

DOI:10.20033/j.1003-7241.(2026)10-0168-06

基于 Android 与 Java 的音乐课程学习云系统优化

刘 佳

(宝鸡职业技术学院,陕西 宝鸡 721000)

摘 要:为解决传统音乐课程学习系统在 Android 环境下运行性能偏低、移动端适配性差、并发承载能力不足等问题,本文采用 Android 与 Java 技术,结合云计算架构完成音乐课程学习云系统的优化设计。硬件层面部署公有云存储环境,重构网络拓扑结构,优化复位、接口等关键连接电路;软件层面完成数据库存储、关系结构与数据表设计,基于 Android 程序架构与消息处理机制,搭建面向教师、学生、管理员 3 类角色的完整功能模块,实现课程管理、在线学习、虚拟乐器、录音考核、资源点播等核心能力。经多并发压力测试验证,所设计云系统在高用户量场景下响应更快、运行更稳定、信息错误率更低,可在手机等移动终端流畅运行,有效提升音乐课程数字化教学效率与移动端使用体验,为高校音乐教育信息化提供可靠技术方案。

关键词:Android 系统;Java 开发语言;音乐课程;学习云系统;云计算;移动学习

中图分类号:TP311.52;G434;TP319

文献标志码:A

文章编号:1003-7241(2026)10-0168-06

Cloud system optimization of music course learning based on Android and Java

Liu Jia

(Baoji Vocational Technology College, Baoji 721000, Shaanxi, China)

Abstract: To address the problems of low operating performance, poor mobile adaptation and insufficient concurrent carrying capacity of traditional music course learning systems in the Android environment, this paper adopts Android and Java technologies combined with cloud computing architecture to implement the optimized design of a music course learning cloud system. At the hardware level, a public cloud storage environment is deployed, the network topology is reconstructed, and key connection circuits such as reset and interface are optimized. At the software level, database storage, relational structure and data table design are completed. Based on the Android program architecture and message processing mechanism, a complete functional module for three roles of teachers, students and administrators is built to realize core capabilities such as course management, online learning, virtual instruments, recording assessment and resource on-demand. Multi-concurrency pressure tests verify that the designed cloud system features faster response, more stable operation and lower information error rate under high user volume scenarios, and can run smoothly on mobile terminals such as mobile phones, effectively improving the efficiency of digital music teaching and mobile experience, providing a reliable technical solution for the informatization of music education in colleges and universities.

Keywords: Android system; Java development language; music course; learning cloud system; cloud computing; mobile teaching

音乐是一种艺术形式和文化活动,具有陶冶情操、调节情绪、娱乐等功能,因此在医疗、教育等方面具有较强的应用价值。为了推动音乐的发展,提高人类的审美水平,在中国的各大高校均开设了音乐专业。音乐课程教学作为素质教育的重要组成部分,逐渐受到各大高校的重视^[1]。在教育数字化、信息化的发展背景下,设计了音乐教学系统,该学科系统与其他学科的教学系统相比,音乐教育需要感性思维,其表达方式也更加具有独特性。目前有效的利用计算机和互联网技术研发音乐课堂教学系统能够扩大受益群体、促进学生的自主学习,同时能够更加全面地改善感官体验与心理感受。

随着世界音乐教育事业的发展,音乐课程学习系统的

发展也得到了社会和个人越来越高的重视。现如今音乐课程学习系统已经成为教育部门以及各个高校音乐专业信息化建设过程中非常基础以及重要的环节,国内外也得出了相对成熟的研究成果。在国外的学习系统研究开发中,更多地聚焦于学生的主动学习、合作学习、创造学习,以及运用虚拟现实在线多媒体环境的特征去扩展学生的学习潜能和研究能力,从而实现高效学习。在国内有学者研究开发并设计了基于云存储的数字化音乐资源库及其教学平台,提高了信息技术在音乐教育方面的便利性。此外,在此方面的研究中还包括 Web 辅助的音乐课程学习系统、基于 Linux 操作系统的音乐课程学习系统以及基于数据挖掘的学习系统等,但是上述的研究成果主要是利用

收稿日期:2025-05-26;录用日期:2025-07-19

基金项目:陕西省职业教育教学改革研究课题研究成果(2024SZX650)

作者简介:刘 佳(1984—),女,硕士,副教授,研究方向为音乐学、职业教育。

引用本文:刘佳.基于 Android 与 Java 的音乐课程学习云系统优化[J].自动化技术与应用,2026,45(10):168-173. (Liu Jia. Cloud system optimization of music course learning based on Android and Java[J]. Techniques of Automation and Applications,2026,45(10):168-173.)

计算机设备进行系统的运行与操作,在移动端的运行性能较差。因此,为了扩大音乐课程学习系统的应用范围,利用 Android 开发工具和 Java 开发语言,在云计算技术的支持下,实现对音乐课程学习系统的优化设计,保证该系统能够在移动终端以及手机客户端上运行。云系统采用了国际上的 HFP 和 HDRDP 技术,提供了安全性更高的数据存储中心。另外在系统优化设计过程中应用了 Android 和 Java 两种工具,其中 Android 是在智能手机以及平板电脑上的运行提供技术支持和开发平台。而 Java 是一种具有强大功能的编程语言,综合 Android、Java 以及云计算技术,在传统音乐课程学习系统的基础上从而进行优化设计,在保证系统运行功能的同时,扩大了系统的受众范围。

1 音乐课程学习云硬件系统设计

音乐课程学习云系统的设计主要是结合音乐课程的教学体系,进行规划的过程,在此次设计中分别运用了现代教育、多媒体技术和云计算技术,并解决了在音乐教学中遇到的和可能发生的问题。在此次优化设计中,主要利用 Android 与 Java 两种语言,实现系统运行环境的转变,而系统的运行原理与课程内容基本不变^[2]。在现有资源和系统研究成果的基础上,按照安卓软件的开发原则,要

求系统的实现效果以直观的界面显示,并考虑对数据进行维护和保护工作,对系统的功能模块进行划分,有利于后续功能的维护与升级。云系统可以分为基础设施服务、平台服务和软件服务 3 个部分。在整个系统架构体系中,基础设施服务层从硬件设备的角度出发,为用户提供计算、存储以及通讯等服务。在传统硬件系统的基础上,结合 Android 云平台的运行特点,对云存储环境、通信网络环境进行转换和优化,并调整对应的连接电路^[3]。

1.1 云环境部署

目前广泛使用的云计算服务分为公有云和私有云两种方式,其中公有云就是由第三方提供商为用户提供的通过互联网络访问的云计算服务,而私有云就是为单一用户使用而建立的云计算服务。由于音乐课程学习系统的用户包括音乐教师、音乐专业学生以及其他相关学者,因此在云环境部署过程中选择公有云作为云环境^[4]。在云环境的配置与部署过程中,利用 Hadoop HDFS 技术实现 GFS 的开源,用来解决分布式系统中的一致性问题。

1.2 网络拓扑架构

此次通信网络的连接选用多层次开发架构设计模式作为基本拓扑结构,这样的拓扑结果能够有效降低服务器的负载,实现系统通信网络内部的负载均衡。音乐课程学习云系统的通信网络结构如图 1 所示。

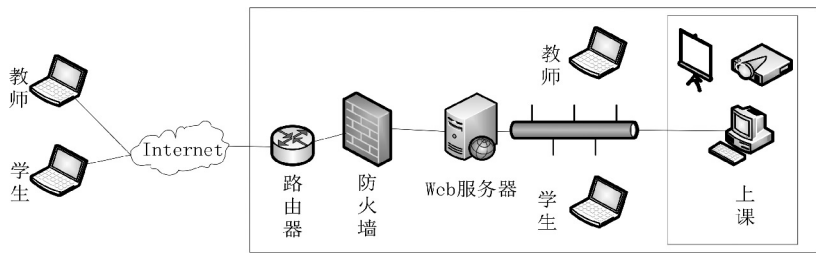


图 1 系统网络拓扑

Fig. 1 System network topology

从图 1 中可以看出,除了一般网络中的路由器设备和防火墙之外,通信网络的主要组成部分分别为服务器、客户终端和通信线路。

1.2.1 服务器端

音乐课程学习云系统使用 Java 语言来开发服务器程序,Java 语言的运行环境为 MyEclipse。利用 MyEclipse 连接数据库,设置数据库名和最大活动连接,保持 10 000 条连接等待,最少保持 30 条连接等待。同时设定数据库的用户名和密码用来获取数据源并保持服务器的长久连接与快速建立连接。

1.2.2 Android 客户端

考虑终端设备的性能与便携性,最适宜用来进行音乐课程学习云系统的运行终端设备是智能手机。智能手机的屏幕大小在 3 寸到 6 寸之间,适中的屏幕大小既能够保证有效的音乐课程内容,同时也具有良好的便携性。目前的 Android 操作系统支持云计算,且内置云服务。在通信网络的支持下,利用服务器和 Android 客户端形成无线网

络覆盖,构成的网络环境如表 1 所示。

表 1 网络环境分析

Tab. 1 Network environment analysis

网络通信	名称	带宽/Mbps	范围
WPAN	蓝牙	1	个人区域 10 m
	ZigBee	250	
WLAN	WiFi	100	局域连接 100 m
WMAN	WiMax	70	城域连接
	2 G	30~80	
	3 G	144~384	
WWAN	4 G	100	广域连接

结合音乐课程学习云系统的受众群体和应用范围选择网络环境的规模,并利用通信链路实现服务器和 Android 客户端之间的连接与通信。

1.3 电路设计

由于音乐课程学习云系统的存储模式转换为云存储

模式,因此需要对连接的电路进行调整。需要调整的电路包括复位电路、I/O 扩展电路以及意义硬件设备接口电路。通过复位电路的运行可以控制系统回到预先设定的硬件状态下,实现系统的重置工作。一般情况下系统的复位主要是在系统软件部分运行故障或网络故障的前提下进行的,最大程度地降低故障对数据造成的影响。在音乐

课程学习云系统中的复位电路上,通过 RESET 引脚控制复位电路是否进入重置状态^[5]。另外接口电路可以分为键盘接口电路、LED 显示接口电路等多个部分,其中键盘的行线与系统主控制器的 P1.3-P1.5 相连,列线分别与 P1.0-P1.2 相连。除此之外 LCD 显示接口电路的连接情况如图 2 所示。

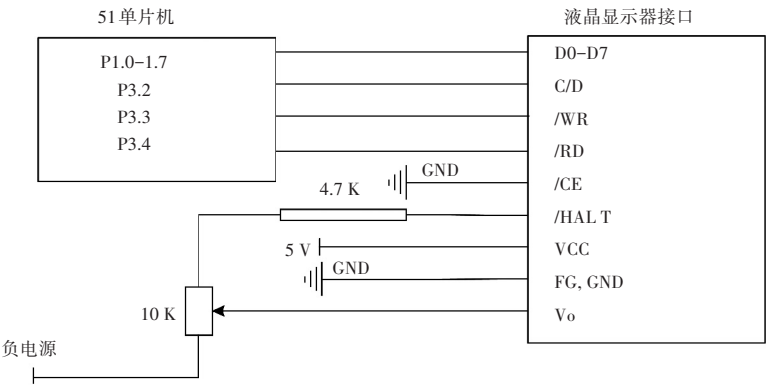


图 2 液晶显示接口电路图
Fig. 2 LCD interface circuit diagram

2 音乐课程云系统数据库设计

音乐课程学习云系统需要永久存储数据,在系统运行沟通用户与界面交互的中间介质,数据库的适用性、稳定性和安全性直接决定了系统以及软件的质量。根据对音乐课程学习系统的用户需求来分析,结合现有教学系统数据库的研究成果,为此次云系统数据库的建设奠定了有效的基础。

2.1 数据库存储设计

由于此次学习云系统改变了以往的数据存储方式,利用云计算技术实现数据的云存储,因此数据的存储方式也要随之发生^[6]。基于 Android 环境选择 MySQL 数据库技术作为数据存储的主要算法。该数据库在运行时需要定义参数 version,该参数用来决定数据库的版本,若当前版本高于之前版本,系统则会自动升级数据库版本,实现音乐课程学习数据的存储与更新。

2.2 数据库关系结构设计

数据库的关系结构设计包括了概念结构、物理结构以及逻辑结构 3 个方面,通常选择实体 E-R 图作为数据库概念设计的模型工具。物理结构也就是根据音乐课程学习云系统本身的无结构和存储方式进行设计,确定每个模块的存储结构、方式以及路径。而逻辑结构就是数据库中各个数据库表之间的类关系。在此次系统数据库中共创建 4 个数据类分别为 UserEntity 类、ReplayEntity 类、ObjEntity 类和 NoticeEntity 类。其中 UserEntity 类数据用于对应用户的各项信息进行记录,主要完成用户信息的管理功能,ReplayEntity 类用来记录系统产生的实时评论数据,ObjEntity 类用来记录音乐课程的相关信息,而 NoticeEntity 类记录了系统的提示相关信息。以 UserEntity 类数据为例,其具体的逻辑关系如表 2~4 所示。

表 2 UserEntity 类逻辑结构功能

Tab. 2 Logical structure function of UserEntity class

类功能描述	处理和用户相关的业务操作
所在包名称	com. classnet. entity
继承对象	无
实现对象	Serializable

表 3 UserEntity 类主要实现方法

Tab. 3 Main implementation methods of UserEntity class

保护属性	方法名	输入参数	输出参数	方法功能描述
public	getId	—	Intger	获取用户 id
public	setId	Intger id	void	设置用户 id

表 4 UserEntity 类属性结构定义

Tab. 4 Attribute structure definition of UserEntity class

保护属性	类型	名称	描述
private	Intger	username	姓名 长度限制 20 字符
private	String	password	密码 长度限制 20 字符
private	String	roletype	角色类型 管理员、教师或学生
private	String	truename	真实姓名 —
private	String	birth	出生日期 —
private	String	email	邮箱 —
private	String	sex	性别 —
private	String	tel	联系方式 —
private	String	img	头像 用户上传的头像突破

2.3 数据库表设计

数据库表包括用户信息表、音乐课程资料表、在线测试习题表、考试设置表等多个部分,其中音乐课程学习资料对应数据库表的具体设计情况,如表 5 所示。

表 5 音乐课程学习资料
Tab. 5 Music course learning materials

字段属性	字段类型	字段说明	是否为主键
MuID	int	音乐 ID 号	是
MuName	char	音乐名称	否
MuArtist	char	演唱者名称	否
MuAlbum	char	所属专辑	否
MuSize	Long int	文件大小	否
MuPath	char	存储路径	否
MuType	char	文件类型	否

同理可以得出其他数据库表的具体存储数据及其类型,并将其存储到云空间中对应的类别中。在数据库与前端的界面触发按钮之间形成链接,实现数据库的连接,方便数据的调用与存储。

3 音乐课程学习云系统功能设计

3.1 Android 程序结构与消息机制分析

Android 是音乐课程学习云系统的基本运行环境,在 Android 应用系统开发时首先需要了解 Android 操作系统的基本架构,一般来讲 Android 系统架构由内核层、系统运行库、应用程序架构和应用程序层 4 个部分组成。内核层主要用来连接系统的硬件和软件模块,通过内核层的连接保证软硬件两个部分的协同工作模式。系统运行库主要借助了 Android 运行时和 C/C++ 库两种模式,在不同的模式下针对的 Android 版本不同,也就是针对的移动终端设备的类型不同。Android 架构中的应用程序框架为底层功能及其相互之间的作用提供框架,实现功能数据之间的共享。而应用程序层是保证系统中功能程序能够顺利运行的关键。在 Android 环境下,每个线程只能拥有一个循环器实例和消息队列,在消息处理的过程中,消息队列会被设置成成员变量,并与处理者管理广播、内容接收器,活动以及服务组件,子线程里的循环器必须直接由循环器的 Prepare 来进行实例的创立与消息处理^[7]。

3.2 用户角色分类

音乐课程学习云系统的操作用户分为学生、教师和管理者,对不同的用户角色分别赋予不同的使用权限。普通课程学习学生的用户权限包括利用 Android 设备浏览系统提供的学习资源、进入音乐课程学习、在答疑讨论模块中提问、参与在线测试等。教师用户主要负责开设音乐课程、回答学生疑问、在线测试评分等功能,用户注册教师角色,需要提供能够证明其教师身份的说明和文件。而系统中的管理员拥有系统的最高权限,实现对系统后台数据库的管理以及用户权限的授予与分配。

3.3 教师功能模块

结合教师角色的功能,教师在注册用户并顺利登录系统后,添加音乐课程并设置课程的人数限制。选择音乐课程的形式分别为直播授课或录制授课,如果为直播授课则需要教师提前将音乐课程资料上传到系统数据库中,并生成资源调用连接,方便音乐课程资源的分发与调用。若选择的授课模式为录制授课,则需要提前录制音乐课程的视

频课件,并作为课程内容发布到系统中。在课程结束后,教师可以布置课后任务并接收学生的作业提交结果。在一个阶段的音乐课程学习后,教师可以发布考核或测试任务,完成整个音乐课程的授课任务。

3.4 学生功能模块

3.4.1 音乐课程云学习

结合系统中教师角色的音乐课程开设情况,可以将音乐课程内容分为五线谱教学、乐器教学、乐理知识教学、音乐欣赏教学等。以乐器教学中的模拟钢琴教学课程为例,该教学内容是实时演奏以及乐谱联动演示,该模块包括模拟钢琴的演示、钢琴键盘的放大缩小和钢琴音节的平移等^[8]。启动虚拟钢琴乐器并初始化钢琴参数,显示虚拟钢琴界面后进行监听界面的控制操作,判断是否需要缩放或平移,若需要则调用对应的界面变换程序,否则可以直接通过外设设备或触屏机制实现对虚拟钢琴的操作。

3.4.2 录音与播放功能

音乐的录音主要用于课后作业和在线考核模块,学生可以对作业或考核内容进行录制,并上传给教师端^[9]。而音乐的播放和暂停功能主要通过相应的功能键或屏幕位置实现。在音乐课程资料的播放和控制过程中需要获得调用音乐的具体资料,当学生用户使用播放器功能时,播放功能与暂停功能会随着用户的使用而依次进行转换。

4 系统测试

系统测试是音乐课程学习云系统开发过程中的最后一部分,是通过系统测试、检测该系统各个功能、流程的设计是否存在问题,若存在及时纠正。系统测试是保证音乐课程学习、教学资源综合信息管理、系统正常高效运行的最后一个环节,在测试过程中遵循系统的测试贯穿整个系统开发过程、考虑全面等原则。在测试程序时,检测程序执行时的工作情况,测试程序是否执行业务所需。

4.1 开发环境与开发工具

基于 Android 与 Java 的音乐课程学习云系统应用 Android 作为开发环境、Java 作为开发语言^[10],因此需要在主测设备上利用 Java 语言开发 Android 客户端,对应的开发环境如图 3 所示。

如图 3 使用的软件为 Myeclipse 软件,该软件与 Java 语言为运行工具,将对应的客户端信息程序导入到软件中,得出 Android 客户端的开发结果。另外系统需要云数据库的支持,因此利用 MySQL 工具实现数据库的开发。为了更加直观地得出系统测试结果,并方便调取系统的后台运行数据,在测试环境中安装夜神安卓模拟器,其主界面如图 4 所示。

如图 4 所示的模拟器软件采用了类似手机界面视觉设计的 PC 端桌面,可以通过计算机设备运行 Android 环境下开发的系统和软件。点击图 4 中标注的按钮,将设计的音乐课程学习云系统在模拟器中运行,并将对应的系统图标添加到模拟器的主界面中。

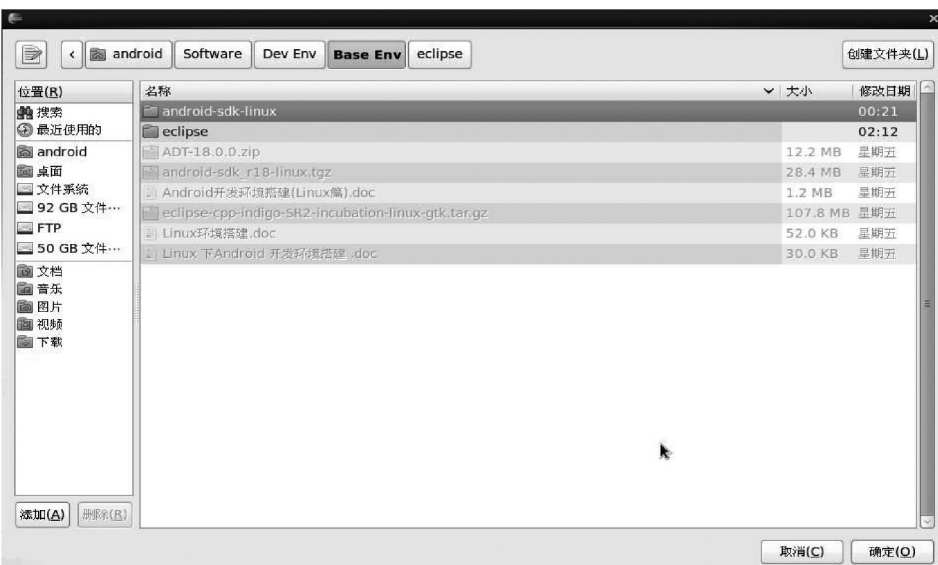


图 3 Android 客户端开发环境

Fig. 3 Android client development environment



图 4 夜神安卓模拟器主界面

Fig. 4 Main interface of nox android simulator

4.2 准备音乐课程素材与实验样本

为了保证音乐课程学习云系统的正常运行,准备大量的音频、视频以及图片数据,其中图片数据也就是音乐课程中的多媒体课件,视频素材就是录制的音乐课程视频,为音频软件具体包括学生上传的作业音频、音频教学资源等,将上述音乐课程素材按照其类型分别存储到云系统的数据库中^[11]。而实验样本也就是系统中的用户,准备 3 000 个 Android 移动终端并创建 3 000 个系统用户,以此作为系统测试的实验样本。

4.3 测试内容与测试过程

此次系统测试分别从基于 Android 与 Java 的音乐课程学习云系统的运行功能和应用性能两个方面进行具体测试^[12],其中运行功能也就是执行不同的操作,系统界面显示的结果与预期结果之间的差别,而系统的应用性能测

试主要就是在不同的负载压力下,系统响应时间的变化情况。为了形成实验对比,除了设计的音乐课程学习云系统外,还设置了传统的学习系统和文献[8]提出的基于数字化音乐课堂教学系统作为测试实验的两个对比系统。在相同测试的环境和工具运行下,得出 3 个系统的主界面。

4.4 系统测试结果分析

4.4.1 单元功能测试结果

单元功能测试是在硬件系统良好的前提下进行的,根据系统的设计需求制定测试用例,并设置对应的预期结果。运行系统的测试用例并得出各个功能模块单元的测试结果。此次系统测试中设置的用例内容包括音乐课程添加、课程视频观看、模拟乐器操作与控制、在线考核以及音乐文件上传等。其中模拟乐器操作的功能实现结果如图 5 所示^[13]。

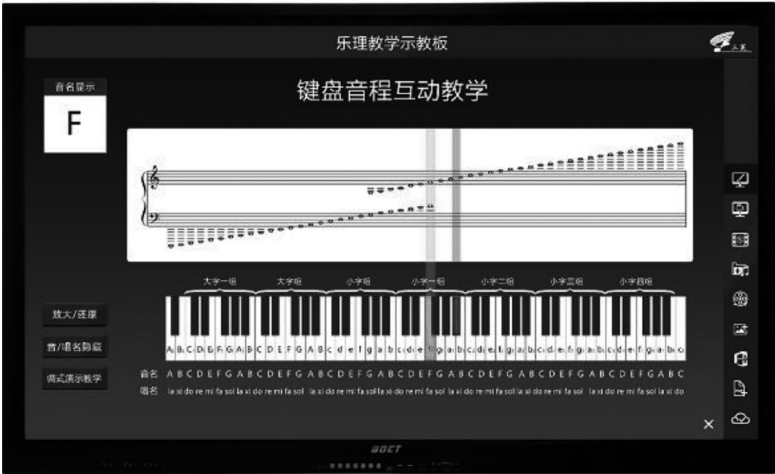


图5 音乐课程教学云系统实现界面

Fig. 5 Implementation interface of music course teaching cloud system

同理整合所有测试用例的界面响应结果,发现3个测试系统基本满足预期结果和测试要求,且在功能方面未发现明显差异。

4.4.2 压力性能测试结果

分别设置了不同的并发数量,也就是系统同时在线的用户数量,执行用户信息添加、修改、课程添加等操作判断系统是否能够按照预期正确的运行和显示^[14-15]。调取不

同环境下的信息错误率、信息吞吐量以及系统的平均响应时速,经过统计与对比得出测试结果,如表6所示。

从表6中可以看出,随着并发数量的增加,音乐课程学习云系统的响应速度明显下降,信息错误率逐渐增加。经过纵向对比发现,相比于两个对比系统,设计基于Android与Java的音乐课程学习云系统的响应时速更快、信息错误率更低^[16]。

表6 系统压力性能测试结果

Tab. 6 System pressure performance test results

并发数量	传统学习系统		文献[8]提出的学习系统		设计的学习云系统	
	信息错误率/%	响应时速/(kB·sec ⁻¹)	信息错误率/%	响应时速/(kB·sec ⁻¹)	信息错误率/%	响应时速/(kB·sec ⁻¹)
200	0.00	3 770.4	0.00	3 772.6	0.00	3 775.3
500	0.00	3 607.2	0.00	3 609.2	0.00	3 629.9
1 000	0.36	3 491.3	0.22	3 644.2	0.00	3 832.3
1 500	1.97	3 114.7	1.37	3 214.8	0.43	3 516.1
3 000	15.6	2 025.9	11.3	2 215.5	7.31	2 463.5

5 结论

通过Android与Java的音乐课程学习云系统的设计与开发改变了传统音乐课程教学单一的教学模式,将音乐教学推到了一个全新的高度。综合多种现代化技术,打破了教学系统在应用设备方面的局限性,从而扩大了课程的受众群体和用户范围。

参考文献

[1]李丹.高校专业课课程思政教学设计研究—以Android应用程序开发课程为例[J].电脑与电信,2022(3):22-25.
[2]贺钊.基于Android的音乐播放器设计与实现[D].南京:南京大学,2021.
[3]邓俊豪.Android本地音乐播放器的设计与实践[J].福建电脑,2021,37(3):89-91.
[4]陈莎莎.基于混合注意力机制的AI智能音乐播放器研究[J].自动化与仪器仪表,2024(12):187-190,195.
[5]孟丽强.基于改进无监督学习的音乐片段自动推荐技术[J].自动化技术与应用,2025,44(9):69-73.
[6]潘伟林.基于翻转课堂和任务驱动的Android教学分析[J].电脑知

识与技术,2020,16(15):168-196.
[7]黄劭涵.基于云课堂的线上线下大学音乐教学模式研究[J].北方音乐,2020(8):192-193.
[8]杨志玲.数字化时代背景下音乐课堂教学模式创新研究—评《数字化音乐教学软件设计与制作》[J].中国油脂,2022,47(11):171.
[9]卢铭君.基于Android与Java的音乐管理系统研究与设计[J].电脑知识与技术,2020,16(13):93-94.
[10]姚嘉怡.AI技术赋能高校音乐类课程教学的应用探索[J].参花,2025(15):149-179.
[11]周国斌.云计算平台下高校音乐教育资源共享云架构设计与性能优化[J].现代教育技术,2024,34(5):78-84.
[12]夏瑞清.深度学习驱动智能化在线音乐学习云平台构建与并发性能研究[J].数字教育,2025,11(2):45-51.
[13]邢文.边云协同架构下大规模在线音乐教学系统负载压力调度方法[J].计算机工程与应用,2024,60(9):192-198.
[14]王飞.基于改进CNN卷积神经网络的音乐识别模型构建[J].自动化技术与应用,2024,43(2):127-131.
[15]魏厚彪.基于公有云的远程音乐教育平台设计与实现[J].电脑知识与技术,2023,19(18):67-70.
[16]李沐阳.高校混合式音乐课程云服务系统吞吐量优化与压力仿真测试[J].信息技术,2025,49(3):112-116.