依赖的第三方库，通过package.json文件管理项目信息，配置脚本， 以及指明项目依赖的具体版本等，其中包含用于网络请求的axios，用于页面渲染的element-ui，用于在html中播放hls流的flv.js，还有项目核心js框架Vue，以及用来路由跳转的vue-router等等，Web前端配置具体依赖如下所示。

|  |
| --- |
| "dependencies": {  "@babel/core": "^7.10.3"，  "@babel/preset-env": "^7.10.3"，  "axios": "^0.19.2"，  "core-js": "^3.6.4"，  "element-ui": "^2.4.5"，  "flv.js": "^1.5.0"，  "video.js": "^7.6.6"，  "videojs-flash": "^2.2.1"，  "vue": "^2.6.11"，  "vue-baidu-map": "^0.21.22"，  "vue-router": "^3.1.5"，  "vuedraggable": "^2.23.2"，  "vuex": "^3.1.2"  } |

接下来Web前端项目中的配置文件采用了多环境配置，将生产环境和开发环境相分离，提高开发效率。首先需要在package.json中指明命令和配置文件的映射关系，这里我们配置了dev开发环境和prod生产环境，具体命令映射代码如下所示。

|  |
| --- |
| "scripts": {  "serve": "vue-cli-service serve"，  "build": "vue-cli-service build"，  "lint": "vue-cli-service lint"，  "dev": "vue-cli-service serve --mode dev"，  "prod": "vue-cli-service serve --mode prod"，  "build:dev": "vue-cli-service build --modev dev"，  "build:prod": "vue-cli-service build --modev prod"  } |

1. 服务器端

下面介绍本文系统搭建服务器端开发环境的具体步骤。Spring Boot基本的开发环境可以由IDEA编辑器来搭建实现，项目中所需要的依赖库由Maven来管理，Maven是一个优秀的项目管理工具，通过pom.xml文件来管理项目依赖，我们可以在构建项目初期通过IDEA来勾选项目中所需要使用的各种库依赖，也可以在pom.xml文件中手动添加依赖库。此外，通过配置YML格式的配置文件，来配置项目中所需要的各种数据，结合多配置文件来在不同的环境中配置不同的参数来便捷的切换配置数据。

Spring Boot有约定大于配置的特点，简化了很多繁琐的配置，所以由Spring Boot官方和其他第三方开源了很多的Spring Boot Starter包可以快速搭建出一个项目的脚手架。这些Starters作为Spring相关技术的引导过程，我们不再需要担心它们之间的依赖关系，它们将由Spring Boot Starters自动管理。Starters包含了许多你需要的依赖项，以使项目快速启动和运行，并且具有一致的、被支持的一组管理传递依赖项。在Maven中配置Spring Boot的代码如下所示。

|  |
| --- |
| <parent>  <groupId>org.springframework.boot</groupId>  <artifactId>spring-boot-starter-parent</artifactId>  <version>2.2.6.RELEASE</version>  <relativePath/> <!-- lookup parent from repository -->  </parent>  <groupId>com.bear</groupId>  <artifactId>bear-springboot</artifactId>  <version>0.0.1-SNAPSHOT</version>  <name>bear-springboot</name>  <description>Demo project for Spring Boot</description>  <properties>  <java.version>1.8</java.version>  <mybatis-spring-boot-starter.version>2.1.2</mybatis-spring-boot-starter.version>  <fastjson.version>1.2.68</fastjson.version>  <hutool-all.version>5.2.3</hutool-all.version>  <commons-lang3.version>3.9</commons-lang3.version>  <pagehelper.spring.boot.starter.version>1.2.5</pagehelper.spring.boot.starter.version>  <druid.version>1.1.14</druid.version>  </properties> |

接下来添加项目开发过程中所需要的依赖库，数据库相关的依赖有jdbc、MySQL、Mybatis、Alibaba数据库连接池，用于更加便捷的操作数据库和持久化数据。用于Web前端和服务器端双向通信的websocket依赖，用于日期数据处理的fastjson依赖等。具体依赖配置文件代码如下所示：

|  |
| --- |
| <dependencies>  <dependency>  <groupId>com.alibaba</groupId>  <artifactId>druid-spring-boot-starter</artifactId>  <version>1.1.14</version>  </dependency>  <dependency>  <groupId>com.github.pagehelper</groupId>  <artifactId>pagehelper-spring-boot-starter</artifactId>  <version>1.2.5</version>  </dependency>  <dependency>  <groupId>org.springframework.boot</groupId>  <artifactId>spring-boot-starter-jdbc</artifactId>  </dependency>  <dependency>  <groupId>org.springframework.boot</groupId>  <artifactId>spring-boot-starter-web</artifactId>  </dependency>  <dependency>  <groupId>org.mybatis.spring.boot</groupId>  <artifactId>mybatis-spring-boot-starter</artifactId>  <version>${mybatis-spring-boot-starter.version}</version>  </dependency>  <dependency>  <groupId>mysql</groupId>  <artifactId>mysql-connector-java</artifactId>  <scope>runtime</scope>  </dependency>  <dependency>  <groupId>org.projectlombok</groupId>  <artifactId>lombok</artifactId>  <optional>true</optional>  </dependency>  <dependency>  <groupId>org.springframework.boot</groupId>  <artifactId>spring-boot-starter-test</artifactId>  <scope>test</scope>  <exclusions>  <exclusion>  <groupId>org.junit.vintage</groupId>  <artifactId>junit-vintage-engine</artifactId>  </exclusion>  </exclusions>  </dependency>  <dependency>  <groupId>com.fasterxml.jackson.core</groupId>  <artifactId>jackson-databind</artifactId>  </dependency>  <dependency>  <groupId>org.springframework.boot</groupId>  <artifactId>spring-boot-starter-websocket</artifactId>  </dependency>  <!-- https://mvnrepository.com/artifact/com.alibaba/fastjson -->  <dependency>  <groupId>com.alibaba</groupId>  <artifactId>fastjson</artifactId>  <version>${fastjson.version}</version>  </dependency>  <dependency>  <groupId>cn.hutool</groupId>  <artifactId>hutool-all</artifactId>  <version>${hutool-all.version}</version>  </dependency>  <dependency>  <groupId>org.apache.commons</groupId>  <artifactId>commons-lang3</artifactId>  <version>${commons-lang3.version}</version>  </dependency>  </dependencies> |

然后添加maven plugin插件和阿里云镜像，在添加了maven plugin插件之后，当运行“mvn package”命令进行打包时，会将系统打包成一个可以直接运行的 JAR 文件，使用“java -jar”命令就可以将项目运行。这在很大程度上简化了应用的部署，只需要安装了 JRE 就可以运行。Maven默认的仓库地址是国外的，配置阿里云镜像之后能够有效地提高依赖库的下载速度，提升开发效率。相关配置代码如下所示：

|  |
| --- |
| <repositories>  <repository>  <id>public</id>  <name>aliyun nexus</name>  <url>http://maven.aliyun.com/nexus/content/groups/public/</url>  <releases>  <enabled>true</enabled>  </releases>  </repository>  </repositories>  <pluginRepositories>  <pluginRepository>  <id>public</id>  <name>aliyun nexus</name>  <url>http://maven.aliyun.com/nexus/content/groups/public/</url>  <releases>  <enabled>true</enabled>  </releases>  <snapshots>  <enabled>false</enabled>  </snapshots>  </pluginRepository>  </pluginRepositories> |

在服务器端使用了多环境配置文件配置Spring Boot项目，开发版本和生产版本的配置相分离，有利于项目的开发与维护，减少配置文件的修改。下面主要介绍生产版本的配置内容。首先对数据库进行配置，数据库连接池采用Druid， Druid是目前较好的数据库连接池，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。由于使用了Mybatis作为持久层框架，需要配置Mybatis映射文件位置以便Spring Boot正确加载映射文件，配置项目中实体类的全类名，简化了Mybatis映射文件中代码的编写。此外添加了自定义配置字段，用来规定视频资源的存放位置和URL组成等等。以下是相关配置代码。

|  |
| --- |
| spring:  datasource:  driver-class-name: com.mysql.cj.jdbc.Driver  type: com.alibaba.druid.pool.DruidDataSource  filters: stat  maxActive: 20  initialSize: 1  maxWait: 60000  minIdle: 1  timeBetweenEvictionRunsMillis: 60000  minEvictableIdleTimeMillis: 300000  validationQuery: select 'x'  testWhileIdle: true  testOnBorrow: false  testOnReturn: false  poolPreparedStatements: true  maxOpenPreparedStatements: 20  mybatis:  mapper-locations: classpath:mapper/\*Mapper.xml  type-aliases-package: com.bear.bearspringboot.entity  server:  port: 80  graduation:  path: E:/graduation/uploadPath  hlsExt: .m3u8  hlsPort: 9988  hlsPre: http  pageHelper:  helperDialect: mysql  reasonable: true  supportMethodsArguments: true  params: count=countSql |

1. 手机端

下面介绍应急演练辅助教学系统手机端的环境搭建。手机端基于Uni-app框架开发，Uni-app框架中以Vue为基础，因此手机端的依赖管理工具也是npm，此处就不一一展开描述了。手机端每个页面和路由路径一一对应，相关映射关系如下所示。

|  |
| --- |
| "pages": [  {  "path" : "pages/login/login"，  "style" : {}  }，  {  "path" : "pages/login/register"，  "style" : {}  }，  {  "path" : "pages/login/forget"，  "style" : {}  }，  {  "path" : "pages/me/index"，  "style" : {}  }，  {  "path":"pages/me/liveHistory"  }，  {  "path" : "pages/live/playVideos"，  "style" : {}  }，  {  "path" : "pages/live/live"，  "style" : {}  }，  {  "path":"pages/live/beforeLive"，  "style": {}  }，  {  "path" : "pages/home/index"，  "style" : {}  }  ，{  "path" : "pages/live/record"，  "style" : {}  }  ]， |

## 5.4 本章小结

本章对应急演练辅助教学系统进行了实现与测试。在系统开发之前，对系统的运行环节和系统部署方案进行了介绍，包含多配置文件。接下来本章介绍了系统的主要功能模块实现，包括应急演练直播模块、应急演练录播模块、应急演练交互模块、应急演练配置模块、应急演练评价模块和用户管理模块，基于实现效果图和具体实现代码逻辑详细介绍了实现过程。最后本章介绍了系统的测试环境和方案，对应急演练辅助教学系统进行了全面的测试，测试结果表明，本文应急演练辅助教学系统具备良好的健壮性和并发性，能够有效地满足业务需求。

# 第6章 总结与展望

## 6.1 工作总结

本文在充分调研用户需求之后，在系统需求分析的基础上，对应急演练辅助教学系统进行了设计与实现，系统主要包括应急演练直播模块、应急演练录播模块、应急演练交互模块、应急演练配置模块、应急演练评价模块和用户管理模块等六个模块。管理员用户可以自定义应急演练方案，根据应急演练方案中的内容进行实地演练并上传直播影像，评价用户实时观看多窗口直播，掌控应急演练全局执行状况，并根据评价指标对应急演练进行具体评价。

本文主要完成的工作如下：

1. 系统需求分析。 通过实地调研用户需求，结合相关文献，总结出应急演练教学中存在的问题，分析了系统的需求，包括可行性分析、功能需求分析和性能需求分析。
2. 系统结构设计。在确定了系统具体的需求之后，结合校园内部网络环境确定了项目的物理架构。通过对比框架和技术，确定了系统的逻辑架构，最后将系统需求具体分模块进行开发。
3. 系统实现与测试。在确定了系统的框架、技术和模块之后，在代码层面对需求进行了实现，然后结合主流测试方案，对系统进行了全面的测试。

## 6.2 研究展望

本文应急演练辅助教学系统满足了基本的预期业务需求，但是在系统开发期间，由于涉及技术较多，本人能力有限，因此，本文所设计与实现的应急演练辅助教学系统仍有一定的不足，存在改进的空间。主要有以下方面：

1. 本文所采用的流媒体服务器为本人手动搭建的服务器，相比于大公司现成的流媒体服务器，存在一定的缺陷，如视频资源存在卡顿的问题。
2. 在观看应急演练直播过程中，采用的协议为HLS协议，其中存在的弊端为延迟较大，大约有10s左右的延迟，导致直播的实时性较差。
3. 管理员用户与摄影用户进行交互时，采用文字交互，功能比较单一。

拟解决方案如下：

（1）使用市场上成熟的流媒体服务器，提升系统的性能，解决观看视频资源卡顿的问

题。

（2）可考虑使用RTMP协议，可将延迟降低至2-3s，但是RTMP协议的播放依赖Flash插件。另外可考虑使用WebRTC技术实现。

（3）用户交互功能可考虑加上语音通话的技术，使得交互指挥功能更具备实时性。

## 致 谢

时光荏苒，又到了一年一度的毕业季，三年期间收获了很多，也有很多遗憾和不足，即将告别校园踏入职场，心里很不舍啊。更多的是感谢。

感谢我的导师单维锋教授，感谢老师在我一无是处的时候接收了我，在我停滞不前的时候鞭策我、在我膨胀的时候打醒我，毫无疑问的是，您是我生命中最重要的贵人，让我明白了做事情要踏踏实实、不能弄虚作假，在这个急功近利的时代，是您让我看到了什么是科研精神，什么是实干精神，我也算从一个。。。转型了计算机的世界，即使我现在还不算一个很好的开发人员，

感谢学校中各个善良的老师

感谢我的同学们，相处融洽，

感觉我的未婚妻秦梦女生，从2013年到2021年，8年长跑，我没给你带去什么，而你却一直陪伴着我，心里有很多愧疚，，千里迢迢

感谢我的父母家人，是他们默默支持着我完成学业。