文本

描述已自动生成

**本 科 毕 业 论 文（设 计）**

|  |  |
| --- | --- |
| 题 目 | 基于HarmonyOS的交友APP的设计与实现 |
| 学生姓名 | 张东庆 |
| 学 号 | 2123855 |
| 学 院 | 信息技术学院 |
| 专业班级 | 计科B21-5 |
| 指导教师 | 郑光远 |
| 交稿日期 |  |

教务处制

**上海建桥学院毕业论文（设计）学术诚信声明**

本人郑重声明：所呈交的毕业论文（设计），是本人在导师的指导下，独立进行研究工作所取得的成果。除文中已经注明引用的内容外，本毕业论文（设计）不含任何其他个人或集体已经发表或撰写过的作品或成果。对本文的研究做出重要贡献的个人和集体，均已在文中以明确方式标明。本人完全意识到本声明的法律结果由本人承担。

作者签名： 日期： 年 月 日

**上海建桥学院毕业论文（设计）版权使用授权书**

本毕业论文（设计）作者同意学校保留并向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅。本人授权上海建桥学院可以将本毕业论文（设计）的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存和汇编本毕业论文（设计）。

**保 密** □，在 年解密后适用本授权书。

本论文属于

**不保密** □。

（请在以上方框内打“**√**”，如作者未做出选择的情况下，按不保密处理。）

作者签名： 指导教师签名：

日期： 年 月 日 日期： 年 月 日

基于HarmonyOS的交友APP的设计与实现

摘 要

随着信息技术的快速发展，校园生活中出现了大量的网络社交平台和活动管理工具。但现有平台通常缺乏个性化的推荐机制，无法将用户与感兴趣的社交活动有效地精准匹配，导致社交效率低下，参与度不高。此外，传统的活动管理系统往往存在着组织和参与活动时信息过载、活动安排不规范等问题，这些问题也进一步影响了活动的效果和用户的体验，而且传统的活动管理系统往往比较繁琐，缺乏高效的管理工具。因此，本研究构建了一个综合平台，将智能推荐、社交、活动管理三个方面进行整合，通过对用户兴趣标签和行为数据的分析，有效提升校园活动的参与度和管理效率，实现精准社交匹配。

课题采用基于DevEco Studio的Harmony OS环境下的Ark Ts前端语言，结合Harmony OS的SDK框架实现客户端的开发，再基于Java的IDEA集成开发环境进行服务端开发，完成客户端和MySQL数据库的连接。

系统主要按照三个模块进行设计：管理员、用户、系统。管理员对用户信息、活动和用户发布的内容进行管理；用户可以进行注册和登录，查询、管理个人信息，参与、评论和点赞社交活动，创建社区活动，添加好友，与好友聊天等操作；系统对用户异常行为和用户安全的管理。

**关键词：Harmony OS；MySQL；智能推荐**

Design and Implementation of a Social Networking Application Based on HarmonyOS

Abstract

With the rapid development of information technology, a large number of online social platforms and activity management tools have emerged in campus life. However, existing platforms often lack personalized recommendation mechanisms, failing to effectively and accurately match users with social activities of interest, resulting in low social efficiency and poor engagement. Additionally, traditional activity management systems frequently suffer from issues such as information overload during organization and participation, as well as irregular activity arrangements, which further undermine the effectiveness of activities and user experience. Moreover, traditional activity management systems tend to be cumbersome and lack efficient management tools. Therefore, this study constructs an integrated platform that combines intelligent recommendations, social interaction, and activity management. By analyzing user interest tags and behavioral data, it effectively enhances participation in campus activities and management efficiency, achieving precise social matching.

The project is implemented using the HarmonyOS environment based on DevEco Studio, with the front-end developed in ArkTS and utilizing the HarmonyOS SDK framework. The back-end is built using Java within the IDEA integrated development environment, and the system establishes a connection between the client application and a MySQL database.

The system is primarily designed around three modules: Administrator, User, and System. The administrator manages user information, activities, and user-published content. Users can register and log in, query and manage personal information, participate in, comment on, and like social activities, create community events, add friends, chat with friends, and perform other operations. The system manages user abnormal behavior and user security.

**Key Words：HarmonyOS; MySQL; Intelligent Recommendation**

目 录

[1 绪论 1](#_Toc24752)

[1.1 研究背景 1](#_Toc15155)

[1.2 目的和意义 1](#_Toc24580)

[1.3 设计目标 1](#_Toc5432)

[2 系统开发 3](#_Toc2199)

[2.1 HarmonyOS开发框架 3](#_Toc15285)

[2.2 Ability框架 3](#_Toc26646)

[2.3 ArkUI 3](#_Toc2422)

[2.4 ArkTS 3](#_Toc2752)

[2.5 CSS 4](#_Toc3023)

[2.6 SpringBoot 4](#_Toc6731)

[2.7 MySQL数据库 4](#_Toc6287)

[2.8 WebSocket 4](#_Toc17547)

[2.9 基于内容的推荐算法 5](#_Toc8304)

[3 基于HarmonyOS的交友APP分析与设计 6](#_Toc12452)

[3.1 需求分析 6](#_Toc2577)

[3.2 系统结构设计 6](#_Toc16025)

[3.3 数据库设计 8](#_Toc452)

[3.3.1 数据库概念结构设计 8](#_Toc27757)

[3.3.2 数据库逻辑结构设计 9](#_Toc19628)

[3.3.3 数据库物理结构设计 10](#_Toc21809)

[3.3.4 数据库连接 14](#_Toc25095)

[4 基于HarmonyOS的交友APP的实现 15](#_Toc8639)

[4.1 系统使用模块 15](#_Toc25502)

[4.1.1 用户登录 15](#_Toc16355)

[4.1.2 用户注册 16](#_Toc5788)

[4.1.3 完善个人信息 16](#_Toc7240)

[4.1.4 用户聊天 17](#_Toc18542)

[4.1.5 添加好友 19](#_Toc15467)

[4.1.6 通讯录 19](#_Toc10073)

[4.1.7 参与活动 21](#_Toc16718)

[4.1.8 创建活动 23](#_Toc5341)

[4.1.9 修改个人信息 24](#_Toc24792)

[4.1.10 管理员管理用户 25](#_Toc26662)

[4.1.11 管理员管理活动 27](#_Toc4850)

[4.1.12 管理员审核活动 28](#_Toc2330)

[4.2 系统主要功能实现 29](#_Toc22765)

[4.2.1 实时接收聊天信息 29](#_Toc20678)

[4.2.2 发送聊天信息 30](#_Toc30898)

[4.2.3 智能推荐 30](#_Toc16949)

[4.2.4 热门活动筛选 31](#_Toc4807)

[4.2.5 信息加密 31](#_Toc1780)

[5 系统测试 33](#_Toc28978)

[5.1 测试目的 33](#_Toc13526)

[5.2 测试方法 33](#_Toc7872)

[5.3 功能测试 33](#_Toc20146)

[5.4 测试结论 36](#_Toc4412)

[结论 37](#_Toc16431)

[参考文献 38](#_Toc18068)

[致谢 39](#_Toc21726)

# 绪论

## 研究背景

如今，信息技术日新月异，互联网、智能手机几乎覆盖了每一个角落，人们的生活方式也随之发生了变化，数字化、智能化在校园里已经不是什么新鲜的概念，而是在日常的学习、工作中深深地融入了其中。学生通过社交平台结识新朋友，教职员工借助管理工具组织活动，这些技术本应让校园生活更便捷、更高效，但现实情况却并不那么理想，现有的社交平台虽然功能越来越多，但真正能“懂”用户的却寥寥无几。他们不会根据你的兴趣爱好推荐合适的活动，也不会分析你的社交习惯来匹配志趣相投的人，结果往往是学生被淹没在杂乱的信息中，好不容易找到一个活动，却发现与自己想要的相差甚远，活动管理起来同样麻烦。传统的系统操作复杂，但功能十分有限，活动主办方往往需要依靠多个工具来策划活动、宣传活动、报名活动、收集反馈等，这在增加管理成本的同时，也容易造成活动整体效果和用户体验受到信息过载、活动安排不规范等影响。因此，如何通过技术手段优化校园社交和活动管理，成为当前校园信息化建设的迫切需要解决的问题。

## 目的和意义

本课题构建的是一个整合了智能推荐、社交互动与活动管理的平台。主要解决目前平台上存在个性化推荐以及活动管理方面的问题。通过加入智能推荐算法，可以更加准确地实现将用户与感兴趣的社交活动匹配。并且，平台将配备活动管理工具，可以为用户创建活动管理平台，简化活动的组织流程，规范活动的安排，避免信息过载。

## 设计目标

本课题要求构建一个集智能推荐、校园社交、活动管理功能于一体的综合性平台，专为大学生和教职工设计，满足他们在校园生活中的实际需求。智能推荐系统基于用户的兴趣和行为数据，精准匹配潜在社交对象和活动，增强校园社交的互动性和参与度，平台提供专属活动模块，专注于校园活动的创建、发布和参与。活动管理模块简化了校园活动的创建、组织和参与流程，有效提高了活动的管理效能。

(1)智能推荐系统的设计与实现：通过用户的兴趣行为数据（如注册时填写的兴趣爱好、浏览历史、活动参与情况等），将志趣相投的好友与活动推荐到添加好友模块上方。

(2)学校频道功能的开发：为用户提供活动模块，支持校内活动（如志愿活动、校园比赛、社团活动等）的发布与参与，用户可以在活动模块内查看、参与并创建学校相关的社交活动。

(3)社交活动的创建与管理：模块化的设计允许用户快速完成活动的设置，通过系统管理，平台可以确保所有活动的创建遵循统一的标准和规则（如设置明确的主题、时间、地点和人数限制），可以避免不规范、不完整的活动信息发布，从而减少用户参与活动时的信息混乱。

(4)异常检测与用户安全保障：使用行为分析和异常检测算法，对用户的行为进行监控，识别异常行为（如频繁发布垃圾信息、虚假内容或违禁词），在信息传输过程对信息加密，以确保消息未被篡改，以确保平台的安全性。

(5)数据传输与实时反馈：通过高速无线技术实现校园内外的信息实时同步。

# 系统开发

本课题采用基于DevEcoStudio的HarmonyOS环境下的ArkTs前端语言，结合HarmonyOS的SDK框架实现客户端开发，再基于Java的IDEA集成开发环境进行服务端开发，完成客户端和MySQL数据库的连接。

## HarmonyOS开发框架

华为推出的HarmonyOS有效弥补了我国在全球移动操作系统领域的空白。平台相比目前主流的移动操作系统，在整合常规系统所有功能的同时，更以创新的方式实现了真正打破设备间壁垒、跨终端协同作战的能力，展现了独特的竞争优势。其核心特性可以概括为：  
(1)采用的分布式框架突破了传统系统的局限，支持各类智能终端之间的流畅交互与高效协作；  
(2)通过优化延时管理和进程通信机制，系统响应速度和运行效率得到显著提升；

(3)精简的微内核结构既增强了安全防护能力还保证了系统运行的实时性；

(4)集成式开发环境可以兼容多种编程语言，开发者只需要编写一次代码就可以适配多种终端设备，促进了跨平台生态的共建共享[1]。

## Ability框架

在鸿蒙系统的架构设计中，Ability作为功能模块化的核心概念，为开发者提供了灵活的应用构建途径。每个应用程序可以集成一个或多个Ability单元，按照功能特性将这些单元分为两种基本类型：Feature Ability(FA)和Particle Ability(PA)。其中，FA作为一个拥有用户界面的交互单元，主要负责前端业务逻辑的实现；PA则是一个以后台数据处理和功能支持为重点的无界面服务单元。两个能力单元通过协同合作，共同构成了鸿蒙应用的基础功能框架，使开发者能够高效地实现复杂的业务需求，这种模块化的设计理念在提升开发效率的同时，也为系统带来更好的可扩展性和维护性，在开发过程中，鸿蒙应用的能力架构将会得到更好的提升[2]。

## ArkUI

ArkUI（方舟UI框架）是专为开源鸿蒙打造的新一代界面开发利器。它用更直观的声明式写法，让开发者能像搭积木一样快速构建应用界面。相比传统开发方式，ArkUI提供了丰富的现成组件和实时预览功能，大大简化了UI开发流程，让界面实现变得轻松高效[3]。

## ArkTS

ArkTS是华为团队对TypeScript进行深度优化和功能扩展的一种开发语言，完美适配ArkUI框架，保留了TypeScript基础语法的同时增加了声明式UI编程范式和组件化开发能力。ArkTS技术架构继承自JavaScript/TypeScript技术栈，方便开发者快速上手多端应用的开发。ArkTS强化了静态类型检查，在编译阶段就能捕获错误，有效提升了代码质量和稳定性。ArkTS兼容现有的JavaScript/TypeScript生态，针对移动端开发场景增加声明式UI描述语法和高效的并发编程模型[4]。

## CSS

CSS是前端开发中重要的样式语言，有着强大的样式控制能力，通过丰富的样式属性和选择器能够实现各种美观、精美的页面设计效果，通过这些样式属性和选择器还可以实现各种精美的视觉元素以及动效的交互，通过CSS的样式表语言不仅能够构建响应式的布局，还可以保证页面的内容在不同终端设备上能够保持统一的显示效果，从而能够提高跨平台用户体验[5]。

## SpringBoot

SpringBoot是Spring框架为简化Java应用开发而设计的解决方案。它主要解决了传统Spring项目中配置复杂的问题，通过预设的自动配置、内置服务器和模块化依赖来减少开发者的手动配置工作。使用SpringBoot可以快速搭建各种规模的企业应用，这些应用不仅运行效率高，还具备可靠的扩展性和安全性[6]。在Spring框架中，使用@Scheduled注解可以轻松实现定时任务功能。开发者只需要在方法上添加这个注解，框架就会自动将任务加入调度队列。应用启动后，Spring会创建一个专用的任务执行线程，这个线程会不断检查队列中的任务，当发现有任务到达预定执行时间时，就会立即执行对应的方法[7]。

## MySQL数据库

MySQL在大规模数据场景下表现出了显著的处理优势，能够有效组织多源异构信息，进行高效的数据采集、汇聚。内置的ETL功能组件可进行数据清洗、格式转换、噪声过滤等数据预处理工作。通过标准化的SQL查询语句完成数据提取、指标运算等工作，为在线分析处理（OLAP）提供数据基础。规整的结构化数据集可用于构建可视化分析看板和机器学习算法训练样本。数据库内置的关联分析功能可在特征工程中发挥优势，提升预测模型准确性[8]。

## WebSocket

WebSocket是一项突破传统HTTP协议限制的创新全双工通信技术。与基于请求-响应模式的HTTP不同，WebSocket建立了持久稳定的通信通道，使服务器和客户端可以随时主动发送数据，真正意义上实现实时双向交互。这一技术的诞生正是为了解决HTTP协议在实时通信场景中的先天不足，为现代网络应用开辟了一条实时数据传输的新途径，大大拓展了Web应用的交互可能性[9]。

## 基于内容的推荐算法

基于内容的推荐系统通过分析用户过往偏好特征，计算内容相似度来生成个性化推荐，这种机制能精准匹配用户兴趣偏好。该算法不依赖群体行为数据，仅需物品本身的特征信息即可完成推荐，特别适合新用户场景。在冷门领域或用户行为数据不足的情况下，系统仍能保持稳定的推荐质量[10]。

# 基于HarmonyOS的交友APP分析与设计

## 需求分析

系统的总体目标是为用户提供一个基于HarmonyOS的交友平台，通过智能推荐、兴趣匹配和活动参与，促进用户之间的互动和社交。为了实现系统设计的总目标，基于HarmonyOS的交友APP应具备以下功能目标，主要包括：

（一）用户可以注册和登录，查询、点赞、评论、管理个人信息，参与社交活动，匹配好友。

1)用户可以注册个人账号并登录APP。

2)用户可以基于兴趣标签、注册时填写的兴趣爱好、浏览历史和活动参与情况，获得好友推荐列表，快速匹配志趣相投的朋友。系统将采用基于内容的推荐算法，优先推荐与用户兴趣相似或参与过相同活动的其他用户。

3)用户可以通过APP中的活动推荐模块，浏览并参与平台推荐的社交活动，如校园内外的聚会、比赛等。系统将分析用户过往的活动参与情况、点赞和评论行为，并结合当前的热门活动，进行个性化推荐。

4)用户可以创建并管理社交活动，其他用户可以参与、评论和点赞，系统将高点赞活动推送至热门推荐列表。

5)用户可以查看和修改个人信息，管理自己的好友列表。该APP使用双向好友形式，即用户需通过对方的好友请求后才能成为好友。

6)用户可以实时同步校园内外的活动信息，确保能够接收到最新的活动通知和消息。

（二）管理员可以管理用户信息、管理活动和审核用户发布的内容。

1)管理员可以封禁、解封和删除用户。

2)管理员可以审核用户发布的活动、评论等信息，确保平台内容的合法合规。

3)管理员可以对平台中的社交活动进行管理，包括发布新的社交活动或对现有活动进行修改和删除。

（三）系统保障功能，包括异常检测和用户安全管理。

1)系统使用行为分析算法，实时监控用户的行为，识别异常活动，如垃圾信息或不良内容发布。

2)系统在信息传输过程中进行加密处理，确保用户数据的安全性和隐私性。

## 系统结构设计

基于HarmonyOS的交友APP的系统结构设计采用了分层架构，分为前端层、后端层和数据层。前端层负责用户界面的展示和交互，后端层处理业务逻辑和数据存储，数据层负责数据的持久化存储。通过合理的模块划分和技术选型，系统能够满足用户的功能需求和非功能需求，确保系统的高效性、安全性和可扩展性。

这是本系统各个功能模块划分。共分为五块：用户模块、活动管理模块、推荐系统模块、管理员模块、系统保障模块，详细的基于HarmonyOS的交友APP系统功能结构图，如图 3.1所示。

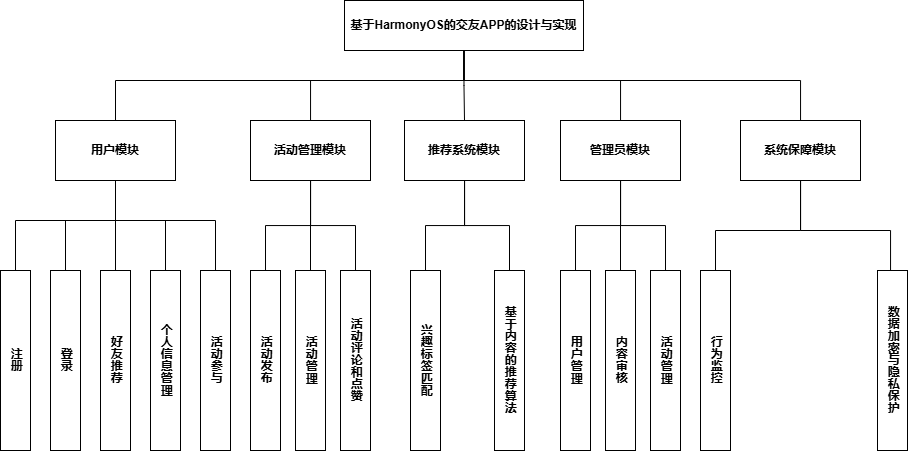


图 3.1基于HarmonyOS的交友APP功能结构图

在用户模块中，实现了用户可以注册和登录自己的账号，而且还可以在APP里面直接点开进入，直接在好友列表当中查看到系统推荐的好友。同时，用户可以查看自己的信息，也可以修改自己的信息，保证自己个人信息的及时更新和修改。并且用户可以自由管理自己的好友列表，可以对好友进行添加或者删除。最后，用户还可以通过用户模块浏览和参与各类社交活动。

用户活动管理模块用于用户创建及管理社交活动，用户可通过此模块发布新的社交活动，并设置活动的时间、标签、人数等信息；同时用户也可对已发布过的活动进行编辑、取消、重新安排；此外用户还可以评论、点赞参加的活动；这些数据不仅可以丰富用户的社交体验，还可以给推荐系统模块提供行为数据。

推荐系统模块是平台的核心功能，根据用户的兴趣标签、行为数据，为用户进行个性化推荐。通过整合用户的兴趣标签、活动参与历史、点赞评论等行为数据，再利用基于内容的推荐算法，向用户推荐有可能感兴趣的好友、活动，并且点赞量高的社交活动会被推荐到活动热门板块，增加活动曝光度和用户参与度。

管理员模块是为平台管理员提供对平台的管理功能，维护平台的秩序，确保平台内容的合规性。管理员可以对用户进行封禁、解封和删除等操作，处理违规用户，维护平台的健康环境；审核用户发布的活动、评论等，确保平台内容的合法性；还可以发布、修改和删除社交活动，确保活动的管理规范有序。

系统保障模块主要是为了确保平台的安全性和稳定性。采用行为分析算法，实时对用户的行为进行监控，发现垃圾信息发布、不良信息传播等异常行为；在信息传输过程中对用户的数据进行加密处理，确保数据的安全性和隐私性。

## 数据库设计

社交平台的数据库建设，作为整个系统的重要组成部分，承载着存储、管理和处理海量数据的使命，能否建设一个高效、稳定、可扩展的数据库，不仅决定着平台的核心功能能否得以正常运转，而且决定着能否为用户提供顺畅的体验，以及能否确保数据的安全性和完整性。

### 数据库概念结构设计

在数据库设计中，实体关系图（Entity-Relationship Diagram,简称ER图）是描述数据模型的重要工具。它将数据库中的实体、属性以及实体之间的关系通过图形化的方式展现出来，帮助开发人员和设计人员对数据的结构和逻辑关联进行直观的了解。对于社交平台来说，ER图的设计不仅可以将用户、活动、评论、点赞等核心数据之间的关系清晰呈现，同时也可以为数据库的实现和优化提供指引。本系统设计的 ER 图如图 3.2所示。

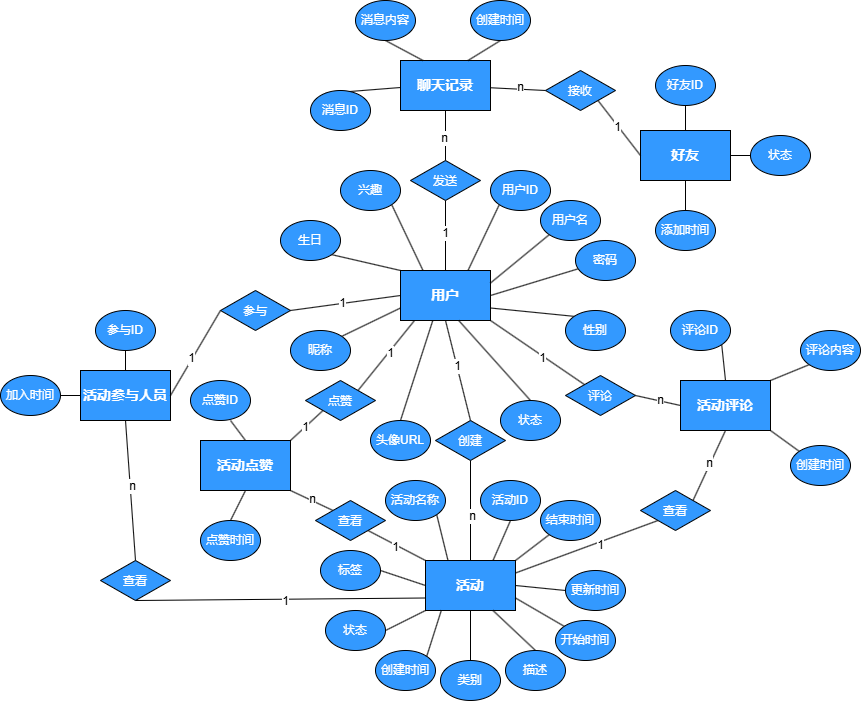


图 3.2基于HarmonyOS的交友APP E-R 图

### 数据库逻辑结构设计

数据库逻辑结构设计是数据库系统开发的核心环节，旨在将概念模型（如ER图）转化为具体的逻辑模型，在社交平台中，逻辑结构设计直接决定了用户管理、活动创建、互动行为、推荐系统等核心功能的数据支撑能力，在这些功能需求的前提下，需要列出以下数据项：

1. USERS 用户表（用户ID，用户名，密码，性别，头像URL，昵称，生日，兴趣，状态）
2. MESSAGES 聊天记录表（消息ID，发送者ID，接收者ID，消息内容，创建时间）
3. FRIENDS 好友表（用户ID，好友ID，状态，添加时间）
4. ACTIVITY 活动表（活动ID，活动名称，类别，标签，描述，开始时间，结束时间，创建时间，更新时间，参与人数，点赞数，评论数，状态，创建者ID）
5. ACTIVITY\_PARTICIPANTS 活动参与人员表（参与ID，活动ID，用户ID，加入时间）
6. ACTIVITY\_COMMENTS 活动评论表（评论ID，活动ID，用户ID，评论内容，创建时间）
7. ACTIVITY\_LIKES 活动点赞表（点赞ID，活动ID，用户ID，点赞时间）

### 数据库物理结构设计

基于HarmonyOS的交友APP使用 MySQL 数据库进行数据存储。这是数据库表和关系表的表示形式，在本系统中一共创建表结构如下图所示。

1. 用户表：主要包含用户ID，用户名，密码，性别，头像URL，昵称，生日，兴趣，状态。如表 3.1所示。

表 3.1用户表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 默认值 | 是否非空 | 主外键 | 备注 |
| ID | INT |  | YES | 主键 | 用户ID |
| USERNAME | VARCHAR(255) |  | YES |  | 用户名 |
| PASSWORD | VARCHAR(255) |  | YES |  | 密码 |
| GENDER | VARCHAR(10) |  | NO |  | 性别 |
| AVATAR\_URL | VARCHAR(255) |  | NO |  | 头像URL |
| NAME | VARCHAR(100) |  | NO |  | 昵称 |
| BIRTHDAY | DATE |  | NO |  | 生日 |
| INTERESTS | TEXT |  | NO |  | 兴趣 |
| STATUS | ENUM | Normal(正常) | YES |  | 状态 |

1. 聊天记录表：主要包含消息ID，发送者ID，接收者ID，消息内容，创建时间。如表 3.2所示。

表 3.2聊天记录表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 默认值 | 是否非空 | 主外键 | 备注 |
| MESSAGE\_ID | INT |  | YES | 主键 | 消息ID |
| SENDER\_ID | INT |  | YES | 外键 | 发送者ID |
| RECEIVER\_ID | INT |  | YES | 外键 | 接收者ID |
| MESSAGE | TEXT |  | YES |  | 消息内容 |
| CREATED\_AT | TIMESTAMP |  | YES |  | 创建时间 |

1. 好友表：主要包含用户ID，好友ID，状态，添加时间。如表 3.3所示。

表 3.3好友表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 默认值 | 是否非空 | 主外键 | 备注 |
| USER\_ID | INT |  | YES | 主键，外键 | 用户ID |
| FRIEND\_ID | INT |  | YES | 主键，外键 | 好友ID |
| STATUS | ENUM | Pending(申请中) | YES |  | 状态 |
| ADDED\_AT | TIMESTAMP |  | YES |  | 添加时间 |

1. 活动表：主要包含活动ID，活动名称，类别，标签，描述，开始时间，结束时间，创建时间，更新时间，参与人数，点赞数，评论数，状态，创建者ID。如表 3.4所示。

表 3.4活动表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 默认值 | 是否非空 | 主外键 | 备注 |
| ACTIVITY\_ID | INT |  | YES | 主键 | 活动ID |
| ACTIVITY\_NAME | VARCHAR(255) |  | YES |  | 活动名称 |
| CATEGORY | VARCHAR(100) |  | YES |  | 类别 |
| TAGS | VARCHAR(255) |  | NO |  | 标签 |
| DESCRIPTION | TEXT |  | NO |  | 描述 |
| START\_TIME | DATETIME |  | NO |  | 开始时间 |
| END\_TIME | DATETIME |  | NO |  | 结束时间 |
| CREATED\_AT | TIMESTAMP |  | YES |  | 创建时间 |
| UPDATED\_AT | TIMESTAMP |  | YES |  | 更新时间 |
| PARTICIPANTS\_COUNT | INT | 0 | YES |  | 参与人数 |
| LIKES\_COUNT | INT | 0 | YES |  | 点赞数 |
| COMMENTS\_COUNT | INT | 0 | YES |  | 评论数 |
| STATUS | ENUM | Active(正常) | YES |  | 状态 |
| CREATOR\_ID | INT |  | YES | 外键 | 创建者ID |

1. 活动参与人员表：主要包含参与ID，活动ID，用户ID，加入时间。如表 3.5所示。

表 3.5活动参与人员表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 默认值 | 是否非空 | 主外键 | 备注 |
| PARTICIPANT\_ID | INT |  | YES | 主键 | 参与ID |
| ACTIVITY\_ID | INT |  | YES | 外键 | 活动ID |
| USER\_ID | INT |  | YES | 外键 | 用户ID |
| JOINED\_AT | TIMESTAMP |  | YES |  | 加入时间 |

1. 活动评论表：主要包含评论ID，活动ID，用户ID，评论内容，创建时间。如表 3.6所示。

表 3.6活动评论表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 默认值 | 是否非空 | 主外键 | 备注 |
| COMMENT\_ID | INT |  | YES | 主键 | 评论ID |
| ACTIVITY\_ID | INT |  | YES | 外键 | 活动ID |
| USER\_ID | INT |  | YES | 外键 | 用户ID |
| COMMENT\_TEXT | TEXT |  | NO |  | 评论内容 |
| CREATED\_AT | TIMESTAMP |  | YES |  | 创建时间 |

1. 活动点赞表：主要包含点赞ID，活动ID，用户ID，点赞时间。如表 3.7所示。

表 3.7活动点赞表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 默认值 | 是否非空 | 主外键 | 备注 |
| LIKE\_ID | INT |  | YES | 主键 | 点赞ID |
| ACTIVITY\_ID | INT |  | YES | 外键 | 活动ID |
| USER\_ID | INT |  | YES | 外键 | 用户ID |
| LIKED\_AT | TIMESTAMP |  | TES |  | 点赞时间 |

### 数据库连接

完成数据导入后，后端需要与数据库建立连接，本系统使用的是MySQL关系型数据库。后端使用的是Java编程语言，Java提供了JDBC（Java Database Connectivity）API，用于连接和操作各种关系型数据库。以下是后端与数据库连接的代码：

private static final String URL = "jdbc:mysql://localhost:3306/makefriends";  
private static final String USER = "root";  
private static final String PASSWORD = "025354";  
private Connection connection;

public DbService() {  
 try {  
 connection = DriverManager.getConnection(URL, USER, PASSWORD);  
 } catch (SQLException e) {  
 e.printStackTrace();  
 }

}

# 基于HarmonyOS的交友APP的实现

## 系统使用模块

### 用户登录

本系统用户分为管理员和普通用户，两者通过统一的登录页面进行身份验证。用户输入用户名和密码并点击登录按钮后，前端将信息发送至后端，后端连接数据库查询用户信息。若信息匹配且账号状态正常，后端返回用户数据，前端根据角色跳转至对应的系统主页；若用户名或密码错误，后端返回错误信息，前端提示用户重新输入；若数据库连接失败或查询超时，后端返回服务器错误信息，前端提示连接失败；若用户账号已被封禁，前端提示账号已被封禁。此外，系统内提供“退出登录”功能，点击后前端清除本地缓存的用户信息，并自动跳转至登录页面，确保用户安全退出。登录功能模块图如图 4.1所示。



图 4.1 登录功能界面

### 用户注册

本系统支持新用户注册功能，用户若未注册账号，可点击登录页面的“没有账号？马上注册”按钮跳转至注册页面。注册流程分为三步：首先输入用户名，用户名只能包含字母和数字，长度为8-16位，若格式不符合要求，前端会提示格式错误；然后设置密码，密码只能包含字母、数字和特殊字符，长度为8-15位，用户需再次确认密码以确保一致性；最后选择性别，若不选择则默认为男。完成以上步骤后，用户点击注册按钮，前端将注册信息发送至后端，若用户名已被注册，前端会提示用户名已存在，若注册信息无误，后端会完成注册并返回成功信息，前端提示注册成功并自动跳转至登录页面，方便用户登录系统。注册功能模块图如图 4.2所示。



图 4.2 注册功能界面

### 完善个人信息

当用户首次登录系统时，系统会弹出“请完善信息”的弹窗，弹窗中要求用户填写自己的昵称、出生日期、兴趣。填写昵称和出生日期可以使用户在系统中有自己的个性化的标识，选择兴趣是为了能够更精准地为用户推荐好友以及活动。完善信息的填写后系统会将这些数据保存下来，并根据兴趣为用户推荐内容和活动。完善信息界面如图 4.3所示。



图 4.3完善信息界面

### 用户聊天

用户完成个人信息完善后，系统会跳转至首页，首页会显示用户的聊天记录列表，包括最近联系的好友及其最新的聊天内容摘要。用户点击某个好友的聊天记录，即可进入与该好友的聊天界面，进行实时对话。聊天记录会实时更新，确保用户可以及时查看和回复最新的消息。首页界面如图 4.4所示，聊天界面如图 4.5所示。

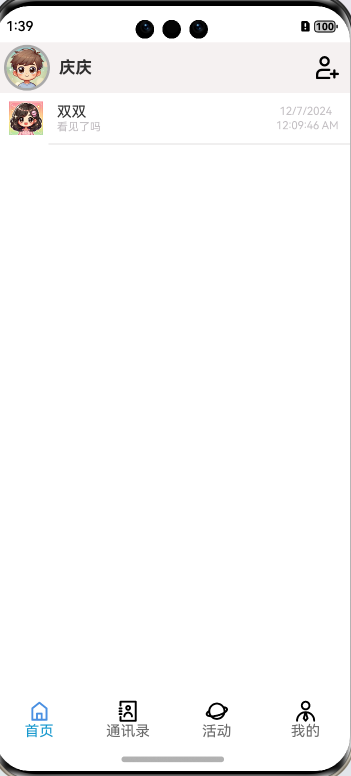


图 4.4 首页界面



图 4.5 聊天界面

### 添加好友

在首页或通讯录页面的右上角，点击图标即可跳到添加好友界面，用户可以通过搜索框输入目标用户的用户名进行搜索，系统会实时显示匹配的结果，找到目标用户后点击“添加”按钮就可以发送好友请求。这个系统采用双向关注机制，需要对方同意好友请求后才能开始聊天。添加好友界面如图 4.6所示。

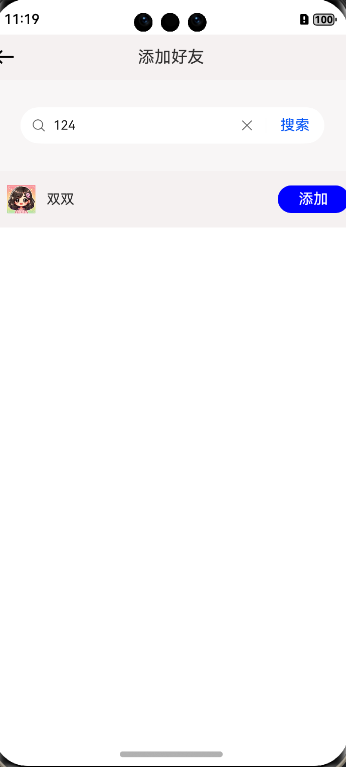


图 4.6 添加好友界面

### 通讯录

在通讯录界面，系统分为两个主要区域：推荐区和我的好友区。在推荐区，用户可以查看系统根据兴趣推荐的好友，点击推荐好友后可以查看其出生日期、兴趣等基本信息，并根据自己的意愿选择是否发送好友请求。在我的好友区，用户可以查看已添加的好友列表，点击某位好友可以查看其详细信息，如昵称、兴趣等，点击“发消息”按钮可直接跳转到与该好友的聊天界面。此外，通讯录界面还提供“新朋友”入口，点击后可跳转到好友申请页面，查看是否有待处理的好友申请。通讯录界面如图 4.7所示，好友详细界面如图 4.8所示，好友申请界面如图 4.9所示。



图 4.7 通讯录界面



图 4.8 好友详情界面

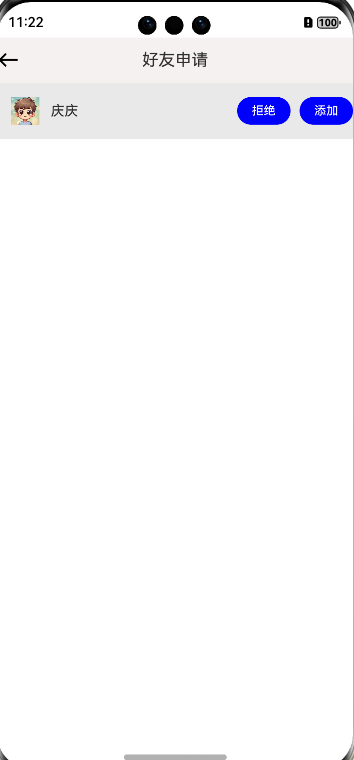


图 4.9 好友申请界面

### 参与活动

活动界面分为多个类别，包括推荐、热门、运动、艺术、娱乐、科技和公益，方便用户快速找到感兴趣的活动。推荐活动是根据用户的兴趣标签进行个性化推荐；热门活动则展示当前点赞数前三名的活动。每个模块下方会显示对应类别活动的简略信息，如活动名称、时间等，用户点击活动后可以进入活动详情界面，查看更详细的信息，包括活动创建者、开始时间、结束时间、参与人数、评论和点赞等。如果用户对活动感兴趣，可以点击“参加”按钮报名参与。用户已报名的活动可以在“我的活动”页面中集中查看和管理，方便用户随时了解自己的活动安排和状态。活动界面如图 4.10所示，活动详情界面如图 4.11所示，我的活动界面如图 4.12所示。



图 4.10 活动界面



图 4.11 活动详情界面

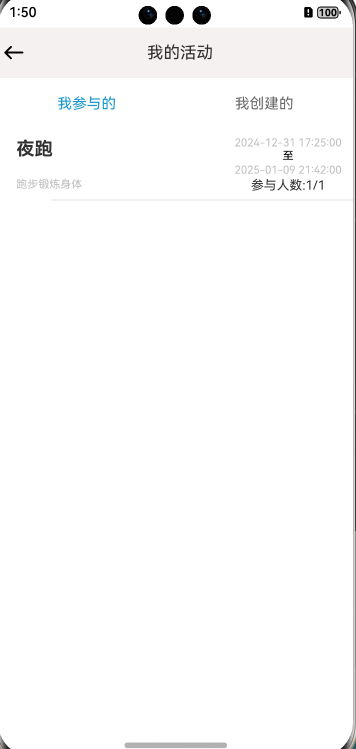


图 4.12 我的活动界面

### 创建活动

活动界面右上角点击“ + ”按钮，即可进入活动创建页面，用户需要填写详细信息，包括活动名称、活动类别、活动标签、活动时间、参与人数等。活动提交后进入审核状态，由管理员审核内容确保合规，审核通过后才会在对应类别中公开展示活动情况，用户可实时查看自己创建活动的审核状态。创建活动界面如图 4.13所示。

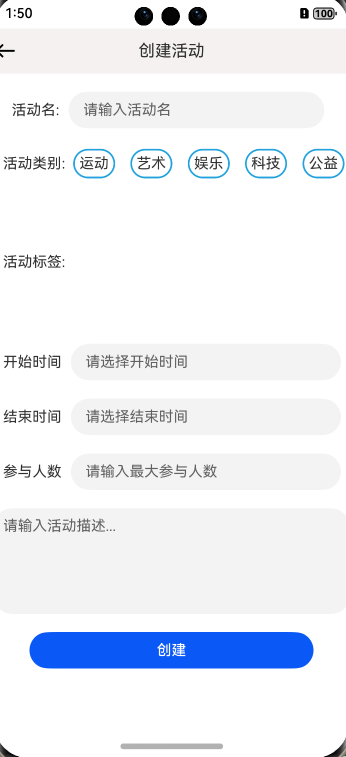


图 4.13 创建活动界面

### 修改个人信息

在“我的”个人中心界面，用户可以方便地管理个人信息，界面提供了昵称、出生日期、性别和兴趣标签四个可修改模块。点击任意模块，系统就会弹出对应的编辑弹窗。用户完成修改后点击确认按钮，前端会立即同步更新数据到后端数据库，保证信息实时更新，所有修改操作均经过数据校验，保证格式规范。个人中心界面如图 4.14所示。



图 4.14 个人中心界面

### 管理员管理用户

在管理员后台，管理员拥有完整的用户管理权限。管理员面板可以展示平台所有用户的列表视图，支持通过用户名或用户ID进行精准搜索定位特定用户。点击任一用户条目后，系统将跳转到该用户的详情管理界面，管理员可通过操作按钮执行账户状态管理：对违规用户点击"封禁"按钮将立即限制其登录权限，对已整改用户点击"解封"按钮即可恢复其正常使用权限，还可以直接删除用户，所有封禁、解封和删除操作都会实时同步至数据库。管理用户界面如图 4.15所示，用户详情管理界面如图 4.16所示。



图 4.15 管理用户界面



图 4.16 用户详情管理界面

### 管理员管理活动

管理员拥有全面的活动管理权限，可查看所有活动或通过活动ID精准搜索，点击进入详情页后能对违规活动执行封禁、解封或删除操作，所有变更实时生效并同步数据库。管理员还享有直接创建活动的特权，其创建的活动自动通过审核。管理活动界面如图 4.17所示，活动详情页如图 4.18所示。



图 4.17 管理活动界面



图 4.18 活动详情页

### 管理员审核活动

用户创建的活动将首先进入待审核状态，仅创建者本人可见。管理员会收到审核通知，在审查活动内容后做出判定：若审核通过，活动将立即公开发布，所有用户可见并可参与；若审核不通过，系统会自动删除该活动。审核界面如图 4.19所示。



图 4.19 审核界面

## 系统主要功能实现

### 实时接收聊天信息

本系统采用单例模式管理Socket连接，确保用户首次访问首页时仅创建一个持久化连接。该连接会持续监听后端信息，当接收到信息时自动触发回调函数，将信息存储至前端并实时渲染展示。这种设计既避免了重复创建连接的开销，又保证了消息的实时处理与展示。以下是实时接收聊天信息功能的主要代码：

private tcpSocket: socket.TCPSocket | null = null;  
onMessageReceived?: (receivedMessage: string) => void;  
this.tcpSocket.on("message", (value) => {  
 let byteArray = new Uint8Array(value.message);  
 let textDecoder = util.TextDecoder.create("utf-8");  
 let State = textDecoder.decodeToString(byteArray).trim();  
 if (State && this.onMessageReceived) {  
 this.onMessageReceived(State);  
 }  
});

### 发送聊天信息

当用户成功登录后，后端会为每个客户端创建一个线程，用 ConcurrentHashMap 安全地存储用户名与对应的PrintWriter，确保多线程环境下的数据一致性。当需要向特定用户发送消息时，系统会根据用户名快速检索对应的输出流，并通过该流高效推送数据，实现精准、可靠的实时通信。以下是发送聊天信息功能的主要代码：

private static Map<String, PrintWriter> clientWriters = new ConcurrentHashMap<>();

PrintWriter targetWriter = clientWriters.get(targetUsername);  
if (targetWriter != null) {  
 DbService.USERINFO user = this.queryUserOne(username);  
 targetWriter.println(user.getId() + "|#|" + message);  
} else {  
 System.out.println("目标用户不在线。");  
}

### 智能推荐

本系统采用动态标签机制进行个性化推荐，用户标签由两部分构成：显式标签（用户自主选择）与隐式标签（系统根据用户行为自动生成）。系统会持续分析用户行为，包括浏览、点赞、评论和活动参与等，根据行为权重动态调整兴趣标签的强度。  
 设标签为，用户对该标签的兴趣权重为，每种行为对应权重如下：

 (4.1)

每次用户行为发生后，兴趣值按照如下公式动态更新：

 (4.2)

当某一标签的兴趣值达到设定阈值时，系统自动将该标签纳入用户的兴趣标签集合，并将其兴趣值归零以便重新累积：

 (4.3)

该机制保证了推荐内容与用户当前兴趣高度匹配，从而优化社交推荐与活动推荐策略。为实现上述逻辑，智能推荐功能的主要代码如下所示：

switch (action) {  
 case "browse":  
 this.interest[labelArray[index]]=this.interest[labelArray[index]]+1;  
 break;  
 case "like":  
 this.interest[labelArray[index]]=this.interest[labelArray[index]]+2;  
 break;  
 case "comment":  
 this.interest[labelArray[index]]=this.interest[labelArray[index]]+3;  
 break;  
 case "join":  
 this.interest[labelArray[index]]=this.interest[labelArray[index]]+5;  
 break;  
}

if(this.interest[labelArray[index]]>=10){  
 const currentInterests = interests || "";  
 this.interest[labelArray[index]] = 0;  
 if (!currentInterests.split(' ').includes(labelArray[index])) {  
 let update = new UpdateNet();  
 let newInterests = interests+' '+labelArray[index]  
 update.updateInterests(newInterests)  
 }  
}

### 热门活动筛选

系统会将点赞数排名前三的活动自动归类至热门板块，并在前端优先展示，以提升高人气活动的曝光度。以下是热门活动筛选功能的主要代码：

ArrayList<ACTIVITY> recommendedActivities = new ArrayList<>();  
int likeCount = getActivityLikes(activityId);  
activity.setLikesCount(likeCount);  
recommendedActivities.add(activity);

recommendedActivities.sort((a1, a2) -> Integer.compare(a2.getLikesCount(), a1.getLikesCount()));

### 信息加密

本系统采用端到端加密机制保障数据安全：所有传输信息在客户端进行加密，后端接收后解密并安全存储于数据库；在信息转发过程中，使用密钥循环的对称加密算法再次加密处理，确保数据在传输链路中的保密性。目标用户接收后解密消息并将其存储于本地前端。该方案在数据传输、存储及转发各环节提供全程保护，有效降低信息泄露风险。以下是信息加密功能的主要代码：

let encryptedMessage = "";  
for (let i = 0; i < message.length; i++) {  
 const charCode = message.charCodeAt(i);  
 const keyCode = this.key.charCodeAt(i % this.key.length);  
 const newCharCode = charCode + keyCode;  
 encryptedMessage += String.fromCharCode(newCharCode);  
}

return encryptedMessage.split("").reverse().join("");

# 系统测试

## 测试目的

软件测试的目的是及时发现和修正软件系统中存在的缺陷和问题，保证系统稳定高效地运行。在设计基于HarmonyOS的交友APP以提升校园社交体验的同时，必须从用户需求出发，确保系统功能完善可靠。为此，用户注册时对错误格式输入的有效拦截、好友消息的实时渲染显示、基于用户行为的智能推荐标签更新、活动结束后的自动下架机制，以及数据加密、异常监测等功能，都需要经过详尽的测试。通过输入测试数据，调用功能模块，在比较实际结果与预期目标、分析系统运行状态的同时，验证数据库更新是否实时同步到界面。如果发现异常或错误，需要及时排查并修复，最终确保系统功能完整，性能达标，为用户提供一个安全高效的社交平台。

## 测试方法

黑盒测试将待测系统视为一个"黑箱"，在不考虑内部代码结构和实现逻辑的情况下，通过输入数据和验证输出结果来评估系统功能是否符合预期要求。本系统主要采用黑盒测试，从用户的角度测试功能是否满足要求。

## 功能测试

本系统的功能较多，本文主要展示用户注册、用户修改个人信息、活动参与等主要使用功能测试。

1. 用户注册功能的测试用例及结果如表 5.1所示。

表 5.1 用户注册功能的测试用例及结果表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 用例编号 | 输入/动作 | 测试方法 | 预期结果 | 测试结果 |
| 1 | 直接点击注册 | 观看、操作 | 系统提示“请输入正确的用户名格式” | 符合预期 |
| 2 | 输入已被注册的用户名 | 观看、操作 | 系统提示“用户名已存在” | 符合预期 |
| 3 | 输入格式不正确的用户名或密码 | 观看、操作 | 系统提示“请输入正确的格式” | 符合预期 |

当用户输入已被注册的用户名并点击注册后会跳出用户名已存在提示。如图 5.1所示。



图 5.1 注册重复用户名图

1. 用户修改个人信息功能的测试用例及结果如表 5.2所示。

表 5.2 用户修改个人信息功能的测试用例及结果表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 用例编号 | 输入/动作 | 测试方法 | 预期结果 | 测试结果 |
| 1 | 修改性别 | 观看、操作 | 系统提示“修改成功” | 符合预期 |
| 2 | 修改昵称 | 观看、操作 | 系统提示“修改成功” | 符合预期 |
| 3 | 修改出生日期 | 观看、操作 | 系统提示“修改成功” | 符合预期 |
| 4 | 清空昵称并尝试保存 | 观看、操作 | 系统提示“昵称不能为空” | 符合预期 |

当用户修改昵称时不输入新昵称，直接点击“确认”按钮，系统会跳出昵称不能为空提示。如图 5.2所示。



图 5.2 尝试保存空昵称图

1. 活动参与功能的测试用例及结果如表 5.3所示。

表 5.3 活动参与功能的测试用例及结果表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 用例编号 | 输入/动作 | 测试方法 | 预期结果 | 测试结果 |
| 1 | 创建一个结束时间为当前时间+1分钟的活动，等待1分钟后刷新页面 | 观看、操作 | 活动不再出现在活动列表 | 符合预期 |
| 2 | 参加人数已满的活动 | 观看、操作 | 系统提示“参与人数已满” | 符合预期 |
| 3 | 参加已经参加过的活动 | 观看、操作 | 系统提示“你已经参加了” | 符合预期 |
| 4 | 100用户同时报名同一活动（未满员） | 观看、操作 | 系统响应时间≤1秒，报名成功人数=活动剩余名额，超限用户收到“参与人数已满”提示 | 符合预期 |

当用户参加人数已满的活动时，系统会提示参与人数已满。如图 5.3所示。



图 5.3 参加人数已满图

## 测试结论

经过全面的黑盒测试验证，基于HarmonyOS的交友APP核心功能完整，用户注册登录、好友互动、活动管理等主要业务流程运行稳定，界面交互流畅，数据同步及时，异常处理机制完善，测试结果表明系统能够为用户提供安全可靠的社交服务。

# 结论

本论文以校园社交需求为背景，分析了当前社交平台在个性化推荐、活动管理效率和用户隐私保护等方面的不足。通过调研和需求分析，结合HarmonyOS的技术优势，设计并实现了一款智能交友APP。系统基于管理员、普通用户两类角色需求，分别设计了用户模块、活动管理模块、推荐系统模块、管理员模块等功能模块，并重点介绍了实时接收聊天信息、发送聊天信息和推荐算法等核心功能的实现原理。此外，对关键技术的代码逻辑进行了详细讲解，并通过测试验证了系统功能的可行性和稳定性。

本系统利用HarmonyOS的ArkTS语言和MySQL数据库结合Java语言，实现了用户注册与匹配、活动创建与管理、内容审核等高效、安全的社交平台功能。在信息技术快速发展的背景下，系统仍需继续完善，目前的不足和改进的方向如下：

1. 目前该系统主要适用于校园场景，未来还可以扩展到更广泛的社交场景，并且还会添加更多的智能化功能，比如基于深度学习的动态推荐算法。
2. 随着用户规模的增长，需要对数据库查询，实时通信进行优化，应对高并发场景，使用户有更好的体验。
3. 系统虽然已经采用加密传输和异常检测机制，但用户隐私保护仍需要加强，比如增加严格的用户身份验证机制、数据脱敏机制等。

在开发系统的过程中，也遇到了一些技术挑战，如信息接收的实时渲染、推荐算法的精准度等，未来需要加强学习、实践，完善系统功能，提高用户体验，让它成为一个更智能、更安全的社交平台。

# 参考文献

1. 王林超.基于HarmonyOS的混合APP中间件研究与实现[D].西南科技大学,2023.DOI:10.27415/d.cnki.gxngc.2023.001334.
2. 刘小芬.鸿蒙系统架构及应用程序开发研究[J].电脑编程技巧与维护,2021,(12):3-5+12.DOI:10.16184/j.cnki.comprg.2021.12.001
3. 范承宇,李竞择,欧阳迪.基于方舟开发框架的智能装备监控应用研究[J].机电产品开发与创新,2024,37(02):114-117.
4. 张欧亚,曾超,黄涛.基于鸿蒙OS手机与智慧屏协同的居家老人上网辅助系统的设计与实现[J].电脑知识与技术,2024,20(15):102-105+112. DOI:10.14004/j.cnki.ckt.2024.0780.
5. 李勤.网页设计中的CSS技术应用[J].电子技术,2024,53(05):58-59.
6. 莫文东,李呈彩,梁彬,等.基于SpringBoot技术的个人博客平台的研究与应用[J]. 科技风,2024(14): 94-96. DOI:10.19392/j.cnki.1671-7341.202414032.
7. 马春晓,叶青,吕明.志愿活动管理系统的设计与实现[J].工业控制计算机,2022,35(01):135-136+139.
8. 贾靖仪.大数据技术与MySQL数据库的应用[J].集成电路应用,2024,41(08):78-79.DOI:10.19339/j.issn.1674-2583.2024.08.029.
9. 吴杰,孙振勇,韩松涛.WebSocket在服务端和客户端之间的应用[J].起重运输机械,2024(13):100-104.
10. 李斌.基于深度学习的个性化推荐模型研究及应用[D].临沂大学,2024.DOI:10.44252/d.cnki.glydx.2024.000020.

# 致谢

在本论文的撰写与系统开发过程中，我深知个人的力量是有限的，能够顺利完成这一课题，离不开许多人的支持与帮助。

首先，我要衷心感谢我的导师郑光远老师。在整个毕业设计期间，郑老师不仅在专业知识方面给予我悉心指导，更在思维方法和项目规范上给予我极大帮助。每次讨论中，老师都能耐心指出我存在的问题，并引导我找到解决思路，使我受益匪浅。

其次，感谢信息技术学院的各位教师，是你们扎实而系统的教学让我夯实了基础知识，也为本项目的实现提供了理论支撑。

同时，也感谢我的同学和朋友们，尤其是在测试和调试阶段给予我积极反馈的人，是你们的陪伴和鼓励让我在遇到困难时不至于退缩，反而更加坚定前行的信心。

最后，我要感谢我的家人一直以来的理解与支持。是你们给予我无条件的鼓励和关心，让我得以在学习与生活中保持平衡，安心完成毕业设计。

再次对所有给予我帮助的人致以诚挚谢意！