

گزارش تکلیف اول

سپهر مقیسه

پاییز ۱۴۰۱

برای این تکلیف اکثرا از کتابخانه flask استفاده شده است

در ابتدا تابع necessaryConnection را صدا میزنیم تا به دیتابیس، صندوقچه و صف وصل بشویم سپس برنامه flask را روشن کرده تا به api call ها گوش بدهد.

تابع adpost

همانطور که در عکس زیر مشخص است، این تابع وظیفه گوش دادن به postcall ها را دارد و سپس از طریق form data اطلاعات و عکس را میگیرد

```
@app.route('/adpost', methods=['POST'])
def postAd():
    email = request.form.get('email')
    des = request.form.get('description')
    image = request.files.get('image')
    what = str(imghdr.what(image))
    return dbManagement(1, email, des, image, what)
```

POST	http://127.0.0.1:5000/adpost
Params Authorization Headers (8) Body Pre-request Script Tests Settings	
<input type="radio"/> none <input checked="" type="radio"/> form-data <input type="radio"/> x-www-form-urlencoded <input type="radio"/> raw <input type="radio"/> binary <input type="radio"/> GraphQL	
KEY	VALUE
<input checked="" type="checkbox"/> email	sepehrmoghiseh@gmail.com
<input checked="" type="checkbox"/> description	a 2009 honda motorcycle
<input checked="" type="checkbox"/> image	images33.jpg X

سپس این پاسخ را به ما میدهد

```
1 your ad has been registered with id 108
```

تابع adget

همانطور که از اسم تابع مشخص است وظیفه برگرداندن اطلاعات آگهی را دارد به این صورت که اگر همچنان در صف باشد این پاسخ را میدهد:

```
1 your ad is not approved yet!
```

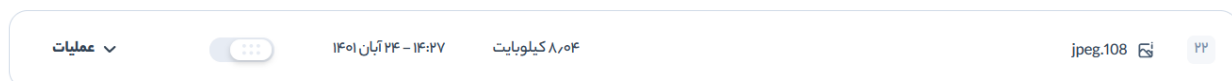
اگر تایید شده بود این پاسخ:

```
1 here is your ad info
2 description: a 2009 honda motorcycle
3 category: car
4 image:
5 https://s3.ir-thr-at1.arvanstorage.com/adproj/110.jpeg?AWSAccessKeyId=65a0c54d-5ad8-429b-989d-ea431d6c92aa&
  Signature=q6L58YA%2BPHdvmY0JGAirITdigRo%3D&Expires=1669115216
6 state: approved
```

و در آخر اگر هم رد شده بود این پاسخ را برگرداند:

```
1 your ad is rejected!
```

همچنین عکس را در صندوقچه با شماره آگهی ذخیره میکند و قابلیت دانلود را نیز دارد



در سرور دوم نیز در تابع call back تابع فرستادن ایمیل و تگ زدن بر عکس را صدا میزنیم:

```
def callback(ch, method, properties, body):
    id = imageTag(str(body)[2:-1])
    emailSend(id)
```

در تابع تگ زدن نیز عکس را از صندوقچه گرفته و سپس چک میکند اگر که شرط لازم را داشت تایید و در غیر این صورت رد میکند:

```
for i in range(len(response.json()['result']['tags'])):
    if 'vehicle' == response.json()['result']['tags'][i]['tag']['en']:
        if response.json()['result']['tags'][i]['confidence'] >= 50:
            id = int(body[:body.index('.')])
            return updateState(id, response.json()['result']['tags'][0]['tag']['en'], 1)
        else:
            break
    else:
        id = int(body[:body.index('.')])
        return updateState(id, response.json()['result']['tags'][0]['tag']['en'], 0)
```

```
if flag == 1:
    value = (tag, "approved", id)
else:
    value = (tag, "rejected", id)
```

و در آخر ایمیل مورد نیاز را میفرستد:

```
if myresult[0][3] == "approved":
    return requests.post(
        f"https://api.mailgun.net/v3/.org/messages",
        auth=("api", "-d854d79b"),
        data={"from": "<mailgun@sandbox4599c.mailgun.org>",
              "to": [email],
              "subject": "your ad status has been updated",
              "text": "hello! your ad with id " + str(id) + " has been
approved!"})
else:
    return requests.post(
        f"https://api.mailgun.net/v3/sa4ddfb06294706.mailgun.org/messages",
        auth=("api", "da93167755c8268001-48c092ba-d854d79b"),
        data={"from": "<mailgun@sandbox459106294706.mailgun.org>",
              "to": [email],
              "subject": "your ad status has been updated",
              "text": "hello! your ad with id " + str(id) + " has been
rejected!"})
```

your ad status has been updated

hello! your app with id 103 has been rejected!

your ad status has been updated

hello! your ad with id 108 has been approved!