# 故事：

昨天从下班开始就同一个女生聊天，直到晚上12点后睡觉，这是一次全新的聊天方式，因为这次在整个聊天过程中都没有20个字，全程斗图。现在不是很流行斗图吗？经过我昨晚上的体验，我发现了这么个问题，就是图片太多，很难找到自己想要的图片！或许对于一些斗图高手来说会对我抛来鄙夷的眼光，呵呵。先说一下我用的是什么斗图软件吧，它的名字就叫做“装Bi神器”，也不知道这是哪家公司开发的，我觉得它里头图片资源丰富，也很多让我眼前一亮的特色功能，比如可以自定义图片、伪造各种娱乐证件等等。总得来说这是一款很不错的软件，但作为一名程序员，我有我们程序员特有的思维方式，怎么才能更快的找到最合适的图片呢？于是我就构思出了这么一个解决方案，或者说自己弄一个app。我觉得这将是一款很有市场的app，因为它可以服务的人是那么的多，想想看，腾讯的qq、微信、陌陌等各种社交软件的用户都是你的用户，必火的赶脚啊！

App的核心思路就是，在用户输入对话内容时，根据输入的内容自动检索匹配的图片！然后列出匹配的供用户选择就是了。对的，或许你会说微信QQ上不是已经有这些功能了吗？是的，但那对于爱斗图的你能有多大帮助呢？功能看似简单，其实底层要做的事可却不少，但简单就是王道，这也预示着我们的app具备被兆民所接纳的可能。我在想，这样的功能给输入法研发公司来做是不是更有优势呢，那样的话我们的app就可以是以这样的一个形式展现给用户了。用户直接在输入框里用输入法打字，然后输入法自动匹配合适的图片，如果有则供用户自行选择。哇塞，这样的体验和选图效率都是很棒的不是吗？事实上我们可以跟输入法公司一同研发这样的功能，也还可以跟腾讯合作！牛逼咯，怎么说呢，就是将这个自动匹配图片的功能交给各种社交软件来完成，比如说微信。用户在使用微信输入聊天内容时，由微信来匹配内容，而正如我前面说过的，微信它本身就已经有这样的功能，只是不够强大，至少没有达到我期望的那样好用，那样强大！如果我们不能跟这些巨头合作开发，那么我们只能自己搞一个app了。只是那样的话在操作上不是那么的方便而已，就是需要用户多点几下手机屏幕这样。下面我们开始设计我们的这款很有前途的app，需要知道的是，这个app暂时没有跟这些公司建立任何的合作开发的关系，我们完全自己独立开发。。

# 重点速记：

好的点子就是那么的可遇不可求，我想我该先把一下关键的点记录下来，以免遗忘，那么暂时有以下几个重点：

* 我们的app是干什么的？很简单，根据用户输入的聊天内容，匹配图片。
* 为了实现这个搜索图片的功能，首先我们要丰富我们的图片资源，如果我们的app资源少的可怜，用户输入十次才成功匹配到一两次那是一种什么样的心情呢？反正我会受不鸟，我会骂：辣鸡app！为此我们需要以下一些策略：

我们总不能依靠自己的时间和精力到网上找图片找设计师来为我们的充实图片资源吧？我觉得这是不科学也不明智的，我们哪有那么多时间和精力或金钱来办这事啊。我们得依靠我们的衣食父母啊，只有让他们来办这事才是可持续发展的啊。我们得依靠广大人民群众啊！然人民群众来丰富我们的图片资源库啊，让人民群众自己为图片配上合适的台词或内心独白啊。这样一来就会出现我们想要的这样的场景——用户A为图片jp1设定了三元素内容，这时用户B在输入用户jp1这张图片的匹配的三元素内容时，我们就自动匹配到jp1这张图片了，是不是很科学呢？我是这么认为的。当然如果图片jp1在我们的服务器上不存在，这时我们需要将这个图片上传到我们的服务器，并将它放到一个以该图片的三元素内容为名的文件文件夹下，如果文件夹已经存在，就直接将图片放入其中就可以了，这样不就实现了，同一种情绪、态度和台词对应出N多合适的图片了吗？偷笑。。其实我偷笑的有点早了，为什么这么说呢？因为我考虑到了一个问题——用户凭啥给你做这个图片分类的工作啊？人家吃饱没事撑着啊！我后来仔细一想不是那么回事，我认为用户会很乐意做这件事！因为从我昨晚上的选择图片的体验来说，我就很想对手机里哪几百张图片进行分类的。为什么啊，选择图片太难了啊因为，一大堆的混在哪儿，直接乱成一锅粥。。为了让用户能够帮我们完成这件工作，我想我们有义务制作一个方便用户对图片进行分类的友好界面，关于这个界面应该花些心思来设计，嗯！我们的分类是很细的，因为我们直接以心情、一句对话内容作为一个分类。讲到这，我想我该说出我们这款app 的一个口号了，或者说是一个目标吧，“让聊天不存在可以复制的文字”，又或者说让聊天全程斗图，讲到这，我为我们的这款app想到了一个名字《easy 斗图》，看官你意下如何呢？

* 根据图片的使用次数，删除图片。为什么这么做，这是因为可能很多图片的配的对白并不合适，这样导致长期没人使用它，故而应该把它删除。
* 首先我们还是要收集一下常用的图片作为系统的默认数据的，这样不至于用户在使用我们的app时找不到一张图片，比如一下代表心情的图片、问候语的图片等等
* 激励用户为表情设置对白，更要激励用户多上传表情 你有什么好法子吗？搞个登记制度吧？最佳贡献奖 斗图等级：菜鸟 斗图大师 神一样的存在？
* 关于allimages文件夹，这个文件夹用来缓存放用户搜索到的表情，它存放的地方在app的缓存目录中，allimages文件夹下会有很多的以对白为子目录的文件夹，这些文件夹中存放相对应的的表情。
* 免注册直接用，但是我们还是会根据用户使用的设备创建一个用户，当有一天用户注册了，并且是使用这一台设备注册的时候，我们将它们关联起来就是了
* 给表情配上独白，同时还可以为表情设定心情、情绪、态度，这样的表情匹配得就太精准了。

# 功能：

### 管理本地表情：

**关键词：**

**三元素：**什么是三元素啊？我们人与人交流的过程中，我们说的每一句话，都一定程度的可以反映出我们的心情和态度的，同理，一个表情更是如此。我是这么想的，我们可以为每一张表情标注相应的情绪（喜、[怒](http://baike.baidu.com/item/%E6%80%92)、[哀](http://baike.baidu.com/item/%E5%93%80/75381)、惊、[恐](http://baike.baidu.com/item/%E6%81%90)、爱、无奈、寂寞）、态度（傲娇、鄙视、冷漠、逗、卑微、责备、开玩笑、认真的）和对白，看到了吗？我们的三元素就是说话时人的情绪、态度和最终说的话！因为下文会反复的提到这三元素，于是我就把它抽取了出来。

**为什么要这个功能？**

干什么用的？顾名思义啊？其实没那么简单，我想象中这个功能的作用是：请容我先讲个我昨晚上聊天的故事，再开始和妹子聊天之前，我就从《装Bi神器》这个app里下载了不少的表情，为什么啊？有备无患啊，你总不能临时抱佛脚的在聊天的时候再去各种地方照图片吧？好了，现在我是找了好多的表情存在手机上了，可是表情多了后你会发现一个问题，图片太TM难找了，你如果记得还好说，不记得你可能还得从网上去找，然而你本地已经存在这个表情了，这就白费劲了。于是昨晚上我就想，我要是能对这一堆乱七八糟的图片进行分类管理就好找了！so，这个管理本地表情的需求就有了。

**怎么实现这个功能？**

那么用户该怎么管理本地的表情呢？很简单，我们只要给一个表情定义好三元素的内容就可以了，关于三元素是什么，请看关键词“三元素”，这样我们就将一个表情做好了分类了，就是这么easy。对于用户操作而言，管理表情，就是简单的为本地的图片定义合适的三元素内容。需要注意的是，同一张表情可以定义多个三元素内容。在给表情标注这些属性前，我们还要根据这张图片的hashcode从服务器查找别人对这个表情设定类似的属性，如果有，而且用户觉得别人定义的三元素内容和自己预期的一样或者相近时，用户这可以直接选择别人设定的内容。那么在用户为表情标注好了三元素内容并点击确定后，我们需要做什么呢？这里要做的事情就多了，这里要分客户端和服务端来说，客户端要做什么呢？具体有：

* 将已经设定好了三元素内容的本地表情从需要管理的图片列表中删除。然而并不是将图片从内存中删除，
* 在allimages文件夹中，用该图片的三元素内容作为文件的名称创建一个文件夹，并将被管理的图片剪切到该文件夹中
* 判断表情是否需要上传，什么情况需要上传呢？如果表情是不存在于服务器的，上传表情，并将配套的三元素内容也提交到服务器并保存，如果表情已经存在于服务器，但用户为表情设定了新的三元素内容，这时也需要将新的三元素内容提交到服务器

服务端要做什么呢？具体有：

其实服务器要做的事情就是，如果有上传的表情，我们将根据表情的三元素内容，在服务器端存放表情的文件夹中创建一个文件夹，然后将表情上传到这个目录；在表情文件信息表里插入一条关于该上传的表情的数据，具体字段有：文件名称（文件名称是表情文件的hash码值加文件格式组成的）、上传时间、上传的用户id、存放路径。。）；将提交的表情的情绪、态度、对白保存到表情表里面，表情表的字段具体有：表情情绪、表情态度、表情对白、表情被检索次数、表情被使用次数、表情文件id、创建时间、创建人id），由于用户为这张表情设定了两条对白，这时则要插入两条记录了。至此，就算完成了管理本地表情的功能了。欧耶

### 搜索功能：

我们的这个app没什么功能，正因为少，所以每一个功能都显得尤其重要，这也是我们要把每一个功能做好做精的原因！搜索功能可是我们这个app的分量非常重的一块啊，下面我们开始说这个搜索功能。一句话概括，根据用户输入的三元素内容，自动的查找与输入的内容相符或相近的表情。需要说明的是，三元素中的情绪和态度不是必须的，我们默认大家的情绪都是愉快的，在这也祝全天下人每天心情愉快！问题的关键是看我们怎么查了，简单的做法就是，根据输入的对话内容，查找表情表的对白字段，两者相同或相近的算为匹配。

需要注意的细节：

* 根据表情的热门（被使用的次数）排序；
* app本地是有缓存的，我们应该考虑将那些表情加入缓存。关于缓存怎么做，我的解决思路是：在设备的本地数据库中也用一个表情表来记录缓存中所具有的表情，这张表具有这么一些字段：三元素内容、表情存放路径、图片文件名（文件名等于hashcode加格式名称）搜索成功匹配次数、使用次数。这样一来，我们的搜索功能就需要重新调整优化一下了，具体做法是：先根据输入的三元素内容在本地的表情表中查询一次，将匹配的结果保存到一个a列表中，然后在发送请求到服务器查询一次，并将查询结果保存到b列表中、计算出b列表中不存在于a列表的表情，并将这些表情下载下来并显示，同时将a列表中的表情也显示出来
* 自动复制输入的内容，这是为了，在没有匹配的表情时，可以快速的将内容自己粘贴到聊天工具的输入框，而不是得重新一个一个字的敲

缓存管理功能：

用户可以对本地的表情缓存进行管理，主要是删除。得有那么一个界面用来展示用户的缓存，同时提供删除的功能。删除表情的时候需要将本地表情表相应的数据予以删除，另外，如果删除的表情有多个对白时，并不删除相应的表情，除非表情对应的所有对白都被删除。

说到缓存，我们还得考虑一下缓存策略，什么策略呢？就是我不认为把用户所有的查询到的表情都缓存起来是个好办法，那样或许台占内存了。或许我们可以只缓存那些用户使用过的表情吧？这需要商榷。