

**2025 届本科生毕业设计（论文）**

**基于Spring Boot的校园心理健康网站的设计与实现**

学 号： 2021005859

姓 名： 邓炀玉

学 院： 软件学院

专 业： 软件工程

班 级： 2125

指导教师： 王禹 软件开发工程师

高程昕 讲师

完成日期： 2025 年 4 月

声明及论文使用的授权

本人郑重声明：所呈交的毕业设计（论文）是本人在指导教师的指导下取得的研究成果，毕业设计（论文）写作严格遵循学术规范。除了文中特别加以标注和致谢的地方外，毕业设计（论文）中不包含其他人已经发表或撰写的研究成果。因本毕业设计（论文）引起的法律结果完全由本人承担，太原理工大学享有本毕业设计（论文）的研究成果。

论文作者签名 年 月 日

本毕业设计（论文）作者和指导教师同意太原理工大学保留使用毕业设计（论文）的规定，即：学校有权保留送交毕业设计（论文）的复印件，允许毕业设计（论文）被查阅和借阅；学校可以上网公布全部内容，可以采用影印、缩印或其他复制手段保存毕业设计（论文）。

论文作者签名： 指导教师签名：

签字日期： 年 月 日 签字日期： 年 月 日

太　原　理　工　大　学

毕业设计（论文）任务书

第1页

|  |
| --- |
| 毕业设计（论文）题目：  基于Spring Boot的校园心理健康网站的设计与实现  毕业设计（论文）要求及原始数据（资料）：  1．综述国内外校园心理健康平台研究现状；  2．深入了解基于Spring Boot框架开发Web应用的相关技术；  3．熟练掌握使用Spring Boot进行Web应用开发的能力；  4．设计并实现一个基于Spring Boot的校园心理健康网站；  5．深入分析基于Spring Boot的Web应用开发过程中的关键技术和挑战；  6．训练检索文献资料和利用文献资料的能力；  7．训练撰写技术文档与学位论文的能力。 |

第2页

|  |
| --- |
| 毕业设计（论文）主要内容：  1．综述心理健康支持系统在校园环境中的应用；  2．了解基于Spring Boot开发Web应用的相关技术；  3．熟悉使用Spring Boot进行Web应用开发的开发环境；  4．设计一个基于Spring Boot的校园心理健康网站的功能模块；  5. 深入分析并描述搭建和构建基于Spring Boot的心理健康网站的关键技术和流程；  6．熟练掌握基于Spring Boot平台的心理健康网站的开发技巧；  7．设计与实现针对特定功能模块（如心理咨询预约系统）的程序。  学生应交出的设计文件（论文）：  1．内容完整、层次清晰、叙述流畅、排版规范的毕业设计论文；  2．包括毕业设计论文、源程序等内容在内的毕业设计电子文档及其它相关材料。 |

　 第3页

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 主要参考文献（资料）：  [1] 胡荣,羊雪玲.基于Spring Boot前后端分离Web系统的设计与实现[J].新能源与智能网联,2024,(01):88-97.  [2] 徐智勇,戴祖旭.基于WebSocket协议的消息安全推送系统设计与实现[J].武汉工程大学学报,2024,46(03):310-316.DOI:10.19843/j.cnki.CN42-1779/TQ.202306003.  [3] 王珊, 萨师煊. 数据库系统概论(第5版)[M]. 高等教育出版社, 2014.  [4] 陈敬宗.计算机软件Java编程特点与技术应用分析[J].中国信息界,2025,(01):162-164.  [5] 雷帅,李娟,陈亮,等.校园心理健康平台“知你”APP设计与实现[J].电脑编程技巧与维护,2021,(10):64-66.DOI:10.16184/j.cnki.comprg.2021.10.024.  [6] 杨家炜.基于Spring Boot的web设计与实现[J].轻工科技,2016,32(07):86-89.  [7] 王鹏,甘琳琳.基于微信平台的大学生校园心理健康服务实践探析——以社会心理服务平台为例[J].农村经济与科技,2020,31(08):280-281.  [8] Kumar S , Sharma A , Mamun K , et al. A Deep Learning Approach for Motor Imagery EEG Signal Classification[C]. 2016 3rd Asia-Pacific World Congress on Computer Science and Engineering (APWC on CSE). IEEE, 2016.  [9] MEHEDI MASUD, SULTAN ALJAHDALI. Concurrent execution of transactions in a peer-to-peer database network[J]. International journal of intelligent information and database systems: IJIIDS,2011,5(5):510-531.  [10] Angelis D M ,Volpi L ,Giusino D , et al.Acceptability and Usability of a Digital Platform Promoting Mental Health at Work: A Qualitative Evaluation[J].International Journal of Human–Computer Interaction,2025,41(2):1246-1259.   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 专业班级 | 软件2125班 | | | 学生 | 邓炀玉 | | 要求设计（论文）工作起止日期 | | | 2025年2月17日~2025年6月6日 | | | | 指导教师签字 | | 高程昕 王禹 | | 日期 | 2025年2月20日 | |

基于Spring Boot的校园心理健康网站的设计与实现

摘　要

随着社会压力的增加和人们心理健康意识的提升，校园心理健康问题日益受到关注。为了更好地帮助学生缓解心理压力，本文设计并实现了一个基于Spring Boot的校园心理健康网站。该网站通过Spring Boot框架构建，结合前后端分离的技术架构，提供了在线心理咨询、心理测评、心理健康文章分享等功能。系统具有良好的可扩展性和高效性，能够为广大学生提供便捷的心理健康服务。通过对系统的设计与实现，本文展示了Spring Boot在Web应用中的优势，并讨论了相关技术的选型、架构设计以及系统安全等问题。最终，本文通过对系统的测试与反馈分析，验证了该系统在实际应用中的可行性和有效性。

关键词：Spring Boot；校园心理健康；网站设计；在线心理咨询；心理测评

Design and implementation of campus mental health

website based on Spring Boot

Abstract

With the increasing social pressure and growing awareness of mental health, campus mental health issues have gained more attention. To better assist students in relieving psychological stress, this paper designs and implements a campus mental health website based on Spring Boot. The website is built using the Spring Boot framework and adopts a front-end and back-end separation architecture. It offers online psychological counseling, psychological assessments, and mental health articles sharing functionalities. The system is highly scalable and efficient, providing convenient mental health services for students. Through the design and implementation of the system, this paper demonstrates the advantages of Spring Boot in web application development and discusses the selection of relevant technologies, architecture design, and system security. Finally, the system’s feasibility and effectiveness in real-world applications are verified through testing and feedback analysis.

**Key words:** Spring Boot;Campus Mental Health; Website Design; Online Counseling; Psychological Assessmen

目录

[摘　要 i](#_Toc21494)

[Abstract ii](#_Toc6552)

[1 绪论 1](#_Toc13685)

[1.1 研究背景与意义 1](#_Toc30533)

[1.1.1 研究背景 1](#_Toc7825)

[1.1.2 研究意义 1](#_Toc7078)

[1.2 国内外研究现状 2](#_Toc24552)

[1.2.1 国内研究现状 2](#_Toc25425)

[1.2.2 国外研究现状 2](#_Toc8317)

[1.3 主要研究内容 2](#_Toc10991)

[1.3.1 系统需求分析 2](#_Toc15653)

[1.3.2 系统设计 2](#_Toc19348)

[1.3.3 核心功能开发 3](#_Toc1031)

[1.3.4 数据安全管理 3](#_Toc22173)

[1.3.5 系统测试与优化 3](#_Toc13690)

[1.3.6系统应用与推广价值 3](#_Toc30130)

[1.4 研究方法 4](#_Toc15503)

[1.5 研究结构 4](#_Toc14236)

[2 系统需求分析 5](#_Toc24467)

[2.1 用户需求分析 5](#_Toc23204)

[2.1.1 管理员 5](#_Toc13493)

[2.1.2 心理医生 5](#_Toc10283)

[2.1.3 学生 6](#_Toc24187)

[2.2 功能需求分析 7](#_Toc2145)

[2.2.1 用户管理模块 7](#_Toc26458)

[2.2.2 内容管理模块 7](#_Toc10976)

[2.2.3 在线心理咨询模块 7](#_Toc7089)

[2.2.4 心理测评模块 8](#_Toc16648)

[2.2.5 AI智能客服模块 8](#_Toc11216)

[2.3 非功能需求分析 8](#_Toc21625)

[2.4 系统流程分析 10](#_Toc29483)

[2.5 总结 10](#_Toc4608)

[3 系统总体设计 11](#_Toc13809)

[3.1 系统架构设计 11](#_Toc13630)

[3.1.1 系统架构概述 11](#_Toc28841)

[3.1.2 系统模块划分 11](#_Toc25182)

[3.2 技术选型 12](#_Toc29068)

[3.2.1 前端技术选型 12](#_Toc26251)

[3.2.2 后端技术选型 13](#_Toc29092)

[3.2.3 数据库技术选型 13](#_Toc23817)

[3.3 系统功能模块设计 14](#_Toc21504)

[3.3.1 用户管理模块 14](#_Toc26137)

[3.3.2 内容管理模块 15](#_Toc9089)

[3.3.3 在线咨询模块 16](#_Toc17947)

[3.3.4 心理测评模块 16](#_Toc27657)

[3.3.5 AI智能客服模块 17](#_Toc13951)

[3.4 数据库设计 18](#_Toc12811)

[3.4.1 E-R图设计 18](#_Toc16266)

[3.4.2 数据库表设计 23](#_Toc9177)

[3.5 部署结构设计 26](#_Toc26873)

[3.6 总结 26](#_Toc14880)

[4 系统详细设计 27](#_Toc16647)

[4.1 系统架构设计 27](#_Toc26987)

[4.2 模块设计 28](#_Toc29799)

[4.2.1 用户管理模块 28](#_Toc21674)

[4.2.2 内容管理模块 28](#_Toc18214)

[4.2.3 在线咨询模块 28](#_Toc12564)

[4.2.4 心理测评模块 29](#_Toc32153)

[4.2.5 AI智能客服模块 29](#_Toc19922)

[4.3 安全性设计 29](#_Toc32192)

[4.4 总结 30](#_Toc1627)

[5 系统实现、测试和部署 31](#_Toc29897)

[5.1 系统实现 31](#_Toc23664)

[5.1.1 注册、登录实现 32](#_Toc27030)

[5.1.2 首页实现 32](#_Toc7946)

[5.1.3 个人中心实现 33](#_Toc11369)

[5.1.4 预约、咨询管理实现 34](#_Toc28101)

[5.1.5 心理测评实现 35](#_Toc21814)

[5.1.6 课程文章实现 37](#_Toc9349)

[5.2 系统测试 38](#_Toc19744)

[5.2.1 功能测试 38](#_Toc22892)

[5.2.2 性能测试 39](#_Toc1315)

[5.2.3 安全测试 40](#_Toc10075)

[5.3 测试结果分析 40](#_Toc19215)

[5.4 系统部署 40](#_Toc4494)

[5.5 总结 42](#_Toc30512)

[6 结论与展望 43](#_Toc28955)

[6.1 结论 43](#_Toc9046)

[6.2 研究不足与局限性 43](#_Toc12589)

[6.3 未来展望 44](#_Toc10345)

[6.4 总结 44](#_Toc20705)

[参考文献 45](#_Toc7076)

[致谢 46](#_Toc25285)

[外文原文 47](#_Toc31568)

[中文翻译 51](#_Toc31354)

1 绪论

1.1 研究背景与意义

1.1.1 研究背景

随着社会节奏的加快和学业压力的不断增加，大学生心理健康问题日益凸显，已成为高等教育中不可忽视的重要课题。据相关调查显示，越来越多的高校学生在学习、人际关系、就业等方面面临着心理困扰，严重者甚至出现焦虑、抑郁等心理障碍，影响其正常的学习与生活。如何有效、及时地发现并干预学生的心理问题，已经成为高校心理健康教育亟需解决的问题[1]。

传统的心理咨询服务主要依赖线下预约和面对面咨询，这种方式在效率、覆盖面和隐私保护等方面存在诸多不足，难以满足当下高校学生多样化、个性化的心理健康服务需求。尤其在疫情期间，远程心理咨询和信息化管理的重要性愈加凸显。因此，借助信息技术手段，构建一个功能完善、安全可靠、使用便捷的校园心理健康管理平台显得尤为迫切。

Spring Boot作为目前广泛应用的轻量级Java开发框架，具有上手快、配置简洁、易于部署等特点，特别适合用于快速开发和部署中小型应用系统。基于 Spring Boot技术开发的校园心理健康网站[2]，能够整合心理测评、咨询预约、在线互动、数据统计与分析等功能，为学生提供一个安全、温暖、易于访问的心理支持平台，同时也为心理辅导员提供高效的管理工具，从而提升整个高校心理健康教育的质量和效率。

1.1.2 研究意义

近年来，大学生心理健康问题日益受到社会各界的关注。随着社会压力的增加以及学业竞争的加剧，越来越多的学生面临心理困扰，如焦虑、抑郁、孤独等。然而，由于传统心理咨询方式的局限性，如资源有限、预约困难、隐私顾虑等，许多学生未能及时获得专业帮助。因此，构建一个基于Spring Boot的校园心理健康网站，能够为学生提供在线心理评估、咨询预约、心理知识科普等服务，从而提高心理健康服务的可及性与便利性。

该系统的构建不仅能够缓解校园心理健康资源不足的问题，还能结合现代信息技术，如大数据分析、人工智能问答系统等，实现智能化的心理健康服务。通过本项目的研究与实施，可为高校心理健康服务提供一种新的解决方案，提升大学生的心理健康水平。

1.2 国内外研究现状

1.2.1 国内研究现状

国内对于心理健康服务的信息化探索已经取得了一定进展。近年来，许多高校逐步建立了心理健康中心，并结合互联网技术推出线上心理咨询预约系统。此外，部分研究者开始探索人工智能在心理健康领域的应用，如情感分析、智能问答、心理测评等。然而，现有系统往往功能较为单一，缺乏智能化分析能力，无法有效满足学生多元化的心理健康需求。

1.2.2 国外研究现状

国外高校在心理健康服务的数字化方面起步较早，部分大学已建立完善的在线心理健康支持系统。例如，美国一些高校推出了结合人工智能的心理健康应用，能够提供智能心理测评、情绪分析等功能。此外，欧洲一些国家也在积极研究利用机器学习和自然语言处理技术优化心理咨询服务。然而，目前大多数系统仍然依赖于传统心理学理论，尚未充分发挥大数据和深度学习的优势。

1.3 主要研究内容

本研究主要围绕以下内容展开：

1.3.1 系统需求分析

为了准确把握高校学生在心理健康服务方面的实际需求，本研究是通过问卷调查和访谈相结合的方式收集用户反馈。问卷设计主要围绕学生的心理困扰类型（如焦虑、压力）、使用习惯以及功能期待咨询（如是否需要匿名）展开，共设计有效问卷10份。访谈部分与学生进行了深入交流，例如有同学提到“不想让别人知道自己有心理问题”，正是隐私保护的重要性。结合调研结果，本研究明确了系统的核心功能模块：心理健康、咨询预约系统和心理知识科普，同时强调隐私保护和高风险访问的支持。此外，通过分析高校现有的心理健康管理模式（如线下咨询），评估了线上系统的咨询评估与改进方向，为后续设计奠定了基础。

1.3.2 系统设计

系统后端采用Spring Boot框架进行开发，前端使用Vue.js实现用户界面。这种设计不仅提高了系统的扩展性和可维护性，还包括为后续功能的迭代升级。数据库方面，采用MySQL以及MongoDB作为数据存储方案，移植在表结构设计中，特别关注用户信息与评估数据的分类存储[3]。

1.3.3 核心功能开发

本系统的核心功能开发围绕学生实际需求展开，具体包括以下模块：

心理健康测评：集成了标准化的心理测试量表，如焦虑量表（GAD-7）和抑郁量表（PHQ-9）。学生可在线完成测试，系统自动计算分数并提供简要反馈，例如“你的焦虑指数偏高，建议适当放松”。本模块主要完成了量表的界面布局与结果展示逻辑，依托现量表提高了开发效率。

预约系统：支持学生在线选择咨询师与预约时间，并提供预约功能。学生可以使用昵称代替真实姓名提交请求，从而降低心理负担，提升咨询使用意愿。

心理知识科普：构建了一个心理健康知识库，内容涵盖常见心理问题的成因（如考试压力导致的焦虑）、应对策略（如深呼吸法）以及自我调节建议。这些内容参考权威资料整理而成，力求语言通俗易懂，以学生理解与应用。

1.3.4 数据安全管理

数据管理是系统稳定运行的关键，本研究对此进行了初步设计。用户信息采用加密技术保护，例如密码通过哈希算法处理，防止数据泄露；敏感信息如咨询记录则进行脱敏感存储。心理测评数据单独保存，既属于学生查看历史结果，也为学校心理咨询中心提供匿名化统计支持，例如分析“近期焦虑学生比例的变化趋势”。咨询记录管理模块支持学生回顾历史建议，形成持续的心理健康跟踪工具。同时，所有数据操作均遵循高效性原则，确保系统在实际应用中的可靠性。

1.3.5 系统测试与优化

系统开发完成后，开展了多轮测试与优化工作。首先进行功能测试，验证各模块运行是否正常，例如提交是否成功提交、测评分数是否准确计算，确保无重大Bug。其次针对性能优化，发现数据库查询响应较慢时，通过添加索引和调整SQL语句子提升了效率，系统响应时间大约80ms。最后，根据用户反馈改进前端界面，例如调整按钮布局、优化字体大小，使交互更加人性化。测试过程较为繁琐，但尽管通过反复进行，系统整体稳定性与用户体验均得到显着提升。

1.3.6系统应用与推广价值

设计与实现不仅完成了系统技术层面的开发任务，还具备一定的实际应用价值。首先，该系统可作为高校心理健康服务的补充工具，解决了咨询资源不足的问题。其次，系统中的测评数据与统计功能能够为心理心理咨询中心提供决策支持，例如通过学生分析此外，该系统具有较好的可移植性，可根据不同高校的需求进行功能定制，如增加心理讲座预约或咨询心理活动管理模块。

1.4 研究方法

文献研究法：通过查阅国内外关于心理健康服务系统、Web系统开发以及Spring Boot框架的相关资料[4]，深入了解心理健康服务的现状、存在的问题及技术发展趋势，为系统的功能设计与架构规划提供理论支持。

需求分析法：通过调研高校心理健康服务现状，结合实际应用场景，对用户角色的功能需求进行系统分析，从而明确系统的功能模块与实现目标。

系统设计与开发法：采用Spring Boot作为系统的核心开发框架，结合前后端分离思想，完成系统的整体架构设计。

测试验证法：对系统的主要功能模块进行系统测试与用户测试，验证其功能的可行性、系统的稳定性及用户体验效果，确保系统能够满足实际使用需求。

1.5 研究结构

本文结构安排如下：

1. 绪论

介绍本文的研究背景、研究意义、研究内容与方法，结合国内外研究现状分析校园心理健康的发展状况，阐明本文的研究目的与研究内容。

1. 系统需求分析

分析校园心理健康平台的用户角色、功能需求与非功能需求，明确系统的整体业务逻辑与服务目标。

1. 系统总体设计

包括系统架构设计、数据库设计、功能模块划分、系统流程设计等，为系统开发提供全面的设计支撑。

1. 系统详细设计

详细介绍各功能模块的实现方式，如用户登录注册、心理测评、在线预约咨询、AI心理客服等，并展示系统的主要界面。

1. 系统实现与测试

说明系统测试的内容与方法，对测试结果进行分析与评价，检验系统的稳定性与可用性。

结论与展望

总结本文研究成果，指出当前系统不足与改进方向，并展望未来的发展趋势。

2 系统需求分析

在系统开发的初期阶段，需求分析是至关重要的一步。为了确保系统能够有效地满足用户需求，我们首先需要详细分析不同用户角色的功能需求及系统应具备的核心功能。本章将对系统的用户需求、功能需求和非功能需求进行深入分析，为后续的系统设计与实现奠定基础。

2.1 用户需求分析

校园心理健康网站的主要用户包括管理员、心理医生和学生三类，每类用户的需求不同，因此系统需要根据不同角色进行针对性设计[5]。

2.1.1 管理员

管理员是系统的核心管理者，负责系统的整体运营与维护。

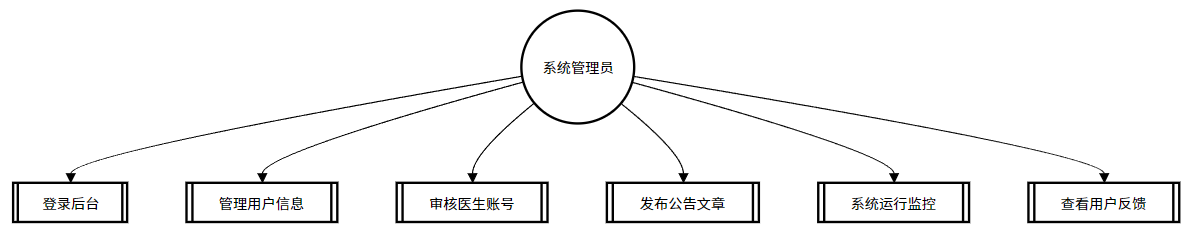


图2-1 管理员相关功能图

管理员的主要需求包括：

用户管理：管理员需要对系统中的用户（包括学生和心理医生）进行管理。具体操作包括用户登录注册、角色分配、用户权限控制（如学生与心理医生权限不同）等。

内容管理：管理员需审核医生提交的内容，确保文章内容的科学性和合法性，避免发布不实信息。

数据统计与分析：管理员可以查看系统的使用情况和用户行为分析报告，例如用户注册数、活跃用户数、咨询次数、测评参与率等数据。这些统计信息将直观的呈现给管理员，以便于优化系统功能和提升服务质量。

系统配置与维护：管理员可以设置系统参数，如工作时间、心理医生咨询时间、文章发布频率等，同时进行系统的日常维护，确保系统的正常运行。

2.1.2 心理医生

心理医生在本系统中的角色主要是为学生提供心理咨询服务，并通过平台发布心理健康资源。

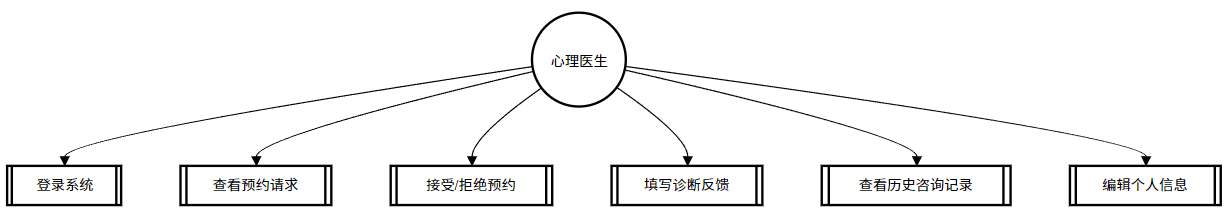


图2-2 心理医生相关功能图

医生的需求主要包括：

在线咨询：医生需要能够通过系统接受学生的预约咨询请求，进行在线文字咨询或视频咨询，并为学生提供个性化的心理干预方案。

内容发布与管理：医生可以通过平台发布心理健康相关文章、课程视频以及其他相关资源。发布的内容需要经过管理员的审批，以确保其符合科学依据。

心理测评管理：心理医生能够设计并发布心理健康测评题目，以帮助学生更好地评估自己的心理状态[6]。医生还可以根据测评结果提供针对性的建议。

记录查询：心理医生可以设置自己的工作时间，方便学生进行预约。同时，医生可以查看历史咨询记录，了解学生的心理健康状况变化。

2.1.3 学生

学生是系统的最终用户，主要目的是获得心理健康服务和支持。

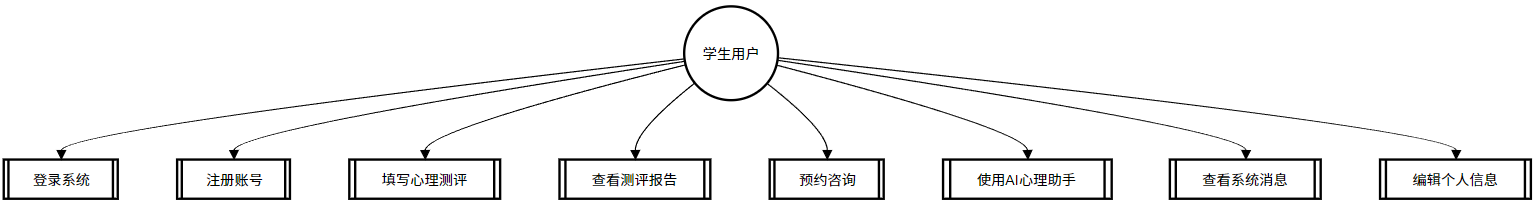


图2-3 学生相关功能图

学生的需求包括：

心理咨询服务：学生需要能够通过平台预约医生，进行在线咨询（文字或视频），获得针对性的心理健康建议与干预。

心理健康资源：学生希望能够通过平台访问到丰富的心理健康资源，包括文章、课程、视频等，帮助其提升心理调适能力。

心理测评与反馈：学生可以通过平台完成心理测评，了解自己的心理状态，并根据测评结果获得相应的建议或资源推荐。

隐私保护与安全性：学生对隐私非常关注，尤其是咨询记录、测评结果等敏感信息。因此，系统必须提供强有力的数据加密和安全保障措施，确保信息的安全性与保密性。

2.2 功能需求分析

为了满足不同类型用户的实际使用需求，本系统围绕学生、心理医生和管理员三种角色，设计并实现了若干核心功能模块，确保用户能够顺利完成各自任务，并保障平台运行的规范性与高效性。系统主要包括以下几个功能模块：

2.2.1 用户管理模块

本模块是整个系统的基础部分，主要进行角色识别和权限的管理。通过角色控制机制，限制各类用户的操作权限，保证系统安全性与使用的有序性。

用户注册与登录：支持用户进行账号注册和登录，并根据角色赋予相应的访问权限。登录时，系统会校验用户的身份信息，并根据其角色（管理员、学生或心理医生）跳转至相应的首页界面，提供个性化服务入口。

用户角色管理：角色可以划分为学生、心理医生、管理员，通过控制每个角色的权限，来提升系统的灵活性与安全性。

账号信息管理：用户登录后可以进入个人中心，修改自己的基本信息，包括头像、用户名、联系方式、修改密码等。

2.2.2 内容管理模块

内容管理模块是系统进行心理健康知识发布的重要模块，管理方面主要面向心理医生和管理员。通过该模块，心理医生可以发布相关文章，管理员负责对文章内容进行审核与维护。

内容分类与标签管理：系统需支持对内容进行分类，并添加相关的标签，方便用户能够快速的筛选，可以根据自己的兴趣和需求进行搜索，大大提升检索效率，提升学生的体验。

内容搜索与推荐：平台提供强大的搜索引擎，支持通过一些关键字词、相关分类、不同的标签等多维度的搜索功能。系统还应根据学生的行为和兴趣进行个性化推荐，带来更好的体验。

2.2.3 在线心理咨询模块

该模块是系统最为重要的功能，主要实现学生与心理医生之间的线上预约、一对一的在线对话以及记录等。

预约咨询：学生可以查看心理医生的排班表，并预约合适的时间进行咨询。

实时聊天与视频咨询：平台支持文字对话和视频交流功能，学生与心理医生之间可以进行实时的交流。

咨询记录管理：系统会自动记录学生每次咨询的内容以及反馈，心理医生和学生可以随时查看他们的历史记录。

2.2.4 心理测评模块

心理测评模块为系统提供科学的心理评估，有助于发现学生的心理问题，并为学生提供个性化的建议和方案。

测评设计与管理：医生可以设计并发布心理健康测评，确保测评内容符合心理学理论。

测评结果分析与反馈：学生完成测评后，系统会自动的生成测评分析报告，并为学生提供相关的心理健康意见。

测评内容审批：管理员对心理医生发布的测评内容进行审核，确保内容的科学性和合理性。

2.2.5 AI智能客服模块

AI智能客服模块为学生提供基础的心理健康咨询服务，提升平台的响应效率，为学生提供7×24小时服务。

自动回复与基础咨询：AI客服能够根据学生的问题提供基础的心理健康知识和常见问题解答，帮助学生更好地了解心理健康的重要性。

情绪识别与反馈：通过大语言模型的训练，AI客服能在与学生交流对话的过程中识别其学生的情绪倾向，比如焦虑、烦躁、心情低落等，并提供相应的情绪支持，帮助学生有良好的对话体验。

智能学习与优化：AI客服会根据用户的提问内容、各种反馈等数据进行不断优化，自动丰富其知识库，提高回答的准确率和好的体验，逐步形成具有独特个性化的服务能力。

2.3 非功能需求分析

在系统开发过程中，除了实现各个功能需求之外，同样也要考虑到非功能性要求的实习，它们从系统整体层面来确保系统的稳定运行、安全保障以及用户满意度。良好的非功能设计能够提升系统的综合质量，并为后续的维护与升级奠定基础。具体非功能需求包括以下几个方面：

系统性能：系统需具备一定的并发处理能力，以保障在大量的用户访问或在高峰期使用时，系统仍然能够保持快速响应并保证系统稳定运行。考虑到学生在心理测评、咨询预约、内容浏览等功能上的集中操作行为，系统后端应支持高并发处理方案，例如引入异步化处理、使用缓存以及采用负载均衡策略，来提升整体吞吐能力。前端页面也要进行性能优化，减少多余的请求，提升页面加载速度，比如采用懒加载等手段，从而增强用户体验。

系统安全性：鉴于系统涉及到学生心理状态、咨询记录等非常敏感的个人数据，系统必须要实现安全性相关的保障。首先，所有用户数据在传输过程中应采用HTTPS协议进行加密，这样能够防止中间人攻击；其次，一些敏感数据在数据库中要加密存储，并结合访问控制机制，来避免非授权访问。同时，系统还应具备完整的权限校验与日志记录功能，一旦发生异常行为或数据泄露风险，管理员能够在第一时间进行追溯并采取应对措施。此外，系统还需定期地进行安全分析，保持系统环境的健康与安全。

系统可用性：系统应保障持续在线，提供7×24小时服务，确保学生在任何时间都能访问网站，尤其是在心理压力突发或夜间情绪低落等关键时刻，系统要保持稳定运行。为此，系统可采用高可用的部署方式，比如主备之间的随时切换、具备常见的容灾方案等，一旦主服务器发生故障，可快速切换到备用节点，最大限度降低因系统停机而对用户产生影响。同时，也要设计友好的错误提示和容错机制，保证即使部分服务短暂的异常，也不会影响用户的核心操作。

系统扩展性：随着系统用户规模的不断扩大以及功能的增加，系统应具备良好的可扩展性，便于后续进行进行迭代和升级。系统在架构上可以采用模块化的设计方案，能确保各功能模块之间相对独立，便于在后期进行修改和维护升级。在硬件资源层面，系统也应支持横向的扩展能力，如通过增加服务器的节点数来提升系统的处理能力。此外，数据库设计也应考虑扩展性，例如分库分表或读写分离，为未来高并发的访问提供技术支持。

用户体验：系统的用户界面应该注重简洁美观且逻辑清晰，功能布局要符合用户的使用习惯，能快速上手使用。尤其是对于学生用户群体，页面要减少操作的复杂度，引导学生能快速找到其所需功能。例如，在首页提供快捷入口至预约咨询、心理测评、AI客服等常用功能，方便使用。同时，在交互设计上应关注系统的响应速度，提升用户使用的流畅性。系统还应适配多终端的访问，包括PC端和移动端，能满足用户在不同场景下的访问需求。

2.4 系统流程分析

依据用户角色进行流程分析：

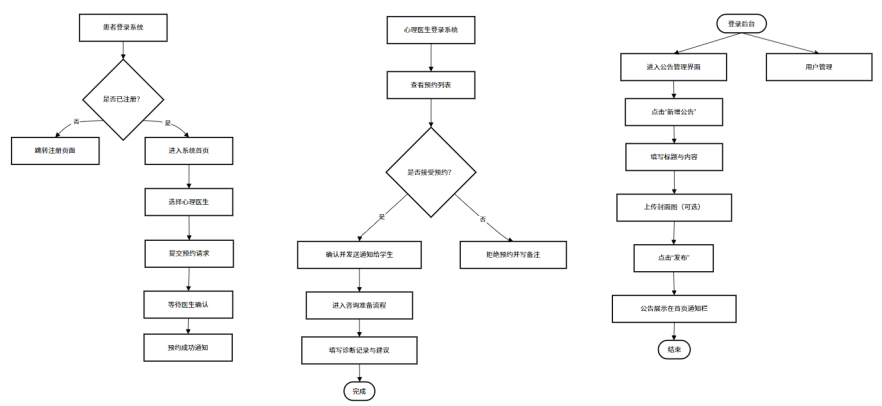


图2-4 学生、心理医生。管理员参与流程图

2.5 总结

本章对系统需求进行了全面而详细的分析，重点从用户需求、功能需求和非功能需求三个方面进行了深入探讨。首先，通过对不同用户角色的需求分析，明确了系统所需要实现的核心功能模块。包括用户管理、在线心理咨询、内容管理、心理测评以及AI智能客服等。

在功能需求方面，系统将根据不同角色的特点，提供个性化的功能访问权限。例如，管理员可以进行角色分配和权限管理；心理医生可发布心理健康内容并提供在线咨询；学生则能通过系统来预约心理咨询、参加心理测评等，进一步促进系统的交互和使用。系统还需实现一些复杂的内容管理功能，如内容发布、分类、标签管理以及根据学生的兴趣提供个性化推荐，增强系统的知识传播能力。

非功能需求方面，本系统不仅关注系统性能的优化，确保在学生大量访问的高峰期依然能够快速响应；同时，也保障学生的隐私安全等。在可用性上，系统将确保24小时服务，并设计容灾备份机制，保证服务的持续性和可靠性。扩展性方面，系统将采用模块化和可扩展的架构，支持未来用户量增长和功能拓展的需求。最后，用户体验作为系统设计的核心之一，将力求实现简洁、直观的界面设计和流畅的操作流程，降低用户的学习门槛，提升用户的使用感受。

通过对这些需求的深入分析，本章为后续的系统设计与实现提供了明确的指导方向，确保系统不仅能够满足用户的基本需求，还能提供高质量的服务和良好的用户体验，为系统的长期发展奠定基础。

3 系统总体设计

在深入进行系统的详细设计之前，首先需要明确系统的总体架构和设计方案。本章将基于第二章中提出的系统需求分析，结合实际的技术实现，全面阐述系统的整体架构设计及各功能模块的实现方式。通过合理的架构设计与技术选型，确保系统具备高效性、可扩展性、稳定性等关键特性，以满足不同用户群体的需求，并为后续的开发和维护提供有力支持。

3.1 系统架构设计

3.1.1 系统架构概述

本系统采用前后端分离的架构，前端负责展示和交互，后端负责处理业务逻辑并进行持久化保存。前后端分离的架构能够提高开发效率，减少不同模块间的耦合度，利于系统的维护与扩展。

系统的架构设计是实现系统各项功能需求的基础，其主要目的是确保系统能够在大量用户访问的情况下保持稳定性和高效性。本系统采用分层架构设计思想，将整体系统划分为若干层次，保证每个层次之间的职责分明，既提升了代码的可维护性，也便于后续功能的扩展。

系统整体架构设计如图3-1所示：

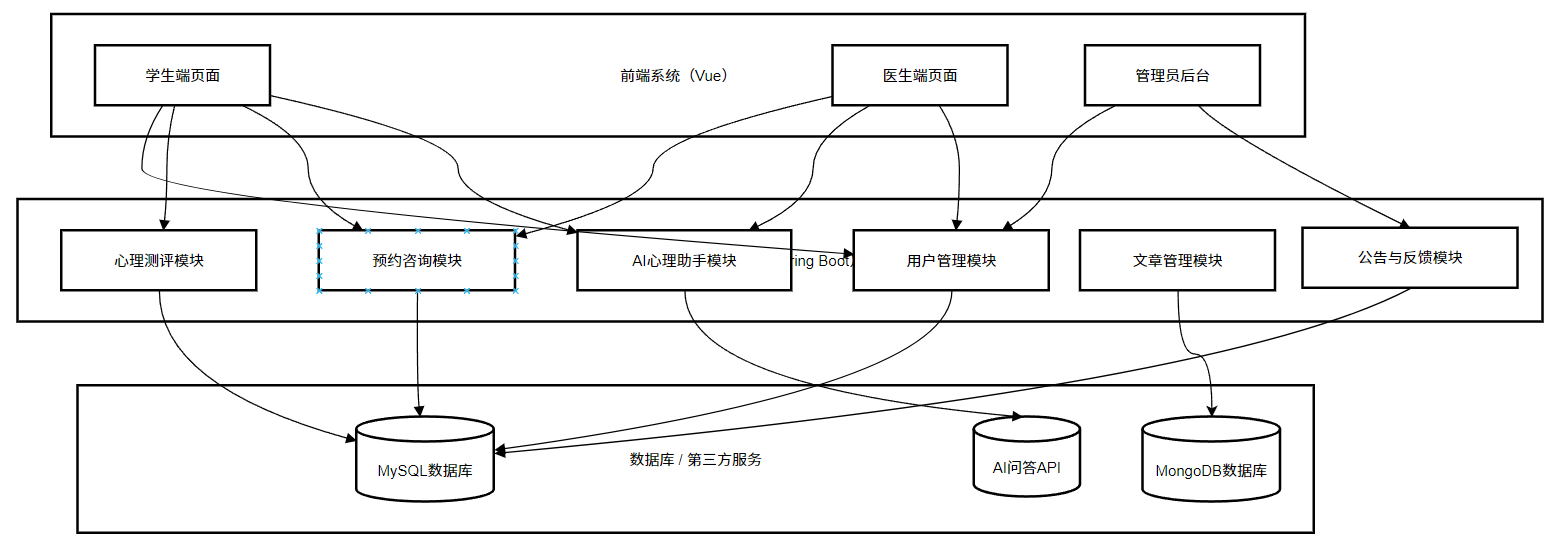


图3-1 系统整体架构图

3.1.2 系统模块划分

根据需求分析，系统可以划分为以下几个核心模块：

用户管理模块：负责系统用户（管理员、心理医生、学生）的注册、登录、信息管理和权限控制。

内容管理模块：包括心理健康文章的发布、审核和管理功能。

在线咨询模块：提供学生与心理医生之间在线咨询、预约、记录管理等功能。

心理测评模块：包括心理测评的创建、发布、结果分析等功能。

AI智能客服模块：提供基础的回答和建议，能够自动回复并进行情感分析，帮助学生快速了解到相关的心理知识。

数据统计与分析模块：对系统使用情况、用户行为进行数据分析，帮助管理员更好的分析用户、系统。

每个模块之间通过相互独立，分别开发，确保各个模块之间的高内聚和低耦合，便于维护和扩展。

3.2 技术选型

3.2.1 前端技术选型

本系统前端部分采用了JavaScript与Vue.js作为主要开发工具，其中 Vue.js 作为轻量级渐进式框架，具备良好的可扩展性与开发友好性，已被广泛应用于现代Web项目中。基于本系统对用户交互性、响应速度及界面可维护性的需求，Vue 的特性能够充分满足开发目标。

Vue最为突出的优势之一在于其响应式的数据绑定能力。该机制允许开发者在数据状态变更时，页面内容能够自动同步更新，避免了传统DOM操作的繁琐步骤。这一特点在实现心理测评答题、在线聊天咨询等高频率交互场景时，能够显著提升用户体验与界面实时性。

此外，Vue 鼓励组件化开发思维。通过将系统中的各功能模块如用户登录、心理咨询卡片、量表答题区、文章浏览页等拆分为独立的组件，能够实现数据、样式与业务逻辑的局部管理，进而增强代码复用性、可读性与协作效率。这种结构对于支持团队协同开发具有显著优势。

在生态系统方面，Vue拥有完善的配套工具与扩展库。配合Vue Router实现前端路由管理，可以支持系统页面间的无刷新跳转，提升用户操作的连贯性；使用 Vuex 进行应用级状态管理，有助于维护如用户登录信息、会话记录等共享数据的一致性，简化组件间的数据交互流程。此外，通过Vue CLI工具可以统一项目的构建流程，提升开发与部署的标准化程度。

页面设计上引入了Element UI作为主要的前端UI组件库。该框架提供了丰富且易用的界面组件，如表单、弹窗、标签页、分页器等，开发者可直接调用现成组件以快速构建页面，避免重复造轮子，提升开发效率。例如在预约功能模块中，结合使用日期选择器组件与时间段选择器可以实现医生排班信息的可视化与交互操作。同时，Element UI提供的响应式设计能力确保系统在多种屏幕设备上均能良好展现，满足本系统对跨终端兼容性的基本需求。

3.2.2 后端技术选型

本系统后端采用了目前广泛应用于企业级开发的Spring Boot框架。作为 Spring 全家桶的重要一员，Spring Boot在继承Spring原有强大功能的基础上，简化了配置过程，并通过集成化设计提升了开发效率，特别适合构建结构复杂、逻辑严密的Web应用。

Spring Boot的“约定优于配置”理念是其提升开发效率的核心优势之一。传统开发中大量XML配置被注解与自动装配机制所替代，使得功能模块的集成变得更加快捷。对于本项目这种模块多、开发周期紧的系统而言，Spring Boot提供了极大的便利。

在功能集成方面，Spring Boot可以轻松整合包括 Spring MVC、Spring Data JPA、Spring AOP以及Spring Security等在内的多种核心模块。系统可借助这些工具高效实现数据库操作、权限认证、日志记录与接口开发等常用功能，降低系统的耦合度，增强整体一致性与扩展性。

在安全性方面，系统引入了Spring Security框架用于处理身份认证与权限控制问题。同时，通过JWT（JSON Web Token）机制实现用户身份信息的加密传输和会话有效管理，有效提升系统的安全防护能力。该方式不仅简化了用户登录的流程，还可在保持无状态的前提下实现权限验证，提高了系统的健壮性和可伸缩性。

总体来看，Spring Boot为系统后端开发提供了稳定、高效且易于扩展的开发环境，是本项目实现复杂业务需求的理想选择。

3.2.3 数据库技术选型

本系统采用了MySQL和MongoDB数据库，主要基于以下考虑：

稳定性与性能：MySQL是开源的关系型数据库，广泛应用于各类系统，具有良好的稳定性和性能表现。

支持复杂查询：MySQL支持SQL标准语法，能够方便地进行复杂的查询操作，便于扩展。

灵活性：MongoDB 支持灵活的JSON格式数据，便于处理非结构化的数据。

可扩展性：MySQL能够支持高并发的请求，且具有良好的数据备份与恢复功能，能够满足系统未来的扩展需求。

为了保证数据的一致性与安全性，数据库设计中将采用适当的索引、事务控制等措施，确保系统在高并发情况下的稳定运行。

3.3 系统功能模块设计

在系统功能模块设计中包含管理员、心理医生以及学生的相关功能，也根据各角色的执行流程所设计出相关的功能模块。

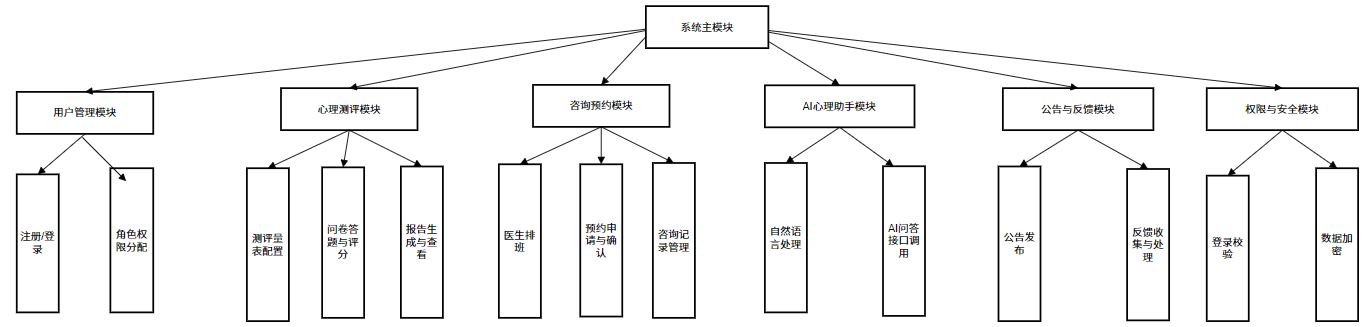


图3-2 系统功能模块图

3.3.1 用户管理模块

用户管理模块涵盖了用户注册登录、角色权限划分以及用户信息维护等核心功能，旨在提供安全高效的用户交互与权限控制机制。具体功能如下：

用户注册与登录：系统允许新用户通过用户名进行账号注册，注册信息需经管理员审核后方可激活账号以登录系统。在登录流程中，系统引入了 JWT（JSON Web Token）机制来生成身份认证令牌。该令牌中包含用户的基本身份信息，并具备有效时限，用于后续接口访问的身份验证，确保用户会话的安全性。

用户角色与权限管理：管理员可以对不同用户（学生、心理医生、管理员）进行角色分配，并设置不同的访问权限。每个角色拥有不同的操作权限，比如学生只能查看自己的个人信息，而医生则能查看学生信息，管理员拥有最高权限，可以管理系统的所有方面，确保系统的安全性。基于角色的访问控制（RBAC）被应用于整个系统，确保每个用户仅能访问与其角色相关的资源。这不仅提高了系统的安全性，也简化了权限管理。

用户信息管理：系统支持用户自主维护个人资料信息，如昵称、头像、联系方式等基本字段。此外，也允许用户拓展个性化信息，如兴趣标签、职业介绍等，以支持更丰富的用户展示与平台互动。

通过这样的设计，用户管理模块不仅能有效地支持系统的基本运作，同时也能为用户提供安全、便捷的服务体验。

对于不同角色的权限控制，采用RBAC的模型，如图3-3所示：

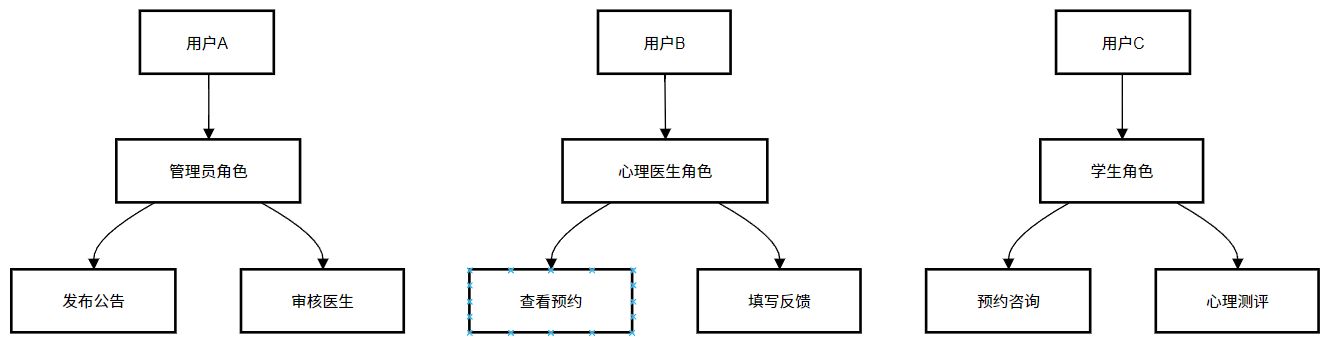


图3-3 权限控制图

3.3.2 内容管理模块

内容管理模块是心理健康教育平台的核心组件之一，它不仅支持内容的创建和发布，还确保了信息的有效传播和用户友好性。

内容管理模块主要包括心理健康文章、课程和视频的发布与管理功能。具体功能如下：

内容发布与审批：心理健康专家或医生可以利用平台提供的工具轻松地撰写文章、录制视频或设计课程。这些内容旨在提供科学的心理健康知识，帮助用户理解和应对各种心理问题。医生发布心理健康相关的内容，管理员负责审核这些内容的合规性与科学性，一旦内容被批准并发布，系统应提供一个渠道让读者能够对内容提出反馈或建议，以便于后续改进。

内容分类与标签：每篇文章、课程和视频可以根据内容主题进行分类，根据心理健康的不同领域（如情绪管理、压力应对、人际关系等），每项内容都应归类到相应的类别中。这有助于学生快速定位他们感兴趣的主题。除了基本分类外，还可以为每篇文章、视频或课程添加多个标签，以反映其特点或涵盖的具体话题。例如，“焦虑缓解”、“儿童心理学”、“职场压力”等标签可以帮助学生更精确地找到所需资源。

内容展示与搜索：学生可以通过关键词、标签或分类进行内容搜索，系统根据用户的行为和兴趣进行个性化推荐。

通过上述措施，内容管理模块不仅促进了高质量心理健康教育资源的共享，也为学生提供了便捷且个性化的学习途径。这样的设计有助于增强学生的学习兴趣和效果，同时也维护了平台的专业形象和权威性。

3.3.3 在线咨询模块

内容管理模块作为校园心理健康平台的重要组成部分，承担着知识传播与资源管理的双重职责。该模块不仅支持心理健康内容的创建、组织与展示，还在确保内容科学性与用户可达性方面发挥着关键作用。主要包括以下功能：

预约咨询：学生可以通过系统查看各个心理医生的空闲时间，每个心理医生的时间排班都清晰地标记出来，包括工作日、周末以及特定时段是否可用等信息，确保双方能够在合适的时间进行咨询[7]。

在线聊天与视频咨询：提供文字聊天和视频咨询功能，学生与心理医生可以实时进行在线一对一的心理健康咨询。文字聊天适合于那些偏好书面表达或者网络条件有限的用户；而视频咨询则更适合需要面对面交流的情况，能够更好地进行对话，增强沟通效果。

咨询记录管理：每次咨询结束后，系统将自动保存所有的聊天记录和视频内容（如果适用）。这些记录不仅有助于心理医生看到之前的咨询过程，也便于学生随时查阅，了解自己的进展。考虑到咨询内容的高度隐私性，所有记录都将被加密保存，并且只有该次咨询的心理医生和学生才能访问。此外，用户还可以设置额外的访问权限，比如限制某些敏感记录的查看期限等。用户在每次咨询后可以提供反馈，评价此次咨询的效果以及对该心理医生服务的满意度。这些建议可以帮助心理医生不断改进服务质量。

通过上述设计，在线咨询模块既为用户提供了一个安全、便捷的心理健康咨询环境，也促进了学生和心理医生之间的有效沟通和信任建立。这种全方位的支持体系有助于提升用户的整体体验，使他们能够更加积极主动地寻求帮助并改善心理健康状况。

3.3.4 心理测评模块

心理测评模块是心理健康教育平台的重要组成部分，这不仅帮助学生更好地了解自己的心理状态，还提供了科学依据来帮助进一步的心理咨询或治疗。心理测评模块能够为学生提供心理状态的自我评估，具体功能如下：

测评设计：医生和心理学专家可以根据不同的心理健康维度（比如抑郁、焦虑、压力管理等）设计测评问卷。这些问题基于科学的理论依据和社会实例，确保其科学性和有效性。测评主要包含选择题和情景题等，能很好地评估学生的心理状况。此外，测评题目还可以根据需要设置开放式问题，让学生能够自由表达他们的情感和想法。考虑到心理健康领域的不断发展，系统应支持定期的更新测评题目和内容，包括添加新的问题或删除过时的问题，确保测评工具始终处于最佳状态[8]。

测评发布与参与：心理医生设计好测评后，需经管理员审核。这一过程旨在保证所有发布的测评都符合规范标准，并且具有高质量的内容。管理员审核通过后，心理测评会发布到平台，学生可以进行测评。

结果反馈：完成测评后，系统会自动生成一份详细的评估报告。这份报告不仅包含测评得分，还会根据学生的回答提供个性化的分析和建议。例如，对于较为焦虑的学生，报告中可能会包含放松技巧和寻求专业帮助的信息。除了直接的心理健康建议外，系统还会为其推荐相关的心理健康的文章、课程或视频资源，来帮助学生深入学习和自我提升。这有助于形成一个完整的支持体系，帮助学生从多个角度改善他的心理健康。系统根据学生的测评结果提供相关的心理健康建议，帮助学生了解自己的心理状态。

通过这些功能，心理测评模块不仅为学生提供了自我评估的机会，也促进了积极主动的心理健康管理。这种全方位的支持不仅增强了用户的自我认知，也为后续的专业干预奠定了基础。

3.3.5 AI智能客服模块

AI智能客服模块在心理健康教育平台中扮演着创新辅助角色，不仅可以快速响应用户的日常咨询，还通过智能技术为用户提供初步的情绪关怀与陪伴。具体功能如下：

自动回复：智能客服系统基于预设的知识体系，能够高效地回应学生在使用过程中的常见问题，例如关于心理健康常识、平台功能操作说明等内容。这一知识体系囊括了心理领域的权威信息及平台相关的问答资料，确保用户在遇到疑问时能够迅速获得清晰、可靠的回应。为了适应不断更新的心理健康研究成果与技术发展，该知识库需定期维护与扩充，以保障信息的时效性与准确性。

情绪分析：AI客服具备情绪识别能力，能够从学生的文字输入中分析出潜在的情感倾向，如忧虑、喜悦、愤怒或孤独等情绪状态。当系统检测到可能存在的不良情绪信号时，会主动推送合适的心理建议，并以共情的语调进行回应，以传递安慰与理解。例如，在识别出用户可能存在焦虑或抑郁迹象时，系统会推荐其联系专业心理服务，并提供相关机构或医生的链接信息。通过对情绪的动态感知，AI客服能够灵活调整沟通方式，使用户在交流中感受到温暖和安全。

智能学习：智能客服还具备持续学习与自我完善的能力。它可以根据每次用户交互的数据记录，分析问题解答的满意度，并据此优化后续的响应策略。例如，对于用户未能完全接受或理解的回答，系统会将其作为改进样本，在后续版本中提升对类似问题的处理能力。此类动态学习机制确保客服系统在使用过程中不断进化，逐步提高其服务水平与人机交互体验。

通过上述功能设计，AI智能客服不仅承担了系统中的信息查询角色，还能在用户心理状态波动时提供初步的情感抚慰，构建起学生与平台之间的第一道心理支持屏障，为问题的早期干预创造了条件。其持续学习的能力也确保了服务质量能够不断迭代与进步，始终保持与现实需求的同步。

3.4 数据库设计

数据库设计是系统实现过程中至关重要的一部分。通过对数据库合理设计，可以确保数据的完整性、一致性和可访问性。系统的数据库设计也是基于功能模块来划分，主要包括了用户管理、内容管理、在线咨询、心理测评等模块的相关数据表。设计过程中，也将各个模块之间的联系和数据的冗余度最小化，能最大化满足系统实现。

在设计数据库时，特别注意了以下几点：

数据的规范化：设计过程中避免了数据冗余，保证数据库的规范化。每个功能模块都独立设计相关表结构，并通过外键或者多表之间通过额外的字段实现关联查询。

表结构的合理性：根据系统的具体需求，确定了每个表的字段和索引设计，保证数据在查询时具备高效性。

数据安全性和一致性：设计了事务管理机制，确保在并发的情况下也能保证数据的一致性和安全性。

3.4.1 E-R图设计

系统的整体E-R 图3-4所示：

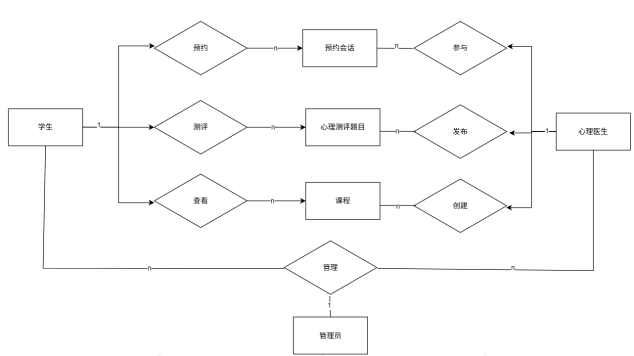


图3-4 整体E-R图

在设计校园心理健康网站时，E-R图的设计理念围绕管理员、学生和心理医生这三个核心实体展开。首先，管理员实体负责整个网站的运营与维护工作，包括用户管理和内容审核等重要职责。该实体的主要属性有管理员ID作为主键用于唯一标识每一位管理员，用户名和密码确保安全登录，以及联系方式便于内部沟通协调。

对于系统的使用者进行统一的管理，通过用户实体对整个系统的所以角色进行统一处理。

用户实体以及用户详情属性图如图3-5所示：

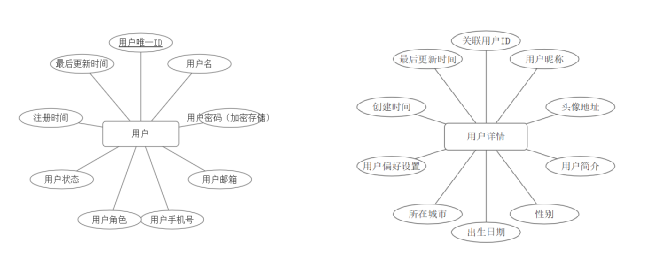


图3-5 用户实体以及用户详情E-R图

学生或称作用户实体，是那些寻求心理健康知识、资源或直接咨询服务的目标群体。每个学生由用户ID这一主键来区分，同时记录学生姓名、年龄、性别等基本信息，比如注册日期记录用户使用的开始时间。通过这个模块，学生可以方便地获取心理健康资料，并预约专业咨询[9]。

心理医生实体代表的是提供专业心理咨询的人员。每位心理医生都有一个唯一的心理医生ID，除此之外，还包括姓名、专业领域以及资格证书编号等相关信息，可以证明他的专业能力以及服务范围。心理医生不仅能够直接为学生提供咨询服务，还能编写一些文章课程或测评供管理员审批后发布，进一步丰富平台的内容资源。这些关系共同构成了校园心理健康网站的复杂而有序的数据模型，保证了系统的高效运行和服务质量。通过合理设计的E-R图，可以清晰地展示出各实体及其属性，以及它们之间复杂的关系，在数据库层面有不错的设计。心理医生实体图如图3-6所示：

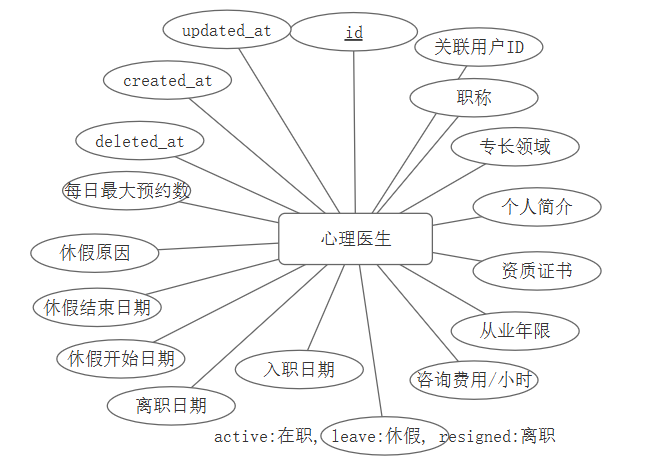


图3-6 心理医生E-R图

心理医生和学生患者的预约流程通过一系列紧密关联的实体来实现高效管理。核心实体包括“用户”、“心理医生”、“预约”及“时间表”。其中，“用户”实体存储了学生的基本信息；“心理医生”实体则详细记录了医生的专业背景、联系方式及其可预约的时间段。“时间表”实体具体划分了每位医生的工作时间和空闲时段，并与“医生”实体相关联，以动态展示其可用性。当进行预约操作时，“预约”实体被创建，它不仅包含了学生所预约的具体时间、选择的心理医生等信息，还记录了咨询需求概要。此外，“预约”实体与“用户”和“医生”实体之间建立了多对一的关系，确保每个预约能及时地关联到相应的心理医生和学生。预约实体图如图3-7所示：

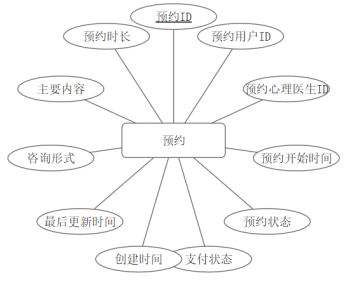


图3-7 预约E-R图

在设计心理健康测评模块的实体结构时，系统主要围绕“测评项目”、“测评题目”、“用户答题记录”以及“测评反馈”四个核心要素展开，目的是构建一个科学、全面且高效的心理评估流程。具体来说，“测评项目”实体用于承载测评工具的基本资料，包括其名称、功能介绍以及适用人群等内容；该实体与“测评题目”建立关联，后者详细列出了每项测评所包含的问题及其备选答案，确保测评在内容上具有专业基础和实用价值。

当学生或用户参与测评时，其作答行为将被记录至“用户答题记录”实体中，涵盖了用户选择的答案、答题时长、测评时间等多个维度的数据，用以支撑后续的数据分析与个体心理状态追踪。在完成答题后，系统会依据用户的作答情况生成“测评反馈”，即评估报告，该结果不仅呈现用户在心理测评中的量化分值，还结合其表现提供有针对性的建议与心理资源推荐，帮助用户理解自身状态并获得进一步的帮助。

测评及测评结果实体图如图3-8所示：

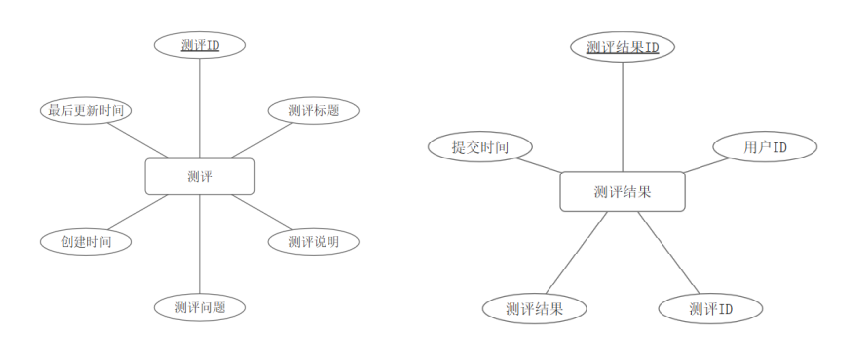


图3-8 测评及测评结果E-R图

在咨询模块的实体结构设计中，关键实体包括“用户”、“医生”、“预约”、“咨询记录”和“反馈”，这些实体共同协作以支持整个咨询流程。具体来说，“用户”实体存储了学生患者的基本信息和个人资料，而“医生”实体则包含了医生的专业背景、联系方式及其可预约的时间段；“预约”实体用于记录用户的预约请求详情，如选定的医生、预约时间及简要咨询目的，并与前两个实体相互关联，确保每位预约都能准确对应到具体的用户和医生；当咨询完成后，“咨询记录”实体详细保存了每次咨询的内容摘要、使用的沟通方式（文字聊天或视频通话）等重要信息，以便后续查阅和跟进；此外，“反馈”实体允许用户对咨询服务进行评价和建议，为服务质量改进提供依据，所有这些实体通过相互间的逻辑关系紧密连接，形成一个完整的数据模型，支撑起从预约到咨询再到反馈的全流程管理，保障了咨询服务的专业性和高效性。

咨询实体图如图3-9所示：

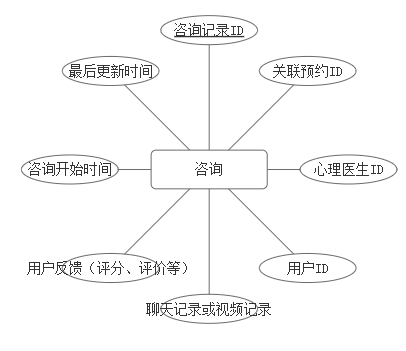


图3-9 咨询E-R图

文章以及内容实体图如图3-10所示：

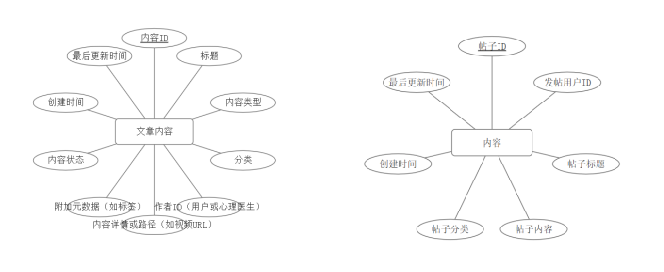


图3-10 文章以及内容E-R图

在系统日志模块的实体结构设计中，核心实体为“日志记录”，该实体围绕平台的操作监控与安全审计功能展开，致力于实现操作全过程的可追溯与数据完整性保障。具体来看，“日志记录”用于详细存储平台上所有关键行为的相关信息，包括操作发生的时间、事件类型（如用户登录、退出、信息更改、内容提交等）、发起操作的用户标识ID以及对应的访问来源IP地址。

对于牵涉到用户敏感数据的操作行为，日志系统会额外保存操作参数及其执行结果，从而为后续可能发生的安全检查或故障诊断提供详实的数据依据。为了增强日志信息的检索效率与数据利用价值，系统引入了日志分类和标记机制，使得管理员可按需通过时间段、用户身份、操作类别等多维度条件进行精确筛选与查询。

此类结构设计不仅有效提升了系统运行过程的可视化程度，还加强了安全威胁的实时识别与响应能力，为保障平台的稳定运行与数据安全提供了坚实的技术支撑。系统日志实体图如图3-11所示：

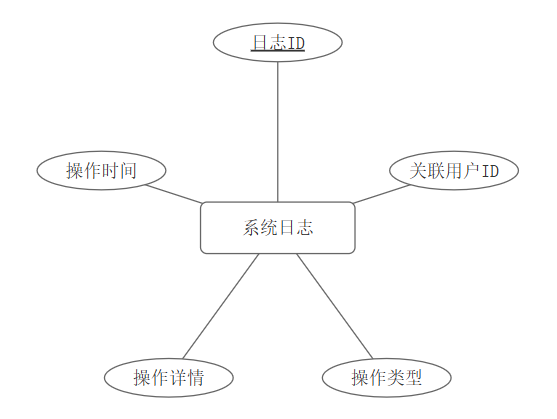


图3-11 系统日志E-R图

3.4.2 数据库表设计

在构建校园心理健康平台的数据库结构时，需要综合考虑系统整体的业务流程与各个功能模块之间的内在联系。管理员相关数据表主要围绕其管理职责进行设计，应包含唯一标识字段（如管理员ID）以及基本属性字段，例如账号名、加密后的密码和联系方式，以支撑其对用户、医生及内容的统一管理操作。学生数据表用于记录学生用户的基本资料，包括姓名、性别、年龄、注册时间等，同时确保使用唯一主键实现用户信息的独立性和可溯源性。

对于心理医生表，设计重点突出其专业资质，除了基本身份信息外，还应包含执业方向、职称资质编号等字段，方便用户根据个人需求进行医生选择与匹配。心理健康资源表（如文章、视频和课程）用于存储医生发布的内容，其字段应涵盖标题、内容、发布时间及审核状态，同时通过外键与医生表进行关联。

咨询预约表在结构上起到了连接学生和心理医生的桥梁作用，需详细记录预约时间、会谈状态等信息，并通过外键链接到学生与医生记录，以实现对预约关系的精确管理。AI客服交互记录表则承担着保存学生与智能客服之间沟通内容的功能，结构中应包括消息内容、交互时间等字段，并关联到具体学生用户，以保障记录的完整性与后续分析的可行性。

整体数据库结构设计遵循规范化原则，合理设定主键与外键以减少冗余数据，从而提升数据的逻辑一致性和完整性。此外，系统架构在设计阶段预留了扩展能力，以便未来增加新功能或调整现有数据模型。同时，针对密码等敏感信息，采用加密方式进行存储处理，并基于用户角色设置访问权限，强化数据安全。

如表3-1至3-5为校园心理健康平台的相关表结构：

表3-1 用户信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 类型 | 长度 | 字段说明 | 约束 |
| id | bigint |  | 用户id | 主键 |
| username | varchar | 50 | 用户名 | 唯一索引 |
| password | varchar | 100 | 密码 |  |
| email | varchar | 100 | 邮箱 |  |
| phone | varchar | 15 | 手机号 |  |
| role | enum |  | 角色 |  |
| status | enum |  | 状态 |  |
| created\_at | datetime |  | 创建时间 |  |
| updated\_at | datetime |  | 更新时间 |  |

表3-2 用户详情表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 类型 | 长度 | 字段说明 | 约束 |
| user\_id | bigint |  | 关联用户id |  |
| nickname | varchar | 50 | 用户昵称 |  |
| avatar\_url | varchar | 255 | 头像地址 |  |
| bio | text |  | 用户简介 |  |
| gender | enum |  | 性别 |  |
| birthdate | date |  | 生日 |  |
| location | varchar | 255 | 位置 |  |
| preferences | varchar | 255 | 爱好 |  |
| created\_at | datetime |  | 创建时间 |  |
| updated\_at | datetime |  | 更新时间 |  |

表3-3 预约表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 类型 | 长度 | 字段说明 | 约束 |
| id | bigint |  | 预约id | 主键 |
| user\_id | bigint |  | 用户id | 唯一索引 |
| doctor\_id | bigint |  | 心理医生id |  |
| schedule\_time | datetime |  | 预约时间 |  |
| status | enum |  | 状态 |  |
| form | enum |  | 预约形式 |  |
| content | varchar | 255 | 内容 |  |
| extra | varchar | 255 | 额外信息 |  |
| created\_at | datetime |  | 创建时间 |  |
| updated\_at | datetime |  | 更新时间 |  |

表3-4 内容信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 类型 | 长度 | 字段说明 | 约束 |
| id | bigint |  | 主键 | 主键 |
| title | varchar | 255 | 标题 |  |
| type | enum |  | 类型 |  |
| category | varchar | 255 | 分类 |  |
| content | varchar | 255 | 内容 |  |
| metadata | json |  | 附加数据 |  |
| status | varchar | enum | 状态 |  |
| created\_at | datetime |  | 创建时间 |  |
| updated\_at | datetime |  | 更新时间 |  |

表3-5 测评信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 类型 | 长度 | 字段说明 | 约束 |
| id | bigint |  | 测评id | 主键 |
| title | varchar | 255 | 标题 |  |
| description | varchar | 255 | 说明 |  |
| questions | json |  | 问题 |  |
| created\_at | datetime |  | 创建时间 |  |
| updated\_at | datetime |  | 更新时间 |  |

3.5 部署结构设计

对于系统的部署，根据前后端分离的特点分别进行部署，如图3-12所示：

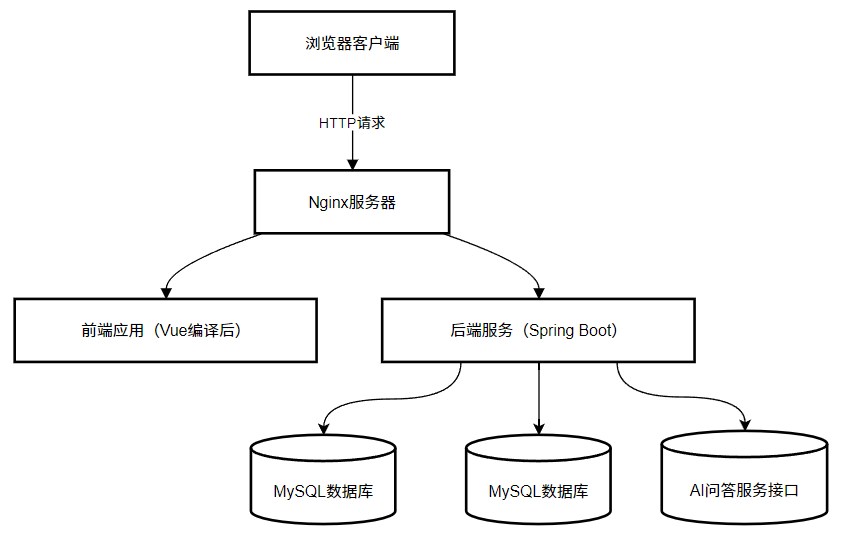


图3-12 部署结构图

3.6 总结

本章从系统架构、技术选型、模块设计等方面进行了系统的总体设计。通过前后端分离架构、Spring Boot框架、MySQL和MongoDB数据库等技术的合理组合，本系统能够高效地满足不同用户的需求。同时，功能模块的设计确保了系统的全面性与可操作性，为后续的详细设计与开发提供了坚实的基础。

4 系统详细设计

在系统的总体设计阶段，我们已经完成了系统的功能需求分析和架构设计。为了确保系统能够高效实现其功能，系统的详细设计必须考虑到如何实现这些功能，并通过模块化的设计保证系统的高效性、可维护性和扩展性。本章将从数据库设计、模块实现、类设计等角度进行详细讨论。

4.1 系统架构设计

系统架构采用前后端分离的方式，前端使用Vue.js框架，后端使用Spring Boot框架进行开发。前后端分离架构能够提高开发效率，简化前后端的协作。前端通过HTTP请求与后端进行数据交互，后端则负责处理业务逻辑和数据存储。

在实现过程中，根据在执行流程中的交互进行设计，如图4-1前后端交互图：

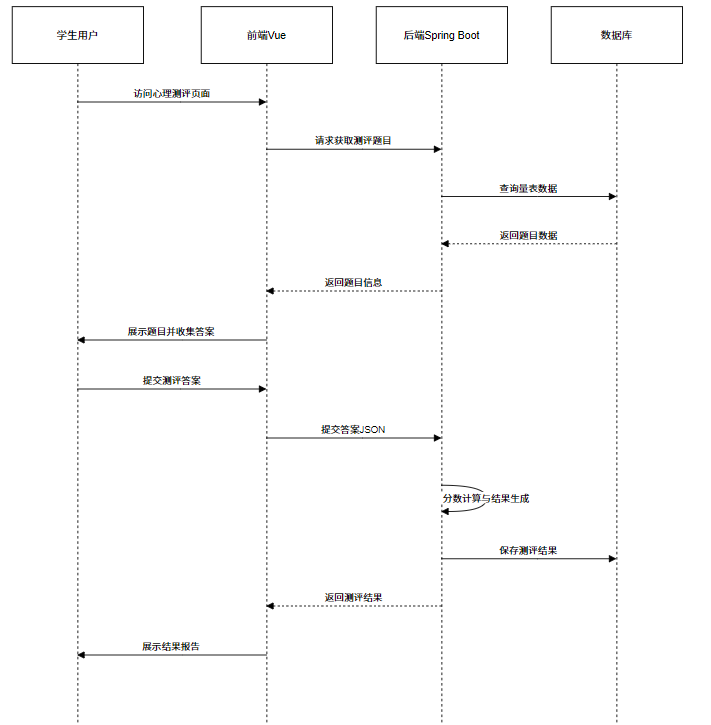


图4-1 前后端交互流程图

系统的后端采用Spring Boot进行开发，提供RESTful风格的API接口，确保系统能够灵活扩展和维护。前端通过调用后端API接口获取数据，展示给用户。

如图4-2是API设计文档（接口调用关系）设计：

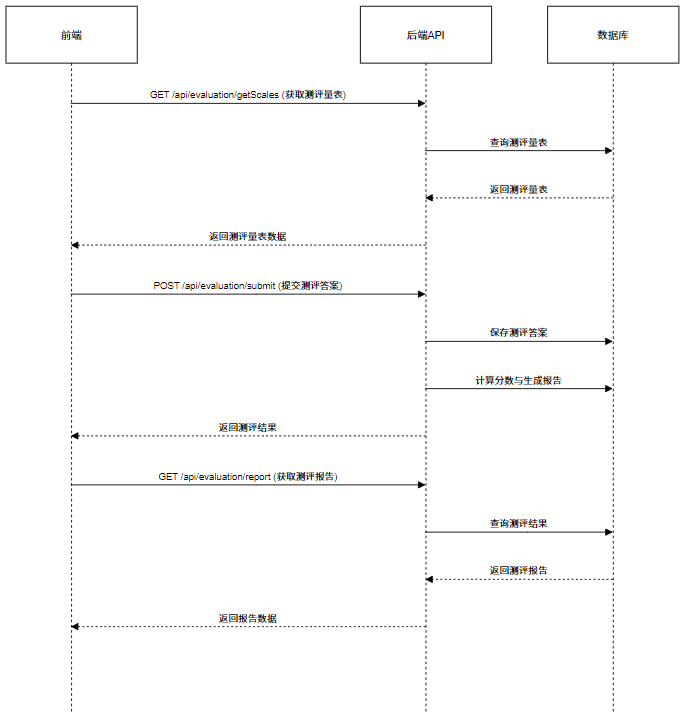


图4-2 调用关系图

4.2 模块设计

4.2.1 用户管理模块

用户管理模块是系统运行的基础，其功能涵盖用户的注册、登录及访问权限的管理。为提升系统的安全防护能力，本模块引入了JWT（JSON Web Token）作为身份认证手段。该机制在用户成功登录后生成一组具有时效性的令牌，用于后续的身份识别与接口访问，有效防止未授权操作的发生。

系统支持三种主要用户角色的配置与管理，分别是学生、心理医生和系统管理员。管理员拥有最高权限，可对新注册用户的信息进行审核、确认其身份，并分配相应的系统角色。心理医生账户审核通过后，具备为学生提供咨询服务的功能，而学生用户则可以访问心理健康类的内容资源，并通过平台发起咨询请求，与医生进行在线沟通。

通过这种分级角色权限体系的设计，系统在保障用户数据安全的同时，也提升了资源分配的效率和操作的规范性，为平台的稳定运行奠定了良好的基础。

4.2.2 内容管理模块

内容管理模块允许医生发布心理健康相关的资源，包括文章、视频和课程等。所有内容在发布前需要经过管理员的审核，确保内容的合规性与科学性。管理员可以对不符合规范的内容进行驳回或修改。这一模块的设计确保了平台内容的高质量和可信度[10]。

4.2.3 在线咨询模块

该模块是本系统中最为关键的功能之一，为学生与心理医生之间的沟通搭建了桥梁。学生可以通过提前预约或实时发起对话的方式，进行心理咨询。平台支持文本消息与视频沟通两种形式，灵活满足不同用户的沟通偏好与心理支持需求。所有的咨询对话内容将自动记录并保存在系统中，以便于日后的复查或管理。

此外，为提升咨询效率，系统集成了一个智能匹配机制，可依据学生提交的心理问题类型，自动推荐最适合的心理医生，提升用户体验。

4.2.4 心理测评模块

心理测评功能支持心理医生在平台上发布多样化的评估问卷，帮助学生完成自我心理状态的科学评估。学生可直接参与各类测评，涵盖情绪波动、压力水平、抑郁倾向等多个心理维度。系统在用户提交答卷后会自动生成测评报告，报告中不仅包括量化数据，还结合结果给出针对性的建议和后续指导。

4.2.5 AI智能客服模块

本模块基于先进的大语言模型技术构建，具备自然语言理解与生成能力，能够为学生提供7×24小时的心理健康相关信息服务。当学生与AI进行对话时，系统不仅能解答常见心理问题，还会结合情绪识别算法，感知学生输入文字中的情感倾向，从而提供个性化的回应策略。

4.3 安全性设计

系统的安全性是整个平台设计中的重要部分，确保数据安全和用户隐私不仅能够保护用户的权益，还能增强用户对系统的信任。为了确保系统的数据安全和用户隐私，系统采用了以下安全措施：

身份认证与授权：系统采用JWT进行用户身份认证，系统为每个成功登录的用户生成一个唯一的token。该token包含了加密后的用户信息，并在每次请求时携带token。这不仅简化了会话管理，还能实现跨域访问。保证每个用户在登录后都能够获得安全的令牌，并能够基于角色进行权限控制，这种基于角色的访问控制保证用户的安全使用。

数据加密：对用户的敏感信息（比如密码、个人信息等）进行了加密处理，避免数据泄露。为了防止用户敏感信息泄露，系统采用了加密算法对密码和个人信息进行加密。对于密码，通常采用基于哈希的加密算法，再加上随机的盐值进行处理，这样做的好处是即使数据库被攻破，攻击者也无法轻易获取原始密码。此外，对于特别敏感的操作（如修改密码），还会要求二次验证（比如短信验证码或邮件确认）以增加安全性。

输入验证：系统对所有来自用户的输入进行了严格的校验和过滤，避免像SQL注入、XSS（跨站脚本攻击）等常见的Web攻击手段。例如，在进行数据库查询之前，所有的输入参数都会经过预编译处理，杜绝输入对SQL造成影响；同时，输出到页面的内容也会经过HTML的编码，避免恶意脚本被执行。除了基本的输入验证外，系统还设置了合理范围的长度及格式限制，防止由于过长的输入导致数据溢出或其他潜在的安全漏洞。

日志审计：系统会记录用户的操作日志，方便管理员进行安全审计和问题排查。系统自动记录所有重要的用户操作行为，比如登录退出、信息修改、内容发布等。这些日志既有助于管理员监控系统的运行状态，也能在发生安全事件时用于追踪和分析。结合日志审计功能，系统可以设置异常行为检测规则，如多次失败的登录尝试或异常的数据访问模式等。当检测到可疑操作，系统将立即触发报警信息通知管理员，及时防护系统。

通过对身份认证、数据加密、输入验证以及日志审计等方面的深入设计和实施，系统建立了一个坚固的安全防护体系。这不仅有效保护了用户的隐私和数据安全，也为平台的稳定运行提供了坚实的基础。

4.4 总结

本章详细介绍了系统的数据库设计、模块设计、架构设计和安全性设计。

在数据库设计方面，结合校园心理健康服务的实际需求，构建了所需的数据表结构。通过规范化的设计原则，确保数据的一致性与完整性，同时优化查询效率，提高系统响应速度。

在模块设计上，系统采用模块化思想进行功能划分。各模块之间职责明确、耦合度低，便于独立开发与后期维护，提升了系统的可扩展性和灵活性。

在系统架构方面，采用了前后端分离的设计模式，增强了系统的可移植性和部署便捷性。

在安全性设计方面，本文充分考虑了用户隐私保护与数据访问控制的问题。系统引入JWT（JSON Web Token）进行身份认证，避免传统Session机制带来的服务器资源消耗；同时结合Spring Security框架实现细粒度的权限管理，保障不同角色用户的操作安全。此外，所有敏感数据在传输过程中均采用HTTPS协议加密，防止中间人攻击，进一步增强系统的安全性。

5 系统实现、测试和部署

在本章中，主要是详细介绍系统的实现过程，并对系统进行功能测试和性能测试，并完成项目的部署。通过系统的实现与测试，验证系统设计的合理性，并确保系统满足预定的功能需求。

5.1 系统实现

系统实现主要涉及前端、后端和数据库方面的开发工作。前端部分使用Vue.js框架，后端采用Spring Boot框架。系统通过前后端分离的架构进行开发，前端通过调用后端提供的RESTful API接口获取数据，后端的主要任务是处理业务逻辑、实现数据存取和用户请求的响应。为了确保系统的高效性和可维护性，后端采用了Spring Boot框架，并使用MySQL和MongoDB数据库进行数据存储。

在后端实现中，重点实现了以下功能：

用户管理功能：实现了用户的登录注册、角色管理、权限控制等功能。用户通过JWT进行身份认证，确保了系统的安全性。

内容管理功能：心理医生能够发布心理健康相关的文章和课程，管理员进行审核后发布。实现了文章的增删改查功能，并保证了内容的合规性。

在线咨询功能：学生可以通过平台发起与医生的在线咨询，医生能够提供心理咨询服务。我们通过WebSocket技术实现了即时消息通信，确保了咨询过程的流畅性。

心理测评功能：医生可以设计心理测评并提供给学生进行填写，学生提交测评后，系统自动生成测评报告。通过Spring Boot的异步任务机制实现了测评结果的生成和反馈。

AI智能客服功能：使用大语言模型训练和推理实现了AI智能客服，能够自动识别学生的咨询内容并提供相关帮助。AI客服系统通过集成第三方API接口和自定义算法，为学生提供情感支持和心理健康建议。

前端采用Vue.js框架，结合Element UI和Axios库，负责系统的界面展示和与后端的交互。前端实现的主要功能包括：

用户界面：用户能够根据其角色（学生、医生、管理员）访问不同的功能模块。学生能够浏览文章、进行测评、预约医生咨询等，医生能够查看学生的咨询记录并发布心理健康文章。

响应式设计：前端采用响应式布局，确保系统在不同设备上的显示效果一致。通过Flexbox和CSS Grid等技术，实现了良好的用户体验。

数据交互：前端通过Axios与后端进行数据交互，所有用户的操作数据都会通过API发送给后端处理。系统支持分页、筛选、搜索等功能，确保数据的高效展示。

交互效果：为了提升用户体验，前端设计了动态效果和提示信息。例如，学生在提交咨询请求时，系统会显示加载动画，并在成功提交后提示用户“咨询请求已成功提交”。

5.1.1 注册、登录实现

用户可输入注册信息及登录信息进行注册及登录。



图5-1 登录注册页面

在登录后，经过身份验证的用户将获得一个由系统生成的临时身份令牌（如JWT），用于维持会话状态。这个令牌会在每次请求时传递给服务器进行验证，以确认用户的身份和权限。令牌具有一定的有效期，过期后需要重新登录获取新的令牌。与此同时，系统还会记录此次登录行为，包括时间、IP地址等信息，以便于后续的日志审计和安全监控。

5.1.2 首页实现

当用户登录后，会进入首页，首页会包含服务项目、联系咨询，功能服务也会在首页一并展示，可以选择相应服务进行查看。



图5-2 首页

5.1.3 个人中心实现

用户根据自己的角色进入到个人中心中，查看自己的个人信息。在个人中心的设计中，核心围绕“用户信息管理”、“活动记录查看”以及“个性化设置”这几个方面展开。首先，“用户信息管理”部分允许用户查看和编辑自己的基本信息，如昵称、头像、联系方式等，并提供修改密码的功能以增强账户安全性。为了保护用户隐私，任何敏感信息的更新都需要经过身份验证，确保操作由用户本人执行。



图5-3 个人中心页面

其次，“活动记录查看”功能让用户能够追踪自己在平台上的所有活动，包括但不限于登录历史、咨询记录、测评结果和个人资料修改记录等。这些记录不仅帮助用户回顾其使用历程，还便于在需要时提供证据或参考。特别是对于心理健康平台而言，用户的咨询记录和个人成长轨迹是非常重要的资源，有助于持续的心理健康管理和治疗跟进。

5.1.4 预约、咨询管理实现



图5-4 预约和预约管理页面

在预约和咨询的设计中，核心围绕“用户”、“医生”、“预约”、“咨询记录”这几个关键实体展开，确保了从预约到实际咨询的整个过程既顺畅又安全。首先，“用户”实体存储了学生患者的基本信息和个人资料，而“医生”实体则详细记录了每位医生的专业背景、联系方式及其可预约的时间段。当用户希望进行心理咨询时，他们会访问“预约”实体，该实体不仅维护了所有可用预约的信息，还负责处理用户的预约请求，包括选择合适的医生及时间，并将这些预约信息关联至相应的用户和医生记录中。一旦预约成功，系统会发送确认通知给用户，并在预约时间接近时提供提醒服务。

到了预定的时间，用户和医生可以通过平台提供的在线聊天或视频通话功能开始咨询，这一过程由“咨询记录”实体管理，它详细保存了每次咨询的内容摘要、使用的沟通方式（如文字聊天或视频通话）等重要信息。此外，为了促进服务质量的持续改进，系统还允许用户对每次咨询服务进行反馈，这些反馈被收集并分析，为医生提供了宝贵的改进建议。整个预约和咨询流程通过这些实体间的紧密协作，构建了一个高效、安全且用户友好的服务体系，不仅增强了用户的体验，也提升了心理健康服务的质量和可达性。

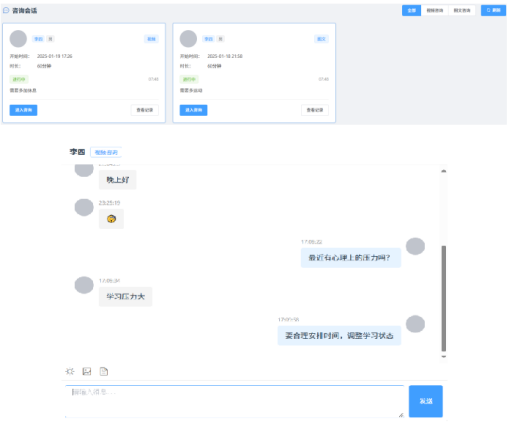


图5-5 预约会话和聊天页面

在预约会话与聊天的设计中，核心实体包括“用户”、“医生”、“预约”以及“聊天记录”，这些实体相互协作确保了从预约到实际咨询过程的顺畅性和安全性。首先，“用户”实体保存了学生患者的基本信息和个人资料，而“医生”实体则包含了医生的专业背景、联系方式及可预约的时间段。当用户希望进行心理咨询时，他们可以通过查看“预约”实体来选择合适的医生及其空闲时间，并提交预约请求。一旦预约被确认，系统不仅会发送通知给用户提醒其即将到来的咨询时间，还会为此次预约生成一个唯一的标识符，用于后续的所有相关操作。

整个预约会话与聊天流程通过上述实体间的紧密集成，构建了一个既保护用户隐私又能提供高质量心理健康服务的框架。这种设计不仅提升了用户体验，也增强了平台的专业性和可靠性，确保每一位寻求帮助的学生都能获得及时且适当的支持。

5.1.5 心理测评实现



图5-6 心理测评

当用户参与测评时，“用户测评记录”实体会实时记录用户的答题过程，包括所选答案、完成时间以及是否中途退出等细节。这不仅有助于分析用户的答题行为，还为后续生成个性化报告提供了数据支持。在用户完成测评后，系统基于其回答自动生成一份详细的“测评结果”，该结果通常包含量化得分、心理健康状况的简要分析以及针对性的建议。例如，对于表现出较高焦虑水平的用户，测评结果可能会推荐放松技巧或相关课程资源，帮助用户采取积极的应对措施。

为了进一步提升用户体验，心理测评模块还集成了智能化功能。例如，系统可以根据用户的历史测评记录进行纵向对比，帮助用户了解自己的心理变化趋势；同时，结合自然语言处理技术，对开放式问题的回答进行情感分析，从而提供更精准的反馈。此外，所有测评数据均经过加密存储，并严格限制访问权限，确保用户隐私得到充分保护。



图5-7 测评结果页面

审核过程中，审核人员需要对测评进行全面评估，包括检查问题的科学性、表述的清晰度、适用人群的准确性以及是否存在潜在的伦理风险（如歧视性语言或不当引导）。为了提高审核效率，系统可以提供辅助工具，例如关键词过滤功能或问题模板对比，帮助快速识别可能的问题区域。同时，“审核记录”实体会详细记录每次审核的操作过程，包括审核人、审核时间、意见及最终决定（通过或拒绝）。如果测评未通过审核，系统会自动生成反馈信息，明确指出需要改进的地方，并将其返回给创建者进行修改。

此外，为了保证审核流程的透明性和可追溯性，所有审核记录都会长期保存，并支持后续查询和审计。管理员还可以设置多级审核机制，例如初审和复审，以进一步提升测评内容的可靠性。对于某些特别重要的测评（如涉及高敏感度的心理健康领域），甚至可以引入外部专家评审，确保其科学性和权威性。

测评审核模块不仅实现了从提交到发布的全流程管理，还为平台提供了强有力的内容质量保障。这种严谨且高效的审核机制，不仅能够维护平台的专业形象，也能让用户更加信任测评结果，从而更好地服务于他们的心理健康需求。



图5-8 测评审核页面

5.1.6 课程文章实现

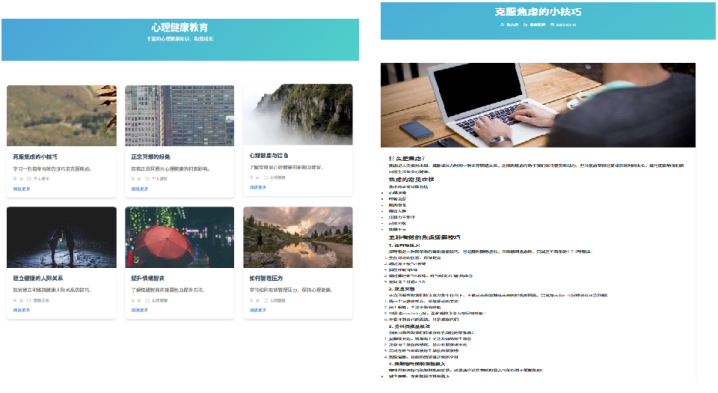


图5-9 文章页面

对于“内容”实体的实现，后端使用MySQL或MongoDB存储文章和课程的基本信息，包括标题、作者、发布时间、正文内容及多媒体资源链接。

“用户交互记录”部分通过JavaScript事件监听器捕获用户的操作行为，如点击收藏按钮、提交评论或评分等。这些行为会触发AJAX请求，将数据发送到后端进行存储。后端使用独立的表记录用户的浏览历史、收藏夹和互动数据，并定期清理过期记录以节省存储空间。同时，这些数据也为个性化推荐提供了基础。

“分类与标签”功能的实现依赖于数据库中的多对多关系表，每篇文章或课程可以关联多个分类和标签。前端通过筛选控件（如下拉菜单或标签云）让用户快速选择感兴趣的主题，而后端则根据用户的筛选条件动态生成SQL查询语句，返回符合条件的内容列表。为了优化查询效率，分类和标签的数据通常会被索引化处理。



图5-10 内容管理页面

5.2 系统测试

系统开发完成后，进行了全面的测试，确保系统的功能和性能达到预期目标。测试分为功能测试、性能测试和安全测试。

软件系统测试是确保应用程序在实际使用环境里能够符合预设的功能、性能及安全和可靠性标准，从而提供给用户高质量且稳固安全的体验的关键步骤。

除了核心的功能验证外，系统测试也涵盖了性能评估，即测试软件在面对高负载、大规模数据处理或复杂操作条件时的速度、资源消耗情况及稳定性。这样的测试确保了软件在实际部署中能持续高效运作，防止因性能不济造成的用户体验恶化或者系统崩溃的风险。

安全性审核同样是系统测试中的关键环节，它致力于找出软件系统中可能存在的安全隐患、隐私泄露风险或是抵御恶意侵袭的能力缺陷。借助模拟各种攻击情形和渗透实验，检测人员可以发现并解决潜在的安全威胁，保护用户信息及系统本身的防护能力。

此外，系统测试还涉及到对软件兼容性、用户体验友好度以及后期维护便利性的考量。例如，兼容性测试保证软件能在多种操作系统、网络浏览器及硬件设备上顺畅运行；用户体验测试专注于界面设计的人性化程度和使用的简便性；而可维护性测试则着眼于软件后续更新升级及故障排查的便捷性。

5.2.1 功能测试

功能测试主要针对系统的各项功能进行验证，确保每个功能模块的正常运行。具体测试内容包括：

用户管理模块：测试了用户的注册、登录、权限分配等功能，确保了不同角色的用户能够正确地访问相应的功能模块。

表5-1 用户登录测试用例表

|  |  |  |  |
| --- | --- | --- | --- |
| 编号 | 用例名 | 预期结果 | 实际结果 |
| CS001 | 输入正确账号和密码 | 登录成功，进入首页 | 登录成功，进入用户首页 |
| CS002 | 输入错误账号和正确密码 | 提示账号或密码错误，不跳转 | 提示账号或者密码错误，不跳转 |
| CS003 | 输入正确账号和错误密码 | 提示账号或密码错误，不跳转 | 提示账号或密码错误，  不跳转 |
| CS004 | 账号和密码均为空 | 提示账号或密码不能为空，不跳转 | 提示账号或密码不能为空，不跳转 |

内容管理模块：测试了文章发布、审核、删除等功能，确保管理员能够对医生发布的文章进行审核和管理。

表5-2 文章发布测试用例表

|  |  |  |  |
| --- | --- | --- | --- |
| 编号 | 用例名 | 预期结果 | 实际结果 |
| CS001 | 标题和内容为空 | 提示标题或内容为空，不能发布 | 发布失败 |
| CS002 | 标题为空，内容非空 | 提示标题为空，不能发布 | 发布失败 |
| CS003 | 标题非空，内容为空 | 提示内容为空，不能发布 | 发布失败 |
| CS004 | 标题与内容均合法 | 发布成功 | 发布成功 |

在线咨询模块：测试了学生与医生之间的实时通信功能，确保了WebSocket连接的稳定性，并验证了即时消息的正确传输。

表5-3 通信测试用例表

|  |  |  |  |
| --- | --- | --- | --- |
| 编号 | 用例名 | 预期结果 | 实际结果 |
| CS001 | 发送文本消息 | 消息发送成功 | 消息发送成功 |
| CS002 | 发送超长文本消息 | 提示不能发送 | 发送失败 |
| CS003 | 发送空消息 | 提示不能发送空消息 | 发送失败 |
| CS004 | 发送图片、文件 | 提示暂不支持 | 发送失败 |
| CS005 | 离线时接收消息 | 上线后可接收 | 消息发送成功 |

心理测评模块：测试了测评的创建、提交、结果生成等流程，确保了学生提交测评后能够快速生成评估报告。

表5-4 测评测试用例表

|  |  |  |  |
| --- | --- | --- | --- |
| 编号 | 用例名 | 预期结果 | 实际结果 |
| CS001 | 创建测评 | 创建成功 | 创建成功，刷新页面 |
| CS002 | 提交测评 | 提交成功 | 提交成功，展示结果 |
| CS003 | 创建空测评内容 | 提示测评内容不能为空 | 创建失败 |
| CS004 | 提交空测评 | 提示测评内容不能为空 | 提交失败 |
| CS005 | 数据准确性检查 | 得分与预期符合 | 得到正确结果 |
| CS006 | 数据不准确 | 得分与预期不相符 | 得到错误结果 |

5.2.2 性能测试

性能测试主要是评估系统在高并发情况下的响应时间和稳定性。为了模拟用户并发操作的情况，我们采用了JMeter工具进行了负载测试。

测试内容包括：

登录模块：测试了系统在多个用户同时登录的情况下，是否能够快速响应并正常处理请求。

在线咨询模块：模拟了多个学生与医生进行在线咨询的情况，测试了WebSocket通信的稳定性。

数据查询模块：测试了大量数据查询时的响应速度，确保了系统在大数据量情况下仍然能够保持较好的性能。

性能测试结果显示，系统在并发访问和高负载情况下，仍能够保持较好的响应速度和稳定性。

5.2.3 安全测试

安全测试主要是对系统的安全性进行评估，确保系统在面临潜在攻击时能够有效防范。主要测试内容包括：

SQL注入攻击：通过模拟SQL注入攻击，测试系统的输入验证机制，确保系统能够有效防止SQL注入。

跨站脚本攻击（XSS）：测试了系统的防止XSS攻击的能力，确保用户提交的内容不会被恶意脚本篡改。

数据加密与存储：测试了系统对用户敏感信息的加密存储方式，确保用户的密码和个人信息不会被泄露。

经过安全测试，系统能够有效抵御常见的攻击类型，具备较高的安全性。

5.3 测试结果分析

通过功能测试、性能测试和安全测试，系统整体表现良好，能够满足预定的功能需求，并在性能和安全性上达到一定的标准。特别是在高并发场景下，系统的响应速度仍然较快，保证了用户的良好体验。同时，系统的安全性得到了有效保障，可以防止常见的攻击类型。

5.4 系统部署

本项目采用 Docker 进行容器化部署，使用 Docker-compose 统一管理各镜像。其 Docker-compose.yml 为：

version: "3.9"  
services:  
 nginx:  
 image: nginx:latest  
 container\_name: mental-heal-nginx  
 restart: always  
 ports:  
 - '443:443'  
 - '80:80'  
 volumes:  
 - /data/mental-heal/web/nginx.conf:/etc/nginx/nginx.conf  
 - /data/mental-heal/web/conf.d:/etc/nginx/conf.d  
 - /data/mental-heal/logs/nginx:/var/log/nginx  
 - /data/mental-heal/web/dist:/opt/web/dist  
 privileged: true  
 mental-heal:  
 image: pig4cloud/java:8-jre  
 container\_name: mental-heal  
 volumes:  
 - /data/mental-heal/images/mental-heal.jar:/opt/app/mental-heal.jar  
 - /data/mental-heal/logs/mental-heal:/opt/logs/mental-heal  
 entrypoint: bash -c 'sleep 30s;java -Xms1024m -Xmx2048m -Djava.security.egd=file:/dev/./urandom -jar /opt/app/mental-heal.jar'  
 restart: always  
 ports:  
 - '8090:8090'

Nginx.conf相关配置：

user root;  
worker\_processes auto;  
  
events {  
 worker\_connections 1024;  
}  
  
http {  
 include mime.types;  
 default\_type application/octet-stream;  
 sendfile on;  
 keepalive\_timeout 65;  
  
 server {  
 listen 80;  
 server\_name mental-heal;  
  
 location / {  
 root /opt/web/dist;  
 try\_files $uri $uri/ /index.html;  
 index index.html index.htm;  
 }  
  
 location /api/ {  
 proxy\_set\_header Host $http\_host;  
 proxy\_set\_header X-Real-IP $remote\_addr;  
 proxy\_set\_header REMOTE-HOST $remote\_addr;  
 proxy\_set\_header X-Forwarded-For $proxy\_add\_x\_forwarded\_for;  
 proxy\_pass http://172.24.54.8:8090/;  
 proxy\_http\_version 1.1;  
 proxy\_set\_header Upgrade $http\_upgrade;  
 proxy\_set\_header Connection "Upgrade";  
 }  
 error\_page 500 502 503 504 /50x.html;  
 location = /50x.html {  
 root html;  
 }  
  
 }  
}

项目的部署：

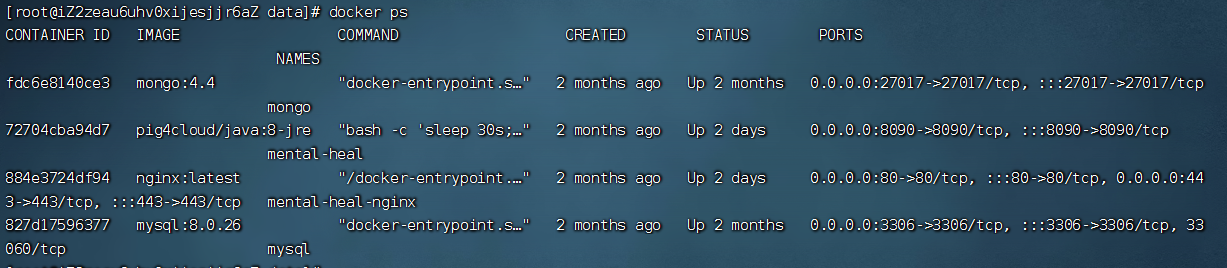


图5-11 部署图

5.5 总结

本章详细介绍了系统的实现过程，并对系统进行了功能、性能和安全方面的测试。测试结果表明，系统在各个方面都表现出较好的稳定性、性能和安全性，能够为用户提供稳定、可靠的心理健康服务。

6 结论与展望

6.1 结论

本文通过研究和设计，成功实现了一个基于Spring Boot的校园心理健康网站。该系统集成了用户管理、内容管理、在线咨询、心理测评、AI智能客服等多个功能模块，旨在为校园中的学生提供便捷、高效的心理健康服务。通过以下几方面的设计和实现，本研究取得了预期的成果：

系统架构与设计：本系统采用了前后端分离的架构，前端使用Vue.js框架，后端使用Spring Boot框架，数据存储采用MySQL和MongoDB数据库，系统具备良好的扩展性和高效性。该架构保证了系统的灵活性，使得前后端开发能够独立进行，减少了开发过程中的相互依赖。

功能模块实现：系统成功实现了多种功能模块，包括用户管理、内容管理、在线咨询、心理测评和AI智能客服。通过这些功能模块，学生能够便捷地寻求心理健康服务，医生能够提供专业的心理健康咨询和测评，管理员能够对系统进行有效管理和维护。

系统测试与验证：在开发完成后，我们进行了系统的功能测试、性能测试和安全测试。测试结果表明，系统能够在高并发场景下保持稳定运行，且能有效防止常见的安全攻击，确保了数据的安全性和用户的隐私保护。

AI智能客服应用：本系统通过集成大于语言模型实现了AI智能客服功能，能够根据学生的输入内容，提供实时的情感支持和心理健康建议。这一功能的实现大大提升了系统的服务效率，并为学生提供了即时的帮助。

综上所述，本文设计和实现的基于Spring Boot的校园心理健康网站具有较高的实际应用价值，能够有效满足学生心理健康需求，且具备较好的系统扩展性和安全性。

6.2 研究不足与局限性

尽管本系统已经具备了基本的功能和稳定的性能，但在实际应用过程中仍然存在一些不足之处，主要表现在以下几个方面：

AI智能客服的智能化程度：虽然AI智能客服能够提供一定的情感支持和心理健康建议，但其智能化程度仍有待提升。未来可以结合更为先进的机器学习算法和情感分析模型，以提升智能客服的服务质量和准确性。

心理测评的科学性：系统中提供的心理测评虽然能根据用户的答案生成一定的反馈，但其科学性和权威性需要进一步加强。目前的测评系统主要依赖于常见的心理学评估工具，未来可以与专业的心理学机构合作，设计更具科学性和针对性的测评问卷，并根据用户的反馈生成更为精准的心理健康报告。

用户数据隐私保护：虽然系统在设计时考虑了数据的加密和隐私保护，但在数据泄露和网络攻击愈加严重的背景下，如何进一步保障用户的隐私仍然是一个挑战。未来可以通过更为严格的安全防护措施，如多重身份验证、数据加密传输等，进一步提高系统的安全性。

系统扩展性和模块化：虽然当前系统已经具备了一定的可扩展性，但随着用户量的增加和功能需求的多样化，系统的扩展性可能会受到一定限制。未来可以采用微服务架构，使得每个模块可以独立部署和维护，提升系统的可扩展性。

6.3 未来展望

随着社会对心理健康问题的关注不断增加，特别是在校园环境中，心理健康问题逐渐成为影响学生成长和发展的重要因素。未来，基于Spring Boot的校园心理健康网站将继续在以下几个方向进行改进和发展：

智能化服务的提升：进一步提升AI智能客服的服务质量，使其能够更准确地识别用户情感，提供更加个性化和专业化的心理健康建议。

跨平台支持与多终端适配：为了满足更多用户的需求，未来可以考虑将系统进行跨平台扩展，支持手机端、PC端和其他智能设备的使用。同时，优化系统的响应式设计，使得用户能够在不同设备上获得流畅的使用体验。

心理健康数据分析：随着用户数据的积累，系统可以利用大数据分析技术，通过对大量用户数据的分析，能够更加精准地为学生提供个性化的心理健康服务，并为学校的心理健康管理提供决策支持。

6.4 总结

本文设计和实现的基于Spring Boot的校园心理健康网站为学生提供了一个便捷、高效、安全的心理健康服务平台。通过系统的功能设计、技术实现和全面的测试，验证了该平台的可行性和有效性。在未来的发展中，随着技术的不断进步，系统将继续完善和优化，为更多学生提供专业化、智能化的心理健康服务。

参考文献

1. 胡荣,羊雪玲.基于Spring Boot前后端分离Web系统的设计与实现[J].新能源与智能网联,2024,(01):88-97.
2. 徐智勇,戴祖旭.基于WebSocket协议的消息安全推送系统设计与实现[J].武汉工程大学学报,2024,46(03):310-316.DOI:10.19843/j.cnki.CN42-1779/TQ.202306003.
3. 王珊, 萨师煊. 数据库系统概论(第5版)[M]. 高等教育出版社, 2014.
4. 陈敬宗.计算机软件Java编程特点与技术应用分析[J].中国信息界,2025,(01):162-164.
5. 雷帅,李娟,陈亮,等.校园心理健康平台“知你”APP设计与实现[J].电脑编程技巧与维护,2021,(10):64-66.DOI:10.16184/j.cnki.comprg.2021.10.024.
6. 杨家炜.基于Spring Boot的web设计与实现[J].轻工科技,2016,32(07):86-89.
7. 王鹏,甘琳琳.基于微信平台的大学生校园心理健康服务实践探析——以社会心理服务平台为例[J].农村经济与科技,2020,31(08):280-281.
8. Kumar S , Sharma A , Mamun K , et al. A Deep Learning Approach for Motor Imagery EEG Signal Classification[C]. 2016 3rd Asia-Pacific World Congress on Computer Science and Engineering (APWC on CSE). IEEE, 2016.
9. MEHEDI MASUD, SULTAN ALJAHDALI. Concurrent execution of transactions in a peer-to-peer database network[J]. International journal of intelligent information and database systems: IJIIDS,2011,5(5):510-531.
10. Angelis D M ,Volpi L ,Giusino D , et al.Acceptability and Usability of a Digital Platform Promoting Mental Health at Work: A Qualitative Evaluation[J].International Journal of Human–Computer Interaction,2025,41(2):1246-1259.

致谢

走到论文写作的尾声，回首这段经历，心中充满了感激和温暖。这份毕业设计虽然署上了我的名字，但它的完成离不开许多人的帮助与支持。正是这些鼓励、指导与陪伴，让我顺利完成了这个关于“基于Spring Boot的校园心理健康网站的设计与实现”的项目，也让我在专业成长的路上走得更加坚定。

首先，衷心感谢我的导师高程昕。在整个毕业设计过程中，老师不仅在技术上给予我细致的指导，在思路与方法上也不断启发我。每一次讨论都让我受益匪浅，导师严谨认真的学术态度也深深影响着我。这段时间的交流与指导，是我大学阶段宝贵的学习财富。

其次，要特别感谢我的同学和朋友们。在论文写作过程中，你们的建议和反馈让我不断完善内容，在遇到难题和瓶颈时，你们总是在第一时间给予我支持和鼓励。有你们在身边，是我前行路上最强的后盾。

同时，也要感谢学校和学院提供的学习平台和丰富的资源支持。从课程体系到实验环境，再到论文撰写的各项服务保障，这些都为我的毕业设计提供了坚实的基础。在这四年里，我不仅掌握了专业技能，更收获了探索和解决问题的能力。

最后，最深的感激留给我的家人。一路走来，家人始终是我最坚强的依靠。无论是生活中的琐碎，还是学业上的挑战，他们给予我最大的包容和支持。正是这份默默的付出，让我能够安心专注地完成这份作品。

感谢所有在这段旅程中给予我帮助的人。这篇论文凝聚着你们的付出与心意。未来的路还很长，而我会带着这份感恩继续前行。

外文原文

Design and Implementation of a Campus Mental Health

Website Based on Spring Boot

Mental health has become a growing concern among college students due to academic stress, social pressure, and personal development issues. As digital solutions advance, developing an intelligent, responsive, and secure online mental health platform becomes increasingly essential. This paper proposes the design and implementation of a campus mental health website based on Spring Boot, focusing on providing psychological assistance, appointment scheduling, and AI-powered chat support for students. The system is structured into three main modules: administrator, counselor, and student, each with specific roles and responsibilities.

1、Introduction

With the increasing prevalence of psychological issues among college students, there is a significant need for accessible and confidential psychological services. Traditional offline counseling services face challenges such as limited resources, scheduling conflicts, and low engagement. Therefore, integrating web technologies into psychological care can improve accessibility and encourage timely help-seeking behaviors.

2、System Design Overview

The system is designed using the Spring Boot framework, which simplifies the development of enterprise-level applications and supports fast integration of services. The backend is built using Java, with MySQL as the database, and front-end development relies on HTML, CSS, and JavaScript (Vue.js). The system follows the MVC (Model-View-Controller) architecture to ensure separation of concerns and enhance maintainability.

3、Functional Modules

**3.1、Administrator Module:**

The administrator is responsible for user management, counselor approval, system settings, and data analysis. Features include user registration review, system log tracking, and content moderation.

**3.2、Counselor Module:**

Psychological counselors can log in, view and manage appointments, access students' case history, and provide feedback after sessions. A calendar-based schedule management tool is integrated for convenience.

**3.3、Student Module:**

Students can register and log in to the platform, submit appointment requests, chat with the AI assistant for initial support, and access mental health resources such as articles, self-assessment tools, and stress-relief exercises.

4、Technical Implementation

**4.1、Backend Development with Spring Boot:**

Spring Boot provides a robust and scalable environment for building microservice-based web applications. Its auto-configuration feature reduces boilerplate code, enabling developers to focus on business logic. RESTful APIs are used to handle communication between the frontend and backend, ensuring a smooth and responsive user experience.

**4.2、Database Design:**

The platform uses MySQL to manage relational data. Tables are created for users, appointments, messages, and articles. A relational schema ensures data consistency and integrity. Hibernate ORM is integrated with Spring Data JPA to simplify database operations.

**4.3、Security and Authentication:**

The system implements JWT (JSON Web Token) for stateless user authentication, enhancing performance and security. Role-based access control (RBAC) ensures that only authorized users can access specific resources. Passwords are securely stored using the BCrypt hashing algorithm.

**4.4、AI Chatbot Integration:**

To provide initial psychological support, an AI chatbot is embedded within the student module. This chatbot uses natural language processing (NLP) to understand and respond to students' inquiries. Although not a replacement for human counselors, it helps reduce the burden on professionals and provides immediate assistance.

5、User Interface and Experience Design

The frontend is developed using Vue.js for its reactive components and efficient data binding. The design prioritizes user-friendliness, with intuitive navigation, responsive layout, and a clean interface. Mobile compatibility is ensured through responsive web design techniques.

6、System Testing and Evaluation

**6.1、Functionality Testing:**

Each module is tested for its core functionalities, including login, appointment scheduling, message delivery, and chatbot interaction. Automated and manual tests are both employed.

**6.2、Usability Testing:**

Selected students and counselors were invited to use the system and provide feedback on the interface, performance, and overall experience. The majority of users reported high satisfaction and ease of use.

**6.3、Performance Evaluation:**

The platform demonstrates stable performance under concurrent access. Load testing using JMeter shows that the system can support over 500 simultaneous users without significant latency.

7、Advantages and Innovations

**7.1、Modular Architecture:**

By separating functionalities into independent modules (student, counselor, admin), the system achieves better scalability and maintainability. Each module can be updated or optimized without affecting the rest of the platform.

**7.2、AI-Assisted Support:**

The integration of an AI chatbot allows students to receive basic psychological support anytime, reducing the pressure on professional counselors and enhancing the user experience. The chatbot also helps collect preliminary user data for follow-up by human experts.

**7.3、Data Security and Privacy Protection:**

Given the sensitivity of mental health data, strict data protection measures are in place. Encrypted communication, role-based access control, and secure database storage help ensure user privacy and comply with data protection regulations.

**7.4、Educational Resource Hub:**

The system includes a curated section with psychological articles, stress-relief exercises, mental health awareness content, and FAQs. This enables students to proactively learn about mental wellness.

8、Future Development Plans

Integrating video conferencing for remote counseling.

Enhancing the AI chatbot with deep learning models for improved emotional understanding.

Expanding support for multilingual users.

9、Conclusion

This paper presents a practical and scalable solution for addressing the mental health needs of university students. By leveraging the Spring Boot framework, the system successfully integrates essential psychological services into an accessible online platform. The inclusion of AI, modular architecture, and security considerations ensures both functionality and user trust. With further development, this system has the potential to be deployed across multiple campuses, contributing to improved mental health support nationwide.

中文翻译

基于Spring Boot的校园心理健康网站的设计与实现

由于学业压力、社交焦虑和自我发展等因素，大学生心理健康问题日益受到关注。随着数字化解决方案的不断进步，开发一个智能、响应迅速且安全的在线心理健康平台变得尤为重要。本文提出了一个基于Spring Boot的校园心理健康网站的设计与实现方案，重点在于为学生提供心理援助、预约管理以及人工智能辅助的聊天服务。该系统主要由三个模块组成：管理员模块、心理医生模块和学生模块，各自承担不同的职责与功能。

1、引言

随着大学生心理问题的日益普遍，提供便捷、保密的心理服务显得尤为重要。传统线下心理咨询面临诸如资源有限、时间安排冲突以及参与度低等问题。因此，将现代Web技术引入心理服务领域，有助于提升服务的可达性，并鼓励学生及时寻求帮助。

2、系统设计概述

本系统基于Spring Boot框架开发，简化了企业级应用的构建过程，并支持服务的快速集成。后端使用Java编写，数据库采用MySQL，前端则结合HTML、CSS与JavaScript（Vue.js）实现。系统遵循MVC（模型-视图-控制器）架构，确保逻辑分离，提升可维护性与可扩展性。

3、功能模块

3.1、管理员模块：

管理员负责用户管理、心理医生审核、系统配置及数据分析。主要功能包括注册审核、系统日志追踪与内容审核等。

3.2、医生模块：

心理医生可以登录系统，查看并管理预约信息，访问学生历史记录，并在会谈后提交反馈报告。同时集成了基于日历的排班管理功能，方便医生进行时间安排。

3.3、学生模块：

学生可注册并登录平台，提交心理预约请求，使用AI智能助手进行初步沟通，并查阅各类心理健康资源，如文章、自测工具及减压练习等。

4、技术实现

4.1、使用Spring Boot开发后端：

Spring Boot为构建基于微服务的Web应用提供了稳定且可扩展的开发环境。它的自动配置特性减少了大量样板代码，使开发者可以更专注于业务逻辑。系统通过RESTful API实现前后端的数据交互，确保用户体验的流畅性与响应速度。

4.2、数据库设计：

平台采用MySQL来管理关系型数据，设计了用户、预约、消息与文章等数据表。通过规范的关系型结构确保数据一致性与完整性。数据库操作整合了Hibernate ORM与Spring Data JPA，大大简化了数据访问层的开发。

4.3、安全与身份验证：

系统采用JWT（JSON Web Token）进行无状态认证，既提升了安全性，也提高了系统性能。同时，基于角色的访问控制（RBAC）机制确保只有授权用户才能访问特定资源。用户密码通过BCrypt哈希算法进行加密存储，保障数据安全。

4.4、AI聊天助手集成：

为提供初步心理支持，系统在学生模块中嵌入了AI聊天助手。该助手利用自然语言处理（NLP）技术理解并响应学生的提问。虽然它无法替代专业心理咨询，但在缓解心理压力、提供即时帮助方面起到了积极作用，并能协助收集初步心理状态信息供医生参考。

5、用户界面与体验设计

前端采用Vue.js开发，其响应式组件与高效的数据绑定能力大大提升了用户操作的流畅性。界面设计注重简洁友好，导航清晰，布局自适应不同设备，确保在移动端和桌面端均具有良好体验。

6、系统测试与评估

6.1、功能测试：

对系统各模块进行了全面功能测试，包括登录、预约管理、消息发送与聊天助手等功能。结合自动化测试与手动测试手段，确保系统稳定运行。

6.2、可用性测试：

邀请部分学生与心理医生使用平台，并就界面、操作流程及响应速度等方面提出反馈。大多数用户表示界面直观，操作简单，整体体验良好。

6.3、性能评估：

平台在高并发情况下表现稳定。通过JMeter压力测试，系统在超过500个用户同时在线访问时仍保持响应良好，系统延迟控制在合理范围内。

7、系统优势与创新点

7.1、模块化架构设计：

通过将系统功能划分为独立的模块（学生模块、医生模块、管理员模块），实现了更强的可扩展性与可维护性。每个模块都可以在不影响整体系统的前提下进行更新和优化。

7.2、AI辅助支持：

系统集成了AI聊天助手，为学生提供全天候的心理支持服务。AI助手不仅减轻了专业心理医生的工作压力，也提升了学生获取心理帮助的便捷性。此外，助手还可收集学生的初步信息，便于后续人工介入。

7.3、数据安全与隐私保护：

鉴于心理健康数据的高度敏感性，系统采取了多重保护措施，包括数据传输加密、基于角色的权限控制及数据库加密存储等，确保用户隐私安全，并满足相关数据合规要求。

7.4、心理教育资源中心：

平台设置了独立的心理健康学习专区，提供精选文章、减压练习、心理健康宣传内容及常见问题解答，鼓励学生自主学习并提升心理素养。

8、未来发展方向

集成远程视频心理咨询功能；

利用深度学习优化AI助手的情绪识别与应答能力；

增加多语言支持，服务更多背景的用户。

9、结语

本文设计并实现了一种实用、可拓展的大学生心理健康服务平台，基于·框架，有效整合了心理援助、AI互动和资源推送等多项功能。系统在架构上保持清晰模块化，在功能上关注用户实际需求，在安全性上严格防护用户隐私。该平台在试点应用中取得了良好效果，未来有望在更广泛的高校推广应用，为大学生心理健康提供强有力的技术支持与保障。