机构图标

001 E 基于安卓的线下交易购物平台

数据库设计报告

|  |  |  |
| --- | --- | --- |
| 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： | X.Y |
| 作 者： | 李达，袁沛霖 |
| 完成日期： | 2015-11-30 |

Company Information

版 本 历 史

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 版本/状态 | 作者 | 参与者 | 起止日期 | 备注 |
| RC1.0 | 李达  袁沛霖 | 李达  袁沛霖 | 2015.11.1～2015.11.30 | 无 |
|  |  |  |  |  |
|  |  |  |  |  |

目 录

[0. 文档介绍 4](#_Toc436567746)

[0.1 文档目的 4](#_Toc436567747)

[0.2 文档范围 4](#_Toc436567748)

[0.3 读者对象 4](#_Toc436567749)

[0.4 参考文献 4](#_Toc436567750)

[0.5 术语与缩写解释 4](#_Toc436567751)

[1. 数据库环境说明 5](#_Toc436567752)

[2. 数据库的命名规则 5](#_Toc436567753)

[3. 逻辑设计 5](#_Toc436567754)

[4. 物理设计 6](#_Toc436567755)

[4.0 表汇总 6](#_Toc436567756)

[4.1 表A 6](#_Toc436567757)

[4.2 表B 6](#_Toc436567758)

[4.3 表C 7](#_Toc436567759)

[4.4 表D 7](#_Toc436567760)

[4.5 表E 7](#_Toc436567761)

[5. 安全性设计 7](#_Toc436567762)

[5.1 防止用户直接操作数据库的方法 7](#_Toc436567763)

[5.2 用户帐号密码的加密方法 8](#_Toc436567764)

[5.3 角色与权限 8](#_Toc436567765)

[6. 优化 8](#_Toc436567766)

[7. 数据库管理与维护说明 8](#_Toc436567767)

# 0. 文档介绍

## 0.1 文档目的

说明本项目的数据库设计

## 0.2 文档范围

本文档主要包括数据库环境搭建，数据库命名规则，逻辑设计，物理设计等。

## 0.3 读者对象

软件测试人员，软件开发人员，项目经理

## 0.4 参考文献

**[1]** 李达，袁沛霖，“001 E基于安卓的线下交易购物平台”开题报告

**[2]** 李达，袁沛霖，“001 E基于安卓的线下交易购物平台”需求报告

**[3]** 李达，袁沛霖，“001 E基于安卓的线下交易购物平台”分析报告

**[4]** 李达，袁沛霖，“001 E基于安卓的线下交易购物平台”设计报告

## 0.5 术语与缩写解释

|  |  |
| --- | --- |
| **缩写、术语** | **解 释** |
| SPP | 精简并行过程，Simplified Parallel Process |
| SD | 系统设计，System Design |
|  |  |
|  |  |
| … |  |

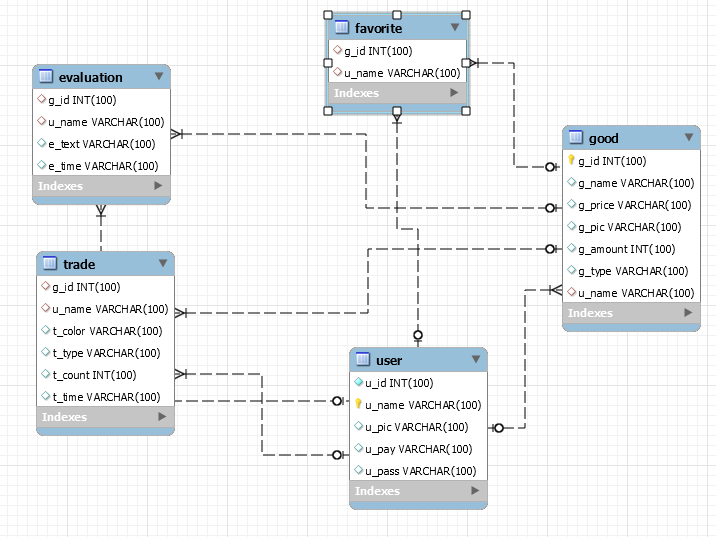
# 1. 数据库环境说明

本项目多采用的数据库系统为MySQL Server 5.6，使用SQLyog作为数据库设计工具。

# 2. 数据库的命名规则

1. 表名全部小写，在数据库表命名时应该用英文单词的单数形式。
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. 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。
4. 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。
5. 不要在数据库表字段（列名）中包含数据类型。
6. 当字段定义为字符串形时建议使用varchar而不用nvarchar。

# 3. 逻辑设计



# 4. 物理设计

## 4.0 表汇总

|  |  |
| --- | --- |
| 表名 | 功能说明 |
| user | 记录用户信息 |
| good | 记录商品信息 |
| favorite | 记录收藏信息 |
| trade | 记录交易购物信息 |
| evaluation | 记录评价信息 |

## 4.1 表A

|  |  |  |  |
| --- | --- | --- | --- |
| 表名 | user | | |
| 列名 | 数据类型（精度范围） | 空/非空 | 约束条件 |
| u\_id | int(100) | not null | auto\_increment |
| u\_name | varchar(100) | not null | primary key |
| u\_pass | varchar(100) | not null |  |
| u\_pay | varchar(100) | not null |  |
| 补充说明 |  | | |

## 4.2 表B

|  |  |  |  |
| --- | --- | --- | --- |
| 表名 | good | | |
| 列名 | 数据类型（精度范围） | 空/非空 | 约束条件 |
| g\_id | int(100) | not null | auto\_increment, primary key |
| g\_name | varchar(100) | not null |  |
| g\_price | varchar(100) | not null |  |
| g\_pic | varchar(100) | not null |  |
| g\_amount | int(100) | not null |  |
| g\_type | varchar(100) | not null |  |
| u\_name | varchar(100) | not null | foreign key |
| 补充说明 |  | | |

## 4.3 表C

|  |  |  |  |
| --- | --- | --- | --- |
| 表名 | favorite | | |
| 列名 | 数据类型（精度范围） | 空/非空 | 约束条件 |
| g\_id | int(100) | not null | foreign key |
| u\_name | varchar(100) | not null | foreign key |
| 补充说明 |  | | |

## 4.4 表D

|  |  |  |  |
| --- | --- | --- | --- |
| 表名 | trade | | |
| 列名 | 数据类型（精度范围） | 空/非空 | 约束条件 |
| g\_id | int(100) | not null | foreign key |
| u\_name | varchar(100) | not null | foreign key |
| t\_color | varchar(100) | not null |  |
| t\_type | varchar(100) | not null |  |
| t\_count | int(100) | not null |  |
| t\_time | date | not null |  |
| 补充说明 |  | | |

## 4.5 表E

|  |  |  |  |
| --- | --- | --- | --- |
| 表名 | evaluation | | |
| 列名 | 数据类型（精度范围） | 空/非空 | 约束条件 |
| g\_id | int(100) | not null | foreign key |
| u\_name | varchar(100) | not null | foreign key |
| e\_text | varchar(100) | not null |  |
| e\_time | date | not null |  |
| 补充说明 |  | | |

# 5. 安全性设计

## 5.1 防止用户直接操作数据库的方法

为防止用户非法登录，在用户创建的同时，每个用户都必须确定自己的密码。用户只有在登录系统以后，才可进行数据操作（比如：查询、删除、添加以及更新）。在用户登录时，如果出现错误给出相应的提示。

此外，为保证数据库内数据的安全性，要求在数据输入和更改时进行数据检测（如：范围、格式等），可以通过正则表达式实现。一致性方面，主要考虑有多个用户进行数据操作时，对数据库更新、修改等要求互斥访问，有系统信息提示。

## 5.2 用户帐号密码的加密方法

在用户密码输入时，要求输入两次密码，只有在两次输入密码一致的时候，才允许密码入库，否则，要求用户重新输入，页面显示为\*号。

在入库前还需要用加密算法，对密码进行加密，将密文存入数据库。在用户登录时，将数据库中密文取出，用相应的解密算法解密以后，与用户输入密码进行匹配，一致则进入系统，不一致则要求用户重新输入。此外，如果用户需要更高的权限，可以通知管理员，与管理员协商，给予相应的权限。

## 5.3 角色与权限

|  |  |  |
| --- | --- | --- |
| 角色 | 可以访问的表与列 | 操作权限 |
| 普通用户 | user | insert |
| good | insert, update, delete, select |
| favorite | insert, delete, select |
| trade | insert, select |
| evaluation | Insert, select |

# 6. 优化

|  |  |  |
| --- | --- | --- |
| 优先级 | 优化对象（目标） | 措施 |
| 无 | 无 | 无 |
|  |  |  |
|  |  |  |

# 7. 数据库管理与维护说明

1. 保证数据被合法使用

关系数据库存储的数据：通过关系数据库系统软件的安全机制来控制数据库、表的操作权限设定来达到数据合法使用的目的，包括对整个数据库的访问；对数据库中某个或多个表的访问等。

用户类型：

普通用户：只能通过应用程序查询相关数据

数据管理员：对数据进行完全控制，具有数据更新、备份、删除、分配权限等功能

未授权的用户拒绝访问系统中的数据。

1. 数据库的联机重做日志，记录对数据库所做的修改操作
2. 定期导出用户数据。