

Server-1.1.0 工程代码使用说明 & 发布流程

（一）使用说明

1. 服务器测试地址:
https://192.168.1.71:443
2. 环境:
Language: Python3.6
Database: MySQL Community Server 8.0.12
IDE: Pycharm 2018 Professional
3. HTTPS 证书:
server.crt
server.cer
4. Package:
Django: Django 2.0.7
Web: requests 2.18.4
requests-unixsocket 0.1.5
MySQL: PyMySQL 0.9.2
HTTPS: django-extensions 2.1.0
pyOpenSSL 18.0.0
django-werkzeug-debugger-runserver 0.3.1
Statistics: django-rank-query 0.1.0
5. 工程项目结构:
 - manage.py: Django 用来管理任务的命令行工具
 - .idea: 项目信息
 - livestreaming: 项目配置信息
 - ① settings.py: 项目配置信息
 - ② urls.py: 配置 http 接口与对应方法的映射
 - userSystem: 用户系统模块
 - ① models.py: 数据库模型
 - ② register.py: 注册接口方法
 - ③ login.py: 登陆接口方法
 - ④ forgetPwd.py: 忘记密码接口方法
 - ⑤ resetPwd.py: 重置密码接口方法
 - ⑥ getUserInfo.py: 获取用户信息接口方法
 - ⑦ uploadPortrait.py: 上传头像接口方法
 - ⑧ checkSession.py: 检查 session 接口方法
 - ⑨ delSession.py: 删除 session 接口方法

- streamingRoom: 直播模块
 - ① startStreaming.py: 开始直播接口方法
 - ② endStreaming.py: 结束直播接口方法
 - ③ refreshRoomList.py: 刷新房间列表接口方法
 - ④ refreshLiveStatus.py: 刷新直播实时状态接口方法
 - ⑤ RefreshReplayList.py: 刷新重播列表接口方法
 - ⑥ userStatistics.py: 获取用户数据接口方法
 - ⑦ anchorStatistics.py: 获取主播数据接口方法
 - ⑧ replay.py: 重播接口方法
- charRoom: 聊天室程序
 - ① server_cz.py: 服务器代码
 - ② concurrency_test.py: 并发测试代码
- protobuf: 数据传输协议
 - ① message.proto: 协议定义格式
 - ② message_pb2.py: 经编译生成消息类
- upload: 存放用户头像文件夹
- server.crt, server.cer, server.key, server.csr: 证书相关文件

6. 使用流程:

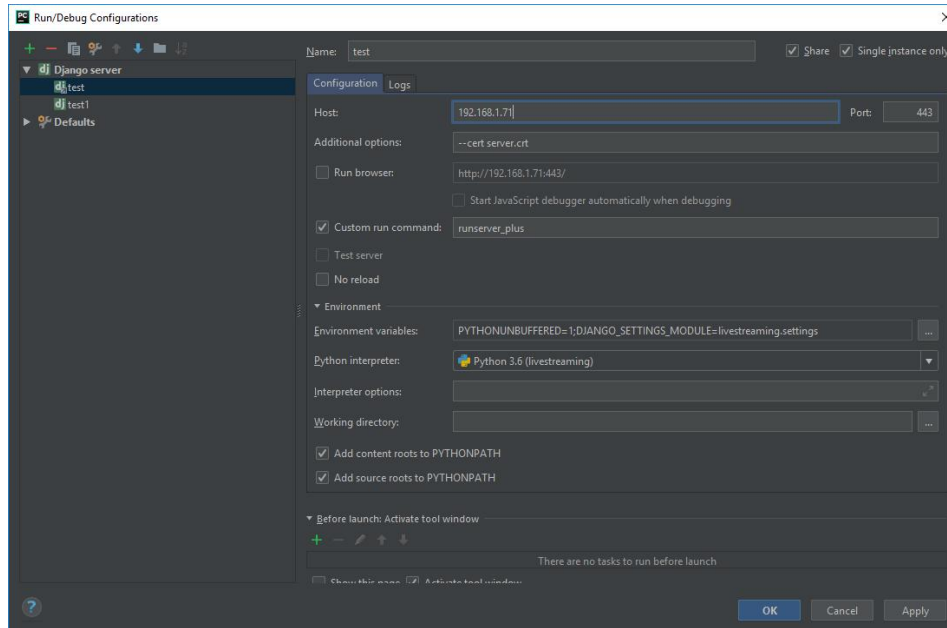
- 1) 使用 Pycharm 打开工程
- 2) 将服务器地址加入 settings.py 的'ALLOWED_HOSTS'中(使用 192.168.1.71 可忽略)
- 3) 配置数据库
在 settings.py 配置文件的'DATABASES'部分填入数据库信息

```

DATABASES = {
    'default': {
        # 'ENGINE': 'django.db.backends.sqlite3',
        # 'NAME': os.path.join(BASE_DIR, 'db.sqlite3'),
        'ENGINE': 'django.db.backends.mysql',
        'NAME': 'test2',
        'USER': 'root',
        'PASSWORD': 'c3daag31',
        'HOST': '127.0.0.1',
        'PORT': '3306',
    }
}

```

- 4) 启动 HTTPS 服务器
 - ① Edit Configuration -> '+' -> Django server
 - ② 配置服务器地址
Host: 服务器地址 Port: 服务器端口
Additional options: --server.crt (服务器证书)
Custom run command: runserver_plus



5) 启动聊天室服务器

- ① Edit Configuration -> '+' -> Python
- ② Script Path: server_cz_2.py 路径

6) 启动 HTTP 回调服务器

- ① Edit Configuration -> '+' -> Django server
- ② Host: 回调服务器地址; Port: 回调服务器端口

7. SRS 系统 Django 服务器代码说明

由于目前 SRS 所在系统为 Ubuntu，Django 服务器在 Windows 下，所以 SRS 服务器录制视频无法直接传到 Django 服务器下，Django 服务器也无法发送回放请求给 SRS，故在 Ubuntu 下另开一个 Django 服务器用于接收视频录制回调和回放请求并返回视频地址。

● 主要文件目录：replay

- ① models.py: 数据库模型 (视频文件存储地址)
- ② refresh.py: HTTP 回调保存视频路径方法
- ③ refreshVideoList.py: 返回视频回放路径列表方法

(二) 发布流程

1. 将工程项目打包，命名为当前 release 版本
2. 上传至 git: intern 下 release 文件夹