
Django microservice request

Release 0.5.3

bandirom

Jun 25, 2022

CONTENTS:

1 Quickstart	3
---------------------	----------

Django application for make sync requests (REST API) between microservices or external APIs.

If you don't have any microservices yet, you can try to use [Django template](#)

QUICKSTART

1.1 Installation

1. Install the package

```
$ pip install django-microservice-request
```

2. Usage (Google Captcha example)

settings.py

```
GOOGLE_CAPTCHA_URL = "https://google.com/recaptcha/api/siteverify"  
GOOGLE_CAPTCHA_SECRET_KEY = os.environ.get("GOOGLE_CAPTCHA_SECRET_KEY")
```

services.py

```
from django.conf import settings  
from microservice_request.services import ConnectionService  
  
class GoogleCaptchaService(ConnectionService):  
    service = settings.GOOGLE_CAPTCHA_URL  
    secret_key = settings.GOOGLE_CAPTCHA_SECRET_KEY  
  
    def validate_captcha(self, captcha: str) -> dict:  
        params = {  
            "secret": self.secret_key,  
            "response": captcha,  
        }  
        response = self.service_response(method="get", params=params)  
        return response.data
```

serializers.py

```
from .services import GoogleCaptchaService  
  
class CaptchaMixinSerializer(serializers.Serializer):  
    captcha = serializers.CharField()  
  
    def validate_captcha(self, captcha: str) -> str:  
        service = GoogleCaptchaService()  
        response_data = service.validate_captcha(captcha)  
        if not response_data.get("success"):
```

(continues on next page)

(continued from previous page)

```
        raise serializers.ValidationError("Captcha validation error")
    return captcha

class LoginSerializer(CaptchaMixinSerializer):
    login = serializers.CharField()
    password = serializers.CharField()
```