

小跳蚤校园二手交易平台

数据库设计说明书

项目团队：努力完成组

组长：吴彪

组员：李杰 彭亮 谢豪

组员：李莹莹 李德瑜

组员：张洁 廖平

指导老师：彭琛

目录

1 引言.....	3
1.1 编写目的.....	3
1.2 项目背景.....	3
1.3 参考资料.....	3
2 外部设计.....	3
3 结构设计.....	4
3.1 概念结构设计.....	4
3.1.1 User (普通用户表)	4
3.1.2 Administrator (管理员表)	4
3.1.3 Goods (商品表)	5
3.1.4 Order (订单表)	5
3.1.5 Comment (评论表)	5
3.1.6 Appeal (申诉表)	6
3.2 逻辑结构设计.....	6
4 验收验证标准.....	7
4.1 数据库基本功能验收.....	7
4.2 数据库数据体的验收.....	7
4.2.1 字段的原子性.....	7
4.2.2 主键设计.....	7
4.2.3 字段使用次数.....	8
4.2.4 NULL 值.....	8
4.2.5 资源存储.....	8
4.3 数据库安全性的验收.....	8
4.3.1 用户标识符和鉴别.....	8
4.3.2 存取控制.....	8
4.3.3 审计.....	8
4.3.4 数据加密.....	9
4.4 数据库运行情况的验收.....	9

1 引言

1.1 编写目的

本说明是校园二手交易项目数据库总体设计和实现说明，指明数据库管理系统、所设计的数据库的中、英文名称，将数据分析的结果进一步整理形成最终的计算机模型，以便开发人员建立物理数据库。

软件开发小组的每一位参与开发成员应阅读本说明，以清楚产品在数据库技术方面的要求和实现策略。

1.2 项目背景

随着消费主义的吹鼓，越来越多人不加以思考，仅凭一时兴起便购买许多对自己实际用处不大的东西，等热度一过，这些东西便会被闲置下来。另一方面，高校毕业生常常要丢弃许多保存完好但携带不便的物品，造成了极大的资源浪费。为此，我们团队准备开发一个针对校内的二手交易平台，来帮助同学们更好的处理闲置品。

校内的同学们可以通过这个平台来进行闲置物品的转让与购买物美价廉的物品，管理员分为超级管理员和普通管理员，不同管理员表现为权限不同，提高了信息交流效率，免于频繁翻阅空间、Q 群信息，节省了学生的交易、管理人员的时间和精力。

1.3 参考资料

[数据库设计说明书编写规范\(国标\)](#)

2 外部设计

数据库版本: MySQL 5.0.X 及以上

服务器: Tomcat

运行平台: Windows10

3 结构设计

3.1 概念结构设计

3.1.1 User (普通用户表)

名	类型	长度	小数点	允许空值 (
qq	varchar	20	0	<input type="checkbox"/>	1
password	varchar	20	0	<input type="checkbox"/>	
school	varchar	20	0	<input type="checkbox"/>	
userid	varchar	20	0	<input type="checkbox"/>	
sex	varchar	10	0	<input checked="" type="checkbox"/>	
sno	varchar	20	0	<input checked="" type="checkbox"/>	
spe	varchar	150	0	<input checked="" type="checkbox"/>	
url	varchar	20	0	<input checked="" type="checkbox"/>	

说明: qq 为用户的 QQ 邮箱, 每个用户唯一, 为主键。password 为用户密码, school 为用户密码,

userid 为用户昵称、sex 为用户性别、sno 为用户编号、spe 为用户的简介描述、

url 为用户头像地址链接、address 为用户的收货地址、phone 为用户收货手机电话号码。

3.1.2 Administrator (管理员表)

名	类型	长度	小数点	允许空值 (
qq	varchar	20	0	<input type="checkbox"/>	1
password	varchar	20	0	<input type="checkbox"/>	
ano	varchar	20	0	<input type="checkbox"/>	

说明: qq 为管理员的 QQ 邮箱, 每个管理员唯一, 为主键。

password 为管理员密码,

ano 为管理员编号

3.1.3 Goods (商品表)

名	类型	长度	小数点	允许空值 (
qq	varchar	20	0	<input type="checkbox"/>	
attribute	varchar	20	0	<input checked="" type="checkbox"/>	
price	varchar	20	0	<input checked="" type="checkbox"/>	
des	varchar	50	0	<input checked="" type="checkbox"/>	
url	varchar	50	0	<input checked="" type="checkbox"/>	
name	varchar	20	0	<input type="checkbox"/>	
gno	varchar	20	0	<input type="checkbox"/>	1
count	varchar	20	0	<input checked="" type="checkbox"/>	

说明：qq 用来确定该商品属于哪个用户、attribute 为商品分类、prices 为价格、des 为商品描述、url 为商品图像的链接地址、name 为商品名字、gno 为商品编号，为主键，唯一。count 为商品数量。

3.1.4 Order (订单表)

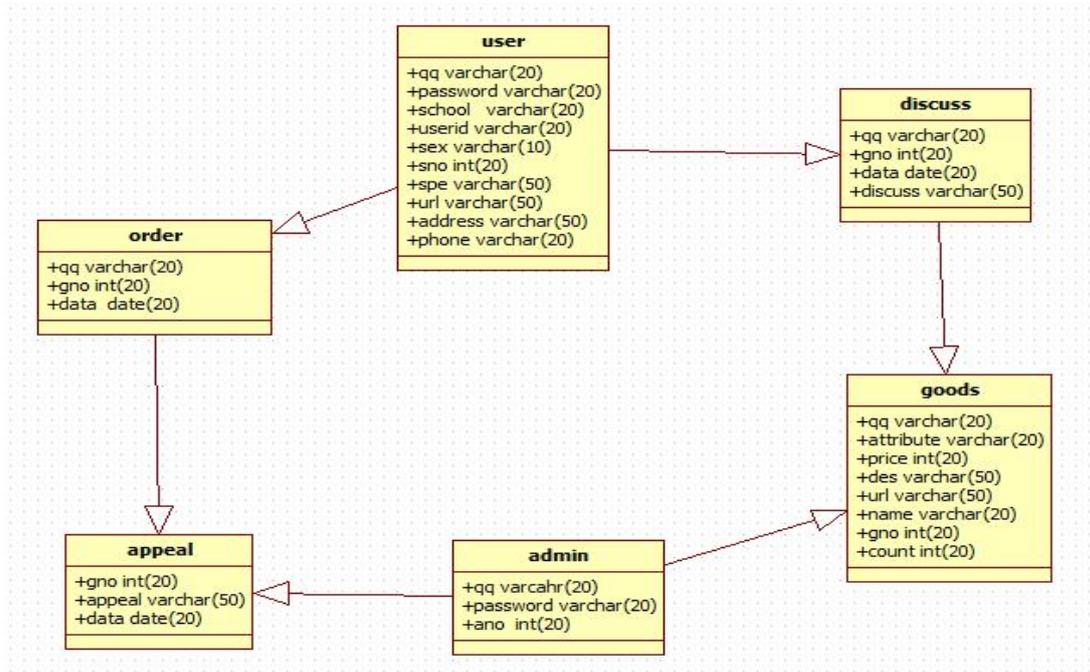
名	类型	长度	小数点	允许空值 (
qq	varchar	20	0	<input type="checkbox"/>	1
gno	varchar	20	0	<input checked="" type="checkbox"/>	
data	varchar	20	0	<input checked="" type="checkbox"/>	

说明：qq 绑定用户，为主键、不为空。、gno 为订单商品的编号、data 为订单时间。

3.1.5 Comment (评论表)

名	类型	长度	小数点	允许空值 (
qq	varchar	20	0	<input type="checkbox"/>	1
gno	varchar	20	0	<input checked="" type="checkbox"/>	
data	varchar	20	0	<input checked="" type="checkbox"/>	
discuss	varchar	50	0	<input checked="" type="checkbox"/>	

说明：qq 绑定用户，为主键，不为空。gno 为评论的商品编号、data 为评论时间、discuss 为评论描述。



4 验收验证标准

4.1 数据库基本功能验收

对每个表都可以实现正常的增、删、查、改即可

4.2 数据库数据体的验收

基于实现以下要求即完成验收

4.2.1 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。

4.2.2 主键设计

主键不要与业务逻辑有所关联。

4.2.3 字段使用次数

对于频繁修改的字段(一般是指状态类字段)最好用独立的数字或者单个字母去表示, 不用使用汉字或长字符的英文。

4.2.4 NULL 值

尽量不要有 null 值, 有 null 值的话, 数据库在进行索引的时候查询的时间更久, 从而浪费更多的时间, 可以在建表的时候设置一个默认值。

4.2.5 资源存储

数据库不要存储任何资源文件, 比如照片/视频/网站等, 可以用文件路径/外链用来代替, 这样可以在程序中通过路径, 链接等来进行索引。

4.3 数据库安全性的验收

实现以下至少两点即完成验收

4.3.1 用户标识符和鉴别

该方法由系统提供一定的方式让用户标识自己的名字或身份。每次用户要求进入系统时, 由系统进行核对, 通过鉴定后才能提供系统的使用权。

4.3.2 存取控制

通过用户权限定义和合法权检查确保只有合法权限的用户访问数据库, 所有未授权的人员无法存取数据。如: 超级管理员可以对管理员账户、权限进行操作分配, 普通管理员只能审核、处理投诉、封禁某个用户、发布公告等。

4.3.3 审计

建立审计日志, 把用户对数据库的所有操作自动记录下来放入审计日志中, DBA 可以利用审计跟踪的信息, 重现导致数据库现有状况的一系列事件, 找出非法存取数据的人, 时间和内容等。

4.3.4 数据加密

对存储和传输的数据进行加密处理，从而使得不知道解密算法的人无法获知数据内容。

4.4 数据库运行情况的验收

实现以下至少两点即完成验收

- 在数据库物理设计时，降低范式，增加冗余，少用触发器，多用存储过程
- 对数据库管理系统 DBMS 进行系统优化，即优化各种系统参数，如缓冲区个数
- 在使用面向数据的 SQL 语言进行程序设计时，尽量采取优化算法。