|  |
| --- |
| **浙江大学城市学院** |
| **数据库设计报告** |
| **H5套圈小游戏** |

|  |
| --- |
| 项目名称 基于html5的休闲小游戏  组长 陈帆  组员 张荣阳、赵伟宏  小组号 G07  专业班级 软件工程1601  指导老师 杨枨 |



版 本 历 史

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 版本/状态 | 作者 | 参与者 | 起止日期 | 备注 |
| 0.1 | 赵伟宏 | 赵伟宏 | 2018-5-6 | 初始化报告 |
| 0.2 | 赵伟宏 | 赵伟宏 | 2018-5-17 | 表结构更改 |
| 0.5 | 赵伟宏 | 赵伟宏 | 2018-6-26 | 最终更新，根据实际情况修改文档信息 |

|  |  |  |
| --- | --- | --- |
| 文件状态：  [ ] 草稿  [√] 正式发布  [ ] 正在修改 | 文件标识： | Company-Project-SD-DATABASE |
| 当前版本： | 0.5 |
| 作 者： | G07赵伟宏 |
| 完成日期： | 2018.06.27 |

目 录

[版 本 历 史 1](#_Toc517903875)

[0. 文档介绍 3](#_Toc517903876)

[0.1 文档目的 3](#_Toc517903877)

[0.2 文档范围 3](#_Toc517903878)

[0.3 读者对象 3](#_Toc517903879)

[0.4 参考文献 3](#_Toc517903880)

[0.5 术语与缩写解释 3](#_Toc517903881)

[1. 数据库环境说明 4](#_Toc517903882)

[2. 数据库的命名规则 4](#_Toc517903883)

[1.操作规范 4](#_Toc517903884)

[2.数据库表命名规范: 4](#_Toc517903885)

[3.表字段命名规范： 5](#_Toc517903886)

[4.表设计规范： 5](#_Toc517903887)

[5.存储过程命名规范 6](#_Toc517903888)

[6.存储过程设计规范 6](#_Toc517903889)

[7.视图命名规范 7](#_Toc517903890)

[8.视图设计规范 7](#_Toc517903891)

[9.触发器命名规范 7](#_Toc517903892)

[10.触发器设计规范 7](#_Toc517903893)

[3. 逻辑设计 8](#_Toc517903894)

[4. 物理设计 8](#_Toc517903895)

[表汇总 9](#_Toc517903896)

[表login 9](#_Toc517903897)

[表user 9](#_Toc517903898)

[5. 安全性设计 9](#_Toc517903899)

[5.1 防止用户直接操作数据库的方法 9](#_Toc517903900)

[5.2 用户帐号密码的加密方法 9](#_Toc517903901)

[5.3 角色与权限 10](#_Toc517903902)

[6. 优化 10](#_Toc517903903)

[7. 数据库管理与维护说明 10](#_Toc517903904)

# 0. 文档介绍

## 0.1 文档目的

具体介绍数据库等相关信息。

## 0.2 文档范围

数据库相关内容

## 0.3 读者对象

开发人员

## 0.4 参考文献

张海潘，牟永敏。软件工程导论（第6版） 北京：清华大学出版社，2013

sinat\_29519243，（转）MySQL数据库命名规范及约定，https://blog.csdn.net/sinat \_2951924 3/article/details/70187040，2018.5.19

## 0.5 术语与缩写解释

|  |  |
| --- | --- |
| **缩写、术语** | **解 释** |
| 暂无 |  |
|  |  |
|  |  |
|  |  |
| … |  |

# 1. 数据库环境说明

数据库系统：mysql

设计工具：Processon

编程工具：visual studio code

# 2. 数据库的命名规则

## 1.操作规范

1. 如无备注，则表中的第一个id字段一定是主键且为自动增长；

2. 如无备注，则数值类型的字段请使用UNSIGNED属性；

3. 如无备注，排序字段order\_id在程序中默认使用降序排列；

4. 如无备注，所有字段都设置NOT NULL，并设置默认值；

5. 如无备注，所有的布尔值字段，如is\_hot、is\_deleted，都必须设置一个默认值，并设为0；

6. 所有的数字类型字段，都必须设置一个默认值，并设为0；

7. 针对varchar类型字段的程序处理，请验证用户输入，不要超出其预设的长度；

8. 建表时将数据字典中的字段中文名和属性备注写入数据表的备注中(“PK、自动增长”不用写)；

9. 如无说明，建表时一律采用innodb引擎；

## 2.数据库表命名规范:

(1)表名前应该加上前缀，表的前缀一个用系统或模块的英文名称缩写，前缀全部大写或首字母大写，表名中包含的单词首字母大写。

(2)数据库表名应该有意义，并且易于理解，最好使用可以表达功能的英文单词或缩写，如果用英文单词表示，建议使用完整的英文单词。

(3)表名不可以太长，最好不要超过3个英文单词长度（22个字母）。

(4)在数据库表命名时应该用英文单词的单数形式，如员工表命名：应该为Employee而不是Employees .

(5)如果是后台表命名时应该在表名基础上加上后缀\_b(back 首字母 )

(6)在表创建完成前，应该为表添加表的注释。

## 3.表字段命名规范：

(1)数据库表字段应该是有意义而且易于理解的，最好是能够表达字段含义的英文字母

（有人认为如果用英文单词作为字段，因为翻译工具不同，而字段不统一建议使用汉语拼音首字母缩写

；有人认为用汉语拼音缩写看起来不直观，老半天也不知道到底这个字段是干什么的）

(2)系统中所有属于内码，即仅用于标识唯一性和程序内部用到的标识性字段，字段名称建议取为 ID ，采用类型为整型或长整型.

(3)系统中属于是业务内的编号字段，代表一定业务信息，建议字段命名为

code , 如工作单编号wf\_code .

(4) 不要在数据库表字段（列名）中包含数据类型，如：datetime

(5)不要在数据库表字段（列名）命名时重复表名，可以使用表名首字母（不包含数据库表名前缀）

注意：

不要在数据库表字段（列名）命名时

不建议使用数据库关键字,如：name,time ,datetime password 等

## 4.表设计规范：

(1)所有字段在设计时，除以下数据类型timestamp、image、datetime、smalldatetime、uniqueidentifier、binary、sql\_variant、binary、varbinary外，必须有默认值。字符型的默认值为一个空字符值串’’；数值型的默认值为数值0；逻辑型的默认值为数值0；

其中：系统中所有逻辑型中数值0表示为“假”；数值1表示为“真”。

datetime、smalldatetime类型的字段没有默认值，必须为NULL。

(2)当字段定义为字符串形时建议使用varchar而不用nvarchar。

注：在MySQL5.0以上的版本中，varchar数据类型的长度支持到了65535，也就是说可以存放65532个字节的数据，起始位和结束位占去了3个字节。

(3)建议在大多数表中（如工作单），应都有以下字段：

字段名说明类型默认值

CreatorID 创建者 int

默认值为0

CreatedTime 创建时间 Datetime

默认值为NULL

(4)字段的描述

a.字段必须填写描述信息（注释）

b.尽量遵守第三范式的标准（3NF）

表内的每一个值只能被表达一次（列名不重复）

表内的每一行都应当被唯一的标示（标识唯一性，如自动增长 主键）表内不应该存储依赖于其他键的非键信息

(5)加索引规则

a.表建好后数据库自动为表生成一个索引（为自动增长的列生成唯一索引），如果在对这列添加索引，数据库会给一个警告,内容大概是，已经为这列添加了索引，建议修改索引名称和自动增长列名保持一致，为了方便使用。

b.如果在添加索引时，建议索引名称和数据库列名保持一致，为了方便使用

c.如果字段事实上是与其它表的关键字相关联而未设计为外键引用，需建索引。

d.如果字段与其它表的字段相关联，需建索引。

e.如果字段需做模糊查询之外的条件查询，需建索引。

f.除了主关键字允许建立簇索引外，其它字段所建索引必须为非簇索引。

## 5.存储过程命名规范

(1)存贮过程的命名请遵循以下命名规范：USP\_ + 系统模块缩写（与表前缀类似）+\_ + 功能标识 + 代表存贮过程操作的主要表名（不带前缀）或功能的英文单词或英文单词缩写。

如果一个存贮过程只对一个表进行操作，建议存贮过程的名称就用存贮过程所操作的表的表名（不带前缀）。这样有利于根据表名找到相应的存贮过程。例如：

用于新增的存贮过程USP\_MESSAGE\_Add\_Model

用于修改的存贮过程USP\_ MESSAGE\_Upt\_Model

用于删除的存贮过程USP\_ MESSAGE\_Del\_ Modele

注：USP是user stored procedure

缩写

## 6.存储过程设计规范

在存贮过程中必须说明以下内容：

(1)目的：说明此存贮过程的作用。

(2)作者：首次创建此存贮过程的人的姓名。在此请使用中文全名，不允许使用英文简称。

(3)创建日期：创建存贮过程时的日期。

(4)修改记录：

修改记录需包含修改顺序号、修改者、修改日期、修改原因，修改时不能直接在原来的代码上修改，也不能删除原来的代码，只能先将原来的代码注释掉，再重新增加正确的代码。修改顺序号的形式为：log1，log2，log3。。。，根据修改次数顺序增加，同时在注释掉的原来的代码块和新增的正确代码块前后注明修改顺序号。

(5)对存贮过程各参数及变量的中文注解。

建议：在数据库中创建一个文本文件保存创建脚本

## 7.视图命名规范

视图的命名请遵循以下命名规范：UV \_ + 系统模块缩写（与表前缀类似）+\_ + 功能标识 + 代表视图查询的主要表名（不带前缀）或功能的英文单词或英文单词缩写。

如果一个视图只对一个表进行查询，建议视图的名称就用视图所查询的表的表名（不带前缀）。这样有利于根据表名找到相应的视图。

注：UV是userView缩写

## 8.视图设计规范

在视图中必须说明以下内容：

(1)目的：说明此视图的作用。

(2)创建者：首次创建此视图的人的姓名。在此请使用中文全名，不允许使用英文简称。

(3)修改者、修改日期、修改原因：如果有人对此视图进行了修改，则必须在此视图的前面加注修改者姓名、修改日期及修改原因。

(4)对视图各参数及变量的中文注解

建议：在数据库中创建一个文本文件保存创建脚本

## 9.触发器命名规范

Insert触发器加'\_i'，Delete触发器加'\_d'，Update触发器加'\_u'

## 10.触发器设计规范

在视图中必须说明以下内容：

(1)目的：说明此触发器的作用。

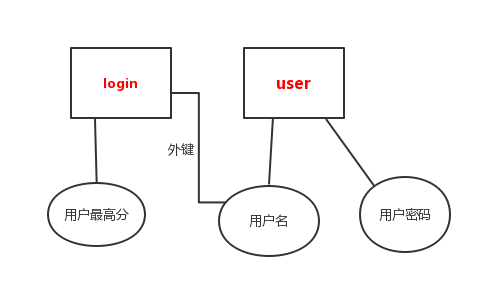
(2)创建者：首次创建此触发器的人的姓名。在此请使用中文全名，不允许使用英文简称。

(3)修改者、修改日期、修改原因：如果有人对此触发器进行了修改，则必须在此触发器的前面加注修改者姓名、修改日期及修改原因。

(4)对触发器各参数及变量的中文注解

# 3. 逻辑设计

***提示：****数据库设计人员根据需求文档，创建与数据库相关的那部分实体关系图（ERD）。如果采用面向对象方法（OOAD），这里实体相当于类（class）。*



# 4. 物理设计

***提示：***

*（1）主要是设计表结构。一般地，实体对应于表，实体的属性对应于表的列，实体之间的关系成为表的约束。逻辑设计中的实体大部分可以转换成物理设计中的表，但是它们并不一定是一一对应的。*

*（2）对表结构进行规范化处理（第三范式）。*

### 表汇总

|  |  |
| --- | --- |
| 表名 | 功能说明 |
| login | 用户id和密码 |
| user | 用户id和最高分 |

### 表login

|  |  |  |  |
| --- | --- | --- | --- |
| 表名 | login | | |
| 列名 | 数据类型（精度范围） | 空/非空 | 约束条件 |
| userid | Char[5] | 非空 | （主键）用户id |
| password | Char[5] | 非空 | 用户密码 |
| 补充说明 | 用户登录信息 | | |

### 表user

|  |  |  |  |
| --- | --- | --- | --- |
| 表名 | user | | |
| 列名 | 数据类型（精度范围） | 空/非空 | 约束条件 |
| userid | Char[5] | 非空 | （主键）用户ID |
| score | int | 非空（初始可为0） | 此用户游戏最高分 |
| 补充说明 | 用户游戏得分信息 | | |

# 5. 安全性设计

密码明文，安全性低。

## 5.1 防止用户直接操作数据库的方法

游戏界面只有登录，注册和开始游戏的按钮，并不能直接进行与访问数据库有关的操作。

## 5.2 用户帐号密码的加密方法

由于时间和小组成员能力均有限，没有设置加密方法。

## 5.3 角色与权限

***提示：****确定每个角色对数据库表的操作权限，如创建、检索、更新、删除等。每个角色拥有刚好能够完成任务的权限，不多也不少。在应用时再为用户分配角色，则每个用户的权限等于他所兼角色的权限之和。*

|  |  |  |
| --- | --- | --- |
| 角色 | 可以访问的表与列 | 操作权限 |
| 用户 | user(userid， score) | 查看 |
| login(userid,password) | 查看，更新 |
| 管理员 | user(userid， score) | 创建、检索、更新、删除 |
| login(userid,password) | 创建、检索、更新 |

# 6. 优化

***提示：****分析并优化数据库的“时－空”效率，尽可能地“提高处理速度”并且“降低数据占用空间”。*

*（1）分析“时－空”效率的瓶颈，找出优化对象（目标），并确定优先级。*

*（2）当优化对象（目标）之间存在对抗时，给出折衷方案。*

*（3）给出优化的具体措施，例如优化数据库环境参数，对表格进行反规范化处理等。*

|  |  |  |
| --- | --- | --- |
| 优先级 | 优化对象（目标） | 措施 |
| 2 | 数据库表 | 减少表结构，减少游戏加载项 |
| 1 | 数据库连接 | 创建数据库池，减少重复连接的创建，降低数据库操作时间 |
|  |  |  |

# 7. 数据库管理与维护说明

***提示：****在设计数据库的时候，及时给出管理与维护本数据库的方法，有助于将来撰写出正确完备的用户手册。*

1. 数据库密码存储应使用密文。
2. 数据库信息应做好备案以防不测。
3. 数据库池的最大连接数较低，有待扩充。