介绍

HttpRunner 是一款面向 HTTP(S) 协议的通用测试框架，只需编写维护一份 YAML/JSON 脚本，即可实现自动化测试、性能测试、线上监控、持续集成等多种测试需求。

核心特性

1.继承 Requests 的全部特性，轻松实现 HTTP(S) 的各种测试需求

2.采用 YAML/JSON 的形式描述测试场景，保障测试用例描述的统一性和可维护性

3.借助辅助函数（debugtalk.py），在测试脚本中轻松实现复杂的动态计算逻辑

4.支持完善的测试用例分层机制，充分实现测试用例的复用

5.测试前后支持完善的 hook 机制

6.响应结果支持丰富的校验机制

7.基于 HAR 实现接口录制和用例生成功能（har2case）

8.结合 Locust 框架，无需额外的工作即可实现分布式性能测试

9.执行方式采用 CLI 调用，可与 Jenkins 等持续集成工具完美结合

10.测试结果统计报告简洁清晰，附带详尽统计信息和日志记录

11.极强的可扩展性，轻松实现二次开发和 Web 平台化

Httprunner安装

命令安装

Pip install httprunner

安装完之后 hrun -V查看是否安装成功，出现版本号说明安装成功

安装 locust

pip install locust

新建项目：httprunner startproject name

项目文件结构：

debugtalk.py（可选）：存储项目中逻辑运算辅助函数

.env（可选）：存储项目环境变量，通常用于存储项目敏感信息（我把配置信息都下写在这里了）

csv（可选）：项目数据文件，用于进行数据驱动（位置与用例同位置，使用示例"${parameterize(testcases/username.csv)}"）

logs：默认生成测试报告的存储文件夹

测试用例（testcases）：对应多个 py 文件，每个文件是一个测试用例，测试用例中可包含多个测试步骤

用例详见py文件

运行命令：

在与api testcases 同级目录下 D:\httprunner

如果想运行某一个case 例如 game\_info

pytest报告

hrun api\login\_test.py --html=reports\report.html

测试报告就会出现在 reports（路径）中名称是report.html，路径和名称都可以更改

allure报告

安装

<http://allure.qatools.ru/> 下载

配置环境变量

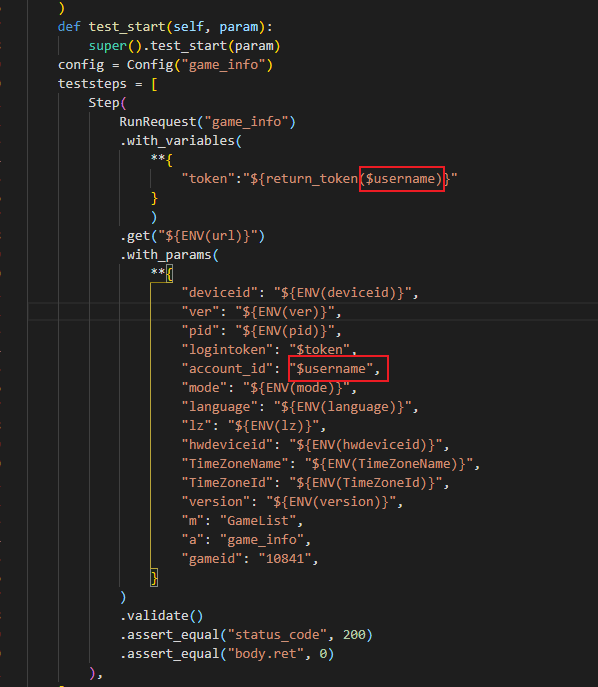
allure-2.8.0\bin文件夹，会看到allure.bat文件，讲此路径设置为系统环境变量path下，这样cmd任意目录都能执行了

（1）hrun api\login\_test.py --alluredir=report命令中的 --alluredir=report 指明了生成的json结果文件存放的目录为当前目录下的report文件夹

（2）运行命令 allure generate report这个命令会将 report 文件夹下的json文件渲染成网页结果，方便观看。生成的网页结果默认保存在当前文件夹下的 allure-report 文件夹内

3）allure open allure-report 渲染报告并打开

性能测试locust



前提：

需要把这两个参数改为实际的账号，例 “22993570”

因为使用这个的时候，会因为取数得时候超时，导致性能测试的时候报错，不能取到username

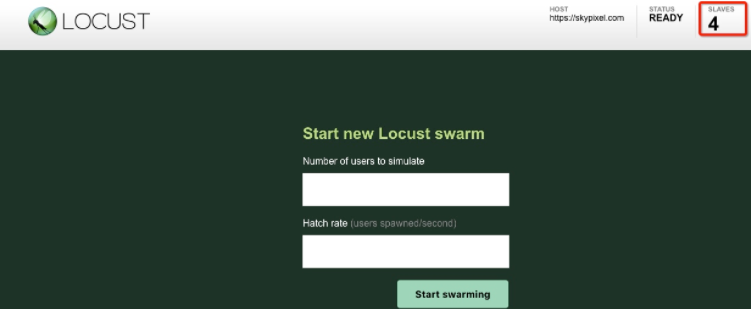
安装

Pip install locust

执行

1.Locusts -f test\_demo.py 或者 locusts -f testcases

2.浏览器打开 localhost:8086



解释：

Number of users to simulate：设置模拟的用户总数

Hatch rate (users spawned/second)：每秒启动的虚拟用户数