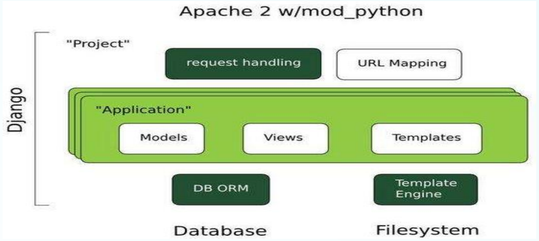
Django and Tornado

By: Right Light Team Group members: Jiarui Song Yucheng Liu Shihao Wang
Ruibo Cheng Yu Han Ao Gao Guangyu Du Kangdao Liu

Django

Introduction

Django is an open source web application framework written by python. The framework mode of MTV is adopted, which includes model M, view v and template t. It was originally developed to manage some news content based websites ,called CMS (content management system) software.



Code Example

Here is a simple “Hello, world” example web app for Tornado:

1.Create view.py

```
from django.http import HttpResponse

def hello(request):
    return HttpResponse("Hello world")
```

2. Edit the urls.py

```
from django.conf.urls.defaults import *
from FirstDjango.view import hello

urlpatterns = patterns('',
    ('^hello/$', hello),
)
```

Tornado

Introduction

Tornado is a Python web framework and asynchronous networking library, originally developed at FriendFeed. By using non-blocking network I/O, Tornado can scale to tens of thousands of open connections, making it ideal for long polling, WebSockets, and other applications that require a long-lived connection to each user.

Components

Tornado can be roughly divided into four major components:

Tornado

- A web framework (including RequestHandler which is subclassed to create web applications, and various supporting classes).
- Client- and server-side implementions of HTTP• An asynchronous networking library including the classes IOLoop and IOStream, which serve as the building blocks for the HTTP components and can also be used to implement other protocols.
- A coroutine library (tornado.gen) which allows asynchronous code to be written in a more straightforward way than chaining callbacks

Code Example

Here is a simple “Hello, world” example web app for Tornado:

```
import tornado.ioloop
import tornado.web

class MainHandler(tornado.web.RequestHandler):
    def get(self):
        self.write("Hello, world")

def make_app():
    return tornado.web.Application([
        (r"/", MainHandler),
    ])

if __name__ == "__main__":
    app = make_app()
    app.listen(8888)
    tornado.ioloop.IOLoop.current().start()
```

Comparison

Django is suitable for the rapid development of beginners or small teams, and it is suitable for the management, blog or website with complex functions.

Tornado is suitable for highly customized websites with large number of visits and many asynchronous situations.

characteristic	Django	Tornado
Size	heavyweight	Lightweight
The community	huge	small and getting larger
Development speed	faster	slower
Uploaded file	in /tmp directory	in memory

Pros and Cons

Advantages of Django
- Big and full (heavyweight framework)
- Fully automatic management background

Disadvantages of Django
- Template is not easy to use (from its own shortcomings)
- It is inconvenient to use NoSQL for database

Advantages of Tornado
- Less and better (Lightweight Framework)
- WebSockets long connection

Disadvantage of Tornado
- Not conducive to encapsulation as a functional module.

Conclusion

Above all, we compared the difference between those two frameworks, It's hard to say which one is better. Django is the most representative of the heavyweights framework, and tornado is a lightweight framework with high performance and high speedSo there is no answer which one is better without a clear need, Both of them have their own advantage. Just choose one more suitable.

References

Tornado Web Server. [online] Available at: https://www.tornadoweb.org/en/latest/.
The advantages and disadvantages of Tornado [online] Available at: https://www.cnblogs.com/zhumengke/articles/11297303.html/.
Django Tutorial [online] Available at: https://www.jianshu.com/p/62833a0b651f.