# 设计思路

用户注册时会生成一个用户唯一的文件夹，登录时会生成一个session用于校验。上传需要识别的发票后，程序会根据发票提供的数据用Python解析识别后生成包含数据的Excel表再返给用户查看或下载使用。主要方法是首先用Python定义类接口，接受PHP发送的文件路径，便于后期存储数据；用户提供图片后会自动存储在生成的文件中，之后向服务器发送请求，将图片存储在文件中后对图片进行简单的类型解析，将不同的发票类型归类；最后就是转码的过程，将图片的格式由API转换识别后存储在列表中，同时调用API接口返回数据，解析返回的数据，转换为我们通常常见的数据格式。最后将解析好的数据写入Excel中，再将Excel表格存到用户对应的文件夹中，方便用户查看。

# 模块设计

* **Python部分**

定义一个类，Get\_Img\_Data：

1. 接收用户上传文件，定义函数get\_img\_file，用来接收用户上传的文件并读取。
2. 对图片格式进行压缩，定义函数decode\_img，把图片格式转化为api能够识别的格式。
3. 将格式化的图片传给api进行识别，定义函数posturl，读取数据保存到网页，并且拿到网页数据。
4. 解析网页数据，定义函数get\_data，对api解析出来的数据进行分析，拿取用户需要的数据。
5. 将数据保存到excel，定义函数to\_excel，将存在列表的数据传入excel
6. 定义run函数，调用上述函数完成一个完整的程序。
7. 在main函数里执行run函数并返回值让php接收。

* **网页部分**

定义getFiles，根据上传的是单张还是多张图片来构建上传文件信息。

1.用户注册，用户登录信息校验成功之后使用mkdir函数来为用户生成用户私人的文件夹。

2.登录时，用户登录数据库数据判断成功以后，使用session存储用户的基本信息。

3.进入请求进入invoice页面以后使用session中的信息和用户数据库中的进行比对，如果比对成功就进入invoice页面，反之就回到登录页。

4.图片判断，定义uploadFile用来判断单张图片的格式，大小，上传方式是否符合条件。

5.定义getUniName得到唯一的字符串，作为图片上传的文件的路径。

6.定义getExt函数，利用strtolower，pathinfo等函数得到文件的扩展名。

* **核心模块：**

Python中主要是图片压缩模块，第三个api解析模块和第四个数据分析模块。

网页部分主要是先调用getFiles函数获得文件信息，然后利用uploadFile来判断图片是否符合上传的标准，最后调用getExt，getUniName来参与上传完成模块。使用fileShow来获得用户之前上传发票进行识别生成的表格，并为用户提供下载。

* 成员分工：

|  |  |  |
| --- | --- | --- |
| 姓名 | 职位 | 职务 |
| 周春芬 | 组员 | 负责提供网页的优化与色彩搭配问题，说明书编写排版和视频的拍摄剪辑。 |
| 徐章含 | 组长 | 负责带领团队确定项目，网页模块的代码编写和内部页面显示 |
| 赵富海 | 组员 | 负责编写发票数据的识别解析的代码和调用阿里云文字识别接口 |

# 重点难点

1.jQuery,bootstrap框架的使用，以及阿里云接口的调用

2.PHP调用python脚本并且进行信息交互

3.使用Session会话实现用户登录校验

4.通过阿里云搭建服务器和数据库

5.Ython的第三方库urllib,json,time,base64,os,openpyxl,sys

6.对用户上传的发票格式进行限制，将图片上传至服务器后进行识别。

7.进行转码操作，解决Python各模块的编码不统- -， 常常出现乱码和无法识别的情况。

8.在调用数据类型时导入模块，解决各模块之间数据类型的冲突问题。

# 开源组件

作品的界面基于HTML5,CSS3,javascript进行开发，使用了bootstrap，jQuery框架，使用了urllib,json,time,base64,os,openpyxl,sys等第三方库，并且调用了阿里云的发票识别接口进行发票识别。

# 安装说明

1.在电脑端打开默认的IE浏览器或者谷歌浏览器等其他浏览器，在地址栏输入<http://xzhxzh.tech/x/index.html>。就可以进入登录页面。

2.在移动端打开浏览器在地址栏输入<http://xzhxzh.tech/x/index.html>。就可以进入登录页面

# 指导老师自评

团队在阿里云相关API的基础上，使用PHP、Python等开发语言，实现了具有一定实用价值的国内常见发票识别工具。在有限的时间内，团队成员默契配合，分别实现各个模块，磨练了专业技能。该作品后期还可以结合具体应用场景，增加更多贴合实际需求、交互性良好的模块

**登录页**

**用户登录**

是否存在该用户

校验失败

保存到指定的用户文件夹中，

并调用python进行数据分析

判断发票是否符合条件

**用户上传发票**

校验成功

**不保存**

不符合

符合

**用户点击下载**