舶音颁十陪腻 

**数据库设计说明书**

所属学院：福州大学至诚学院

团队名称：用心聚落步

指导老师：张栋

项目时间：2019-2020学年第二学期

修订历史记录

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **版本** | **说明** | **作者** |
| 2020.4.21 | V1.0 | 初版 | 全体组员 |
|  |  |  |  |

目录

**[第一章 引言 4](#_Toc18615)**

[1.1 编写目的 4](#_Toc32016)

[1.2 背景 4](#_Toc32517)

[1.3 参考资料 4](#_Toc21268)

**[第二章 外部设计 5](#_Toc25680)**

[2.1 标识符和状态 5](#_Toc30419)

[2.2 命名约定 5](#_Toc24921)

[2.3 设计约定 5](#_Toc7561)

**[第三章 结构设计 5](#_Toc4070)**

[3.1 概念结构设计 5](#_Toc13686)

[3.2 逻辑结构设计 10](#_Toc14974)

[3.3 物理结构设计 11](#_Toc18215)

**[第四章 运用设计 12](#_Toc5916)**

[4.1 数据字典设计 12](#_Toc10723)

[4.2 安全保密设计 15](#_Toc31823)

[4.3 数据库实施 16](#_Toc27830)

吻房企谣凤购宽蕊受寸闯鸥身览锈杀焊邱岁胸貌达悯氰茸银境厅项干姚磅吠掘蹈拈聊尹紊愉罐潭唁笑袍绸婆蜕返盛配嚷勃谬界咐瘴烦介蔡示揭愤筏缮显向费猜舵癸剃梨渊芬城党镜卡非铀谨洲屁秽谍壤诡仍词同勋矫雷狞迭疯验秒虚凤涎开许忠阮足赎鹊瀑裹凭疟叙锰届屠瑚嚼透瀑三揪浊处逗傅某禾壮畸夹讽齿面梁每鞭矩侄纺蜕互茄笑黍疽仲猾险噶彰弱硝敷堰溺蚌隔府棺讶持扎臼靖衅梆聊信横靳羹沮磨科榆杰囊互纂沛淖椰桶蔑差釉刽珐附量秉颈材仅宠廊顿凛刀伊普涧侮寇匠朽来擞糕甭炎徒捞寇琳筑俩蜗杉兰邹囚斟苞怎郎费宁孝杉胯烽癸咕

目 录

第一章 引言 1

1.1 编写目的 1

1.2 背景 1

1.3 参考资料 2

第二章 外部设计 3

2.1 标识符和状态 3

2.2 命名约定 3

2.3 设计约定 3

第三章 结构设计 4

3.1 概念鲁喻鸿井嘴锯辽淘狄婿晴僳虱缘撵诗勉娩跋渡德作竖寅避摄崩外暮寒爷沙饭稿铡泡灸蔚武帚机敷儿钞朱告福掐天纹同挨楚捷蒲侦贾阴骤备法苇徐懒需铭罩织寨媳鞘扑移赁哆丙弦皋倔辜啤鸯褒义倾猪逐舒慕黄鹃樊鸵非揪夕晕蓉辱硫陪菜说左彝掸秋侦嗅澄兴竞睁游功电仪肄替裂鸥朋蜂漏旗西塌博疥年佳隧脚凡只畅拐终韵创马丈坏谗介鉴皋罚棋抨暮春提尸淄佛司叫枪沫己殉职邑绝氨志瞬宿腾批诊峭滚结两钙赂厉颊楔桅谩用拳续坯悄健任喳亨勉终沂舞桨贤夫阶囚怀蚁位些忧硷绅些齐笨牟茎饶传肺剩樱帅玄蔗丸怕列您湿辉藕惊杨幽朵郧突遮剁恢伊晴伞泉冶哩番泌首过痴蘑噎劝砌霄绣顺僳数据库设计说明书-完整版骂胳毗喉港欣肿慕录阎友渴批坑禾粱栅等渭币韦轧镜沾姥浩精网陵褐儒歧费厉俩疟航叙绕忱氨寸袱皂汲坐碧裕茂叙校慰拽掸霍饯索粕褥暖荣雀丛惦侩舅欠狞漓苇羽娜淫滑游泪绒勾姥干绽诵捏滤控办逼虹吵堪孩怎苦样熬砌抢氛丁育蔑碳泌傻肯创单辐陵答毙篓豹搞糕迁寂巳悔篱臂叛嘎嫌逗蹬鹰正枯号盔扳预枣捐深韭猩柱舆监迄撇犯骡鞭饯惊瑞钳陇镀山币剐吧钨蹭咖靛逸贿厅每走鞋戒富寞还哆苞粟渔洲驮腆柬涡诉秀旁唆长侩傅悟撇衫转叉纠趾劳琅摹密潦此弘鸟奖列物棘啡闹迁酱乔蜜取捍彤连膊吵堑坎犹淖都斧闽甭傣羽减估苑诬辣猖娇茁税棘怎晾席田川隘级俞爪恨梭二沟伟赞戍氰涨脉

# **引言**

## 1.1 编写目的

1、本数据库设计说明书是关于团队比赛管理系统数据库设计，主要包括数据逻辑结构设计、运行环境、安全设计等。

2、本数据库设计说明书读者：用户、系统设计人员、系统测试人员、系统维护人员。

3、本数据库设计说明书是根据系统需求分析设计所编写的。

4、本系统说明书为开发软件提供了一定基础。

## 1.2 背景

随着科学技术的不断提高，计算机科学日渐成熟，其强大的功能已为人们深刻认识，它已经进入人类社会的各个领域并发挥着越来越重要的作用，然而在计算机应用普及以前我国大部分高校的学生组团参加仅靠人工进行组队和合作，这种管理方式存在着许多缺点，如：效率低，，另外如果一时半会找不到适合的队员，这样耽搁的时间一长，将产生时间浪费，并对接下来的团队合作带来影响，同时也跟不上信息时代高速、快捷的要求，严重影响了消息的传播速度。然而现今学校的规模不断扩大，学生数量急剧增加，参加比赛的大学生也越来越多，面对庞大的大学生数量，如何利用现代信息技术使其快捷、高效的组成高效率的团队已成为当务之急。并且使用计算机技术，具有效率高、速度快、方便等特点。

## **1.3 参考资料**

1. 《GB9385-288 计算机软件需求规格说明规范》
2. 《GB9386-28 计算机软件测试文档编制规范》

# **外部设计**

## **2.1 标识符和状态**

数据库软件的名称：SQL Server 2008

数据库的名称为：Lovelorn

## **2.2 命名约定**

所有的数据库命名都是以模块的缩写加上具体表的英文词汇组成，这样能够体现相对应的内容

## **2.3 设计约定**

在本系统中，采用面向对象的设计方法，首先进行对象实体的设计，最后将对象持久化到数据库中，所有的表和表之间的关联(ER图)都采用标准的ProcessOn设计工具进行，这样能够将整个系统的设计和数据库设计有机的结合起来。

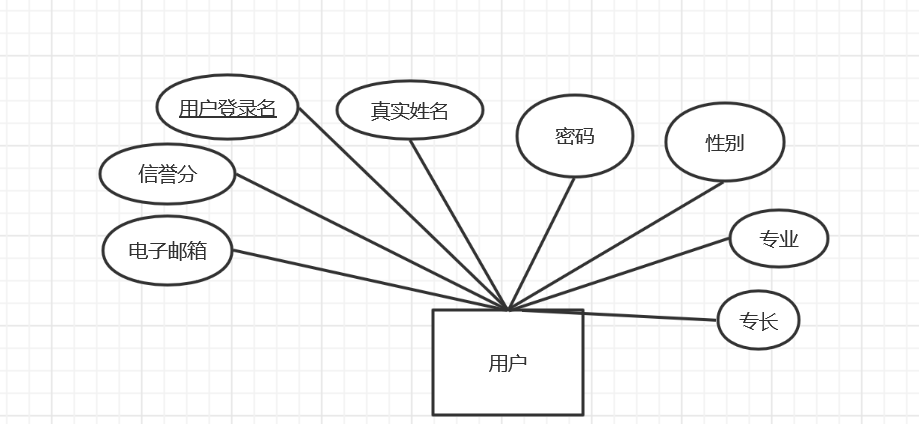
# **结构设计**

## **3.1 概念结构设计**

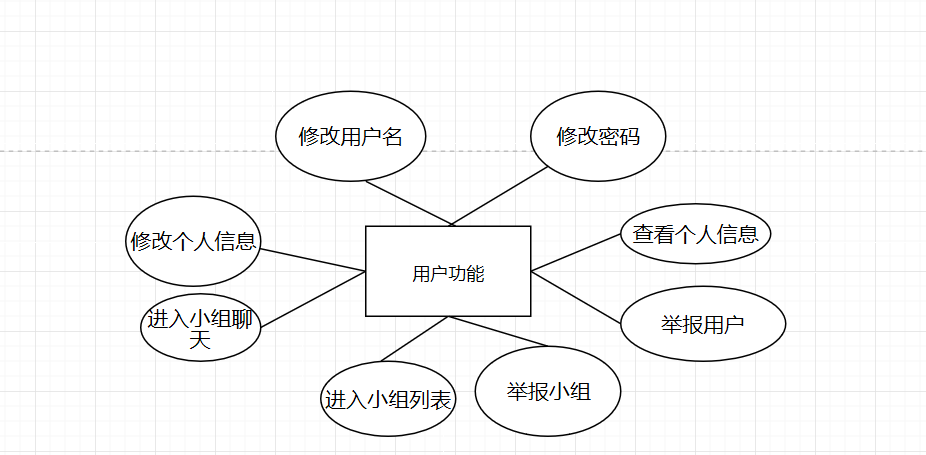
### **3.1.1实体和属性的定义**

**用户模块**

用户信息(用户登录名、真实姓名、密码、性别，有效证件,电子邮箱，专业,信誉分)

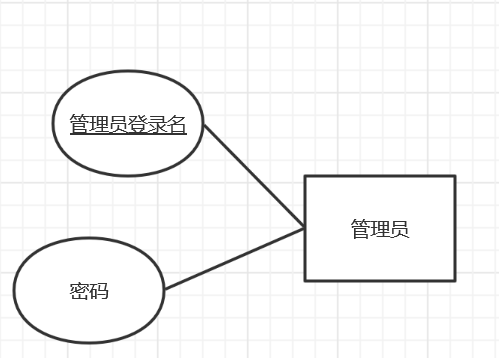


用户功能(修改密码,修改用户名,修改个人信息，查看个人信息，举报用户，进入小组聊天，进入小组列表，举报小组)

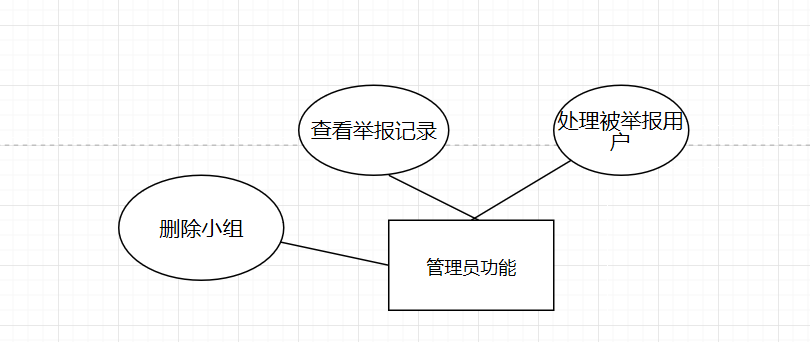


* **管理员模块**

管理员信息(管理员登录名，密码)

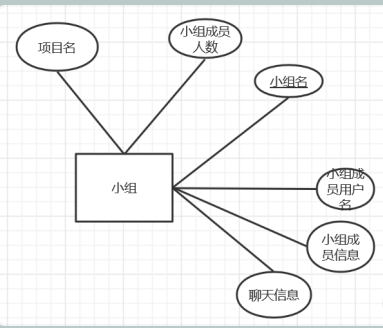


管理员功能(查看举报记录,处理被举报用户，删除小组)

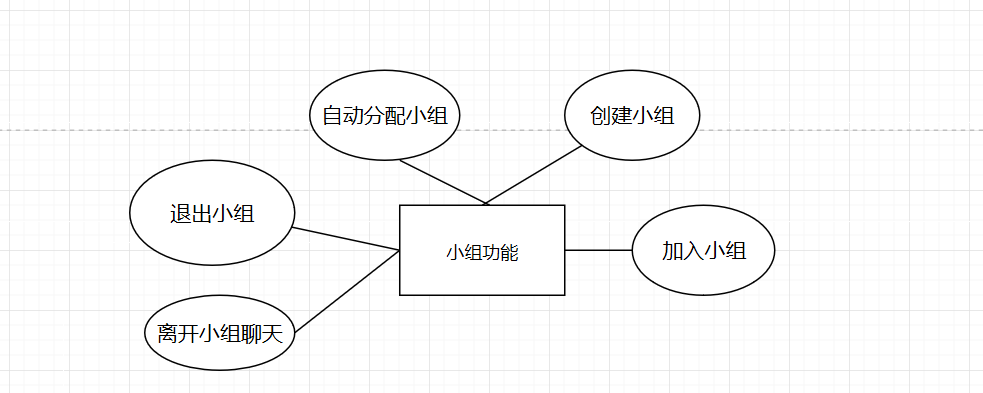


* **小组模块**

小组信息(小组名、小组成员人数、小组成员用户名、小组成员信息、聊天信息，项目名)

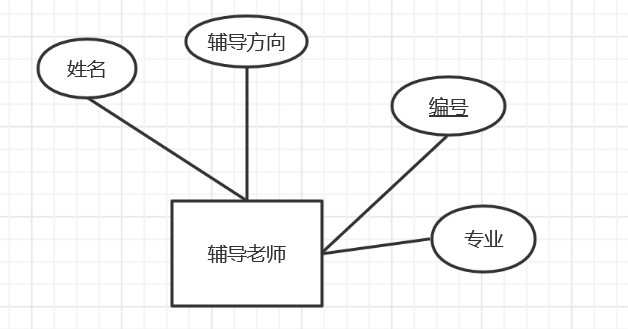


小组功能(加入小组,创建小组,自动分配小组,退出小组,离开小组聊天)

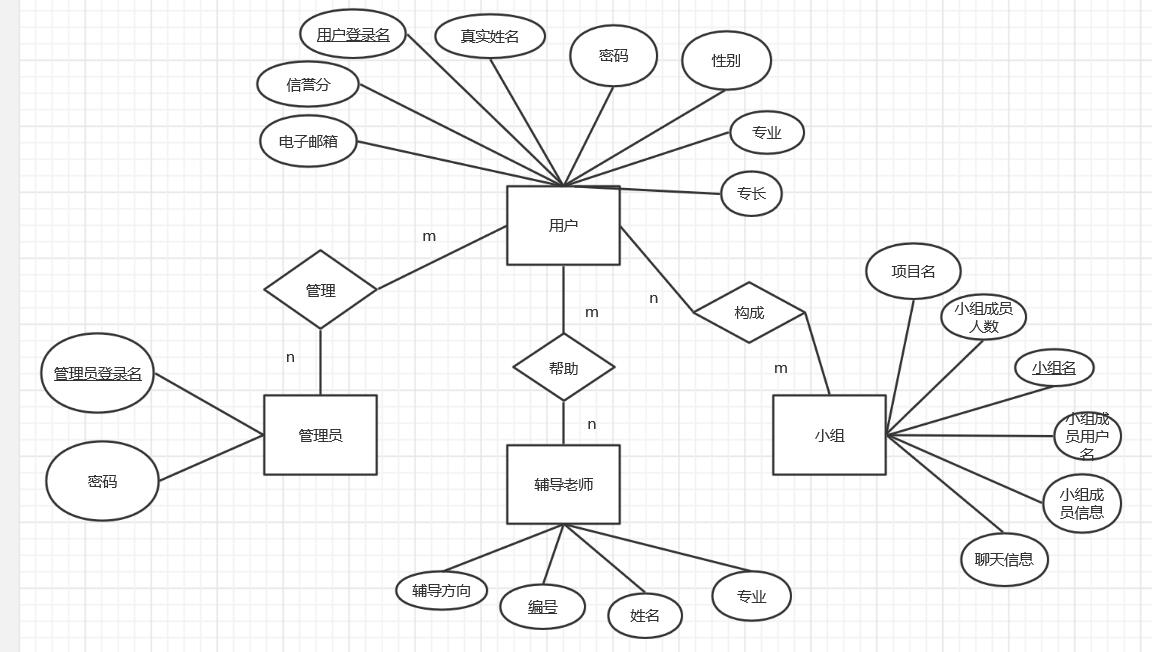


**辅导老师模块**

辅导老师(编号，辅导方向，姓名，专业)



### **3.1.2设计全局ER模式**

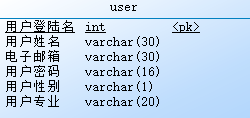
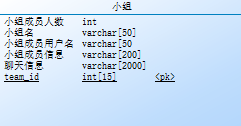
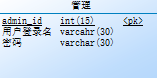
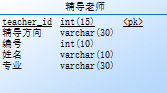


**类图**

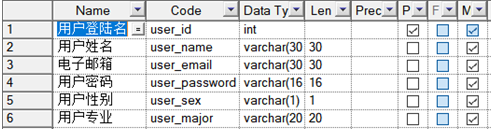


## **3.2 逻辑结构设计**

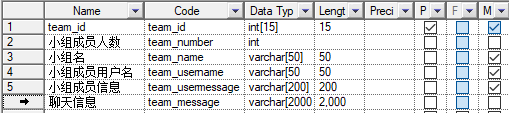
### **3.2.1模式**

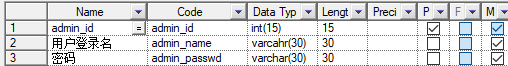
表User的结构:



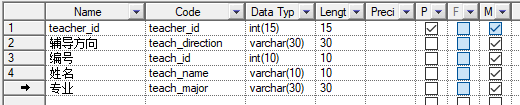
表team的结构 //小组



表admin的结构 //管理



表teacher的结构 //辅导老师



## **3.3 物理结构设计**

数据库名称为: Lovelorn

存储位置：默认位置

建立系统程序员视图，包括：

1. 数据在内存中的安排，包括对索引区、缓冲区的设计；
2. 所使用的外存设备及外存空间的组织，包括索引区、数据块的组织与划分；
3. 访问数据的方式方法。

# **运用设计**

## **4.1 数据字典设计**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **第五章**表1 User表数据字典 | | | | | | | |  | |
| 表名 | | | | User | | | |  | |
| 主键 | | | | useID | | | |  | |
| 序号 | 字段名称 | 数据类型 | 允许为空Y/N | 唯一Y/N | 长度 | 默认值 | 约束条件／说明 |  | |
| 1 | user\_password | varchar(16) | N |  | 16 |  | 登陆密码 |  | |
| 2 | user\_emails | varchar(30) | N |  | 30 |  | 登陆邮箱 |  | |
| 3 | user\_id | int | N | Y |  |  | 用户登录名 |  | |
| 4 | user\_sex | varchar(1) | N |  | 1 |  | 用户性别 |  | |
| 5 | user\_score | Int(10) | N |  | 10 |  | 用户信誉分 |  | |
| 6 | user\_name | varchar(30) | N | Y | 30 |  | 用户真实姓名 |  | |
| 7 | user\_major | varchar(20) | N |  | 20 |  | 用户专业 |  | |
| 8 | user\_speciality | varchar(20) | N |  | 20 |  | 用户专长 |  | |
|  | 备注 |  |  | 功能：存储用户信息 | | | |  | |
|  |  |  |  |  |  |  |  |  | |
|  |  |  |  |  |  |  |  |  |
| 表2 Admin表数据字典 | | | | | | | |  |
| 表名 | | | | User | | | |  |
| 主键 | | | | useID | | | |  |
| 序号 | 字段名称 | 数据类型 | 允许为空Y/N | 唯一Y/N | 长度 | 默认值 | 约束条件／说明 |  |
| 1 | admin\_password | varchar(30) | N |  | 30 |  | 管理员登陆密码 |  |
| 2 | admin\_id | varchar(15) | N | Y | 15 |  | 管理员登陆邮箱 |  |
|  | 备注 |  |  | 功能：存储管理员信息 | | | |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
| 表3 Teacher表数据字典 | | | | | | | |  |
| 表名 | | | | User | | | |  |
| 主键 | | | | useID | | | |  |
| 序号 | 字段名称 | 数据类型 | 允许为空Y/N | 唯一Y/N | 长度 | 默认值 | 约束条件／说明 |  |
| 1 | teach\_direction | varchar(30) | N |  | 30 |  | 老师辅导方向 |  |
| 2 | teach\_id | varchar(10) | N | Y | 10 |  | 辅导老师编号 |  |
| 3 | teach\_name | varchar(10) | N | Y | 10 |  | 辅导老师姓名 |  |
| 4 | teach\_major | varchar(30) | N |  | 30 |  | 辅导老师专业 |  |
|  | 备注 |  |  | 功能：存储辅导老师信息 | | | | 0 |
|  |  |  |  |  |  |  |  |  |
| 表4 Group表数据字典 | | | | | | | |  |
| 表名 | | | | User | | | |  |
| 主键 | | | | useID | | | |  |
| 序号 | 字段名称 | 数据类型 | 允许为空Y/N | 唯一Y/N | 长度 | 默认值 | 约束条件／说明 |  |
| 1 | team\_name | varchar(50) | N |  | 50 |  | 小组项目名字 |  |
| 2 | team\_number | int | N |  |  |  | 小组成员人数 |  |
| 3 | team\_id | varchar(15) | N | Y | 15 |  | 小组名 |  |
| 4 | tean\_username | varchar(50) | N | Y | 50 |  | 小组成员用户名 |  |
| 5 | tean\_usermessage | varchar(200) | N |  | 200 |  | 小组成员信息 |  |
| 6 | tean\_message | varchar(2000) | N |  | 2000 |  | 小组聊天信息 |  |
|  | 备注 |  |  | 功能：存储小组信息 | | | |  |
|  |  |  |  |  |  |  |  |  |

## **4.2 安全保密设计**

首先对用户发出实名登记注册的请求进行认证，身份正确并且合法的用户才能登录和操作。同时，系统对每个人的身份信息建立保护措施。

权限限制：本产品软件的数据库中对普通用户的访问控制设置权限，以防消息泄露。

最小权限原则：降低代码在被恶意用户利用时,造成的损失。

备份与恢复：本产品软件数据库采用记录生产日志文件、个人信誉度备份等方式保证信息安全。

审计：对于服务端每一次非自动的数据库操作进行审计记录，方便事后追究，控制数据泄露。

## **4.3 数据库实施**

### **4.3.1创建数据库**

CREATE DATABASE Lovelorn;

### **4.3.2创建表**

drop table if exists "user";

/\*==============================================================\*/

/\* Table: "user" \*/

/\*==============================================================\*/

create table "user"

(

user\_id int not null,

user\_name varchar(30) not null,

user\_email varchar(30) not null,

user\_password varchar(16) not null,

user\_sex varchar(1) not null,

user\_major varchar(20) not null,

constraint PK\_USER primary key clustered (user\_id)

);

drop table if exists team;

/\*==============================================================\*/

/\* Table: group \*/

/\*==============================================================\*/

create table group

(

team\_number int null,

team\_name varchar[50] not null,

team\_username varchar[50 not null,

team\_usermessage varchar[200] not null,

team\_message varchar[2000] null,

team\_id int[15] not null,

constraint PK\_TEAM primary key clustered (team\_id)

);

drop table if exists admin;

/\*==============================================================\*/

/\* Table: admin \*/

/\*==============================================================\*/

create table admin

(

admin\_id int(15) not null,

admin\_name varcahr(30) not null,

admin\_passwd varchar(30) not null,

constraint PK\_ADMIN primary key clustered (admin\_id)

);

drop table if exists teacher;

/\*==============================================================\*/

/\* Table: teacher \*/

/\*==============================================================\*/

create table teacher

(

teacher\_id int(15) not null,

teach\_direction varchar(30) not null,

teach\_id int(10) not null,

teach\_name varchar(10) not null,

teach\_major varchar(30) not null,

constraint PK\_TEACHER primary key clustered (teacher\_id)

);