策划书

1. **作品设计的创意来源和开发理念**
   1. **创意来源**

随着21世纪的到来，网络这一工具变得越来越发达，相比以前我们能更容易的接触到更多国外的内容。但这时就出现了一个问题 ，并非所有人都有足够的外语能力来看懂这些内容。幸运的是，汉化组的出现使得这些不再成为障碍。他们对文本、音频抑或是视频进行翻译，让人们能够更好的理解这些内容。但所谓“汉化组”其实多为个人汉化或者是小团体汉化，那么就出现了几个缺点：

1. 一般翻译者较少，很难达到高效快速的进行汉化。（因为语言文件可能进行频繁更新，如果不能快速进行新版本汉化的话就会出现前一个版本还没汉化完，下一个版本就已经推出的尴尬问题）
2. 汉化组众多，无法统一其汉化内容，尤其是一些专有词汇的汉化无法统一，可能造成使用不同汉化组的汉化的用户在交流时产生障碍。
3. 因为汉化组多为小汉化组，所以无法有效的推广，有很多汉化组已经推送了上百次更新，但是其努力却无人问津。
4. 在每次版本更新后汉化组需要自己去下载语言文件进行翻译，而用户需要等到汉化完后自行去下载汉化文件并进行替换或补充，这是一个十分没有效率的方法。

而本应用为游戏、软件的语言文件汉化提供了一个从获取到翻译到提供给用户的便捷的平台。

**2、开发理念**

本应用旨在：

1. 快速获取原作者更新中的语言文件。
2. 整合各汉化组，提高汉化效率。
3. 用户能更方便的获取汉化文件。
4. **作品的实用价值和主要创新点**

**1、作品的实用价值**

（1）整合后的汉化组能快速的获取到原作者的更新

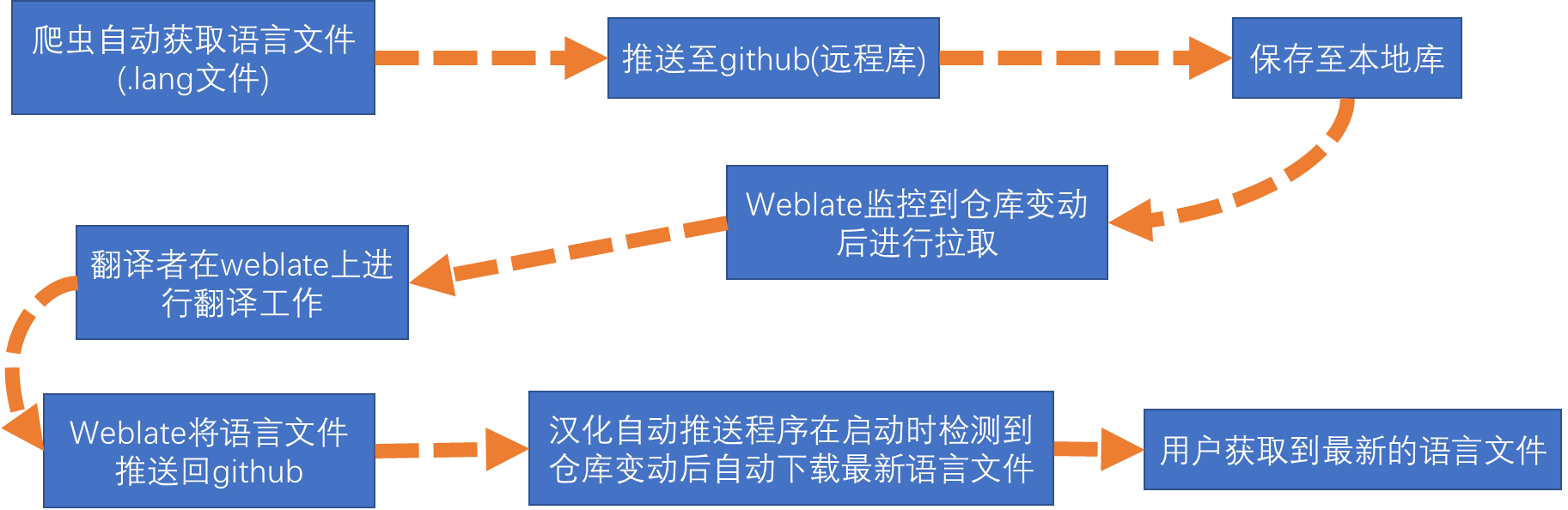
（2）管理者能更好的对汉化内容进行管理、修改，更方便的查看汉化的进度与翻译者的工作状况。

（3）用户能通过本应用快速获取更新后的汉化文件。

通过以上

通过以上三条整个过程的耗时较以前能大幅度减少，为翻译者与用户带来许多便利。

**2、主要创新点**



1. 通过爬虫，当原作者进行更新时能快速的将语言文件进行获取并推送至远程库，减少了翻译者们需要自行检查原作者更新并进行下载的不便。
2. 通过weblate将整个大的语言文件（.lang）中的每一行进行分离，提取应被汉化的部分（词条）后交予翻译者，翻译者将会在其工作页面上看见一句单词或者一句话，之后翻译者可以在输入行中输入翻译后的内容后进行提交，提交后可以得到下一个翻译内容。

简单地说，通过这一过程我们将翻译者分配给了词条，而非将词条分配给翻译者，这一点为翻译者们的高效协同工作提供了保障。

1. 在weblate中项目管理者可以为翻译者提供专有词汇的推荐翻译以防部分专有词汇翻译错误（比如glowstone应翻译成萤石，但有可能被翻译者误翻为荧石，这时管理者就可以推荐将glowstone翻译为萤石，当翻译者收到含有glowstone的翻译内容的时候，系统会在翻译界面的下方显示“glowstone”与其推荐翻译“萤石”）。
2. 当翻译者提交翻译后，该次翻译便会记录在案，管理者可以通过翻译内容查找对应翻译者，并通过翻译者页面获取其信息。

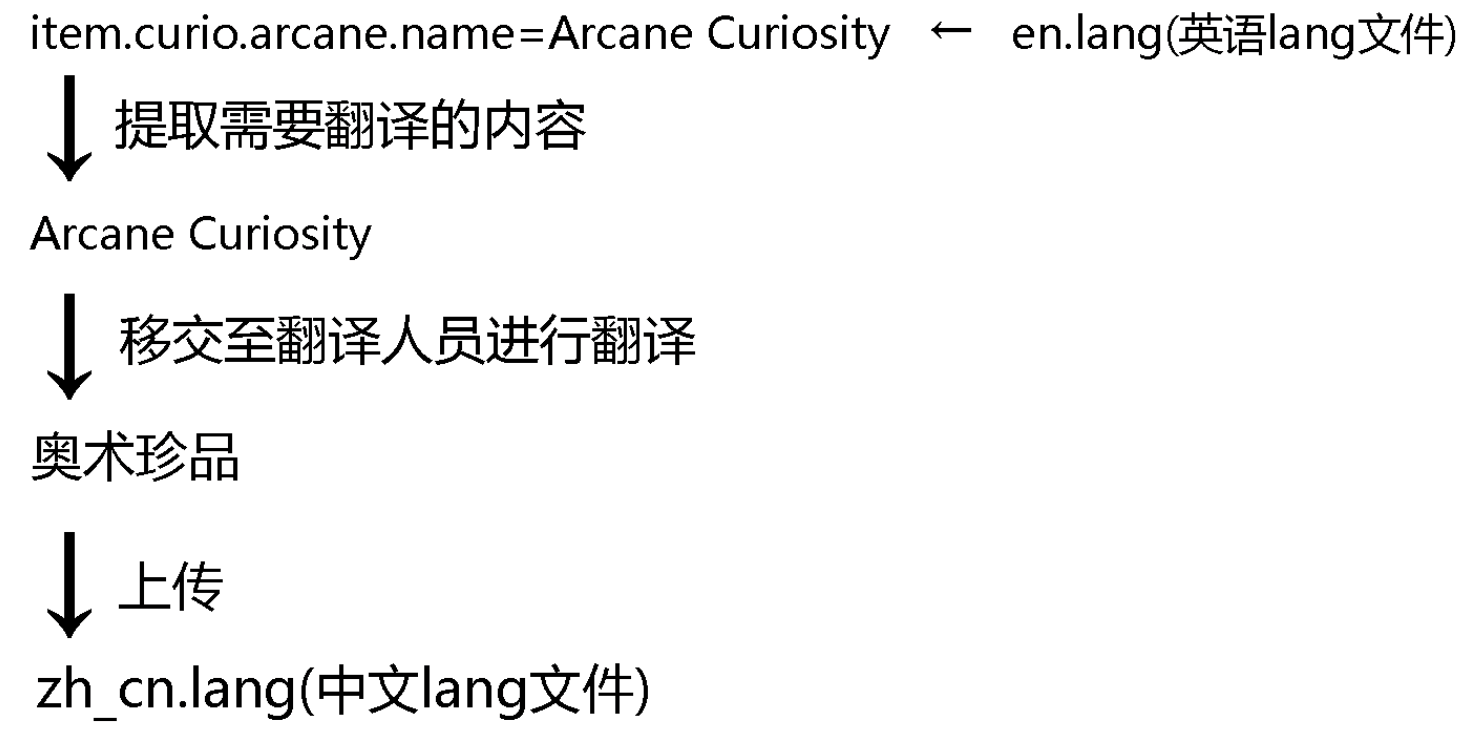
包括：

1. 其个人信息（如：名称、联系方式、注册邮箱、UID等等）
2. 贡献度（翻译词条总个数、在所有翻译者中的翻译数量排名）
3. 所翻译内容（即可以查询到该翻译者所翻译过的所有原文极其对应翻译内容）

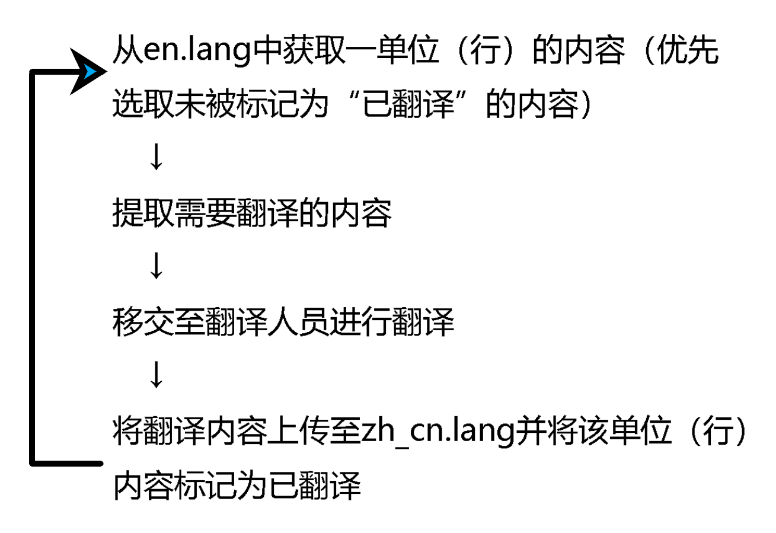
如果出现有人通过直接机翻进行翻译来刷翻译数量的情况，可以快速找到该人，并快速的删除其翻译内容，并对其进行封号等措施。

1. 在用户每次开启软件之前，本应用会先检查是否有新的汉化文件被推送至远程库，若有则进行下载，然后再进行一次汉化版本与原作者版本的对比，如果最新的汉化还没有推送则进行一次警告提示用户最新的汉化还未完成。

图一、一次具体翻译的举例



图二、翻译工作的简要循环过程



1. **作品的主要功能和功能板块**
2. **主要功能**
3. 爬虫自动爬取最新语言文件。
4. 翻译者通过weblate进行汉化。
5. 用户通过客户端自动获取最新语言文件。
6. **翻译者与管理者界面（weblate）**

**1、用户界面**

**、新用户注册**

需填写用户名、邮箱、密码、确认密码、名称，并进行验证码测试。

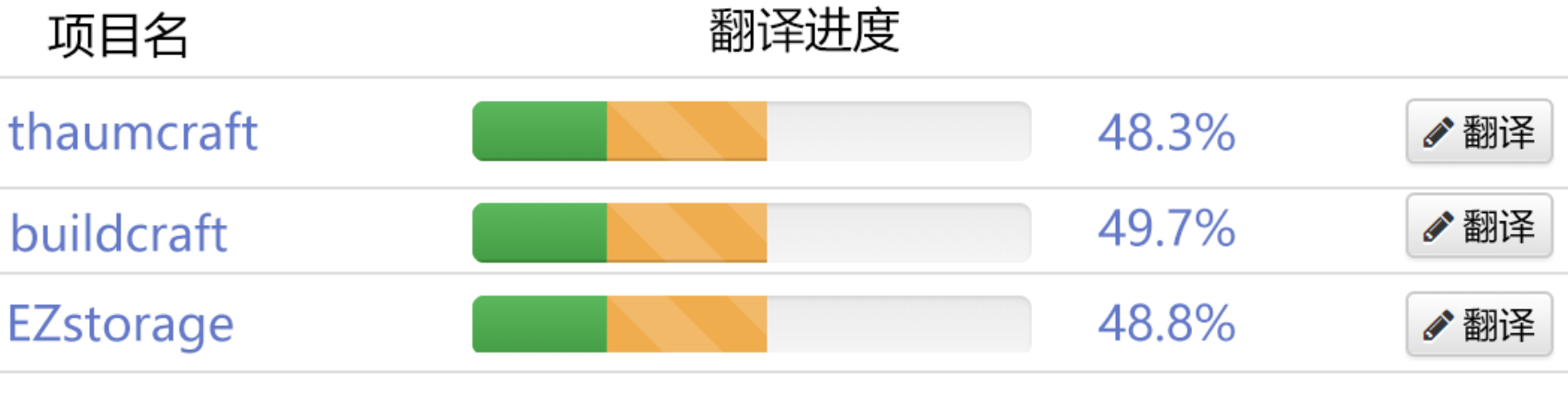
**、用户所属工程管理**

新建（成为工程的拥有者）、加入（加入工程需工程邀请链接）、退出工程。

**、用户信息修改**

**2、工程内界面**

**、项目界面**



在项目界面中翻译者可以选择自己想翻译的项目，点击右边的翻译按钮即可进入翻译界面。同时，工程中的所有工作人员可以通过该界面查看翻译进度（绿色为“已翻译”、黄色为“尚待修改”、灰色为“未翻译”，下面翻译界面功能介绍中会给出介绍）。

**、翻译界面**

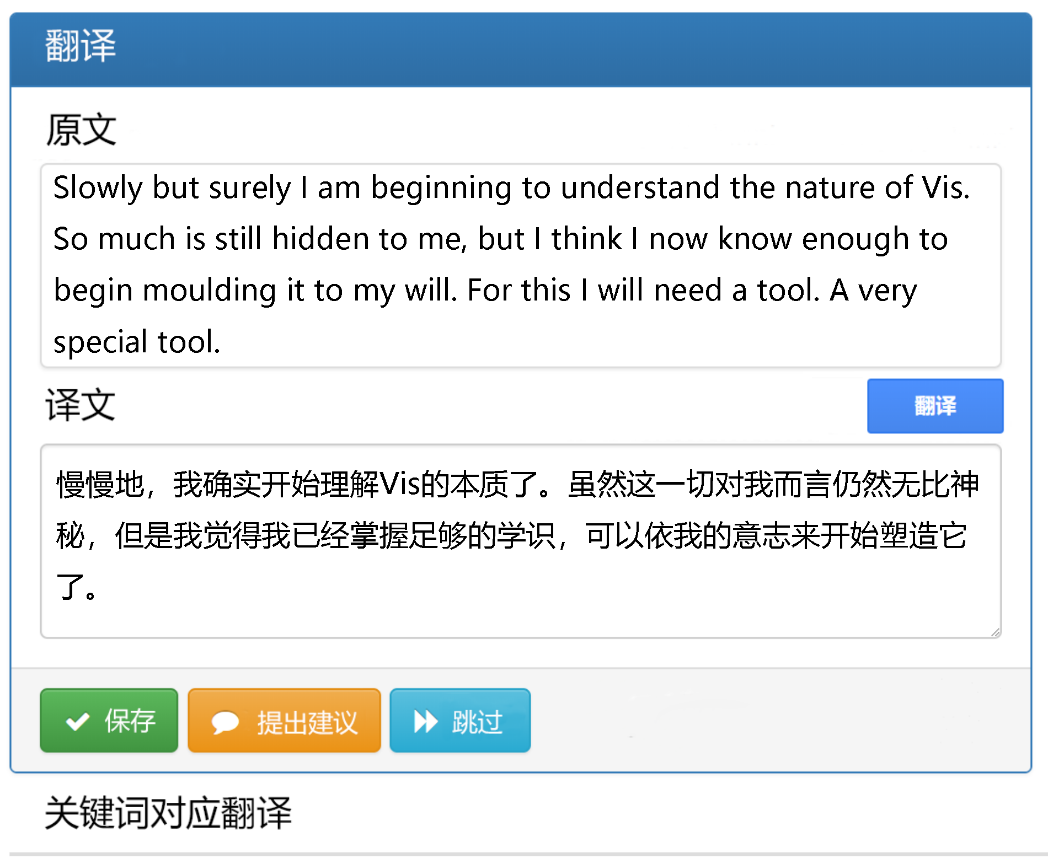
由两部分组成，上半部分为翻译者进行翻译的地方，最开始原文处会给出原文，然后翻译者可在译文处输入译文，然后上传。

上半部分的底部有三个按钮，分别为保存、提出建议、与跳过，那么接下来我将说明这三个功能的作用。

1. 保存：当翻译者确认自己翻译无误时选择此项，此项将译文上传，并将该词条标记为“已翻译”。
2. 提出建议：当翻译者不确定自己翻译是否正确时可以选择此项，此项将译文上传，并将该词条标记为“已翻译”，但当不存在“未翻译”词条时，其他翻译者仍会接受到这个词条，但在译文处会显示之前上传的译文。
3. 跳过：当翻译者无法翻译该词条时可以选择此项，此项将会标记原文为“未翻译”，且不会上传任何文本。

**以下为2个翻译的实例**



****

上半部分的中部有一个“翻译”功能，该功能为将原文中的文本直接放入谷歌翻译中进行翻译后给出译文，以辅助翻译者进行翻译。

下半部分为管理者提供的关键词对应翻译，翻译者应尽量参考提供的关

键词翻译，以使得整个翻译工程中关键词的一致性。

**、翻译者界面**

该页面仅本人与管理者可以查看

1. 查看到翻译者的名称、联系方式、注册邮箱等个人信息
2. 修改个人信息与密码
3. 查看该翻译者翻译的所有条目
4. 查看该翻译者贡献度（翻译词条数目、与在所有翻译者中的排名）
5. 将翻译的词条批量重新标记为“未翻译”
6. 删除账户（可以选择是否将该账户所翻译的内容全部重新标记为“未翻译”）

**、管理者界面**

该界面仅管理者本人与工程拥有者可以查看。

1. 新建、删除项目

在新建项目中管理者通过设置项目名、上传项目文件、设置

词条提取规则来启动新项目。

删除项目则可以直接删除现有的项目。

1. 翻译关键词添加

添加关键词及其对应译文。

1. 邀请链接复制（用于让用户加入该工程，即成为该工程的翻译者）
2. 查看翻译进度总览（工程总进度与工程下各项目进度）
3. 贡献度总览（用于查看各翻译者的词条翻译数）
4. 添加管理者（仅工程拥有者可见）
5. **用户自动获取语言文件功能**
6. 用户安装客户端后在启动需要汉化的程序之前，本应用会先对比远程库与用户电脑上的语言文件是否一致，如果不一致下载远程库上的最新版汉化文件进行替换。
7. 同时远程库会对比远程库与软件最新语言文件的版本是否一致，若不一致则会提示用户“最新版本的汉化尚未完成，当前汉化版本可能会出现汉化不完全的情况”，以让用户了解情况。
8. **作品的可行性、可操作性、实用性分析。**

应用场景：

* 1. 在以前用户需要手动去网上查找汉化文件，但因为汉化组众多，且汉化对应版本也不同用户很难判断应该下载哪个汉化组的汉化文件。

而本应用推出后会对小汉化组进行整合，便可以让汉化文件永远保持全网最新，且用户端会自动进行汉化的更新，无需用户费时费力自行上网搜索。

* 1. 翻译者们不必再自行去寻找软件最新的语言文件，只需要通过网页端进行翻译工作即可。
  2. 当翻译者想在碎片时间中进行翻译时，面对大片的词条是一个十分降低效率的选择。而使用本应用，每次翻译者只会接收到一个词条并对其进行翻译，而非对着大片繁复的文本进行翻译工作，这对在碎片时间进行翻译的效率有很大的提升。
  3. 在每个翻译者上传自己负责的项目之前，想要与其他人翻译的专有名词保持一致是一件不是很方便的事情，最后可能导致最终的成品中，一个名词在成品中有多种翻译。而使用本应用，专有名词可在管理者的设置后，在翻译者进行汉化遇到该专有名词后立刻得到其对应翻译，保证一致性的同时加大翻译效率。
  4. 当翻译者在自己负责的内容中发现不确定或不会翻译的词条时，只能向其他人询问或者空出来在校对时进行翻译。而使用本应用，不确定或者不会翻译的词条可以选择“建议”或者“跳过”，等到项目接近尾声时统一由能够翻译的翻译者进行翻译。

可行性分析：

用户分析：

本应用的主要用户群体为游戏汉化组的组织者、管理者、翻译者，因为游戏中出现的词条更为零散，不需要上下文进行辅助。

当前市场：

各汉化组与用户确实面临着我所叙述的问题，而本应用能够很好的解决其面临的种种问题，且市场上没有相同类型的应用，因此推出本应用后短期内是没有竞争的，此时便可以扩大用户群体，并从用户身上得到反馈，并予以改进。

前景：

同时本应用最终能与软件的作者达成一致，在软件中加入本应用的功能，这样就能为各种软件更好的提供翻译，为软件的本地化工作做出贡献

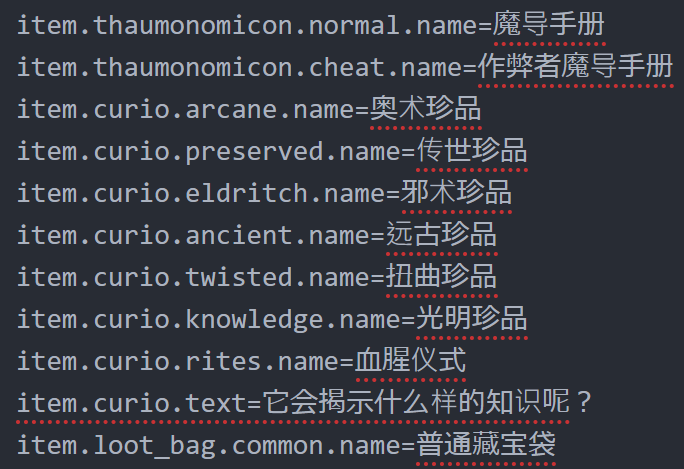
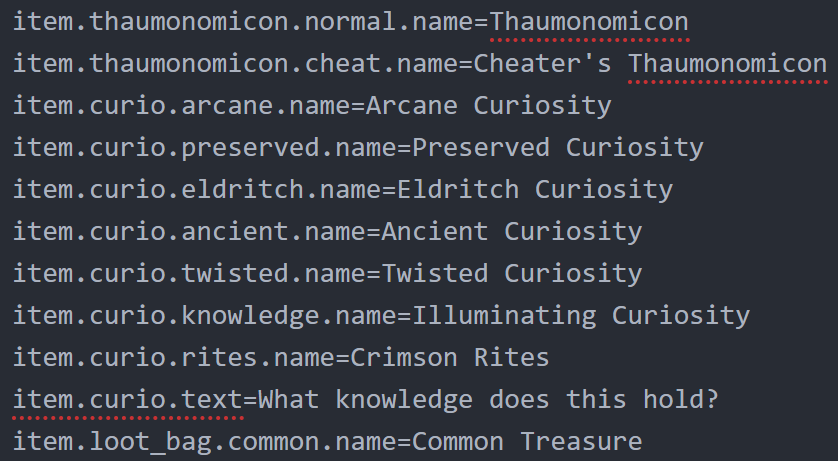
**五、作品的局限性**

本作品最初设计是用于Minecraft中非官方Mod汉化工作，因其lang文件的特殊性，可以很方便的将项目中的词条分离开。且Mod作者都会将其Mod的代码上传至Github因此爬虫可以很方便的爬到语言文件。

同时本作品比起文献，更倾向于游戏汉化，因为不难看出游戏汉化对上下文要求不高，就算只给出长文本中的一部分进行翻译，也不会出现偏差，而文献作品（尤其是小说等文学作品）更需要上下文来进行辅助翻译，同时也需要一个更好的算法来对词条进行提取。

因此本应用的用户群体主要局限于游戏汉化组。

附：Mod的原lang文件en\_us.lang与汉化后的文件zh\_cn.lang



小组成员：周翰飞

学院：计算机信息与工程学院