

《数据库系统原理》大作业

系统设计报告

题目名称：一研为定—服务于北航学生的考研指南网站

学号及姓名：__21373035__金楷茗__

__21373233__王怡萱__

__21373075__应卓航__

2023 年 12 月 14 日

组内同学承担任务说明

学生姓名	工作内容			工作量 占比
	子任务 1: 系统功能设计与数据库设计	子任务 2: 系统服务器端开发	子任务 3: 系统客户端开发	
金楷茗	想法讨论 需求分析 构造数据表	后端架构设计与实现	前端架构实现 前后端代码测试	40
王怡萱	想法讨论 需求分析 构造数据表	数据库连接 数据库存储实现 后端测试	前端程序架构设计 前后端代码测试	35
应卓航	想法讨论 需求分析	后端测试	前端架构实现 前后端代码测试 前端美化	25

一. 需求分析

1. 需求描述

1.1 概述

随着当下大学生越来越意识到攻读研究生的重要性，考研已经成为了时的一个热门话题。根据教育部官方数据，自 2015 年以来，我国硕士研究生报名人数屡创新高，2019 年达到 290 万，2020 年首次突破 300 万，而在刚刚过去的 2022 年，报考人数更是达到 457 万，再度创下历史新高。

面对持续不退的考研热度，每年也同样有很多北航的学生在积极地备战研究生入学考试。然而考研与大家都很熟悉的高考不同，考研的流程、数据等都不似高考那样清晰易懂、透明公开，并且在大学也不会像高中一样有老师引领着考研。我们关注到，其实有很多学生虽然定下了考研的目标，但并不知道考研的具体流程，也不知道该如何选择适合自己的学校、专业。

基于如上背景，我们决定设计出一个针对于北航同学的考研指南网站，旨在帮助同学们更方便地获取信息，制定适宜的备考计划。

基于上述的需求分析，可以将用户分为三个角色，并且对各个角色进行单独的分析，见下表。

备考学生	上岸学生	管理员
获取学校数据	分享考研经验	修改学校信息
向上岸学生获取资讯	回答备考学生问题	更新录取数据
指定备考计划	分享考研资料	管理用户与论坛

1.2 功能需求

1.2.1 用户功能

- 用户注册，输入学号、用户名、密码等。如果信息合法则注册成功，反之报错
- 用户登录
- 展示个人信息，其中包括：头像、个人简介、所属学院、年级排名等
- 在主页获取自己的信息，并能修改头像，密码等信息。
- 根据需求检索学校、专业与对应的信息
- 收藏、点赞、评论帖子
- 根据帖子的 tag 搜索帖子

1.2.2 管理员功能

- 管理学校信息，如每年考研结束后更新录取信息等
- 管理普通用户，对于违法的用户进行删号处理
- 管理帖子，具有删帖功能
- 管理 tag，可以增加删除 tag，也可以修改帖子的 tag

1.2.3 交流贴功能

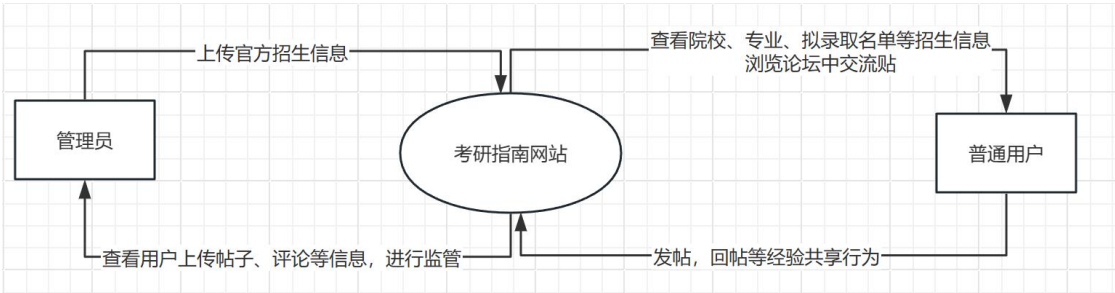
- 创建帖子
- 具有标题、内容、图片、tag、评论区等属性
- 具有评论功能
- 帖子创建人有权力修改帖子
- 帖子应当记录发帖时间、被点赞数量等数据

1.2.4 通知功能

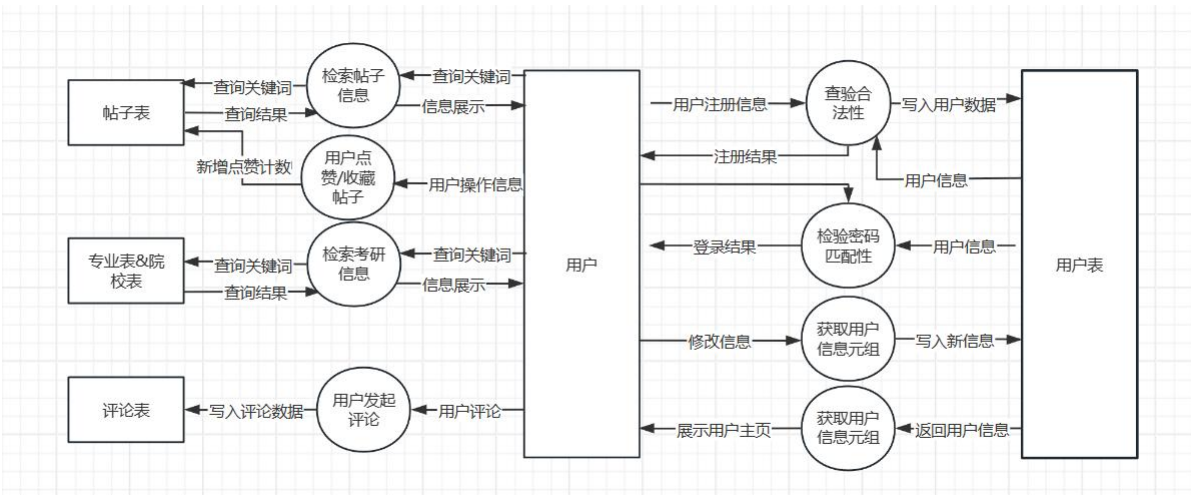
- 用户发的帖子被点赞、评论时，需要收到通知

2. 数据流图

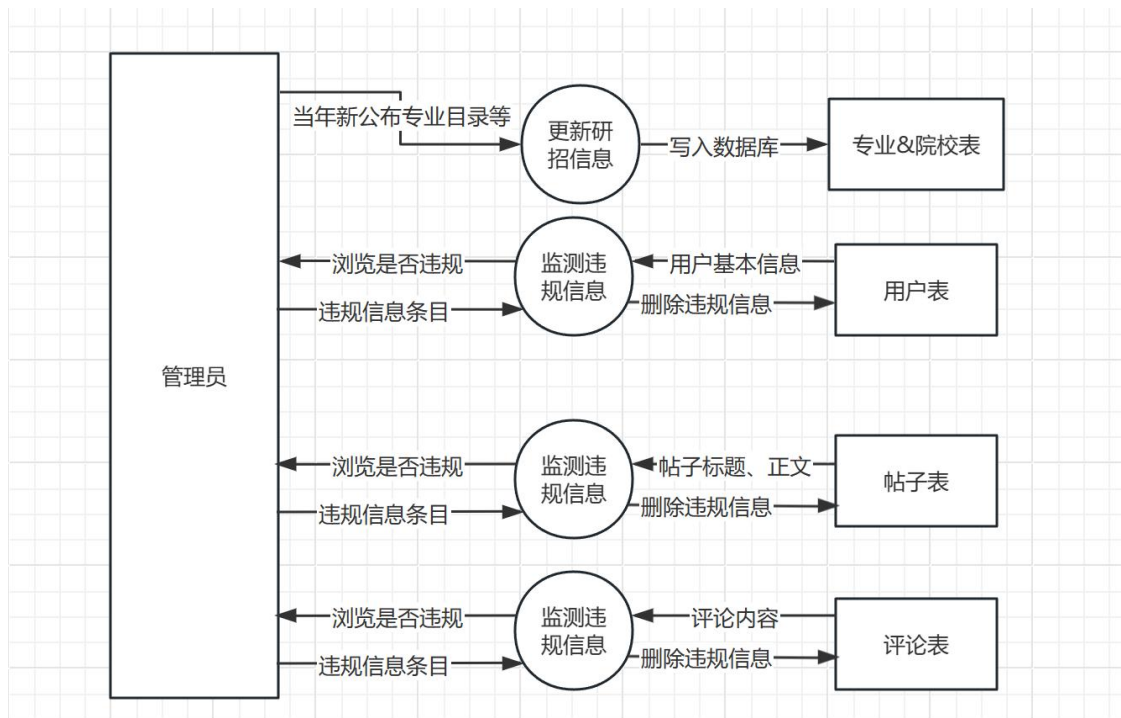
2.1 顶层数据流图



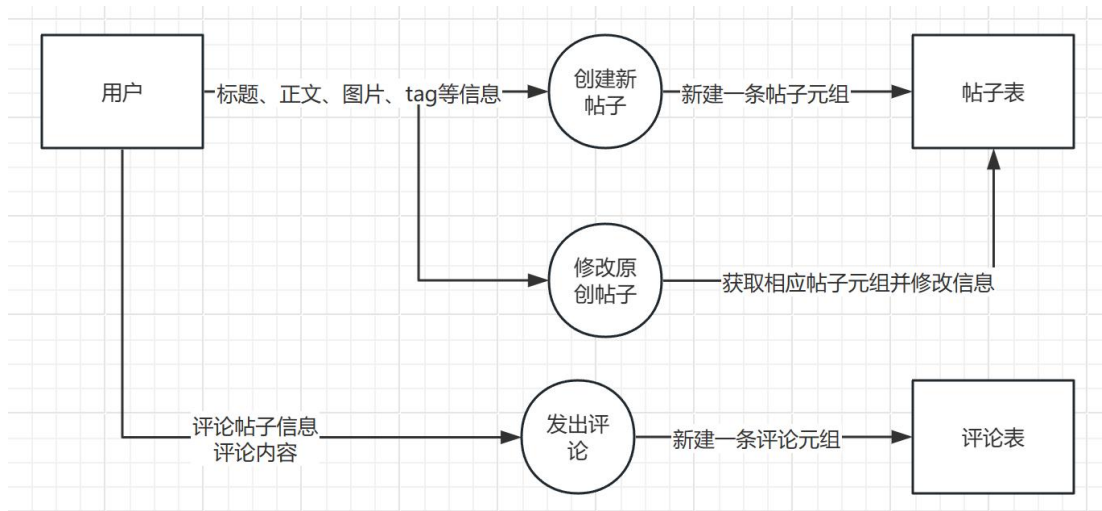
2.2 用户功能数据流图



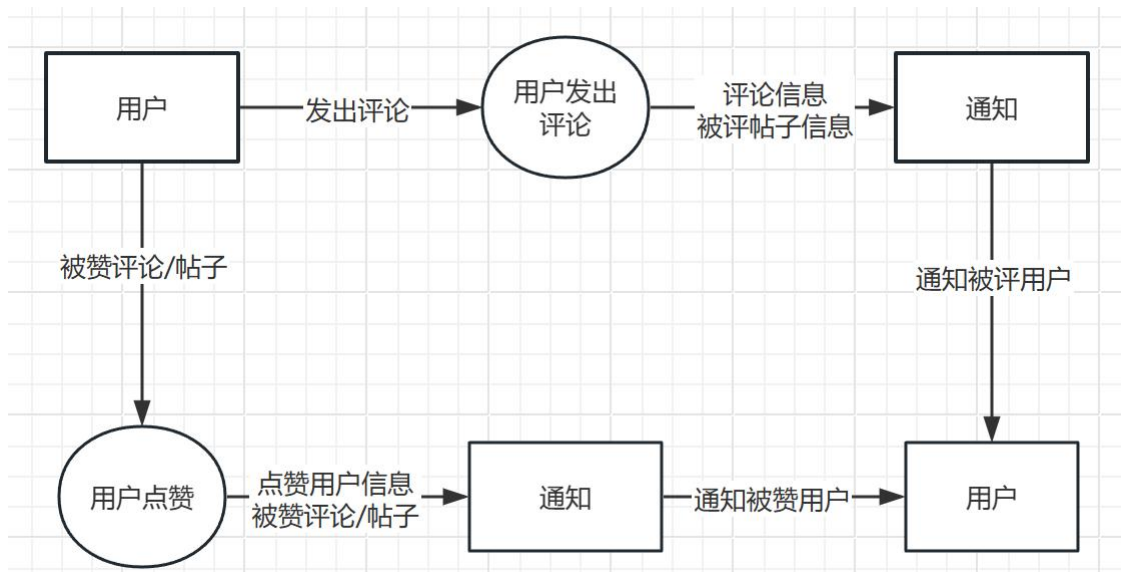
2.3 管理员功能数据流图



2.4 交流贴功能数据流图



2.5 通知功能数据流图



3. 数据元素表

3.1 用户表

属性名	含义	数据类型	备注
id	用户唯一 ID	int	主键
name	用户名	char	非空
password	密码	char	非空
major	本科专业	smallinteger	非空
image	头像	image	
introduction	简介	char	
gender	性别	smallinteger	非空
age	年龄	integer	非空

3.2 专业表

属性名	含义	数据类型	备注
id	专业代码	char	主键
name	专业名称	char	非空
campus	所属学院	smallinteger	
type	专业类型	char	
english_exam	外语科目	char	非空
politic_exam	政治科目	char	非空
professional_exam1	专业课一	char	非空
professional_exam2	专业课二	char	非空

3.3 学校表

属性名	含义	数据类型	备注
id	院校代码	int	主键
name	学校名称	char	非空

location	位置	char	非空
contact	联系方式	char	
label	标签	char	

3.4 录取信息表

属性名	含义	数据类型	备注
id	信息 ID	int	主键
total_score	总分	decimal	非空
score1	初试分数	integer	非空
score2	复试分数	decimal	非空
rank	排名	smallinteger	非空
year	录取年份	int	非空

3.5 帖子表

属性名	含义	数据类型	备注
id	帖子 ID	int	主键
title	标题	char	非空
create_time	创建时间	date	非空
content	正文	char	
tag	关键词	char	

3.6 评论表

属性名	含义	数据类型	备注
id	评论 id	int	主键
content	评论内容	char	非空
create_time	评论创建时间	date	非空

3.7 通知表

属性名	含义	数据类型	备注
id	通知 id	char	主键
message	标题	char	非空
content	内容	char	非空
time	时间	date	非空

3.8 帖子点赞动作

属性名	含义	数据类型	备注
id	点赞 ID	int	主键
user_id	点赞用户 ID	int	外键
post_id	被赞贴 ID	int	外键
time	点赞时间	date	非空

3.9 用户发出帖子

属性名	含义	数据类型	备注
-----	----	------	----

post_id	帖子 ID	int	主键
author	作者 ID	int	外键

3.10 用户发出评论

属性名	含义	数据类型	备注
comment_id	评论 ID	int	主键
author	用户 ID	int	外键

3.11 评论发起通知

属性名	含义	数据类型	备注
comment_id	评论 ID	int	主键
notification_id	通知 ID	int	外键

3.12 通知提醒用户

属性名	含义	数据类型	备注
notification_id	通知 ID	int	主键
owner_id	被评用户 ID	int	外键

3.13 帖子收到评论

属性名	含义	数据类型	备注
comment_id	评论 ID	int	主键
post_id	帖子 ID	int	外键

3.15 专业对应学校

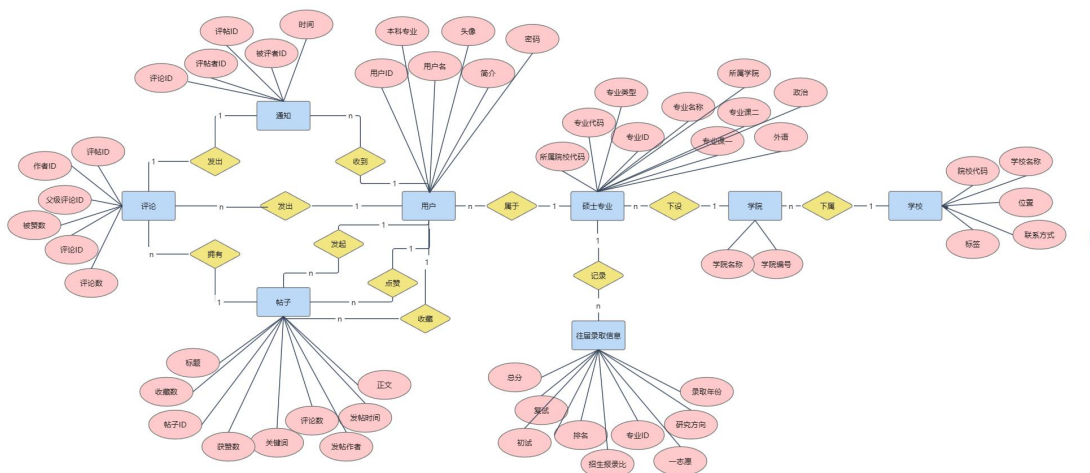
属性名	含义	数据类型	备注
id	专业 id	int	主键
major_id	专业代码	int	外键
university_id	学校代码	int	外键

3.15 专业录取信息

属性名	含义	数据类型	备注
id	录取信息 ID	int	主键
major_id	专业 id	int	外键

二. 数据库概念模式设计

1. 系统初步 E-R 图

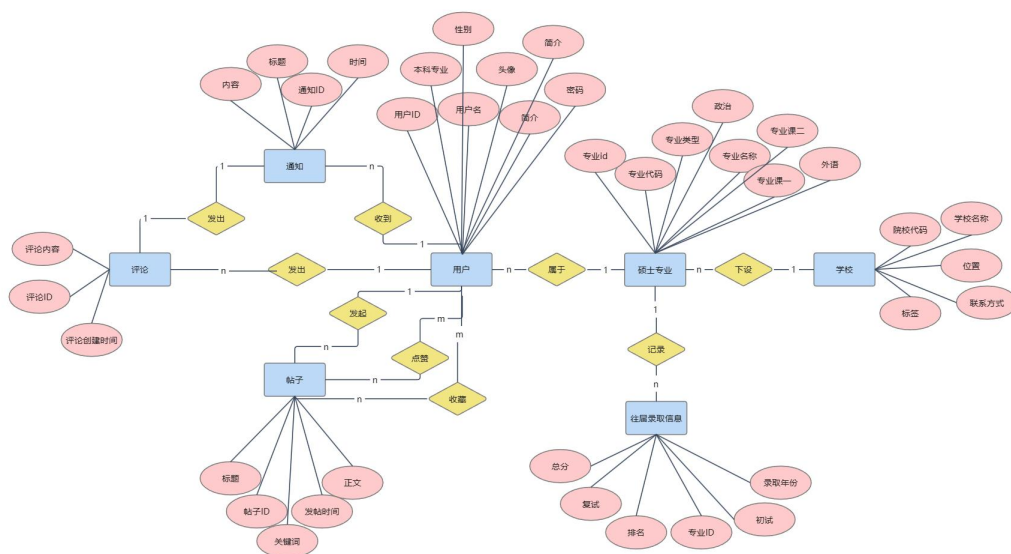


2. 系统基本 E-R 图

对初步 e-r 图中实体属性进行完善，如添加了用户的性别，年龄等属性以保证用户定义完整性。

消除初步 e-r 图中的冲突，主要解决结构冲突，将各实体中的外部属性替换为与对应实体的联系。如通知。

消除冗余联系，如帖子拥有的评论，可以通过帖子—用户—通知—评论间接获得



三、数据库逻辑模式设计

1. 数据库关系模式

1.1 实体

1.1.1 用户

$U = \{\text{用户 ID}, \text{用户名}, \text{密码}, \text{本科专业}, \text{头像}, \text{简介}, \text{性别}, \text{年龄}\}$

$D = \{\text{Str}, \text{Int}\}$

$\text{DOM} = \{\text{DOM}(\text{用户 ID}) = \text{DOM}(\text{本科专业}) = \text{DOM}(\text{性别}) = \text{DOM}(\text{年龄}) = \text{Int}, \text{DOM}(\text{用户名}) = \text{DOM}(\text{用户密码}) = \text{DOM}(\text{简介}) = \text{DOM}(\text{头像}) = \text{Str}\}$

$F = \{\text{用户 ID} \rightarrow \text{用户名}, \text{用户 ID} \rightarrow \text{密码}, \text{用户 ID} \rightarrow \text{本科专业}, \text{用户 ID} \rightarrow \text{头像}, \text{用户 ID} \rightarrow \text{简介}, \text{用户 ID} \rightarrow \text{性别}, \text{用户 ID} \rightarrow \text{年龄}\}$

1.1.2 研究生专业

$U = \{\text{专业 ID}, \text{专业代码}, \text{专业名称}, \text{学院名称}, \text{专业类型}, \text{外语}, \text{政治}, \text{专业课 1}, \text{专业课 2}\}$

$D = \{\text{Str}, \text{Int}, \text{Bool}\}$

$\text{DOM} = \{\text{DOM}(\text{专业 ID}) = \text{DOM}(\text{专业代码}) = \text{Int}, \text{DOM}(\text{专业类型}) = \text{Bool}, \text{DOM}(\text{专业名称}) = \text{DOM}(\text{学院名称}) = \text{DOM}(\text{外语}) = \text{DOM}(\text{政治}) = \text{DOM}(\text{专业课 1}) = \text{DOM}(\text{专业课 2}) = \text{Str}\}$

$F = \{\text{专业 ID} \rightarrow \text{专业代码}, \text{专业 ID} \rightarrow \text{专业类型}, \text{专业 ID} \rightarrow \text{学院名称}, \text{专业 ID} \rightarrow \text{英语}, \text{专业 ID} \rightarrow \text{政治}, \text{专业 ID} \rightarrow \text{专业课 1}, \text{专业 ID} \rightarrow \text{专业课 2}\}$

1.1.3 学校

$U = \{\text{院校代码}, \text{学校名称}, \text{位置}, \text{联系方式}, \text{标签}\}$

$D = \{\text{Str}, \text{Int}\}$

$\text{DOM} = \{\text{DOM}(\text{院校代码}) = \text{Int}, \text{DOM}(\text{学校名称}) = \text{Str}, \text{DOM}(\text{位置}) = \text{Str}, \text{DOM}(\text{联系方式}) = \text{Str}, \text{DOM}(\text{标签}) = \text{Str}\}$

$F = \{\text{院校代码} \rightarrow \text{学校名称}, \text{院校代码} \rightarrow \text{位置}, \text{院校代码} \rightarrow \text{联系方式}, \text{院校代码} \rightarrow \text{标签}\}$

1.1.4 往届录取信息

$U = \{\text{ID}, \text{总分}, \text{初试}, \text{复试}, \text{排名}, \text{录取年份}\}$

$D = \{\text{Str}, \text{Int}, \text{Float}\}$

$\text{DOM} = \{\text{DOM}(\text{ID}) = \text{DOM}(\text{总分}) = \text{DOM}(\text{初试}) = \text{DOM}(\text{复试}) = \text{DOM}(\text{录取年份}) = \text{Int}\}$

$F = \{\text{ID} \rightarrow \text{总分}, \text{ID} \rightarrow \text{初试}, \text{ID} \rightarrow \text{复试}, \text{ID} \rightarrow \text{排名}, \text{ID} \rightarrow \text{录取年份}\}$

1.1.5 帖子

$U = \{\text{帖子 ID}, \text{标题}, \text{关键词}, \text{发帖时间}, \text{正文}\}$

$D = \{\text{Str}, \text{Int}\}$

$DOM = \{DOM(\text{帖子 ID}) = \text{Int}, DOM(\text{标题}) = DOM(\text{关键词}) = DOM(\text{正文}) = DOM(\text{发帖时间}) = \text{Str}\}$

$F = \{\text{帖子 ID} \rightarrow \text{标题}, \text{帖子 ID} \rightarrow \text{关键词}, \text{帖子 ID} \rightarrow \text{发帖时间}, \text{帖子 ID} \rightarrow \text{正文}\}$

1.1.6 评论

$U = \{\text{评论 ID}, \text{评论内容}, \text{评论创建时间}\}$

$D = \{\text{Str}, \text{Int}, \text{Date}\}$

$DOM = \{DOM(\text{评论 ID}) = \text{Int}, DOM(\text{评论内容}) = \text{Str}, DOM(\text{评论 ID}) = \text{Date}\}$

$F = \{\text{评论 ID} \rightarrow \text{评论内容}, \text{评论 ID} \rightarrow \text{评论创建时间}\}$

1.1.7 通知

$U = \{\text{通知 ID}, \text{标题}, \text{内容}, \text{时间}\}$

$D = \{\text{Str}, \text{Int}, \text{Date}\}$

$DOM = \{DOM(\text{通知 ID}) = \text{Int}, DOM(\text{标题}) = DOM(\text{内容}) = \text{Str}, DOM(\text{时间}) = \text{Date}\}$

$F = \{\text{通知 ID} \rightarrow \text{标题}, \text{通知 ID} \rightarrow \text{内容}, \text{时间}, \text{通知 ID} \rightarrow \text{时间}\}$

1.2 联系

1.2.1 多对多联系

- 帖子点赞动作

$U = \{\text{点赞 ID}, \text{帖子 ID}, \text{用户 ID}\}$

$D = \{\text{Int}\}$

$DOM = \{DOM\{\text{点赞 ID}\} = DOM\{\text{帖子 ID}\} = DOM\{\text{用户 ID}\} = \text{Int}\}$

1.2.2 一对多联系

- 用户发出帖子

$U = \{\text{帖子 ID}, \text{用户 ID}\}$

$D = \{\text{Int}\}$

$DOM = \{DOM\{\text{帖子 ID}\} = DOM\{\text{用户 ID}\} = \text{Int}\}$

- 用户发出评论

$U = \{\text{评论 ID}, \text{用户 ID}\}$

$D = \{\text{Int}\}$

$\text{DOM} = \{\text{DOM}\{\text{评论 ID}\} = \text{DOM}\{\text{用户 ID}\} = \text{Int}\}$

- 评论发起通知

$U = \{\text{评论 ID}, \text{通知 ID}\}$

$D = \{\text{Int}\}$

$\text{DOM} = \{\text{DOM}\{\text{评论 ID}\} = \text{DOM}\{\text{通知 ID}\} = \text{Int}\}$

- 通知提醒用户

$U = \{\text{通知 ID}, \text{被评者 ID}\}$

$D = \{\text{Int}\}$

$\text{DOM} = \{\text{DOM}\{\text{通知 ID}\} = \text{DOM}\{\text{被评者 ID}\} = \text{Int}\}$

- 帖子收到评论

$U = \{\text{评论 ID}, \text{帖子 ID}\}$

$D = \{\text{Int}\}$

$\text{DOM} = \{\text{DOM}\{\text{评论 ID}\} = \text{DOM}\{\text{帖子 ID}\} = \text{Int}\}$

- 专业对应学校

$U = \{\text{专业 ID}, \{\text{专业代码}, \text{院校代码}\}\}$

$D = \{\text{Int}\}$

$\text{DOM} = \{\text{DOM}\{\text{专业 ID}\} = \text{DOM}\{\text{专业代码}\} = \text{DOM}\{\text{院校代码}\} = \text{Int}\}$

- 专业录取信息

$U = \{\text{录取信息 ID}, \text{专业 ID}\}$

$D = \{\text{Int}\}$

$\text{DOM} = \{\text{DOM}\{\text{录取信息 ID}\} = \text{DOM}\{\text{专业 ID}\} = \text{Int}\}$

2. 关系模式范式等级的判定与规范化

2.1 实体表

2.1.1 用户

每个用户有唯一的 id 作为主键，无其他候选码，所有函数依赖左部均为用户 id，因此不存在部分依赖，传递依赖，属于 3NF，同时左部均含有码，属于 BCNF

2.1.2 研究生专业

每个专业以统一考研招生专业代码为主键，无其他候选码，所有函数依赖左部均为专业代码，因此不存在部分依赖，传递依赖，属于 3NF，同时左部均含有码，属于 BCNF

2.1.3 学校

每个院校以院校代码作为主键，无其他候选码，所有函数依赖左部均为院校代码，因此不存在部分依赖，传递依赖，属于 3NF，同时左部均含有码，属于 BCNF

2.1.4 往届录取信息

为每条录取信息提供存储自增 id 作为主键，无其他候选码，所有函数依赖左部均为录取信息 id，因此不存在部分依赖，传递依赖，属于 3NF，同时左部均含有码，属于 BCNF

2.1.5 帖子

每个帖子有唯一的创建 id 作为主键，无其他候选码，所有函数依赖左部均为帖子 id，因此不存在部分依赖，传递依赖，属于 3NF，同时左部均含有码，属于 BCNF

2.1.6 评论

每个评论有唯一的创建 id 作为主键，无其他候选码，所有函数依赖左部均为评论 id，因此不存在部分依赖，传递依赖，属于 3NF，同时左部均含有码，属于 BCNF

2.1.7 通知

每个通知有唯一的创建 id 作为主键，无其他候选码，所有函数依赖左部均为通知 id，因此不存在部分依赖，传递依赖，属于 3NF，同时左部均含有码，属于 BCNF

2.2 关系表

实体间一对多联系与多对多建立数据表。

- 对于多对多联系：帖子点赞动作，选取候选码（帖子 ID, 用户 ID）为主键，属于 BCNF；专业-学校关系，选取候选码（学校 ID, 专业 ID）为主键，专业的其他属性均依赖于该主键。

- 对于一对多联系：每个联系均采取联系中“多”的实体 id 作为码，另一实体 id 依赖于码，均属于 2NF，且不存在传递依赖，属于 3NF，关系模式中仅存在一条函数依赖：“多”实体 id \rightarrow “一”实体 id，因此属于 BCNF

3. 数据库设计优化

为提高数据库查询效率，我们做了以下优化

3.1 建立高频索引

对贴子（大学）的标签建立索引，在根据标签进行搜索时有更高的查询效率

3.2 更多外键引入

通过在实体表中引入更多外键，适量提高冗余度提高查询实体关系时的效率，减少表间的关系运算次数。

3.3 优化查询语句

- 尽量使用连表查询而不是嵌套查询；
- 避免将查询语句放在循环内；
- 尽可能的复用查询结果。

3.4 适当合并表格

可以减少连表消耗的代价，适当增加表的冗余程度来换取效率。