**软件工程实践**

**《数据库设计说明书》**

**项目名称：七日记忆**

**团队成员：221701123 洪楷滨（组长）**

**221701101 林露**

**221701108 李波**

**221701120 黄筱宇**

**221701122 黄毅**

**221701133 林海峰**

**221701139 陈炀**

**221701202 陈如滨**

**编写：RATE-MAX团队**

**2020年4月9日**

目录

[1 引言 3](#_Toc37342191)

[1.1编写目的 3](#_Toc37342192)

[1.2项目背景 3](#_Toc37342193)

[1.3文档约定 3](#_Toc37342194)

[1.4 参考资料 3](#_Toc37342195)

[2 数据流图 4](#_Toc37342196)

[3 E-R分析与表结构设计 4](#_Toc37342198)

[3.1七日记忆——实体及其基本属性 5](#_Toc37342199)

[3.2七日记忆——实体属性图 6](#_Toc37342200)

[3.3七日记忆——实体联系图 10](#_Toc37342201)

[4数据库设计 10](#_Toc37342202)

[5 安全性设计 16](#_Toc37342203)

[5.1防止用户直接操作数据库 16](#_Toc37342204)

[5.2用户账号密码的加密 16](#_Toc37342205)

[5.3权限设置 16](#_Toc37342206)

[5.4视图控制 16](#_Toc37342207)

[5.5信息存取控制 16](#_Toc37342208)

[5.6保密安全设置 17](#_Toc37342209)

**修订记录：**

|  |  |  |  |
| --- | --- | --- | --- |
| **版本号** | **修订人** | **修订日期** | **修订描述** |
| V 1.0 | 洪楷滨 | 2020.4.6 | 大体结构约束 |
| V 1.1 | 林露、陈如滨 | 2020.4.7 | 整合，复审并提出问题 |
| V1.2 | 林露、陈如滨 | 2020.4.9 | 修改之前存在问题的部分，整理文档 |
| V1.3 | 洪楷滨 | 2020.4.9 | 对照复审问题重新复审文档，核实解决 |
|  |  |  |  |
|  |  |  |  |

# 1 引言

## 1.1编写目的

本阶段将在系统体系结构设计的基础上，对“七日记忆”项目进行数据库设计。 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。本文档旨在详细描述工作任务管理与评价系统中的数据库结构与设计。本文的读者可为工作任务管理与评价系统的开发者,也可以是工作任务管理与评价系统的维护人员。

## 1.2项目背景

|  |  |
| --- | --- |
| **名称** | **说明** |
| **数据库名称** | Hebdo |
| **数据库系统** | MySQL 8.0.19 |
| **任务提出者** | RATE-MAX |
| **开发者** | RATE-MAX |
| **用户** | 所有人群 |

## 1.3文档约定

本文档按以下要求和约定进行书写：

（1）页面的左边距为3.17cm，右边距为3.17cm，装订线靠左。

（2）标题最多分三级：

1）一级标题：1 2 3……（宋体加粗二号）

2）二级标题：1.1 1.2 1.3……（宋体加粗三号）

3）三级标题：1.1.1 1.1.2 1.1.3……（宋体加粗四号）

（3）正文字体为宋体小四号，单倍行距。无特殊情况，字体颜色均采用黑色。

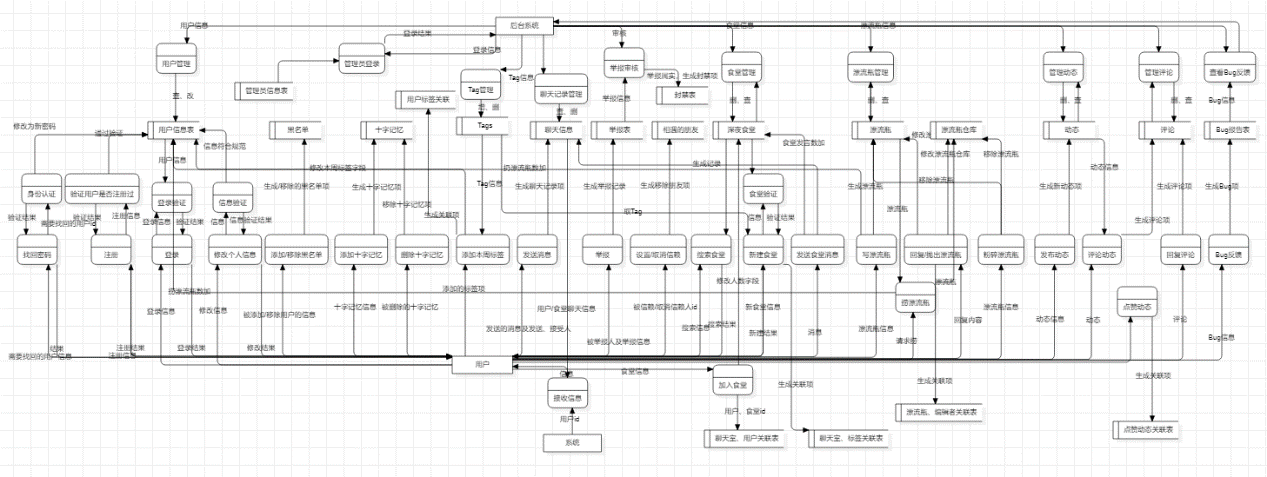
（4）出现序号的段落不采用自动编号功能而**采用人工编号**，各级别的序号依次为（1）、1）、a)等

（5）段落首行缩进 2 个字符

## 1.4 参考资料

[1]数据库设计说明书（GB8567——88）

# 2 数据流图



本系统涉及的实体共有三个：用户、系统、后台系统。

涉及的数据源有：用户表、黑名单表、十字记忆表、Tag表、聊天信息表、举报表、朋友表、深夜食堂表、漂流瓶表、漂流瓶仓库表、动态表、评论表、Bug报告表、管理员信息表、用户标签关联表、封禁表、聊天室用户关联表、聊天室标签关联表、漂流瓶编辑者关联表、点赞动态关联表。

注册：用户输入注册信息给注册功能，注册功能调用验证是否注册功能，如果通过验证则向用户表中添加项并返回注册结果。

找回密码：用户输入找回密码需要得相关信息，找回密码功能调用身份认证功能，如果通过身份认证则将新得密码写入用户信息表并返回认证结果。

登录：用户输入登录信息，登录功能调用登录认证功能，登录认证功能读取用户信息，返回认证结果。

修改个人信息：用户输入修改后得个人信息，修改功能调用信息合法性验证功能，如果合法则将新的用户信息写入用户信息表，并将验证结果返回。

添加黑名单：用户提供被添加黑名单的用户信息，如果该用户以在黑名单中则不做任何改变，否则往黑名单表中添加项。

移除黑名单：用户提供移除黑名单的用户信息，如果该用户不在黑名单中则不做任何改变，否则将该用户移除黑名单。

添加十字记忆：用户输入十字记忆信息，添加十字记忆功能往十字记忆表中添加新的项。

删除十字记忆：用户提供被删除的十字记忆的信息，删除功能将该十字记忆项移除。

添加本周标签：用户输入要添加的本周标签项，该功能修改用户表中的本周标签字段并生成标签、用户关联项。

发送消息：用户输入要发送的信息，该功能添加聊天信息项。

接受信息：系统周期性调用接受信息功能，如果有新的聊天信息记录则返回信息记录。

举报：用户输入举报人信息，该功能生成举报项。

设置信赖：用户提供设置信赖的人的信息，该功能生成相遇的朋友项。

取消信赖：用户提供取消信赖的人的信息，该功能移除相遇的朋友项。

搜索食堂：用户输入搜索食堂的关键字，该功能返回符合条件的食堂。

新建食堂：用户输入新建食堂的信息，该功能调用食堂验证功能，如果通过验证则生成新的食堂项，并返回创建结果。

加入食堂：用户提供想要加入的食堂信息，该功能修改食堂的人数项，并生成聊天室、用户关联项。

发送食堂信息：用户输入需要发送的信息，该功能生成聊天信息项。

写漂流瓶：用户输入漂流瓶中的信息，该功能生成漂流瓶项，并修改用户的扔漂流瓶数字段。

回复漂流瓶：用户输入回复内容，该功能生成漂流瓶、编辑者项，更新漂流瓶和漂流瓶仓库的相关内容。

抛出漂流瓶：用户抛出漂流瓶，该功能将漂流瓶仓库中的相关项移除，并更新漂流瓶表中的相关项。

粉碎漂流瓶：用户使用粉碎漂流瓶功能，该功能将漂流瓶表和漂流瓶仓库中的相关项移除。

发布动态：用户输入动态内容，该功能生成相应的动态项。

评论动态：用户输入评论内容，该功能生成读取该动态信息并生成相关的评论项。

点赞动态：生成相关的点赞动态关联项。

回复评论：用户输入评论内容，生成相应的评论项。

Bug反馈：用户输入Bug信息，生成Bug项。

管理登录：后台系统提供登录的管理员信息，该功能读取管理员信息进行验证，并返回验证结果。

用户管理：后台输入更新后的用户信息，并更新用户表。

Tag管理：后台更新Tag信息，更新Tag表。

聊天记录管理：后台更新聊天记录，并更新聊天记录表。

举报审核：后台审核举报项，如果属实则生成封禁项，否则更新举报表中的相关项。

食堂管理：后台读取食堂信息，如果进行更新则更新食堂表。

漂流瓶管理：后台读取漂流瓶信息，如果有更行则更新漂流瓶表中的相关项。

动态管理：后台读取动态信息，如果有更新则更新动态表。

评论管理：后台读取评论信息，如果更新则更新评论表。

查看Bug反馈：读取Bug报告表并返回。

# 3 E-R分析与表结构设计

## 3.1七日记忆——实体及其基本属性

用户（用户id、手机号码、用户昵称、密码、本周标签、头像、个人简介、性别、打捞数量、抛出数量）

标签（标签id、标签名）

朋友（我的id、朋友的id、朋友的备注、信赖等级）

黑名单用户（我的id、被添加用户的id）

十字记忆（十字记忆id、用户id、标题、内容、创建时间）

深夜食堂（聊天室id、聊天室名称、聊天室标签、聊天室人数、聊天室发言个数、聊天室热度、创建者id、创建时间、结束时间）

消息（消息id、发送内容，发送时间，发送者id）

动态（动态id、动态内容、发送图片、发送人id、发送时间、动态emoji、点赞人数）

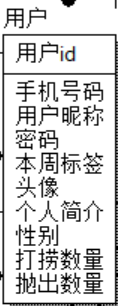
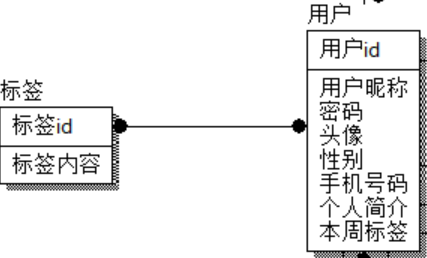
评论（评论id、评论者id、动态id、评论内容、评论表情）

管理员（管理员id、账号、密码）

## 3.2七日记忆——实体属性图

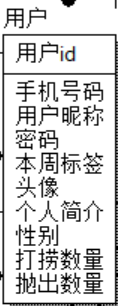
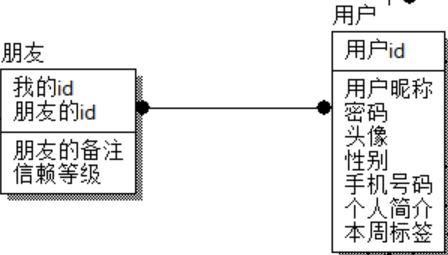
**一**个用户可以选择**多**个（本周）标签，上限3个。

**一**个标签可以被**多**个用户关联使用。



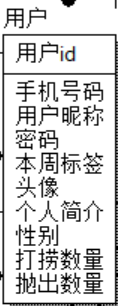
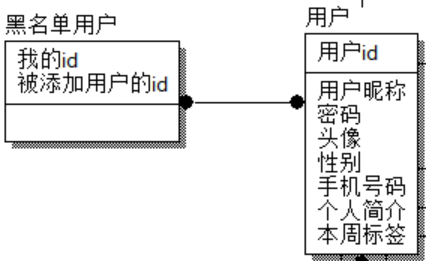
**一**个用户可以有**多**个朋友。

**一**个朋友（上文所指）可以是**多**个用户的朋友。



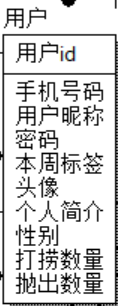
**一**个用户可以有**多**个黑名单用户。

**一**个黑名单用户（上文所指）可以出现在**多**个用户的黑名单里。



**一**个用户可以创建**多**个十字记忆。

**一**个十字记忆只能属于**一**个用户。

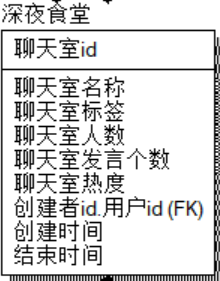
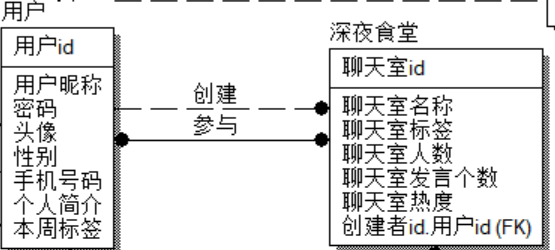
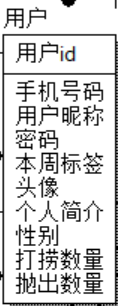


**一**个用户可以参与**多**个聊天室。

**一**个聊天室可以有**多**个用户参与。

**一**个用户可以创建**多**个聊天室。

**一**个聊天室只能被**一**个用户创建。



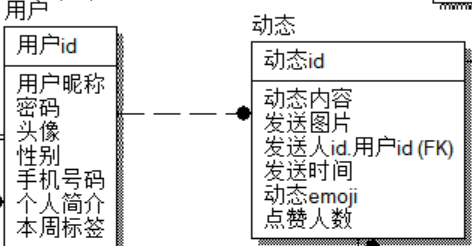
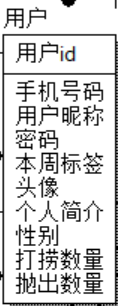
**一**个聊天室可以有**多**个标签。

**一**个标签可以被**多**个聊天室使用。



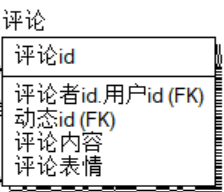
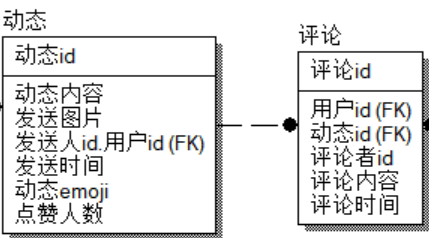
**一**个用户可以发表**多**条动态。

**一**条动态只属于**一**个用户。



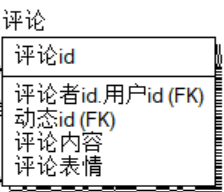
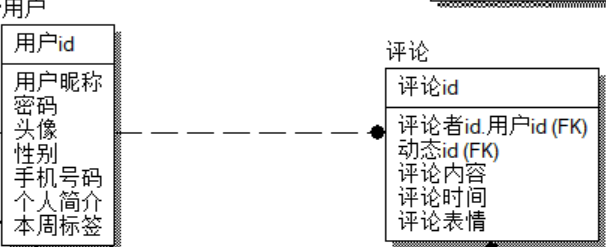
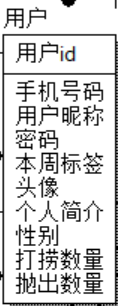
**一**条动态对应**多**条评论。

**一**条评论对应**一**条动态。



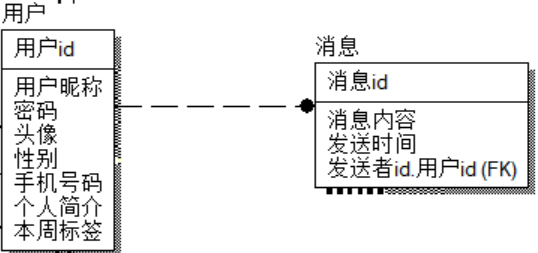
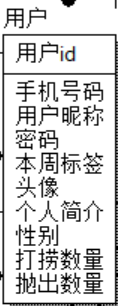
一个用户可以发表**多**条评论。

**一**条评论只能由**一**个用户发表。

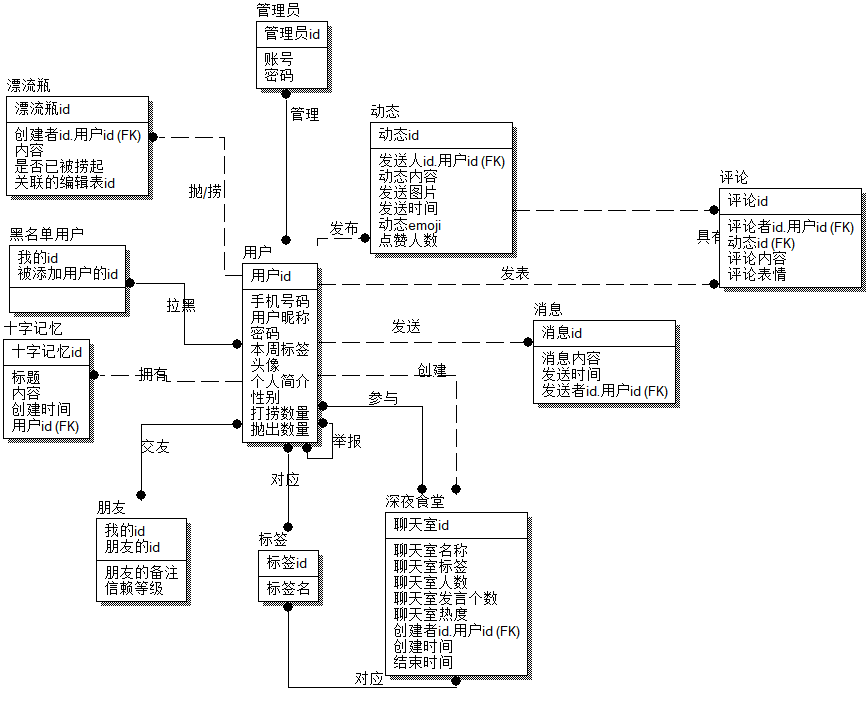


**一**个用户发送**多**条消息。

**一**条消息由**一**个用户发送。



## 3.3七日记忆——实体联系图



注：实线表示标识关系，虚线表示非标识关系。实心圆端所在的那端为一对多关系中的多的那端。

# 4数据库设计

说明:以下的null为无指定默认值,其中自增的字段基本都是主键

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **用户表:user** | | | | | | |
| 简介 | 这是一个存放所有用户的表 | | | | | |
| 编号 | 字段名 | 字段类型 | 是否为主键 | 是否允许为空 | 默认值 | 备注信息 |
| 01 | user\_id | int(11) | 是 | 否 | 自增 | 用户id |
| 01 | telephone | varchar(14) | 否 | 否 | null | 用户登陆账号 |
| 02 | nickname | varchar(20) | 否 | 否 | 由系统生成 | 用户昵称 |
| 03 | password | varchar(32) | 否 | 否 | null | 登陆密码 |
| 04 | this\_week\_tag | varchar(32) | 否 | 是 | null | 本周标签 |
| 05 | icon | varchar(256) | 否 | 是 | 由系统给定 | 用户头像 |
| 06 | profile | varchar(256) | 否 | 否 | null | 用户简介 |
| 07 | gender | varchar(10) | 否 | 否 | 保密 | 性别 |
| 08 | salvage\_number | int(4) | 否 | 否 | 0 | 打捞数量 |
| 09 | throw\_number | int(4) | 否 | 否 | 0 | 抛出数量 |
| 补充说明 | 1. 密码采用md5加密 2. 设定用户每周标签可以设定最多3个,之间用逗号分隔开 3. 初始用户昵称为根据手机号随机生成 4. 头像是在数据库中存储路径 5. 本周标签用逗号分割 6. 抛出和打捞是有上限的 | | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **封禁表:ban** | | | | | | |
| 简介 | 用来管理用户封禁的表 | | | | | |
| 编号 | 字段名 | 字段类型 | 是否为主键 | 是否允许为空 | 默认值 | 备注信息 |
| 01 | user\_id | int(11) | 是 | 否 | 自增 | 用户id |
| 02 | ban\_stime | timestamp | 否 | 否 | null | 封禁开始时间 |
| 03 | ban\_etime | timestamp | 否 | 否 | null | 封禁结束时间 |
| 补充说明 | 1. 封禁操作在后台进行 | | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **标签表:tag** | | | | | | |
| 简介 | 这是个是存放所有标签的表 | | | | | |
| 编号 | 字段名 | 字段类型 | 是否为主键 | 是否允许为空 | 默认值 | 备注信息 |
| 01 | tag\_id | int(11) | 是 | 否 | 自增 | 标签id |
| 02 | tag\_name | varchar(20) | 否 | 否 | null | 标签名 |
| 补充说明 | 1. 聊天室和用户共用一套标签 2. 标签可以存放在Redis中 3. 标签要定期更新 | | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **用户标签关联表:user\_tag** | | | | | | |
| 简介 | 这个是存放用户与标签关系的关联表 | | | | | |
| 编号 | 字段名 | 字段类型 | 是否为主键 | 是否允许为空 | 默认值 | 备注信息 |
| 01 | user\_id | int(11) | 是 | 否 | 自增 | 用户id |
| 02 | tag\_id | int(11) | 是 | 否 | 自增 | 标签id |
| 03 | tag\_number | int(11) | 否 | 否 | 0 | 过往标签出现次数 |
| 补充说明 | 1. 本周标签存在了user表 2. user\_id和tag\_id都有外键约束 | | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **朋友表:friend** | | | | | | |
| 简介 | 这是存放一个用户与其他用户朋友关系的表 | | | | | |
| 编号 | 字段名 | 字段类型 | 是否为主键 | 是否允许为空 | 默认值 | 备注信息 |
| 01 | user\_id | int(11) | 是 | 否 | 自增 | 添加别人的id |
| 02 | added\_id | int(11) | 是 | 否 | 自增 | 被设置信赖的id |
| 03 | remark | varchar(64) | 否 | 是 | null | 设置朋友的备注 |
| 04 | level | int(1) | 否 | 否 | 0 | 信赖等级 |
| 补充说明 | 1. 这个表里面有记录就代表了设置信赖 2. 信赖等级取值为0,1,2 | | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **黑名单表:blacklist** | | | | | | |
| 简介 | 这个是多对多关系,用来表示用户与其他用户的黑名单关系 | | | | | |
| 编号 | 字段名 | 字段类型 | 是否为主键 | 是否允许为空 | 默认值 | 备注信息 |
| 01 | user\_id | int(11) | 是 | 否 | 自增 | 自己的用户id |
| 02 | added\_id | int(11) | 是 | 否 | 自增 | 被添加的用户id |
| 补充说明 | 1. 黑名单是双向的 2. user\_id是添加别人的用户id,added\_id是被添加的id以此区分主动和被动 | | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **十字记忆表:memory** | | | | | | |
| 简介 | 这是存放所有十字记忆的表 | | | | | |
| 编号 | 字段名 | 字段类型 | 是否为主键 | 是否允许为空 | 默认值 | 备注信息 |
| 01 | memory\_id | int(11) | 是 | 否 | 自增 | 十字记忆id |
| 02 | memory\_title | varchar(20) | 否 | 否 | null | 十字记忆标题 |
| 02 | memory\_content | varchar(100) | 否 | 否 | null | 十字记忆内容 |
| 03 | memory\_date | datetime | 否 | 否 | null | 十字记忆日期 |
| 04 | user\_id | int(11) | 否 | 否 | 自增 | 发送记忆的用户 |
| 补充说明 | 1. user\_id为外键 2. 十字记忆定位内容简短的日记,所以内容不会很长 | | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Bug报告表:bug** | | | | | | |
| 简介 | 这是反馈bug的表 | | | | | |
| 编号 | 字段名 | 字段类型 | 是否为主键 | 是否允许为空 | 默认值 | 备注信息 |
| 01 | bug\_id | int(11) | 是 | 否 | 自增 | bug的id |
| 02 | bug\_content | varchar(128) | 否 | 否 | null | bug的内容 |
| 补充说明 | 1. 这里为了简单没有加入图片 | | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **食堂类:chatroom** | | | | | | |
| 简介 | 这是一个聊天室的表 | | | | | |
| 编号 | 字段名 | 字段类型 | 是否为主键 | 是否允许为空 | 默认值 | 备注信息 |
| 01 | chatroom\_id | int(11) | 是 | 否 | 自增 | 聊天室id |
| 02 | chatroom\_name | varchar(64) | 否 | 否 | null | 聊天室名字 |
| 03 | chatroom\_tag | varchar(64) | 否 | 否 | null | 聊天室标签 |
| 04 | chatroom\_number | int(2) | 否 | 否 | 0 | 聊天室人数 |
| 05 | chatroom\_statement | int(10) | 否 | 否 | 0 | 聊天室发言个数 |
| 06 | chatroom\_hot | int(4) | 否 | 否 | 0 | 聊天室热度 |
| 07 | create\_id | int(4) | 否 | 否 | 创建者的id | 创建者id |
| 08 | chatroom\_start | datetime | 否 | 否 | null | 创建时间 |
| 09 | chatroom\_end | datetime | 否 | 否 | 创建时间+24h | 结束时间 |
| 补充说明 | 1. 聊天室等动态属性可以存放在内存数据库或者是数据结构里面 2. 聊天人数限制为50人 3. 聊天室默认存在24小时 4. 聊天室标签也定为3个,用逗号分隔开存储 | | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **聊天室与用户关联表:chatroom\_user** | | | | | | |
| 简介 | 这是一个聊天室和聊天室相关用户的表 | | | | | |
| 编号 | 字段名 | 字段类型 | 是否为主键 | 是否允许为空 | 默认值 | 备注信息 |
| 01 | chatroom\_id | int(4) | 是 | 否 | 自增 | 聊天室id |
| 02 | user\_id | varchar(64) | 是 | 否 | 自增 | 成员id |
| 补充说明 | 1. 建议使用redis存储之后持久化到mysql中 | | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **聊天室标签关联表:chatroom\_tag** | | | | | | |
| 简介 | 这个是聊天室和聊天室标签的表 | | | | | |
| 编号 | 字段名 | 字段类型 | 是否为主键 | 是否允许为空 | 默认值 | 备注信息 |
| 01 | chatroom\_id | int(11) | 是 | 否 | 自增 | 标签id |
| 02 | tag\_id | int(11) | 是 | 否 | null | 标签名 |
| 补充说明 | 1. 建议使用Redis存储,聊天室消失之后存到mysql中 2. redis中存储格式大致为   “chatroom\_id”:  “chatroom\_name”:\*\*\*\*  “chatroom\_hot”:\*\*\*\*  “chatroom\_member”: {1, 2, 3, 4, 5....}  “chatroom\_tag”:{\*\*,\*\*\*....} | | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **聊天消息表:msg** | | | | | | |
| 简介 | 这是存放所有聊天消息的表 | | | | | |
| 编号 | 字段名 | 字段类型 | 是否为主键 | 是否允许为空 | 默认值 | 备注信息 |
| 01 | msg\_id | int(11) | 是 | 否 | 自增 | 聊天记录id |
| 02 | msg\_content | varchar(128) | 否 | 否 | null | 聊天记录内容 |
| 03 | msg\_datetime | datetime | 否 | 否 | null | 发送时间 |
| 04 | send\_id | int(11) | 否 | 否 | 自增 | 发送人id |
| 补充说明 | 1. 会话的id存放到redis中 2. send\_id为外键 | | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **动态表:post** | | | | | | |
| 简介 | 这是存放所有动态的表 | | | | | |
| 编号 | 字段名 | 字段类型 | 是否为主键 | 是否允许为空 | 默认值 | 备注信息 |
| 01 | post\_id | int(11) | 是 | 否 | 自增 | 动态id |
| 02 | post\_content | varchar(128) | 否 | 否 | null | 动态内容 |
| 03 | post\_image | varchar(512) | 否 | 是 | null | 发送图片 |
| 04 | send\_id | int(11) | 否 | 否 | 自增 | 发送人id |
| 05 | post\_date | datetime | 否 | 否 | null | 动态发送时间 |
| 06 | post\_emoji | varchar(256) | 否 | 是 | null | 动态emoji |
| 07 | post\_like | int(11) | 否 | 否 | 0 | 动态点赞人数 |
| 补充说明 |  | | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **评论表:comment** | | | | | | |
| 简介 | 这是存放所有评论的表 | | | | | |
| 编号 | 字段名 | 字段类型 | 是否为主键 | 是否允许为空 | 默认值 | 备注信息 |
| 01 | comment\_id | int(11) | 是 | 否 | 自增 | 评论id |
| 02 | post\_content | varchar(128) | 否 | 否 | null | 评论内容 |
| 03 | post\_emoji | varchar(256) | 否 | 是 | null | 评论表情 |
| 04 | send\_id | int(11) | 否 | 否 | 自增 | 评论人id |
| 05 | post\_id | int(11) | 否 | 否 | 自增 | 动态id |
| 补充说明 | 1. send\_id和post\_id为外键约束 | | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **动态点赞关联表:post\_like** | | | | | | |
| 简介 | 一个动态与点赞的关联表 | | | | | |
| 编号 | 字段名 | 字段类型 | 是否为主键 | 是否允许为空 | 默认值 | 备注信息 |
| 01 | post\_id | int(11) | 是 | 否 | 自增 | 发表动态的id |
| 02 | user\_id | int(11) | 是 | 否 | 自增 | 点赞的id |
| 补充说明 |  | | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **管理员表:admin** | | | | | | |
| 简介 | 存放所有管理员的表 | | | | | |
| 编号 | 字段名 | 字段类型 | 是否为主键 | 是否允许为空 | 默认值 | 备注信息 |
| 01 | admin\_id | int(4) | 是 | 否 | 自增 | 管理员id |
| 02 | admin\_name | varchar(20) | 否 | 否 | null | 管理员账号 |
| 03 | admin\_password | varchar(20) | 否 | 是 | null | 管理员密码 |
| 补充说明 | 1. 管理员仅仅用来后台登陆 | | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **举报表:report** | | | | | | |
| 简介 | 存放所有举报信息的表 | | | | | |
| 编号 | 字段名 | 字段类型 | 是否为主键 | 是否允许为空 | 默认值 | 备注信息 |
| 01 | report\_id | int(11) | 是 | 否 | 自增 | 举报id |
| 02 | report\_reason | varchar(20) | 否 | 否 | null | 举报原因 |
| 03 | report\_content | varchar(128) | 否 | 是 | null | 举报内容 |
| 04 | report\_date | datetime | 否 | 否 | null | 举报时间 |
| 05 | send\_id | int(11) | 否 | 否 | null | 举报人id |
| 06 | report\_type | varchar(3) | 否 | 否 | null | 举报类型 |
| 补充说明 | 1. send\_id为外键约束 2. 举报原因会让用户选择 | | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **漂流瓶表:drift** | | | | | | |
| 简介 | 存放所有漂流瓶信息的表 | | | | | |
| 编号 | 字段名 | 字段类型 | 是否为主键 | 是否允许为空 | 默认值 | 备注信息 |
| 01 | bottle\_id | int(11) | 是 | 否 | null | 漂流瓶id |
| 02 | content | varchar(128) | 否 | 否 | null | 漂流瓶内容 |
| 03 | flag | boolean | 否 | 否 | false | 已被捞起 |
| 04 | user\_id | int(11) | 否 | 否 | 自增 | 创建者id |
| 05 | edit\_id | int(11) | 否 | 是 | 自增 | 关联的编辑表 id |
| 补充说明 | 1. edit\_id,user\_id为外键约束 | | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **漂流瓶与编辑人的关联表:drift\_editor** | | | | | | |
| 简介 | 漂流瓶与编辑他的用户关联表 | | | | | |
| 编号 | 字段名 | 字段类型 | 是否为主键 | 是否允许为空 | 默认值 | 备注信息 |
| 01 | edit\_id | int(11) | 是 | 否 | null | 编辑表 id |
| 02 | user\_id | int(11) | 否 | 否 | null | 编辑者 id |
| 03 | comment | varchar(128) | 否 | 否 | null | 新增内容（回复） |
| 04 | edit\_date | datetime | 否 | 否 | null | 编辑的时间 |
| 补充说明 | 1. user\_id为外键约束 | | | | | |

# 5 安全性设计

## 5.1防止用户直接操作数据库

用户只能通过给定的外部接口操作数据库：外部接口向内部接口传递参数，然后进行预编译sql语句后才能操作数据库，这从根本上杜绝了用户直接操作数据库的可能。

同时，系统对用户发出的访问请求进行身份和授权认证，具有合法权限的用户才能进入系统操作环境，对数据库进行权限范围内的访问；同时将认证及访问情况记录在日志中，便于日志审计。

## 5.2用户账号密码的加密

账号不加密，密码后期进行相关加密技术进行加密（md5）。

## 5.3权限设置

定义用户的权限，限制个别用户对某些功能的使用。如用户在与人聊天是收到举报并在管理员进行封号处理后，用户有一段时间不能与他人进行相遇的朋友界面的对话。在某个聊天室的被管理员进行关闭时，用户无权限访问聊天室聊天。在动态页面的某条动态收到管理员处理时，该条动态用户无权限查看。

同时，只有具有管理员资格的用户才有权限对用户、聊天室、动态、举报信息进行管理。

## 5.4视图控制

系统定义视图，为不同的用户定义不同的视图，把数据对象限制在一定的范围内， 通过视图机制把要保密的数据对无权用户隐藏起来。

## 5.5信息存取控制

后期开发过后时间充裕的化进行对数据库中有关表和字段信息的加密、解密。初步决定采用用对称加密算法中的分组密码算法DES实现。如果有更进一步的健壮性要求，采用三重DES、IDEA等加密算法。

## 5.6保密安全设置

后台设置拦截器防止同一IP在短时间内进行大量的恶意请求，造成服务器资源紧张，瘫痪的现象。

后端设置过滤机制，使用过滤器对没有注册登录用户的请求进行拦截，不予放行，防止非法用户恶意操作，只有经过常规途径注册并登录的用户才能使用系统。