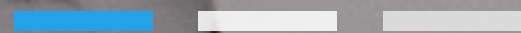


# 系统设计和数据库设计答辩

福大周润发队



# 目录

---

01

系统设计

02

数据库设计

# 系统设计



引言



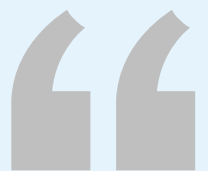
接口设计

总体设计



非功能特  
性设计





## 引言

本软件用于休闲娱乐，主要支持window系统，是一个安全，益智，考验画工和猜谜能力的益智桌游





# 总体设计

## 01

### 设计背景

本软件的建设背景，为fzu的大三软件工程学生共9人主要致力于打造一款可联网的益智类桌游。



## 02

### 集中开发环境

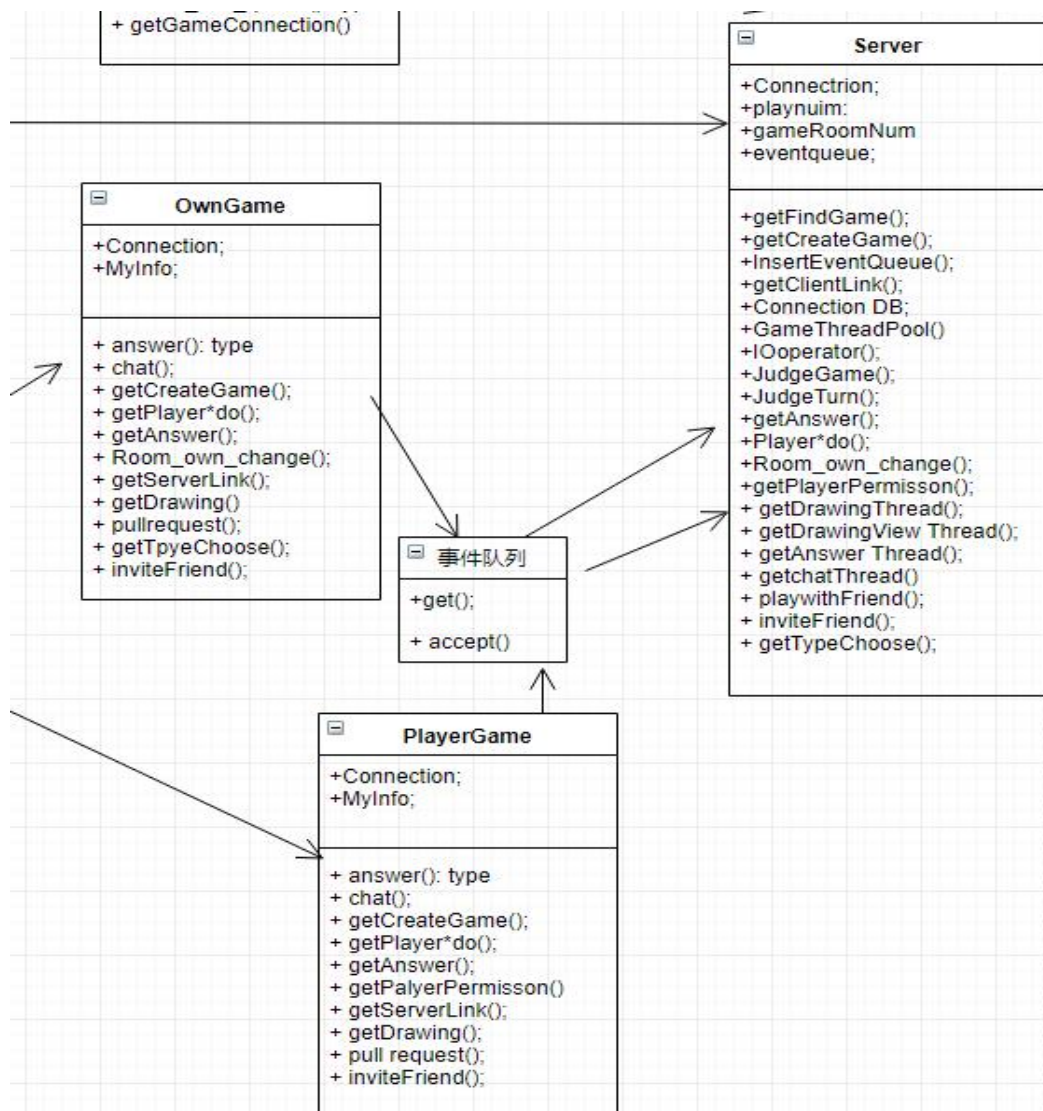
统一使用eclipse java8 , mysql8开发，涉及到多线程，服务器/客户端开发，javafx。服务端架构在linux。

# 总体设计

## 3 游戏内设计和实现

- 使用一个线程管理画板，实现画板的同步使用java中 canvas，每次绘画的时候，都会把画的数据记录下来，发给服务器。服务器再发送给房间内的其他玩家和监管。精确到每个像素。
- 如果要减轻服务器的压力，可以设置传递的方式，如果希望流畅，则可以设置采样的质量，（不同像素间的顺序超过某一值的时候，打包传递不同点的数据，再在其他玩家客户端上连接起来。）要实现清晰的画板视图，则采样的密集度就大一点，精确到像素级别。
- 再用一个线程控制送礼，用一个线程控制聊天，再用一个线程控制回答问题。由各自线程和事件队列交互。再由各自的线程完成各自的请求。

# 游戏内设计 UML图



# 总体设计

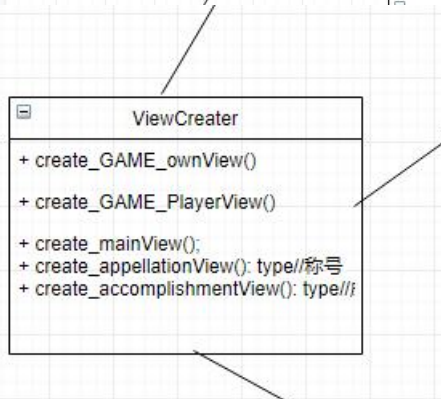
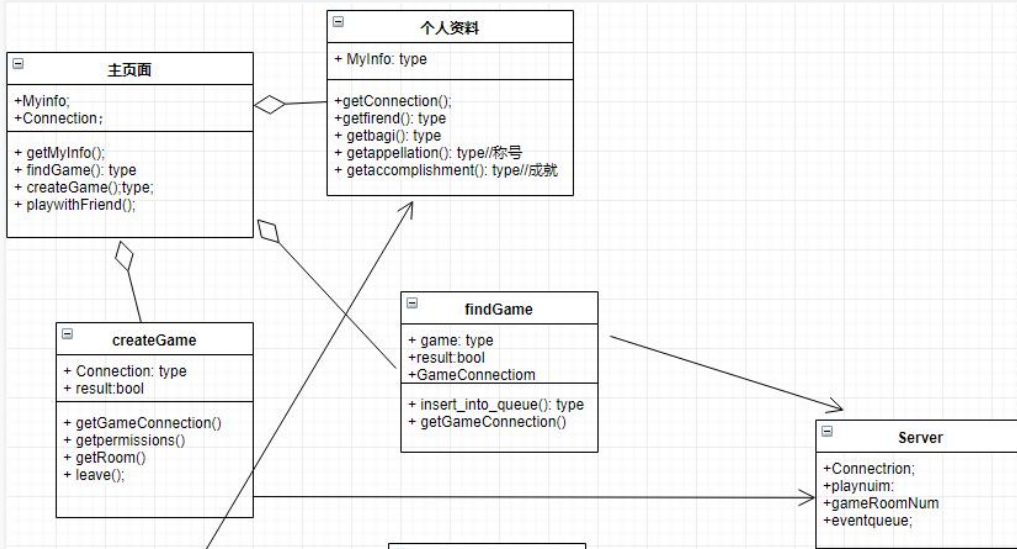
## 4 游戏外设计

- 游戏外的实现，主要通过createview获取各种不同的视图，不同的资料通过事件队列像服务器获取，通过不同类封装的方法，例如findgame来实现，事件的处理。

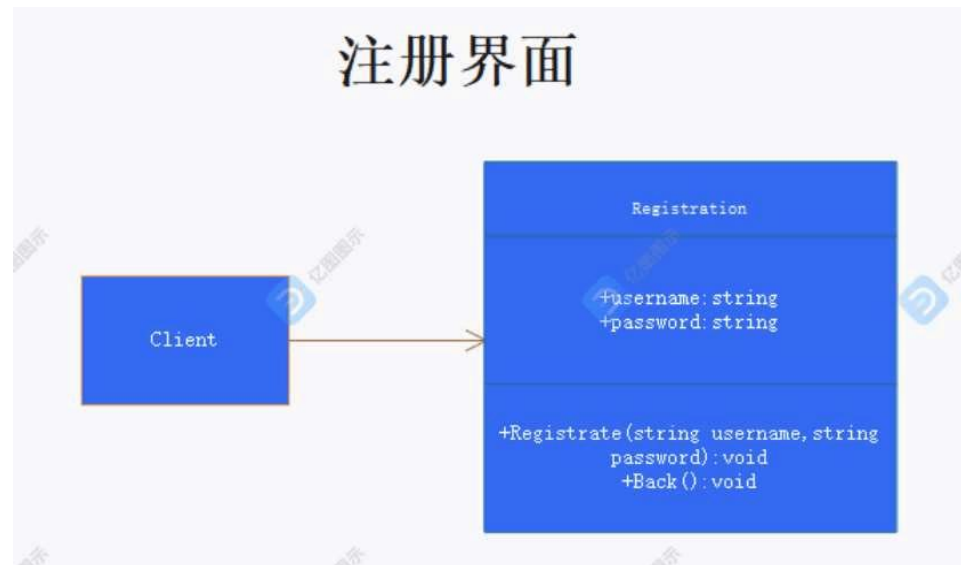


# 游戏外设计

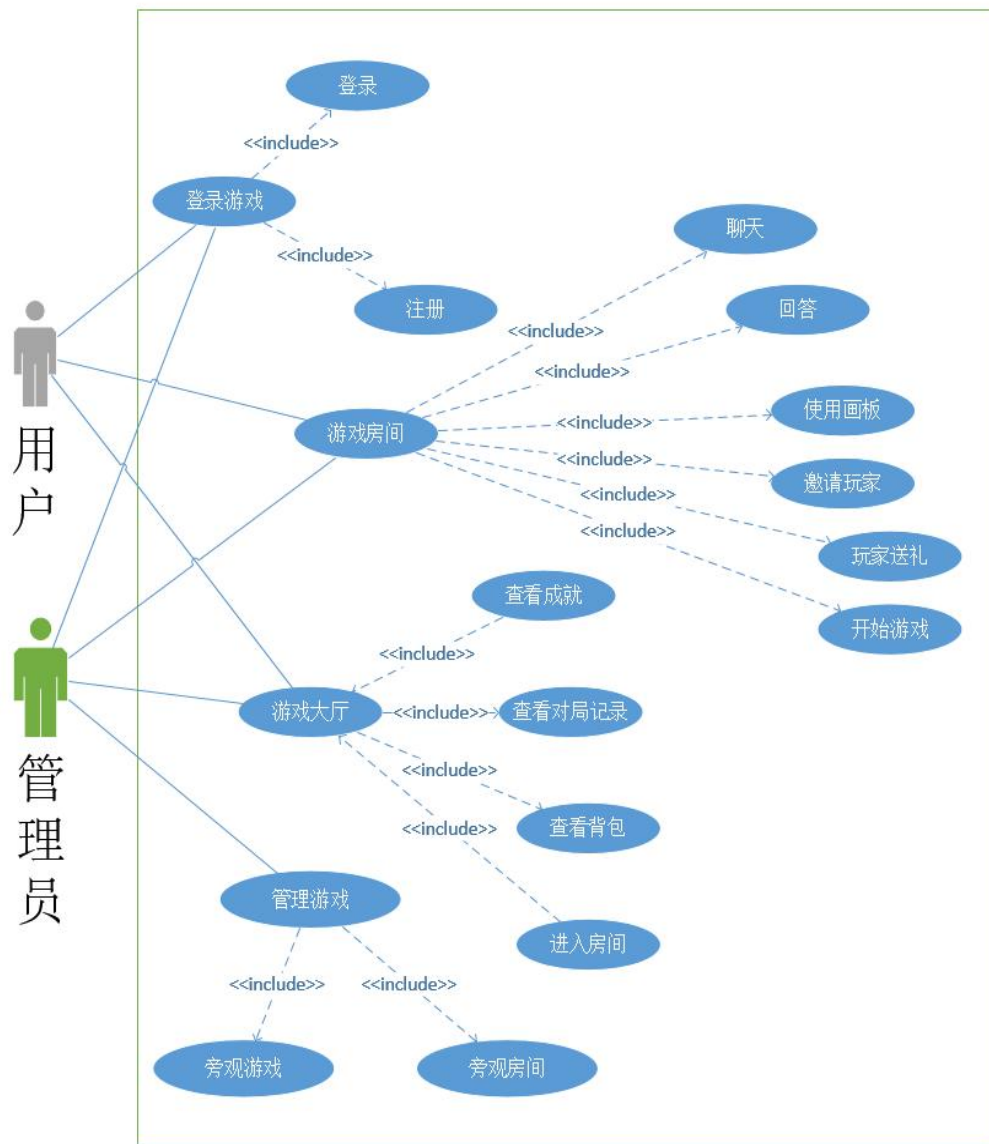
## UML图



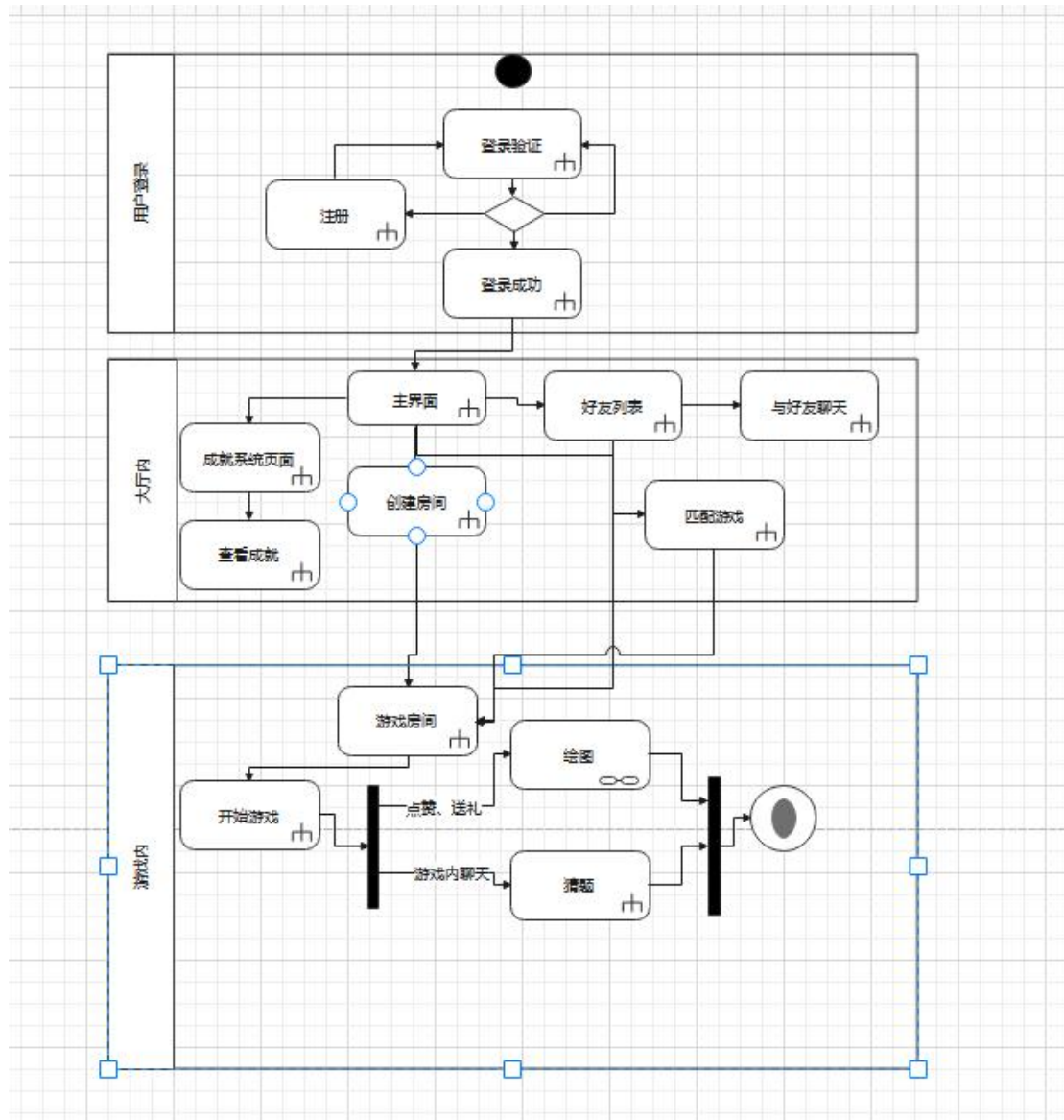
## 注册界面



# 其他UML图



# 其他UML图



# 接口设计

01

外部接口

02

内部接口

# 接口设计

- 外部接口：打算设计一个包含玩家信息的接口，封装了玩家的基本信息，可供玩家在非客户端登陆的时候，访问自己的信息，比如（网页，微信小程序）；
- 内部接口：在客户端和服务端，分别都有事件接受队列，事件发送队列。用于发送请求和处理请求，防止用户发送请求和接收请求而导致卡死。设计一个事件队列接口 `EventfQueue`；然后分别由发送队列和接收队列去继承它，用于处理客户端和服务端的交互。

# 非功能特性设计

01

兼容性

02

安全性

03

运行效率

04

拓展能力



# 非功能特性设计

## 一. 兼容性

- 系统可在window7 , window8, window10, window xp;甚至是具有图形化界面的linux系统中运行。

## 二. 安全性

- 登陆传输的账号资料, 通过某种加密方式传输 (比如 AES) , 防止电脑上的监听软件直接获取明文密码。

# 非功能特性设计

## 三、运行效率

通过，本地缓存，缓存常用用户的较大数据资料（例如用户头像，主题），这样在载入的时候，如果你不曾获取，此资料，则发出申请获取资料，当下一次再次载入的时候，再次发起获取，但传输的数据是时间，由服务器判断是否发生过更改，如果不曾发生更改，则直接通过缓存导入，否则更新数据。这样可以提升平均载入的效率，防止因为加载过大的资料，而发生卡顿；

# 非功能特性设计

## 四．拓展能力

如果后期用户量大，系统的服务端事件队列太频繁插入（入队比出队快）会导致事件处理延迟，可将事件队列拓展为队列池，入队的时候就会由队列池判断入哪个队伍，由队列池负责事件入队安排。即可有多个事件队列



# 数据库设计



- 概述
- 约定设计
- 结构设计
- 非功能特性设计

# 概述

---

你画我猜数据库主要用于处理和你画我猜游戏有关的所有数据存储，信息记录等



# 约定设计

## 1 标识符与状态

数据库标识符 dng 用户名：admin，密码：admin  
权限：全部

## 2 约定说明

除了描述性文本之外，其他的属性不允许为空，描述性文本，比如item表中的comment



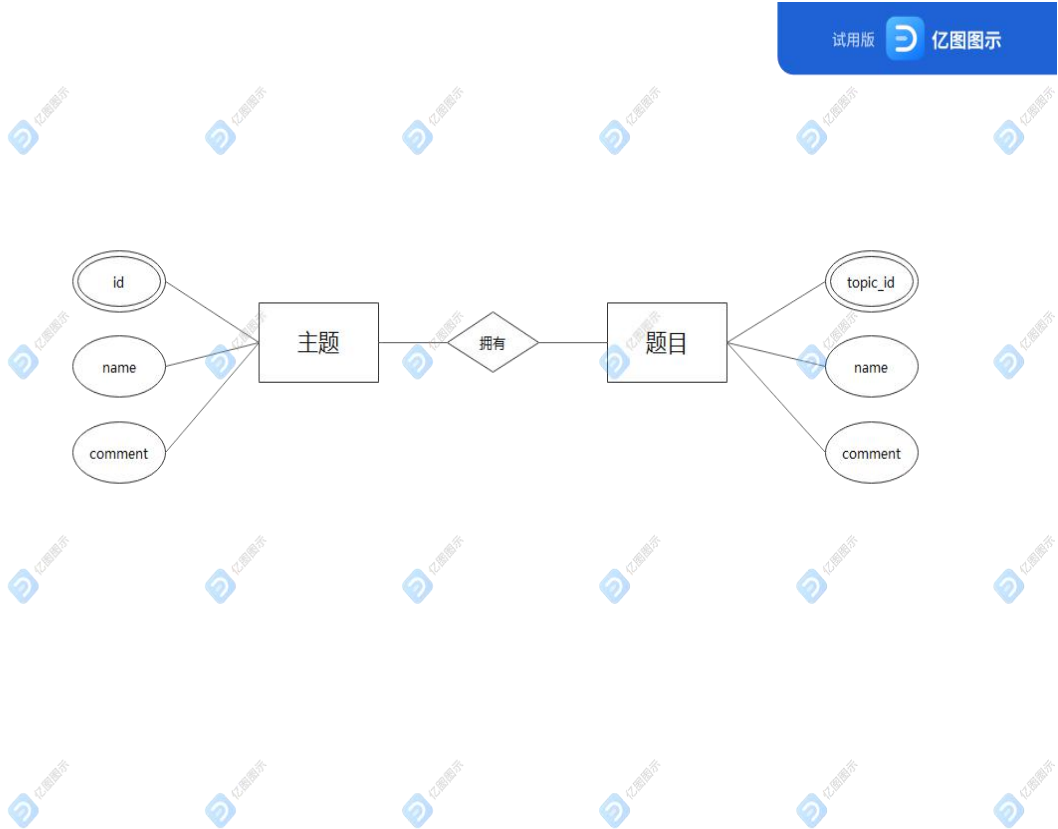
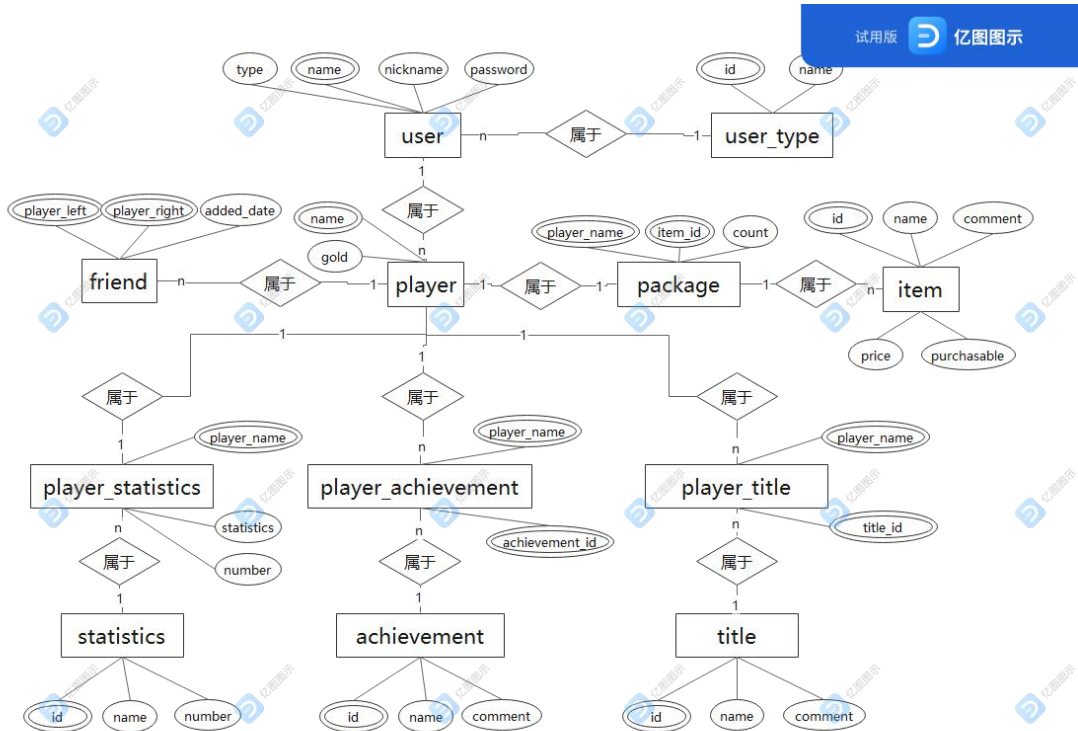
# 结构设计

- 概念结构设计

- 表设计

- 表关联设计

# 概念结构设计E-R图



# 表设计

- statistics
- achievement
- item
- package
- player
- player\_achievement
- player\_statistics
- player\_title
- title
- user
- user\_type



Table Name: user

Charset/Collation: utf8mb4

Comments:

Column Name	Datatype
name	CHAR(20)
nickname	VARCHAR(45)
password	CHAR(32)
type	TINYINT(1)



Table Name: statistics

Charset/Collation: utf8mb4

Comments:

Column Name	Datatype	PK	NN
id	INT(10)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
name	VARCHAR(45)	<input type="checkbox"/>	<input checked="" type="checkbox"/>
number	INT(11)	<input type="checkbox"/>	<input checked="" type="checkbox"/>



Table Name: achievement

Charset/Collation: utf8mb4

Comments:

Column Name	Datatype	PK	NN
id	INT(10)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
name	VARCHAR(45)	<input type="checkbox"/>	<input checked="" type="checkbox"/>
comment	TINYTEXT	<input type="checkbox"/>	<input type="checkbox"/>



Table Name: item

Charset/Collation: utf8mb4

Comments:

Column Name	Datatype	PK	NN	UQ	B	UN
id	INT(10)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
name	VARCHAR(45)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
comment	TINYTEXT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
price	INT(10)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
purchasable	TINYINT(3)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

# 表设计



Table Name: package

Charset/Collation: utf8mb4

Comments:

Column Name	Datatype	PK	NN
💡 player_name	CHAR(20)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
💡 item_id	INT(10)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
💎 count	INT(10)	<input type="checkbox"/>	<input checked="" type="checkbox"/>



Table Name: player\_statistics

Charset/Collation: utf8mb4

Comments:

Column Name	Datatype	PK	NN
💡 player_name	CHAR(20)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
💎 statistics	INT(10)	<input type="checkbox"/>	<input checked="" type="checkbox"/>
💎 number	INT(11)	<input type="checkbox"/>	<input checked="" type="checkbox"/>



Table Name: player

Charset/Collation: utf8mb4

Comments:

Column Name	Datatype	PK	NN
💡 name	CHAR(20)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
💎 gold	INT(11)	<input type="checkbox"/>	<input checked="" type="checkbox"/>



Table Name: player\_title

Charset/Collation: utf8mb4

Comments:

Column Name	Datatype	PK	NN
💡 player_name	CHAR(20)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
💡 title_id	INT(10)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>



Table Name: player\_achievement

Charset/Collation: utf8mb4

Comments:

Column Name	Datatype	PK	NN
💡 player_name	CHAR(20)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
💡 achievement_id	INT(11)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>



Table Name: title

Charset/Collation: utf8mb4

Comments:

Column Name	Datatype	PK	NN
💡 id	INT(10)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
💎 name	VARCHAR(45)	<input type="checkbox"/>	<input checked="" type="checkbox"/>
💎 comment	TINYTEXT	<input type="checkbox"/>	<input type="checkbox"/>

# 表设计

Table Name:

Charset/Collation:

Comments:

Column Name	Datatype	PK	NN
id	TINYINT(1)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
name	VARCHAR(45)	<input type="checkbox"/>	<input checked="" type="checkbox"/>

Table Name:

Charset/Collation:

Comments:

Column Name	Datatype	PK	NN
player_left	CHAR(20)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
player_right	CHAR(20)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
added_date	DATETIME(1)	<input type="checkbox"/>	<input checked="" type="checkbox"/>

Table Name:

Charset/Collation:

Comments:

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI
id	INT(10)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
name	VARCHAR(45)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
comment	TINYTEXT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Table Name:

Charset/Collation:

Comments:

Column Name	Datatype	PK	NN	UQ	B	UN
topic_id	INT(10)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
name	VARCHAR(45)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
comment	TINYTEXT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

# 表之间的关联设计

Player\_achievement 表中存储的是 玩家id 具体获得 成就id，在achievement中存储的则是 所有的成就和成就id 这两张表中的achievementid 属性是相关联的，player\_title表和title同理。player\_statistics表和statistics表存储的是玩家的信息，通过player\_statistics中的具体信息和信息条数，例如n个小红花，来记录某个id的玩家已拥有的道具属性。玩家id相关联的表有friend和user表，一对好友left right和data来记录成为好友的时间，删除好友时只需要删除对应的关系表即可。



# 系统兼容性

安全性

运行效率

可扩展能力

权限设计

# 安全性

---

如果数据库保存了敏感的数据，如密码等，你可能想将这些数据以加密的形式保存在数据库中。这样即使有人进入了你的数据库，并看到了这些数据，也很难获得其中的真实信息。在应用程序的大量信息中，密码不应该以明文的形式保存，它们应该以加密的形式保存在数据库中。一般情况下，大多数系统，这其中包括MySQL本身都是使用哈希算法对敏感数据进行加密的。哈希加密是单向加密，也就是说，被加密的字符串是无法得到原字符串的。这种方法使用很有限，一般只使用在密码验证或其它需要验证的地方。在比较时并不是将加密字符串进行解密，而是将输入的字符串也使用同样的方法进行加密，再和数据库中的加密字符串进行比较。这样即使知道了算法并得到了加密字符串，也无法还原最初的字符串。本次的用户密码数据库，就打算采用这种加密方式

防止用户的账户密码泄露

# 运行效率

---

运行效率方面，由于用户大部分的信息都由两张表来映射，检索数据量小的player\_xx表可以直接返回用户是否拥有某条信息，比都放在一张表上检索速度相对的快。通过不同表之间的关系，可以更快的更新查找用户的信息。

# 可扩展能力

---

因为不同的称谓是用两张表关联关系来实现记录的，当一个玩家的已拥有称谓更新的时候，只需要添加对应的记录即可，不需要改变表的具体结构，也不需要新增哪些表。如果玩家消耗了某些道具，也只需要在对应的play\_xxx中减去对应消耗数量的number即可

# 权限设计

---

权限设计分为，一般用户，和管理员用户，管理员用户可在游戏内旁观游戏，游戏内部也会具体设计出非法绘画的批判标准，但还需要具体的管理员去到现场验证，只有管理员有权限，更改房间状态，更改玩家状态，例如永久封号，一般玩家只有一般的权限，哪怕是作为房主，也只有在开始游戏之后才会临时被赋予某些权限，但这些权限只和游戏有关，和数据库无关。在user表中有对应的type用于存储用户类型判断是否是管理员用户。

The background features several abstract geometric shapes: a large pink circle on the left, a large blue circle at the top center, a large grey circle at the top right, a large blue circle at the bottom center, and a large pink circle at the bottom right. There are also several small triangles in various colors (pink, blue, grey, black) scattered across the white background.

# Thanks