

# 《卡卡颂》项目

## 数据库设计说明书

unity 从入门到入土项目组

2021 年 4 月 22 日

# 目录

1.引言.....	2
1.1 编写目的.....	2
1.2 项目背景.....	2
1.3 定义.....	2
1.4 参考资料.....	2
2. 外部设计.....	3
2.1 标识符和状态.....	3
2.2 使用它的软件.....	3
2.3 约定.....	3
3. 结构设计.....	4
3.1 概念结构设计.....	4
3.2 逻辑结构设计.....	5
3.3 物理结构设计.....	5
4. 应用设计.....	6
4.1 数据字典设计.....	6
4.2 安全保密设计.....	8
5. 数据库验证验收标准.....	8
5.1 数据库数据体的验收.....	8
5.2 数据库安全性的验收.....	8

# 1. 引言

## 1.1 编写目的

编写本文档的目的是详细说明“卡卡颂”数据库设计。提供详细的数据库设计描述。用于指导该项目的开发与部署，作为重要的技术参考资料，为未来的运营维护及功能扩展作开发参考。

该文档适用于开发人员、测试人员、维护人员进行系统前期开发、后期数据库设计方向、服务器维护及功能扩展作参考。

## 1.2 项目背景

- 项目名称：《卡卡颂》游戏
- 项目发起人：“unity 从入门到入土”团队
- 项目开发者：“unity 从入门到入土”团队
- 数据库操作平台：
- 数据库名：carcassonne
- 适用平台：PC

## 1.3 定义

- 数据库(carcassonne)：保存系统数据的后台应用软件
- 表：不同记录的集合
- 记录：包含数个关联字段数据的集合
- 字段：详细数据项
- 类型：数据列的 type

## 1.4 参考资料

参考书籍：《构建之法》—邹欣，  
《unity 从入门到精通》—unity technologies  
参考资料：unity 官方教程—2D 游戏创作  
参考游戏：三国杀、卡卡颂实体版

## 2. 外部设计

### 2.1 标识符和状态

本系统采用 Navicat 12 为基本开发工具，数据库名称为 Carcassonne。数据库共有 8 个表，表名分别为 card 表，edge 表，game\_record 表，log 表，user 表

### 2.2 使用它的软件

本数据库支持 MySql 等数据库管理系统

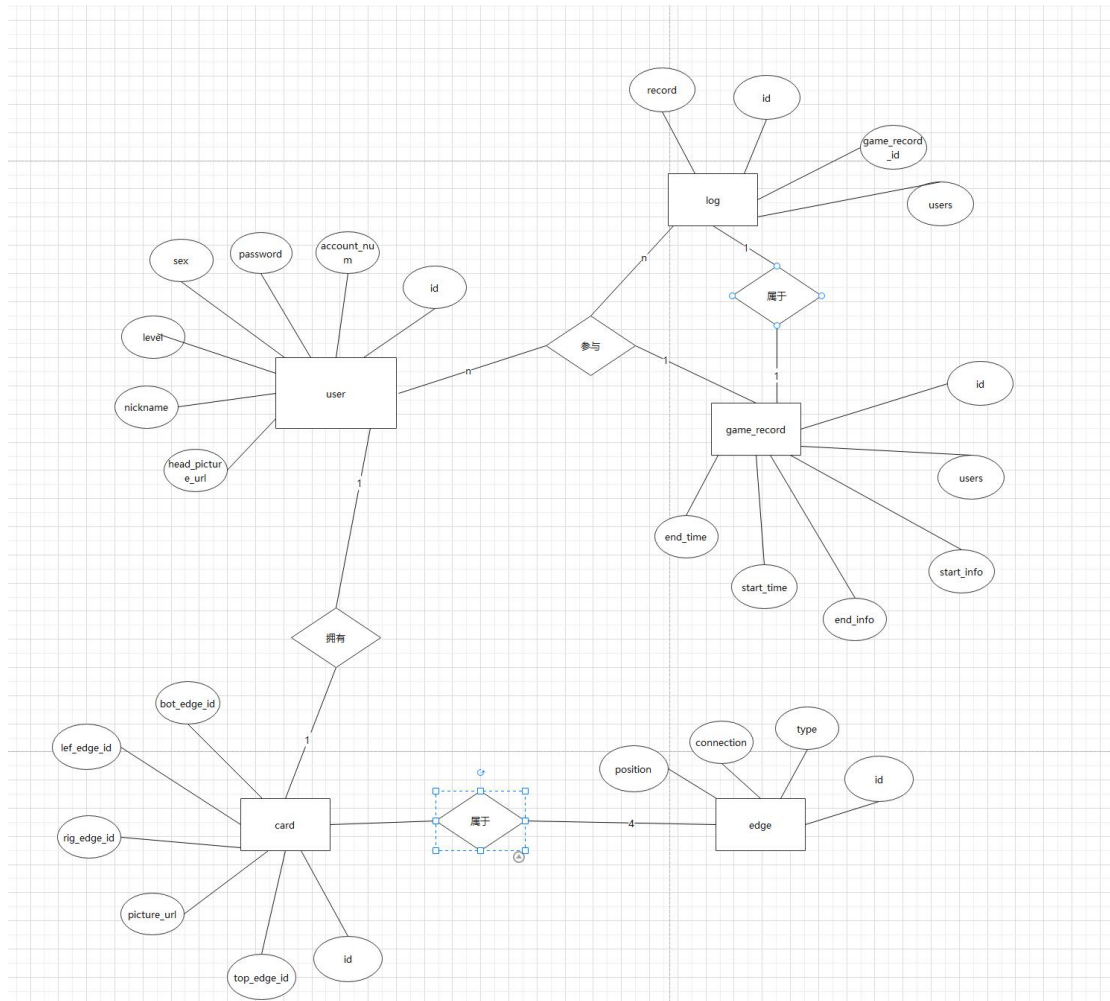
### 2.3 约定

字段名：一般以中文英译为名，多重意思每个英文单词间用“\_”隔开。所有数据表第一个字段都是系统内部使用主键列，自增字段，不可空，名称为：id，确保不把此字段暴露给最终用户。

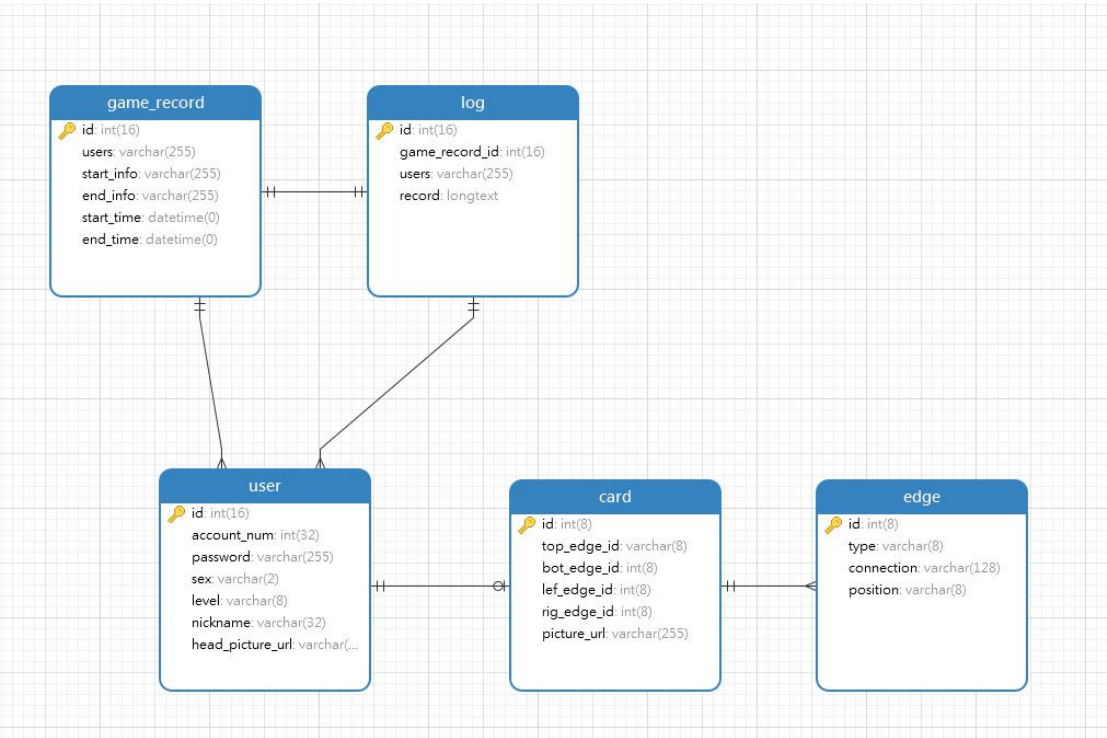
## 3. 结构设计

### 3.1 概念结构设计

#### E-R 图



### 3.2 逻辑结构设计



### 3.3 物理结构设计

#### 物理结构设计

卡片基础信息表

字段	索引	外键	触发器	选项	注释	SQL 预览
名					类型	
▶ id					int	长度 8, 小数点 0, 不是 null <input checked="" type="checkbox"/> , 虚拟 <input type="checkbox"/> , 键 <input checked="" type="checkbox"/> 1, 注释 card id 主键, 该表数据目前为只读
top_edge_id					varchar	长度 8, 小数点 0, 不是 null <input checked="" type="checkbox"/> , 虚拟 <input type="checkbox"/> , 注释 上边的id
bot_edge_id					varchar	长度 8, 小数点 0, 不是 null <input checked="" type="checkbox"/> , 虚拟 <input type="checkbox"/> , 注释 下边的id
lef_edge_id					varchar	长度 8, 小数点 0, 不是 null <input checked="" type="checkbox"/> , 虚拟 <input type="checkbox"/> , 注释 左边的id
rig_edge_id					varchar	长度 8, 小数点 0, 不是 null <input checked="" type="checkbox"/> , 虚拟 <input type="checkbox"/> , 注释 右边的id
picture_url					varchar	长度 255, 小数点 0, 不是 null <input checked="" type="checkbox"/> , 虚拟 <input type="checkbox"/> , 注释 该卡牌的图片资源 URL

卡片边类型表

字段	索引	外键	触发器	选项	注释	SQL 预览
名					类型	
▶ id					int	长度 8, 小数点 0, 不是 null <input checked="" type="checkbox"/> , 虚拟 <input type="checkbox"/> , 键 <input checked="" type="checkbox"/> 1, 注释 边id 目前为只读
type					varchar	长度 8, 小数点 0, 不是 null <input checked="" type="checkbox"/> , 虚拟 <input type="checkbox"/> , 注释 边的类型Grass: 草地, Road: 路, City: 城
connection					varchar	长度 128, 小数点 0, 不是 null <input checked="" type="checkbox"/> , 虚拟 <input type="checkbox"/> , 注释 连接情况, 表示除了边自身, 其他可能的连接,
position					varchar	长度 8, 小数点 0, 不是 null <input checked="" type="checkbox"/> , 虚拟 <input type="checkbox"/> , 注释 top, bot, lef, rig 分别表示上下左右

对局结果记录表

字段	索引	外键	触发器	选项	注释	SQL 预览
名						
id					类型	长度
users					varchar	255
start_info					varchar	255
end_info					varchar	255
start_time					datetime	0
end_time					datetime	0

对局日志记录表

字段	索引	外键	触发器	选项	注释	SQL 预览
名						
id					类型	长度
game_record_id					int	16
users					varchar	255
record					longtext	0

用户表

字段	索引	外键	触发器	选项	注释	SQL 预览
名						
id					类型	长度
account_num					varchar	32
password					varchar	255
sex					varchar	2
level					varchar	8
nickname					varchar	32
head_picture_url					varchar	255

## 4. 应用设计

### 4.1 数据字典设计

表名	卡片基础信息表（card）			
描述	记录所有卡片的基本信息			
字段	中文名	类型	能否为空	备注
id	卡片 id	Int(8)	N	卡片 id，主键，该表数据目前为只读
top_edge_id	上边 id	varchar(8)	N	上边的 id
bot_edge_id	下边 id	varchar(8)	N	下边的 id
lef_edge_id	左边 id	varchar(8)	N	左边的 id
rig_edge_id	右边 id	varchar(8)	N	右边的 id

picture_url	卡片图形 url	varchar(255)	N	该卡牌的图片资源 url
-------------	----------	--------------	---	--------------

表名	卡片边类型表（edge）			
描述	记录卡片边的信息			
字段	中文名	类型	能否为空	备注
id	边界 id	Int(8)	N	边 id，主键，目前为只读
type	种类	varchar(8)	N	边的类型 Grass：草地，Road：路，City：城
connection	连接	varchar(128)	N	连接情况，表示除了边自身，其他可能的连接，示例 JSON 保存：若自身 position 为 top，他的连接 { “ bot ” : “ true ” , “ lef ” : “ false ” , “ rig ” : “ false ” } 表示该边为上边，与底边连接，不与左边与右边连接
position	位置	varchar(8)	N	top, bot, lef, rig 分别表示上下左右

表名	对局结果记录表（game_record）			
描述	记录对局结果的信息			
字段	中文名	类型	能否为空	备注
id	对局结果 id	Int(16)	N	对局结果的 id
users	用户	varchar(255)	N	参与的玩家，JSON 格式保存，示例： {"user1":3,"user2":12,"user3":1,"user4":14,"user":0}表示参与玩家为四名，id 分别为 3，12，1，14；0 表示空位
start_info	开始信息	varchar(255)	N	开始对局时，各种参数，玩家信息等，格式待定
end_info	结束信息	varchar(255)	N	结束对局时的结算情况，以及玩家信息
start_time	开始时间	datetime	Y	开始时间的时间戳



end_time	结束时间	datetime	Y	结束时间的时间戳
----------	------	----------	---	----------

表名	对局日志记录表 (log)			
描述	记录对局中的信息			
字段	中文名	类型	能否为空	备注
id	对局日志 id	int(16)	N	对局日志的 id
game_record_id	种类	int(16))	N	对应的战绩 id
users	连接	varchar(255)	N	参与玩家，与 game_record 的 users 一致，为方便查看多设置一个。
record	位置	longtext	N	日志，本局对战的所有操作

## 4.2 安全保密设计

对重要信息如用户密码使用 SHA-1 加密保存

# 5. 数据库验证验收标准

## 5.1 数据库数据体的验收

- 1、保证每列的原子性，即要符合第一范式。
- 2、表中应该避免可为空的列
- 3、表中记录应该有一个唯一的标识符。

## 5.2 数据库安全性的验收

- 1、登录鉴别：用户要求进入系统时，由系统进行核对，通过鉴定后才提供系统的使用权。
- 2、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。

- 3、数据加密：对部分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的内容。