Django的目录结构整理

1. 添加requirements.txt文件，列出项目中所有使用到的python包。创建requirements文件是为了让其他开发者 拷贝你的项目代码后可以快速地根据此文件中内容安装好必须的python依赖包。这样他们可以方便地运行你的代码，而不必煞费苦心地猜测项目依赖包的版本。

例如：requirements.txt内容如下

Django==2.0.5

Apscheduler==3.0.3

scrapy == 1.0.5

scrapyd == 1.2.0

scrapd-api == 2.1.2

目录结构

Djangoproject/

\_\_init\_\_.py #个文件是django创建project所必须的，不做过多说明

manage.py

settings.py

urls.py

apps/ #即使是“小”工程，也建议分成多个app，每个app足够简单，只解决某一个方面

app1/

app2/

extra\_apps/ #引用的其他app

libs/ #加载第三方模块，以避免版本冲突，按照标准的site-packages管理

site-packages/ #依赖的包

tests/ #project级别的测试，对于每个app，还要有自己的测试代码

test1/

test2/

static/ #静态内容

css/

js/

images/

uploads/ #上传文件所在目录

templates/ #模板目录

app1/

app2/

log/ log目录