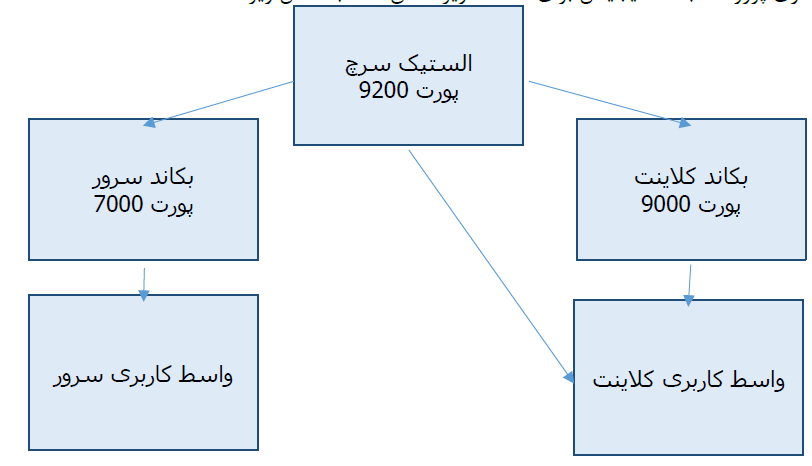


مستند OCR نسخه Clone

1. مقدمه

هدف از این گزارش توصیف کد backend سرور می باشد. این api نسخه ایی از ocr می باشد که قادر است یک عکس را دریافت کند و آن را ocr کند و درواقع نسخه ای از کد سرور کتابدار الزهرا می باشد که قابلیتهای آن کم شده اند.



مطابق این شکل ما هم کد backend سرور و واسط کاربریی سرور و هم کد کلاینت و واسط کاربری کلاینت داریم. همچنین از سرویس elasticsearch جهت ایندکس کردن مستندات و جستجوی سریع استفاده می کنیم. سرور در پورت 7000 و کلاینت در پورت 9000 و الستیک در پورت 9200 درحال listen کردن می باشند. فلش ها نشان دهنده ارسال و دریافت درخواست از یک سرویس به سرویس دیگر می باشند.

کد سرور از دو فایل config .py و clone.py تشکیل شده است.

درفایل config.py تنظیمات مربوطه انجام می شود. این تنظیمات مربوط به IP و پورتی است که کد سرور و الستیک روی آن ها listen می کند.

مربوط به سرویس الستیک:

HOST = '192.168.96.88'

PORT = 9200

مربوط به کد سرور:

server\_Host = '192.168.96.88'

server\_Port = 7000

رابط کاربری این سامانه به شکل زیر است که در سمت راست عکس دریافت می شود و با انتخاب زبان مربوطه و در نهایت با کلیک روی دکمه OCR ، عکس مربوطه ocr می شود.



توضیحات مربوط به کد سرور:

درحقیقت اطلاعات مربوط به زبانی که با استفاده از آن فایل OCR می شود و آدرس عکس مربوطه از سمت UI برای کد سرور ارسال می شود و در کد سرور این اطلاعات دریافت می شوند.

file\_name = recv\_data['file\_name']

language = recv\_data['language']

address = recv\_data["address"]

البته لازم به ذکر است که doc\_type نوع فایل دریافت شده است به عنوان مثال doc\_type می تواند پایان نامه، مقاله، کتاب و ... باشد که در این کد ما آن را بصورت هاردکد student قرار داده ایم.

با استفاده از زبان موردنظری که در کد مشخص شده است و با استفاده از تسرکت ورژن4 عکس موردنظر OCR می شود.

بعد از این مرحله مقادیری که میخواهیم در الستیک ذخیره شوند را مشخص می کنیم. در اینجا این مقادیر را در یک متغییر جیسونی doc ذخیره می کنیم.

doc = {

'author': author,

'total\_page': 0,

'image': "",

'content': "",

'text': ""

}

در نهایت براساس اینکه آدرس ورودی حاوی چه نوع عکسی است، کد مربوط به همان قسمت اجرا می شود.

if address.endswith('.pdf'):

imageBlobs\_pdf = read\_pdf(address)

print(len(imageBlobs\_pdf))

doc['total\_page'] = len(imageBlobs\_pdf)

elif address.endswith('.jpg') or address.endswith('.JPG') or

address.endswith('.jpeg') or file\_name.endswith('.png') or

file\_name.endswith('.tiff'):

imageBlobs\_list.append(address)

doc['total\_page']=1

else:

imageBlobs\_list = read\_dir(address)

سپس در یک حلقه تمام تصویر خوانده می شود و با base64 انکد می شود و در الستیک ذخیره می شود index می شود.