
校园帮帮网 数据库设计说明书

软工实践第二小组

2020 年 4 月 12 日

目录

第一章	1
第二章 引言	1
2.1 编写目的	1
2.2 背景	1
2.3 参考资料	1
第三章 外部设计	2
3.1 标识符和状态	2
3.2 命名约定	2
3.3 设计约定	2
第四章 结构设计	4
4.1 概念结构设计	4
4.1.1 实体和属性的定义	4
4.1.2 设计局部 ER 模式	8
4.1.3 设计全局 ER 模式	13
4.2 逻辑结构设计	13
4.2.1 模式	13

第二章 引言

2.1 编写目的

数据库的设计是为了以后编码、测试以及维护阶段的后台数据的存储做准备。应用于系统开发前期，为后期数据库设计指引方向。

预期的读者：系统开发人员、系统测试人员和系统维护人员

2.2 背景

同学们在日常生活中往往会有一些关于学校资讯方面的疑惑（比如选课、学分、绩点、学术方面的问题等等），而百度、知乎等方式很难获取到这些方面信息，而寻找学长学姐帮忙作为同学们的主要途径不仅效率较低，而且对于很多性格较为内向的同学不太合适。在此背景下我们提出了一个专注于校园内容的问答网站——校园帮帮网来解决同学们的需求。本文是校园帮帮网后台存储数据库的设计方案。

2.3 参考资料

软件工程导论（第五版） 张海藩编著 清华大学出版社
数据库系统概论（第四版） 王珊，萨师煊编著 高等教育出版社

第三章 外部设计

3.1 标识符和状态

团队使用的数据库：5.5.62 MySQL

数据库的名称为：team_work

3.2 命名约定

①在校园帮帮网的数据库命名过程中我们要求表名与类图设计中对应实体类的名字相同，字段名也与类中的属性也基本相同，唯一不同的是在数据库命名字段时单词和单词之间采用下划线的方式区分，但在程序类的设计中我们采用的是驼峰命名法。在程序中我们采用驼峰命名法是因为这个比较符合我们小组中大多数人的命名习惯，同时也符合行业的主流命名方式。但在 mysql 数据库中不区分大小写，用驼峰命名法会带来后期的不便，因此我们数据库表字段的命名采用下划线的命名划分单词。

②通过字段的名字可以大致看出字段的数据类型，如创建时间字段使用 `create_time` 可以看出该字段是采用 `Date` 或 `time` 等类型（这里采用的是 `time`），点赞人数则使用 `like_num` 可以直观看出是 `int` 数据类型数据等等。

3.3 设计约定

一：字段的原子性

保证每列的原子性，不可分解，意思表达要清楚，不能含糊，高度概括字段的含义，能用一个字段表达清楚的绝不使用第二个字段，必须要使用两个字段表达清楚的绝不能使用一个字段，通过满足字段的原子性从而满足数据库设计的第一范式。

二：主键设计

主键尽可能不与业务逻辑有所关联，最好是毫无意义的一串独立不重复的数字并且自增的整型数据；如在 `user` 表中不采用 `account` 作为用户表的主键，而是单独去建立一个 `id` 字段作为 `user` 表的主键

三：关于外键

尽量不建立外键，保证每个表的独立性。如果非得保持一定的关系，最好是通过 `id` 进行关联

四：关于数据量较大，且较少修改的字段尽可能单独设立表

原因以问题表为例，mysql 在查询数据，走聚簇索引遍历时，假如内容在文章表，指针要移动的位置就很长，导致性能低下，同时由于这部分字段的内容较少改变，因此将内容单独存放一个表，在 question 表中用外键关联可以提高查询更新效率

第四章 结构设计

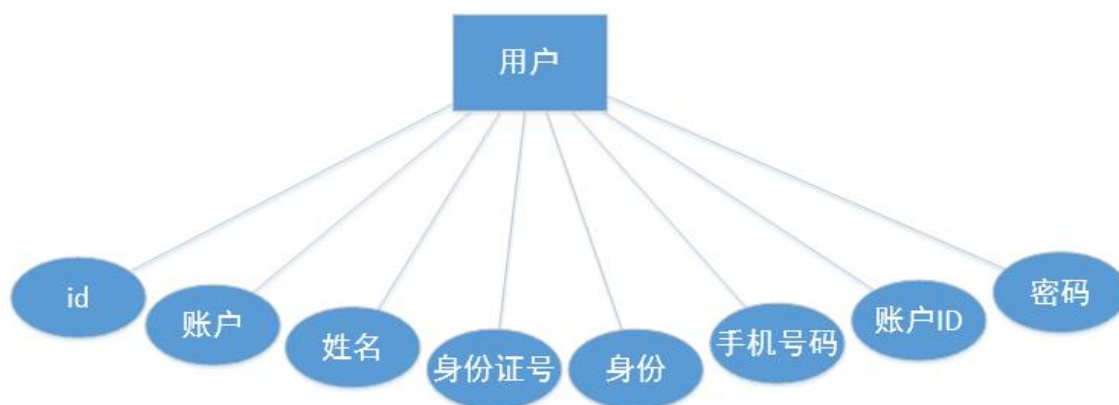
4.1 概念结构设计

4.1.1 实体和属性的定义

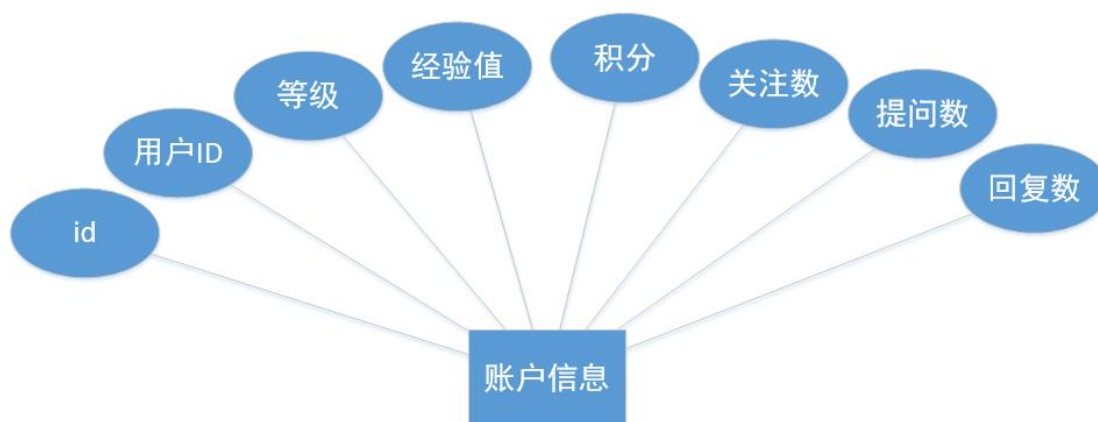
用户实体：(id 编号，账号密码，姓名，身份证号，身份，电话号码，账户信息 ID)

属性约束如下：

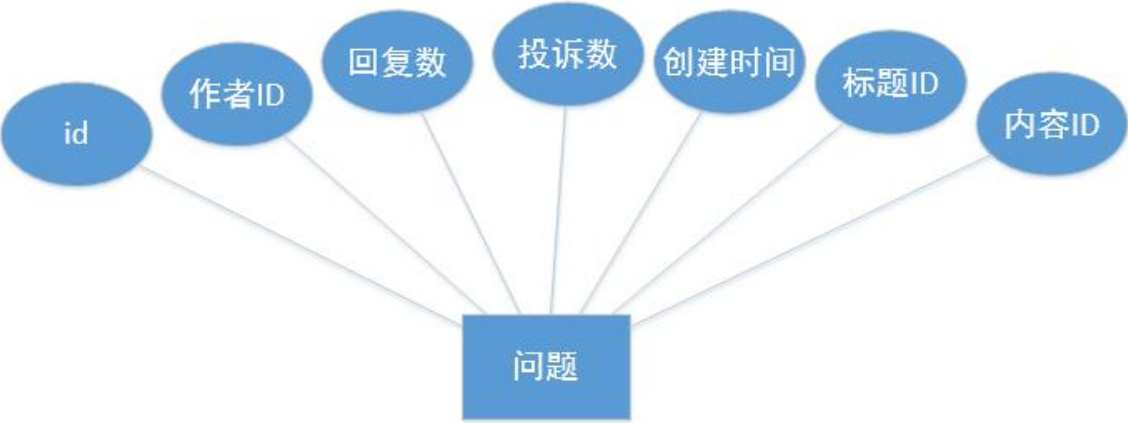
身份：student，teacher，admin



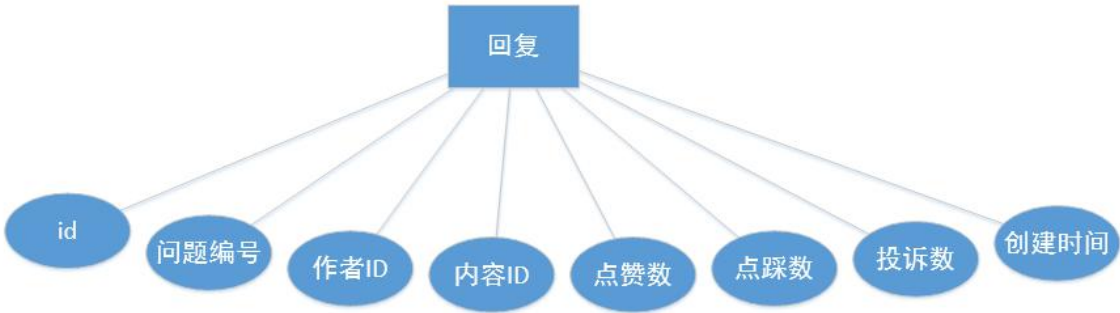
账户信息：(id 编号，用户 id，等级，经验值，积分，关注数，提问数，回复数)



问题实体：（id 编号，作者 id，回复数，投诉数，创建时间，标题 id，内容 id）



回复实体：（id 编号，问题 id，作者 id，内容 id，点赞数，点灭数，投诉数，创建时间）



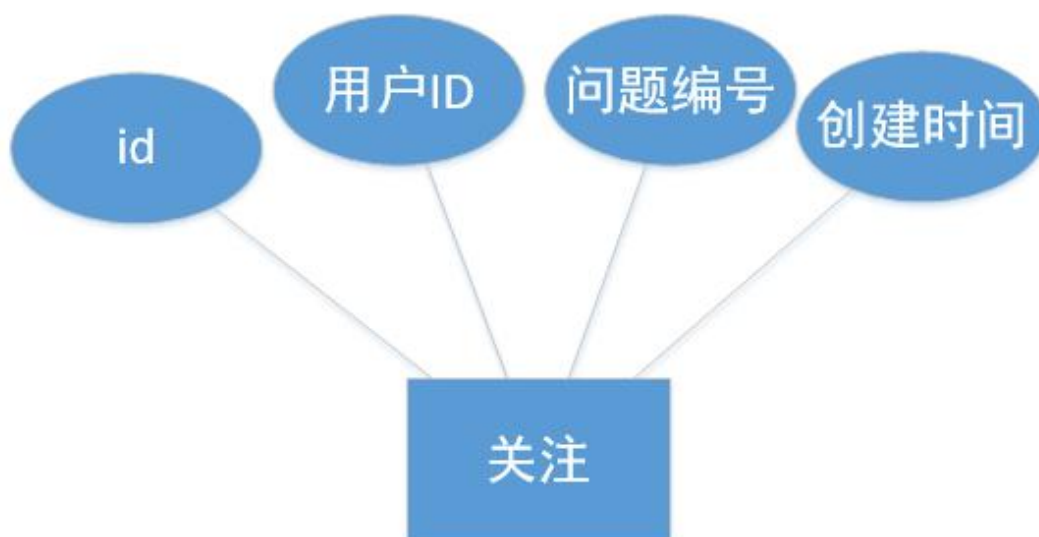
标题表：（id 编号，问题 id，标题内容）



内容表：（id 编号，内容）



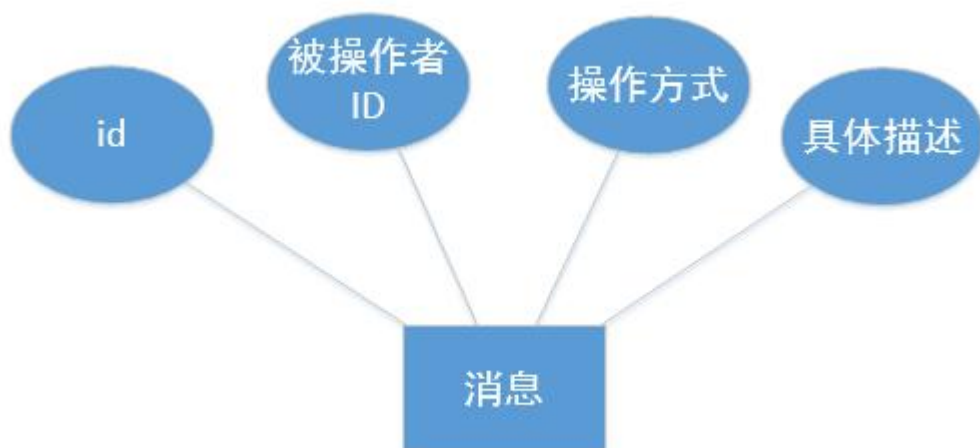
关注实体：（id 编号，关注用户 id，问题 id，关注时间）



消息实体：（id 编号，操作对象用户 id，操作方式，消息描述信息）

属性约束如下：

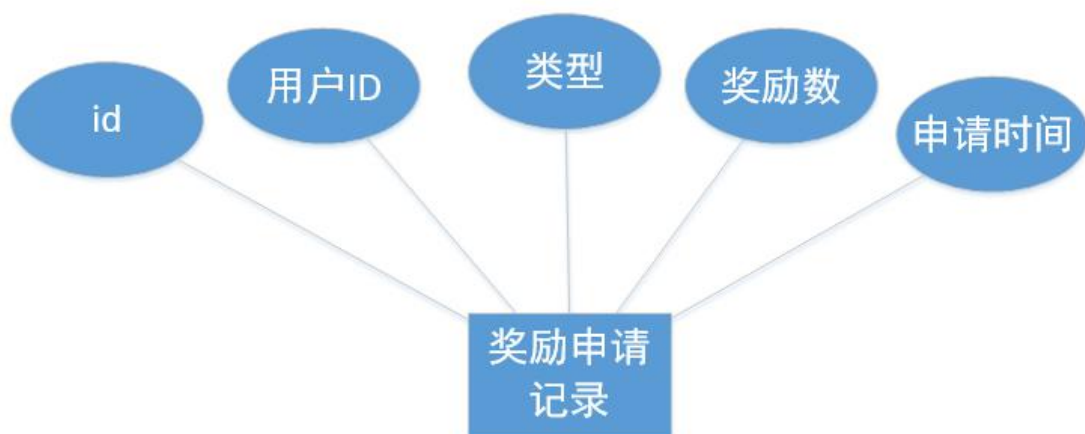
操作方式：like(点赞) dislike(踩灭) focus（关注） delete（删除） report（投诉）
response（回复）



奖励申请记录：（id 编号，申请用户 id，申请类别，申请数量，申请时间）

属性约束如下：

申请类别：SyntheticTest（综测） ServiceTime（时长）



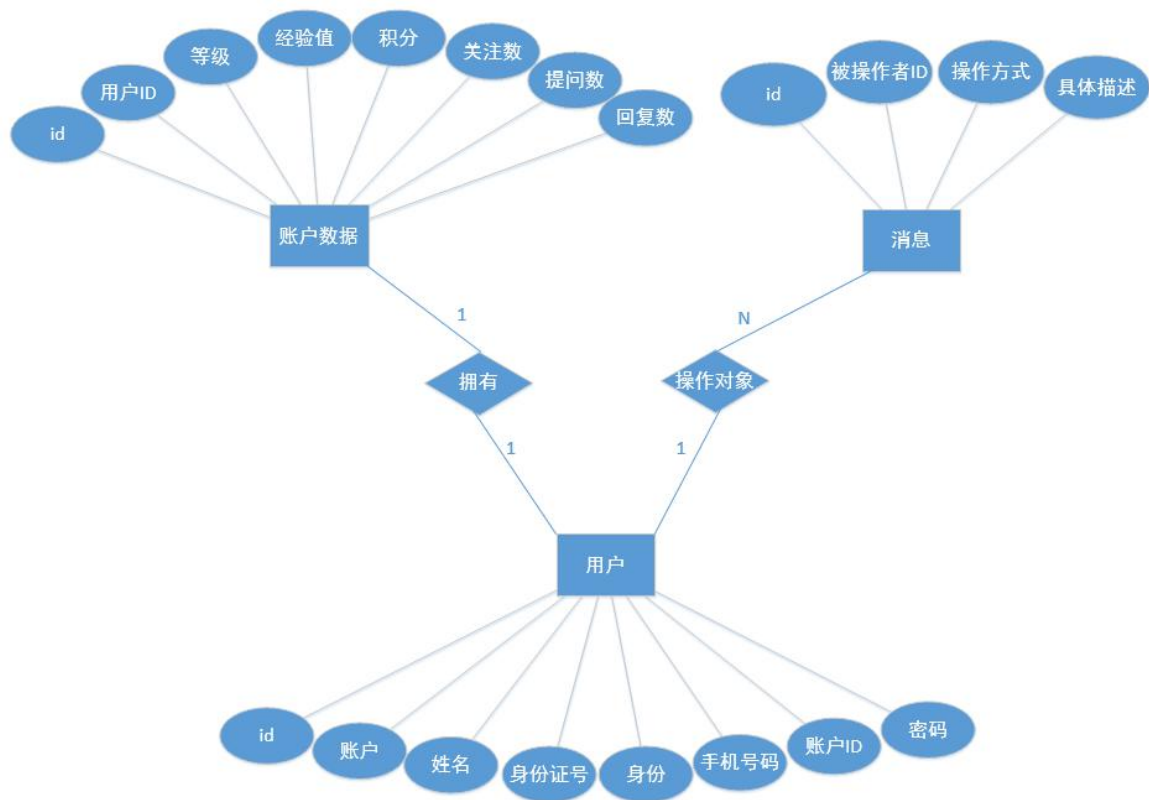
临时板块实体：（id 编号，临时板块名，关键字）



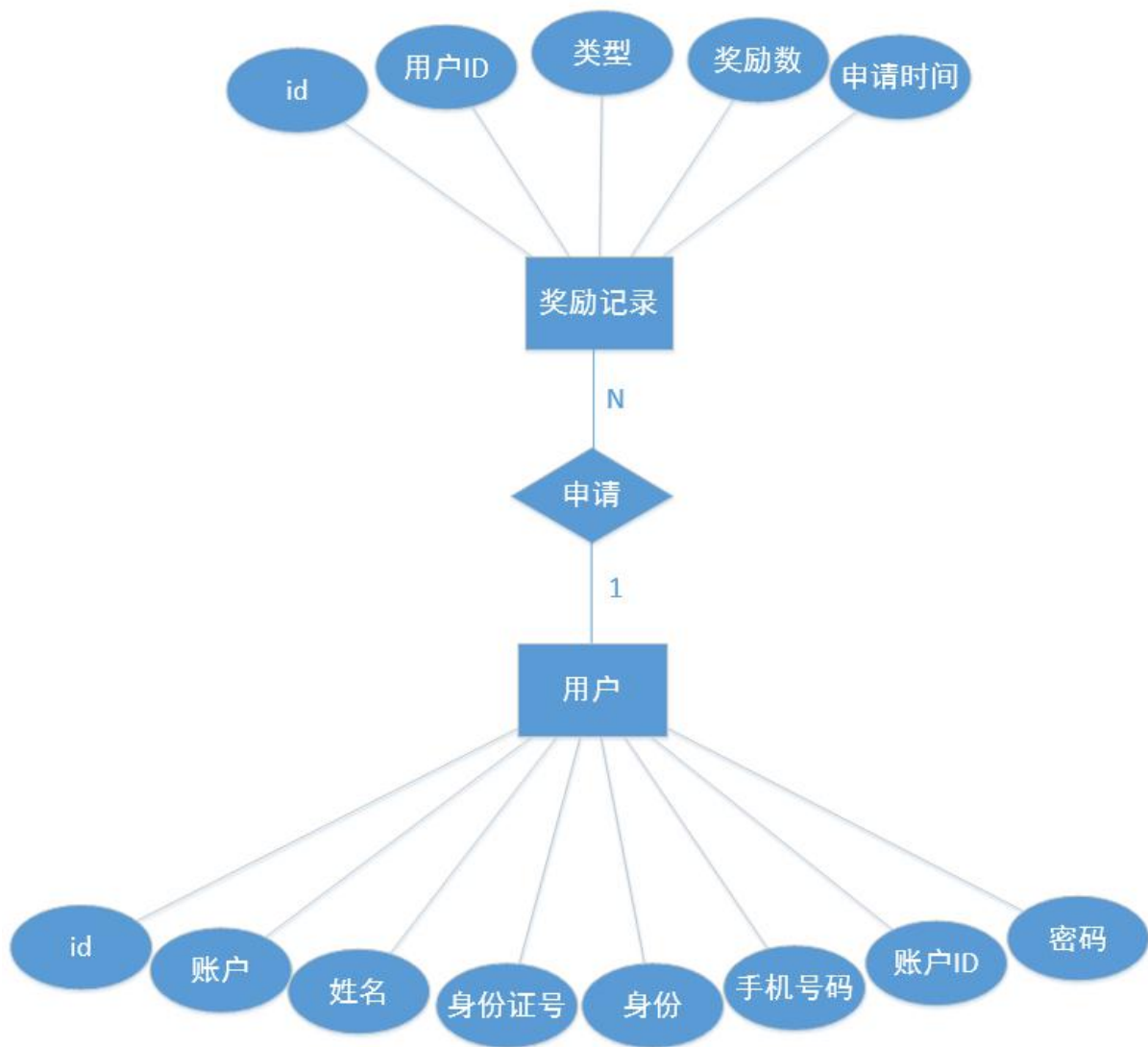
4.1.2 设计局部 ER 模式

用户信息局部 ER 图

我们将账户数据（如积分，经验值等）从用户表中剥离出来，主要是为了使得表所对应的实体类能更清晰，不包含一些不必要的信息（如管理员用户不需要积分，经验值等账户数据）；所对应的代码设计是将账户数据作为一个类成员对象作为用户类的属性，当不需要该属性时，该属性可以为空。在表中二者体现为拥有关系。在数据库中存放的消息与用户之间的关联主要是通过消息表中的被操作者 id，用户作为表中的操作对象，用于前台页面的消息展示。



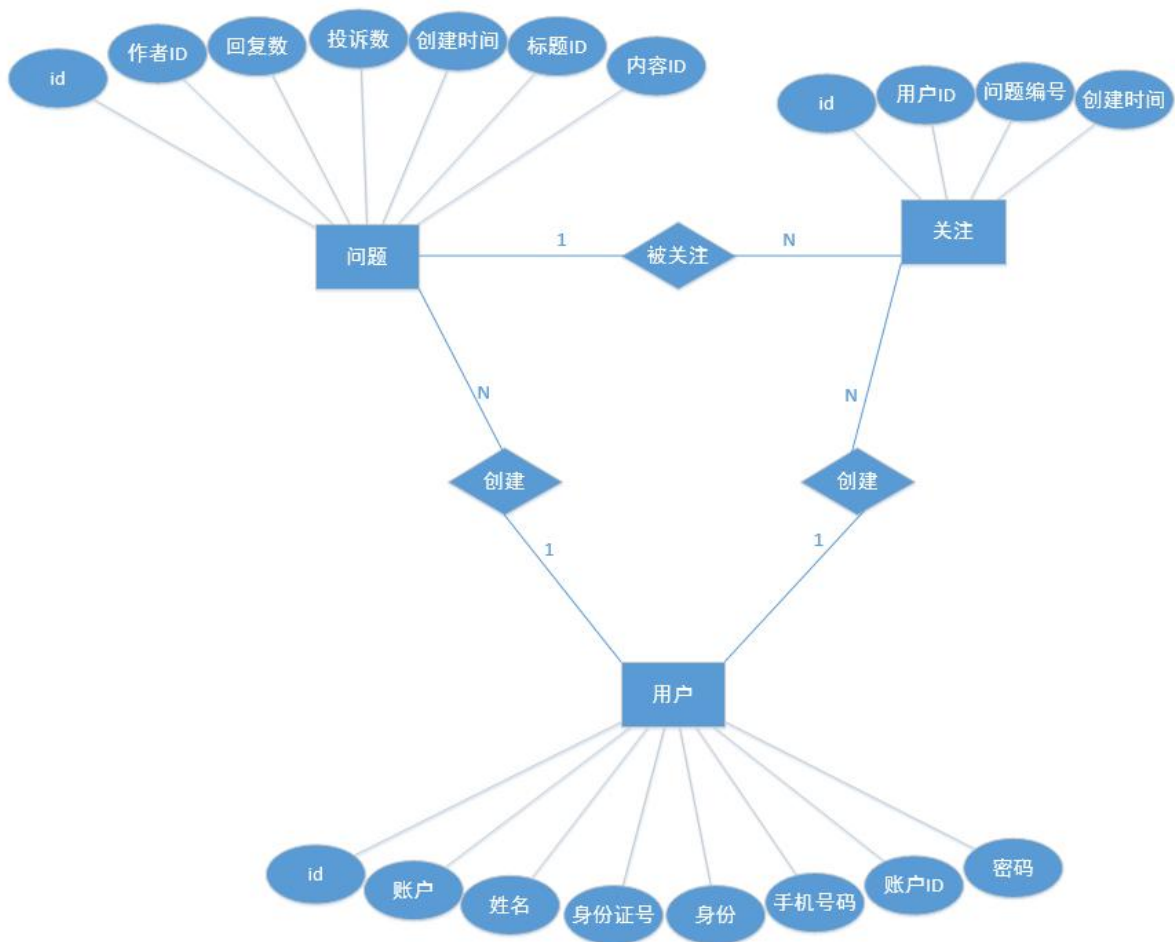
在表的设计中，用户和奖励记录建的关系是申请关系，即用户申请后创建的奖励记录，二者通过奖励记录中的用户 id 外键关联，一个用户可以有多个奖励记录，因此二者之间是一对多的关系。



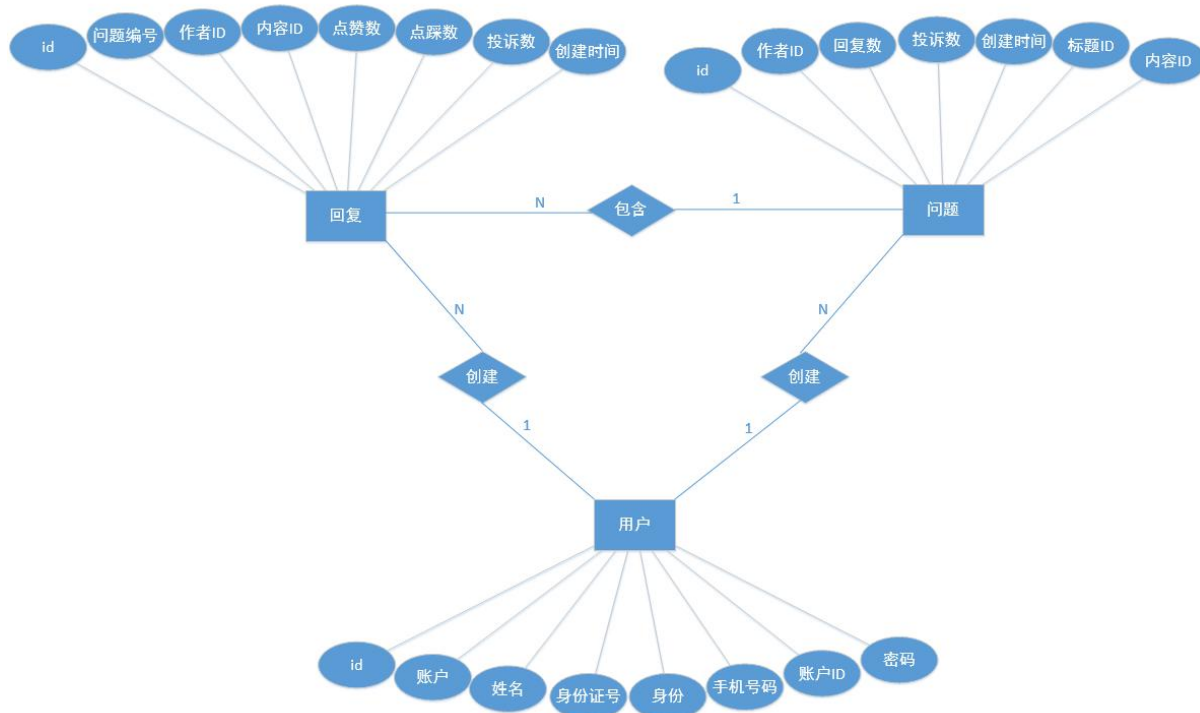
用户和问题之间主要有两种方式关联：

一是通过问题的作者 id 外键关联，体现为创建关系，是用户创建问题，因为一个用户可以创建多个问题，因此二者间体现的是一对多的关系。

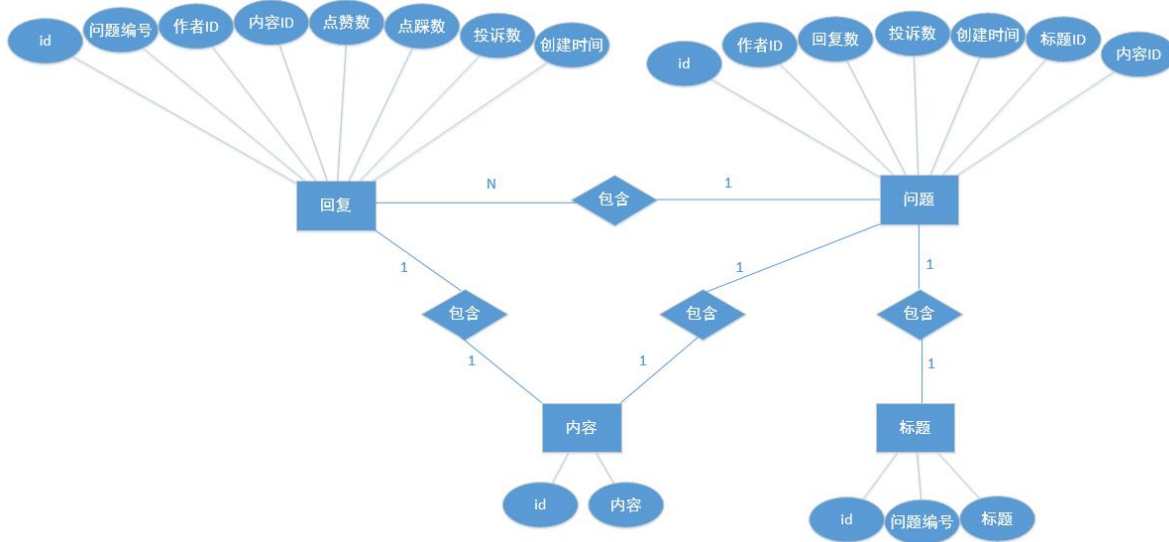
二是通过关注，用户通过关注问题可以与问题产生关联，这里因为别人的问题和用户之间没有直接字段上的关联，因此我们在中间加入了一个关注表，用于记录用户和问题之间的关注信息，用户和关注间体现 1 对多的关系（一个用户可以关注多个问题），问题和关注之间也是一对多的关系（一个问题可以被多个用户关注）



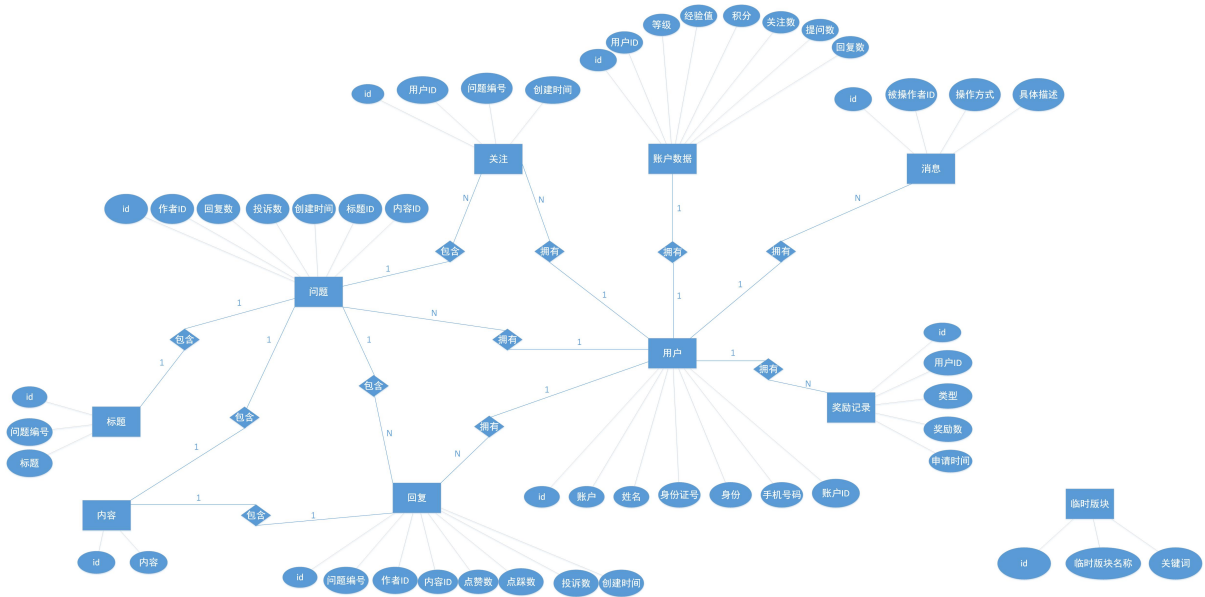
用户和问题，回复之间主要是通过实体中的作者 id 作为外键关联，体现的是创建关系，回复与问题之间通过回复表的问题编号产生多对一的关联，体现一个问题可以有多个回复。



在问题和回复中都有一些数据较为庞大的字段——内容/标题，该部分字段不仅数据较大，而且很少更新（主要更新的是点赞数，评论数等数据），根据设计约定第四条，我们将标题和内容从表中抽离，单独形成一张表，通过回复/问题的外键相关联，提高表的查询，更新效率。



4.1.3 设计全局 ER 模式



4.2 逻辑结构设计

4.2.1 模式

用户表：存储用户基本个人信息

表 User 的结构:

字段名	数据类型	长度	主键	非空	描述
ID	int	11	是	否	自增 ID
account	varchar	20		否	账号，唯一标识 id
password	varchar	50			密码
name	varchar	50			用户真实姓名
id_card	varchar	20			身份证号
identity	varchar	10			身份
phone_num	varchar	12			电话号码
account_data_id	varchar	int			账户信息 id（外键）

账户表存储账户相关信息

表 account_data 的结构:

字段名	数据类型	长度	主键	非空	描述
ID	int	10	是	否	自增 ID
user_id	int	11		否	用户 id（外键）
level	int	11			等级
experience_value	int	11			经验值
score	int	11			积分
focus_num	int	11			关注问题数
question_num	int	11			提问题数
response_num	int	11			回复数

问题表存储问题相关数据

表 question 的结构:

字段名	数据类型	长度	主键	非空	描述
ID	int	10	是	否	自增 ID
author_id	int	11		否	作者用户 id(外键)
response_num	int	11			问题回复数
report_num	int	11			投诉数
create_time	time				创建时间 (yyyy-MM-dd hh:mm:ss)
title_id	int	11			标题 id(外键)
content_id	int	11			内容 id(外键)

标题表用于单独存储问题标题

表 title 的结构:

字段名	数据类型	长度	主键	非空	描述
ID	int	11	是	否	自增 ID
question_id	int	11		否	标题对应问题 id（外键）
title	varchar	100			标题内容

内容表用于单独存储问题或回复的内容

表 content 的结构:

字段名	数据类型	长度	主键	非空	描述
-----	------	----	----	----	----

ID	int	11	是	否	自增 ID
content	varchar	1000		否	问题内容

回复表用于存储用户创建的回复

表 response 的结构:

字段名	数据类型	长度	主键	非空	描述
ID	int	11	是	否	自增 ID
question_id	int	11			问题 id(外键)
author_id	int	11			回复作者 id (外键)
content_id	int	11			内容 id(外键)
like_num	int	11			点赞人数
dislike_num	int	11			点灭人数
report_num	int	11			投诉人数
create_time	time				创建时间 (yyyy-MM-dd hh:mm:ss)

关注表用于存储用户的关注信息

表 attention 的结构:

字段名	数据类型	长度	主键	非空	描述
ID	int	11	是	否	自增 ID
user_id	int	11			关注用户 id(外键)
question_id	int	11			关注问题 id(外键)
create_time	time				创建时间 yyyy-MM-dd hh:mm:ss

消息表用于记录被操作对象的消息，用于后期通知

表 message 的结构:

字段名	数据类型	长度	主键	非空	描述
ID	int	11	是	否	自增 ID
object_id	int	11			被操作对象 id(外键)
way	varchar	10			操作方式
description	varchar	255			消息描述信息

奖励申请记录表用于存储奖励申请记录

表 reward 的结构:

字段名	数据类型	长度	主键	非空	描述
ID	int	11	是	否	自增 ID
user_id	int	11		否	申请用户 id(外键)
type	varchar	10			奖励申请类别 (综测、时长)
reward_num	double				申请数量
apply_time	time				申请时间 yyyy-MM-dd hh:mm:ss

临时模块表用于存储临时模块信息

表 block 的结构:

字段名	数据类型	长度	主键	非空	描述
ID	int	11	是	否	自增 ID
block_name	varchar	20			临时板块名
key_word	varchar	100			关键字
