上传：

http://192.168.11.170/cgi-bin/icon

参数：

action=put

md5=xxx

cross\_uuid=xxx

另外就是post方法传个图片过来

put$%8\*&%^%/rsfsRx00

下载：

//校验md5

static const char key[] = "get98I\*&%^%/rsfsRx>L";

char md5\_src\_str[128];

snprintf(md5\_src\_str, sizeof(md5\_src\_str), "%s%s", key, cross\_uuid.c\_str());

result:resultcode,statuscode,byte\_dtas[optional]

"%s%s%s", key.c\_str(), cross\_uuid.c\_str(), status.c\_str()

0是从正式库找，非0是从备份库

检查状态：

http://192.168.11.170/cgi-bin/icon?action=check&md5=xxxx&cross\_uuid=xxxx

返回那个人的图片的审核状态，1是在审核中，0是不在审核中。如果是0的话，也有一种可能是那个人从来没上传过图片，-1没传过图片

md5的算法跟get的那个接口一样，key也是一样

result:status,md5

"%s%s%s", key.c\_str(), cross\_uuid.c\_str(), status.c\_str()

0的时候从正式库取md5，非0就是从备份库取md5

修改审核状态

http://192.168.11.170/cgi-bin/icon?action=change\_status&md5=xxxx&cross\_uuid=xxxx&status=x

其中那个status是修改的状态，1是改成审核中，0是改成非审核

//校验md5

static const char key[] = "get98I\*&%^%/rsfsRx>L";

char md5\_src\_str[128];

snprintf(md5\_src\_str, sizeof(md5\_src\_str), "%s%s%s", key, cross\_uuid.c\_str(), status.c\_str());

返回的值有2个：

错误码,是否设置成功

“错误码”是0的时候表示玩家有传过值，1的时候表示没传过值

“是否设置成功”这个暂时可以不管，可以认为“错误码”是0的时候就是成功的

Result:resultcode,resultcode

头像列表：

http://192.168.11.170/cgi-bin/icon?action=list&md5=xxxx&start=x&count=x

//校验md5

static const char key[] = "get98I\*&%^%/rsfsRx>L";

char md5\_src\_str[128];

snprintf(md5\_src\_str, sizeof(md5\_src\_str), "%s%s%s", key, start.c\_str(), count.c\_str());

1,审核

2，没通过

3，通过

4，没有传过

获得的图片：

http://192.168.11.170/cgi-bin/icon?action=list&md5=xxxx&start=x&count=x

//校验md5

static const char key[] = "get98I\*&%^%/rsfsRx>L";

char md5\_src\_str[128];

snprintf(md5\_src\_str, sizeof(md5\_src\_str), "%s%s%s", key, start.c\_str(), count.c\_str());

返回值格式应该是：

cross\_uuid:图片字符串:审核状态

cross\_uuid:图片字符串:审核状态

...

http://tools.yhphs.1aishouyou.com/admin/bjui.php

吴培健

123456

本地上传的图片保存的地方

Application.persistentDataPath/Photos/takingphoto.png

从服务器下载到本地的地方

Application.persistentDataPath/Server/cross\_uuid.png

拍照上传流程图

57b1b1056e8d6c758ff72dafb388c748 01

1. 拍照，设置拍完的图
2. 从图库里选择，设置选择的图
3. 检查状态，获得md5码
4. 上传图片
5. 加载图片

开始

安卓端

Unity端

1. 自己：选择“拍照”或“相册”上传图片。别人：选择“私聊”或“加为好友”。
2. 删除，上传按钮功能。
3. 上传之后，显示审核中
4. 审核通过，显示图片。
5. 如果拉伸比例超过20%，提示。

拍照/相册返回 ->显示Photos图片，

上传，加载图片到server/cross\_uuid.png目录

查看自己的图片：

1. 检查自己有没有传过图片：check
   1. 没有传过，未领取宝箱
   2. 传过，根据md5操作
      1. 审核中
      2. 审核未通过
      3. 审核通过

查看其他人的图片：

1. 检查是否有传过图片：check
   1. 没有传过。
   2. 审核中
   3. 审核未通过，显示审核中
   4. 审核通过

拍照、相册，上传图片，显示选中的图片：

1. 上传：上传图片
2. 删除：显示其他（如果已经领取宝箱，则显示未上传图片，否则显示宝箱）

提供接口：

1. 拍照，从相册获取图片。
2. 根据cross\_uuid检查是否有传过图片
3. 下载资源。该接口放到XPhoto的SetPhoto,setPhoto首先check，拿到md5,进行比较，比较不一致（并且本地不存在）才会重新下载图片，否则直接使用本地的缓存
4. 上传资源。

结果码，状态码，图片1，图片2

需求更新：

get：参数控制，从正式库拿图片，从备份库拿图片。

upload：上传一张图片。

check：返回状态码code,md5,md5会随code变动，md5只要保证备份库的正确就可以了。

功能：

正式库：审核中，或者其他状态的图。

备份库：审核通过的图。

查看别人：检查状态备份库状态，记下md5码，检查本地MD5，一致则读取本地，否则从备份库拿图片，拿到，拿不到都可以处理。

查看自己：检查正式库状态：不通过状态，记下md5码（如果有，则该md5是备份库的md5码），转支线1;图片未上传（以及所有其他）状态，显示前端图；审核中状态（如果有，则该md5是审核中的md5码），转支线1；审核通过

支线1：检查本地MD5，一致则读取本地，否则从备份库拿图片，拿到，拿不到都可以处理。

check的返回结果处理：

0：从正式库中取，将md5记到缓存(如果有)，将状态返回给Lua层，由Lua层决定后续处理。

非0：从备份库中取，不通过状态，将md5记到缓存，(如果有)；未上传状态，什么都不做；审核中状态，记下md5到审核md5，审核通过，将md5记到缓存。将状态返回给Lua层，有Lua层决定后续操作。

Upload处理，没有改变

Get处理，没有改变

//直接下载，保存md5

查看其它玩家图，备份库有图，备份库没图。没图一定是没传过，显示本地缓存或者下载的图。还是没有显示基础图，

//需要先check,

查看自己的图：检查状态：

审核中：下载正式库的图，有，正常；没有，显示本地最新的缓存图。还是没有显示宝箱。

审核通过：下载

审核不通过：--去下载，没有显示宝箱

没传过：不用下载，显示宝箱