识别逻辑：

Idfa：手动刷新可变。唯一

Idfv：删除所有厂商应用可变，统一开发商应用唯一（不同开发商之 间不同，会来回变，要注意）

本地或者钥匙串存储id：应用不删除情况下唯一不变

还有要做任何一个key的数据异常时的处理

（ps：ios 钥匙串

<https://www.cnblogs.com/worldtraveler/p/4745064.html>

https://www.cnblogs.com/rayshen/p/5072850.html）

但是android 系统的是否要和ios的分开建表

账号系统

根据不同游戏来做账号系统的表，当绑定到第三方账号或者手机的时候，以账号来确定用户，没有第三方账号的时候根据设备识别来确定用户（比如游戏记录的识别逻辑）

游戏记录

制作一个通用的游戏记录存储表。以json字符串形式来保存游戏记录（为那种单机或者不需要实时存储的游戏而定）。

每个人的游戏记录，存储的时候都有一个唯一性id，用来判定多个设备之间设备记录覆盖还是合并的判断处理。

信息收集 之后返回本地唯一值

之后出现不同的要进行信息合并

登录分为自动登录本地账户 和选择登录账号

1.大致思路：获取本地idfa，idfv等唯一识别字符串到服务器，然后服务器自主生成唯一码idlocal，并将idlocal返还给客户端。

2.当本地不存在idlocal时候，收集手机信息到服务器，并通过信息识别是否已经注册过该设备，（1）如果没有注册过，生成idlocal并返还客户端，客户端保存在本地。（2）如果手机注册过，则找到对应的idlocal返还客户端，并保存本地

3.如果本地存在idlocal的时候，以idlocal为判断依据，更新服务器上的idfa等信息。

4.idfa等信息有个异常全是0的，需要单独判断

5.ios和android个选3个唯一识别码作为判定，没有的先控制。

6.实现信息整合功能，另起线程对现有的设备信息进行整合，判断出是统一用户的整合到一起，应该是个耗时操作吧。