

# 用户/好友/关注体系设计

## 一、用户体系

### 用户类型

类型	status	说明	权限
游客用户	g	仅微信登录，未绑定手机号	只能围观比赛
正式用户	a	已绑定手机号	完整功能（创建比赛、建球队等）
占位用户	p	被他人手工添加（昵称+手机号）	无法登录，待认领合并
已合并用户	m	占位用户被认领后的状态	数据已迁移，仅保留记录
禁用用户	d	违规或异常账号	无法登录，无法被搜索

### 用户类型判断逻辑

判断用户类型（直接通过 status 字段）：

- status='g' → 游客用户 (guest)
- status='a' → 正式用户 (active)
- status='p' → 占位用户 (placeholder)
- status='m' → 已合并用户 (merged)
- status='d' → 禁用用户 (disabled)

### 禁用场景

- 用户违规（发布不当内容、恶意行为等）
- 账号异常（疑似被盗、批量注册等）
- 用户主动申请注销

### 用户表设计 user

```
CREATE TABLE `user` (
  `id` INT PRIMARY KEY AUTO_INCREMENT,
  `openid` VARCHAR(64) DEFAULT NULL COMMENT '微信openid',
  `unionid` VARCHAR(64) DEFAULT NULL COMMENT '微信unionid',
  `phone` VARCHAR(20) DEFAULT NULL COMMENT '手机号',
  `nickname` VARCHAR(50) NOT NULL COMMENT '昵称',
  `avatar` VARCHAR(255) DEFAULT NULL COMMENT '头像',
  `status` CHAR(1) DEFAULT 'g' COMMENT 'g:游客 a:正式用户 p:占位用户 m:已合并
d:禁用',
  `created_by` INT DEFAULT NULL COMMENT '占位用户的创建者ID',
  `is_vip` CHAR(1) DEFAULT 'n' COMMENT 'y:VIP用户(解锁更多隐私设置)',
  `created_at` DATETIME DEFAULT CURRENT_TIMESTAMP,
  `updated_at` DATETIME DEFAULT CURRENT_TIMESTAMP ON UPDATE
```

```

CURRENT_TIMESTAMP,
UNIQUE KEY `uk_openid`(`openid`),
UNIQUE KEY `uk_phone`(`phone`)
) COMMENT='用户表';

```

## 关于 UNIQUE 约束与 NULL :

- `openid` 和 `phone` 字段设置为 UNIQUE, 但允许 NULL
- MySQL 中多个 NULL 值不违反 UNIQUE 约束, 这是预期行为
- 占位用户(p)没有 `openid`; 已合并用户(m)的 `phone` 会被置空

## 占位用户合并逻辑

当用户A绑定手机号X时 :

1. 查找是否存在 `phone=X` 且 `status='p'` 的占位用户B
2. 如果存在, 执行数据迁移 :
  - 比赛记录、成绩数据 → 迁移到A
  - B被他人关注的关系 → `target_id` 改为A (保留 `is_special` 特别关注状态)
  - B关注他人的关系 → 删除 (占位用户不能主动关注)
  - 他人对B的备注名 → `target_id` 改为A
3. 将B的 `status` 改为 'm' (已合并), `phone` 置空 (释放手机号)
4. A绑定手机号X, 完成合并

## 二、关注/好友体系

### 关系说明

A 关注 B → A是B的粉丝, B是A的关注对象  
 B 也关注 A → 互相关注 = 好友关系

### 关注的前置条件

关注者	被关注者	是否允许	说明
正式用户(a)	正式用户(a)	√	正常关注
正式用户(a)	游客用户(g)	√	可以关注游客
正式用户(a)	占位用户(p)	√	可以关注, 合并后自动迁移到正式用户
正式用户(a)	禁用用户(d)	×	禁用用户不可被关注
正式用户(a)	已合并用户(m)	×	已合并用户不可被关注
游客用户(g)	任何用户	×	游客不能主动关注他人
占位用户(p)	任何用户	×	占位用户不能主动关注

## 禁用用户的特殊处理

- 用户被禁用后，**已有的粉丝关系保留**，但粉丝无法查看其内容
- 禁用用户**无法被搜索**，无法被新用户关注
- 如需完全清理，管理员可手动删除关注关系

## 关注表设计 user\_follow

```
CREATE TABLE `user_follow` (
  `id` INT PRIMARY KEY AUTO_INCREMENT,
  `user_id` INT NOT NULL COMMENT '关注者',
  `target_id` INT NOT NULL COMMENT '被关注者',
  `is_special` CHAR(1) DEFAULT 'n' COMMENT 'y:特别关注 n:普通关注',
  `created_at` DATETIME DEFAULT CURRENT_TIMESTAMP,
  UNIQUE KEY `uk_follow` (`user_id`, `target_id`),
  KEY `idx_user_id` (`user_id`),
  KEY `idx_target` (`target_id`)
) COMMENT='关注关系表';
```

## 特别关注

用户可以将已关注的人标记为"特别关注"，用于：

- 列表中优先展示
- 快速筛选查找

## 查询好友（互关）

```
-- 查询用户A的好友列表（互相关注）
SELECT u.* FROM user u
WHERE u.id IN (
  SELECT f1.target_id FROM user_follow f1
  INNER JOIN user_follow f2
  ON f1.user_id = f2.target_id AND f1.target_id = f2.user_id
  WHERE f1.user_id = :userId
);
```

## 备注名

用户可以给**已关注的人**设置备注名（类似微信备注功能）。

类型	说明	可见范围
昵称	用户自己的名字（微信昵称或自己修改）	所有人可见
备注名	A给B设置的名字	只有A自己可见

**显示优先级：备注名 > 昵称**

**占位用户合并**：合并后昵称被微信昵称覆盖，但他人设置的备注名保留

**取消关注时**：保留备注名记录，重新关注后自动恢复显示

### 备注名表设计 user\_remark

```
CREATE TABLE `user_remark` (
  `id` INT PRIMARY KEY AUTO_INCREMENT,
  `user_id` INT NOT NULL COMMENT '设置者',
  `target_id` INT NOT NULL COMMENT '被备注者',
  `remark_name` VARCHAR(50) NOT NULL COMMENT '备注名',
  `updated_at` DATETIME DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  UNIQUE KEY `uk_remark` (`user_id`, `target_id`),
  KEY `idx_target` (`target_id`)
) COMMENT='用户备注名表';
```

### 显示名称逻辑

获取用户B的显示名称（对A而言）：

1. 查询 user\_remark 中 A 对 B 的备注名
2. 如果有备注名 → 返回备注名
3. 如果没有 → 返回 B 的昵称

## 三、拉黑体系

### 黑名单表 user\_block

```
CREATE TABLE `user_block` (
  `id` INT PRIMARY KEY AUTO_INCREMENT,
  `user_id` INT NOT NULL COMMENT '操作者',
  `blocked_id` INT NOT NULL COMMENT '被拉黑者',
  `created_at` DATETIME DEFAULT CURRENT_TIMESTAMP,
  UNIQUE KEY `uk_block` (`user_id`, `blocked_id`),
  KEY `idx_blocked` (`blocked_id`)
) COMMENT='黑名单表';
```

### 拉黑效果

当 A 拉黑 B 时：

- B 无法查看 A 的：比赛、历史成绩、动态
- B 无法关注 A
- 双向解除关注：A 对 B 的关注、B 对 A 的关注都会自动删除

- A 对 B 的备注名保留（方便识别），B 对 A 的备注名删除

## 取消拉黑

- 取消拉黑后，双方需要**重新关注**才能恢复好友关系
- 之前的关注关系不会自动恢复

## 四、隐私设置

### 隐私设置表 `user_privacy`

```
CREATE TABLE `user_privacy` (
  `user_id` INT PRIMARY KEY,
  `phone_searchable` CHAR(1) DEFAULT 'y' COMMENT 'y:可被搜索 n:禁止',
  `profile_visible` CHAR(1) DEFAULT 'y' COMMENT 'y:主页对陌生人可见 n:隐藏 [VIP]',
  `game_history_visible` CHAR(1) DEFAULT 'y' COMMENT 'y:比赛记录对陌生人可见 n:隐藏 [VIP]',
  `allow_stranger_follow` CHAR(1) DEFAULT 'y' COMMENT 'y:允许陌生人关注 n:禁止 [VIP]',
  `updated_at` DATETIME DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP
) COMMENT='隐私设置表';
```

### VIP 与隐私设置的关系

设置项	普通用户	VIP用户
<code>phone_searchable</code> 手机号搜索	✓ 可设置	✓ 可设置
<code>profile_visible</code> 主页可见性	✗ 默认公开	✓ 可设置
<code>game_history_visible</code> 比赛记录可见性	✗ 默认公开	✓ 可设置
<code>allow_stranger_follow</code> 陌生人关注	✗ 默认允许	✓ 可设置

#### 说明：

- 所有用户都有 `user_privacy` 记录，首次登录时自动创建（使用默认值）
- 非VIP用户的 VIP 专属设置会被忽略，按默认值处理
- 用户升级VIP后，之前设置的值立即生效

## 五、添加关注的场景

场景	入口	说明
搜索手机号	通讯录/搜索页	需对方开启手机号搜索
扫码	扫一扫	扫描用户二维码

场景	入口	说明
比赛中	比赛详情页	点击参赛者头像 → 关注
广场围观	广场/比赛列表	点击任意用户 → 关注
好友推荐	推荐列表	基于共同球友等推荐

## 六、关系图总览

