

# Тестирование программного обеспечения

## Лабораторная работа 6

### Разработка сценария

#### нагрузочного тестирования с использованием

#### Locust

Дорогин Никита ИП-312

---

## ЧАСТЬ 1: Настройка окружения.

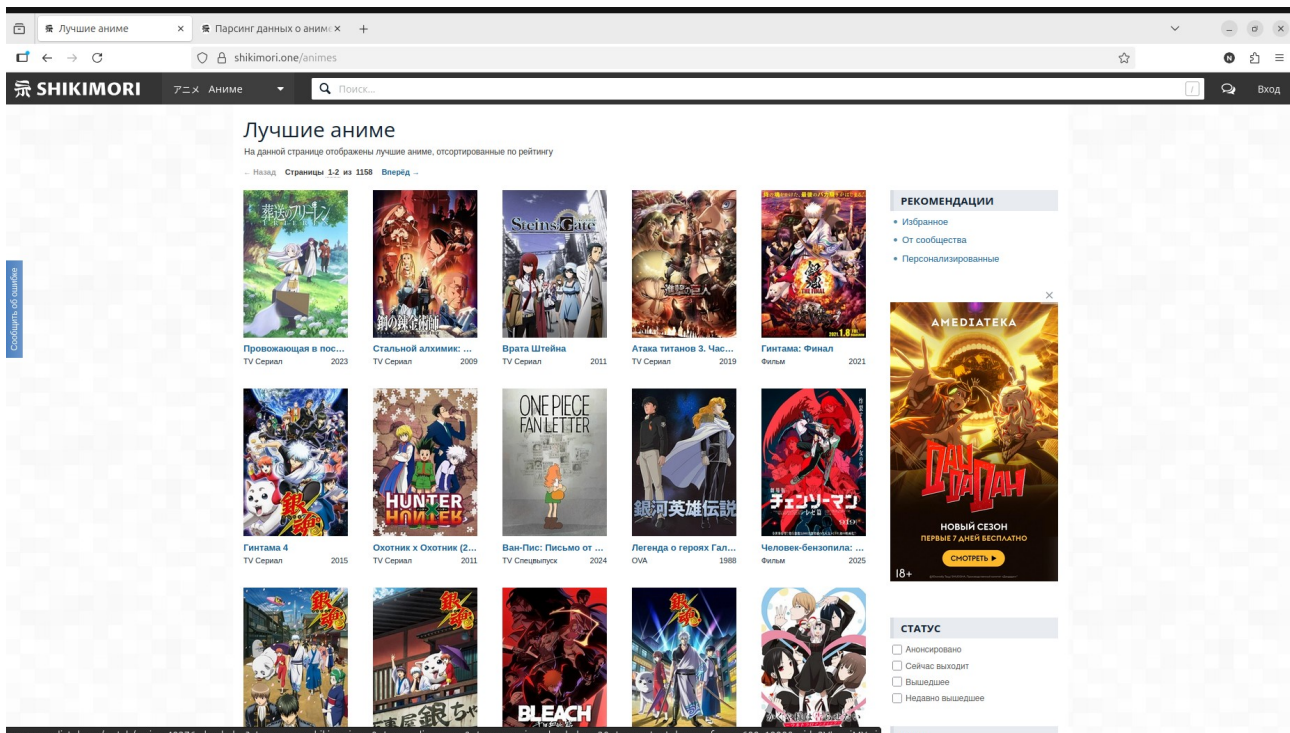
Установка зависимостей Locust:

```
gastello123@gastello123:~$ sudo apt install python3 locust
[sudo] пароль для gastello123:
Чтение списков пакетов... Готово
Построение дерева зависимостей... Готово
Чтение информации о состоянии... Готово
Заметьте, вместо «locust» выбирается «python3-locust»
Уже установлен пакет python3 самой новой версии (3.12.3-0ubuntu2).
Будут установлены следующие дополнительные пакеты:
  fonts-font-awesome fonts-lato libev4t64 libnorm1t64 libpgm-5.3-0t64
  libsodium23 libzmq5 python3-asgiref python3-brotli
  python3-charset-normalizer python3-configargparse python3-flask
```

Проверка доступности:

```
gastello123@gastello123: ~
gastello123@gastello123:~$ curl -k -u root:openBmc https://10.0.2.15/redfish/v1/
curl: (28) Failed to connect to 10.0.2.15 port 443 after 134427 ms: Couldn't connect to server
gastello123@gastello123:~$ curl -k -u root:openBmc https://10.0.2.2/redfish/v1/
curl: (28) Failed to connect to 10.0.2.2 port 443 after 134203 ms: Couldn't connect to server
gastello123@gastello123:~$ curl -k -u root:openBmc https://127.0.0.1:2443/redfish/v1/
{
  "@odata.id": "/redfish/v1",
  "@odata.type": "#ServiceRoot.v1_15_0.ServiceRoot",
  "AccountService": {
    "@odata.id": "/redfish/v1/AccountService"
  },
  "Cables": {
    "@odata.id": "/redfish/v1/Cables"
  },
  "CertificateService": {
    "@odata.id": "/redfish/v1/CertificateService"
  },
  "Chassis": {
    "@odata.id": "/redfish/v1/Chassis"
  },
  "EventService": {
    "@odata.id": "/redfish/v1/EventService"
  },
  "Id": "RootService",
  "JsonSchemas": {
    "@odata.id": "/redfish/v1/JsonSchemas"
  },
  "Links": {
    "ManagerProvidingService": {
      "@odata.id": "/redfish/v1/Managers/bmc"
    }
  },
}
```

## Выбор публичного API для тестирования:



## Часть 2: Разработка сценария нагрузочного тестирования.

Locust-скрипт (locustfile.py) с двумя классами нагрузки:

```
1 from locust import HttpUser, task, between
2 from requests.auth import HTTPBasicAuth
3
```

### 1. Тестирование OpenBMC API

```
4 class openBmcTester(HttpUser):
5
6     host = 'https://127.0.0.1:2443'
7     auth = HTTPBasicAuth('root', 'OpenBmc')
8
9     def on_start(self):
10         pass
11
12     def on_stop(self):
13         pass
14
```

- Запрос на получение информации о системе (/redfish/v1/Systems/system):

```
@task(1)
def sys_info(self):
    try:
        response = self.client.get('/redfish/v1/Systems/system', auth=self.auth, verify= False)
        data = response.json()
        has_Status = 'Status' in data
        has_Power = 'PowerState' in data
        if(not has_Status): print("NO STATUS")
        if(not has_Power): print("NO POWER")
        return response != None and has_Status and has_Power
    except Exception:
        print("OpenBMC API DOESN'T ANSWER")
        return False
```

- Запрос состояния питания (PowerState):

```
@task(2)
def power_info(self):
    try:
        response = self.client.get(self.host + '/redfish/v1/Systems/system', auth=self.auth, verify= False)
        data = response.json()

        try:
            power = data('PowerState')
        except Exception:
            print("THERE'S NO FIELD PowerState in the API")
            return False

        return (power != None)
    except Exception:
        print("OpenBMC API DOESN'T ANSWER")
        return False
```

## 2. Тестирование публичного API

```
48 class shikimoriTester(HttpUser):
49     host = 'https://shikimori.one'
50
51
52     #-----
53     # Имитация браузера и поведения человека в нём, чтобы Шикимори не думал, что мы бот
54     wait_time = between(1, 3)
55     def on_start(self):
56         self.headers = {
57             "User-Agent": "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4
58             "Accept": "text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8",
59             "Accept-Language": "ru-RU,ru;q=0.8,en-US;q=0.5,en;q=0.3",
60             "Accept-Encoding": "gzip, deflate, br",
61             "Connection": "keep-alive",
62             "Upgrade-Insecure-Requests": "1",
63             "Sec-Fetch-Dest": "document",
64             "Sec-Fetch-Mode": "navigate",
65             "Sec-Fetch-Site": "none",
66         }
67     #-----
68     def on_stop(self):
69         pass
70
```

- Запрос списка аниме:

```
70
71 @task(1)
72 def anime_list(self):
73     try:
74         response = self.client.get('/animés', verify= False, headers=self.headers )
75         data = response.json()
76         has_Name = 'name' in data
77         has_Rate = 'rate_score' in data
78         has_Eps = 'episodes' in data
79         has_Kind = 'kind' in data
80         if(not has_Name): print("LIST: NO NAME")
81         if(not has_Rate): print("LIST: NO RATE SCORE")
82         if(not has_Eps): print("LIST: NO EPISODES")
83         if(not has_Kind): print("LIST: NO KIND")
84         return response != None and has_Name and has_Kind and has_Eps and has_Rate
85     except Exception:
86         print("Shikimori API DOESN'T ANSWER")
87         return False
88
```

- Запрос первого аниме в списке (на самом деле сомнительный тест, так как он опирается на сравнение с тем названием, которое стоит на первом месте в данный момент и которое позже может измениться, сделав тест нерабочим):

```
89
90 @task(2)
91 def get_first(self):
92     try:
93         response = self.client.get('/animés', verify= False, headers=self.headers)
94         data = response.json()
95         first = data[0]
96         has_Name = 'name' in first
97         has_Rate = 'rate_score' in first
98         has_Eps = 'episodes' in first
99         has_Kind = 'kind' in first
100         if(not has_Name): print("FIRST: NO NAME")
101         if(not has_Rate): print("FIRST: NO RATE SCORE")
102         if(not has_Eps): print("FIRST: NO EPISODES")
103         if(not has_Kind): print("FIRST: NO KIND")
104
105         if(first['name'] == "Провожающая в последний путь Фрирен"): return True
106         else: return False
107     except Exception:
108         print("Shikimori API DOESN'T ANSWER")
109         return False
110
```

### Часть 3: Запуск нагрузочного тестирования.


PROBLEMS   OUTPUT   DEBUG CONSOLE   TERMINAL   PORTS

```
gastello123@gastello123:~/Desktop/ТестированиеПо/testP0/Lab6$ locust
[2025-10-18 23:16:54,106] gastello123/INFO/locust.main: Starting web interface at http://0.0.0.0:8089
[2025-10-18 23:16:54,114] gastello123/INFO/locust.main: Starting Locust
```

Oct 18 23:18

Новая вкладка x Locust x +

← → ↻ Не защищено 0.0.0.0:8089

 **Locust**

HOSTSTATUSRPSFAILURES  
READY00%

Start new load test

Number of users (peak concurrency)

10

Ramp Up (users started/second)

2

Host (setting this will override the host for the User classes)

Advanced options


▼

START SWARM

Oct 18 23:21 en

Новая вкладка x Locust x +

← → ↻ Не защищено 0.0.0.0:8089/tab=stats

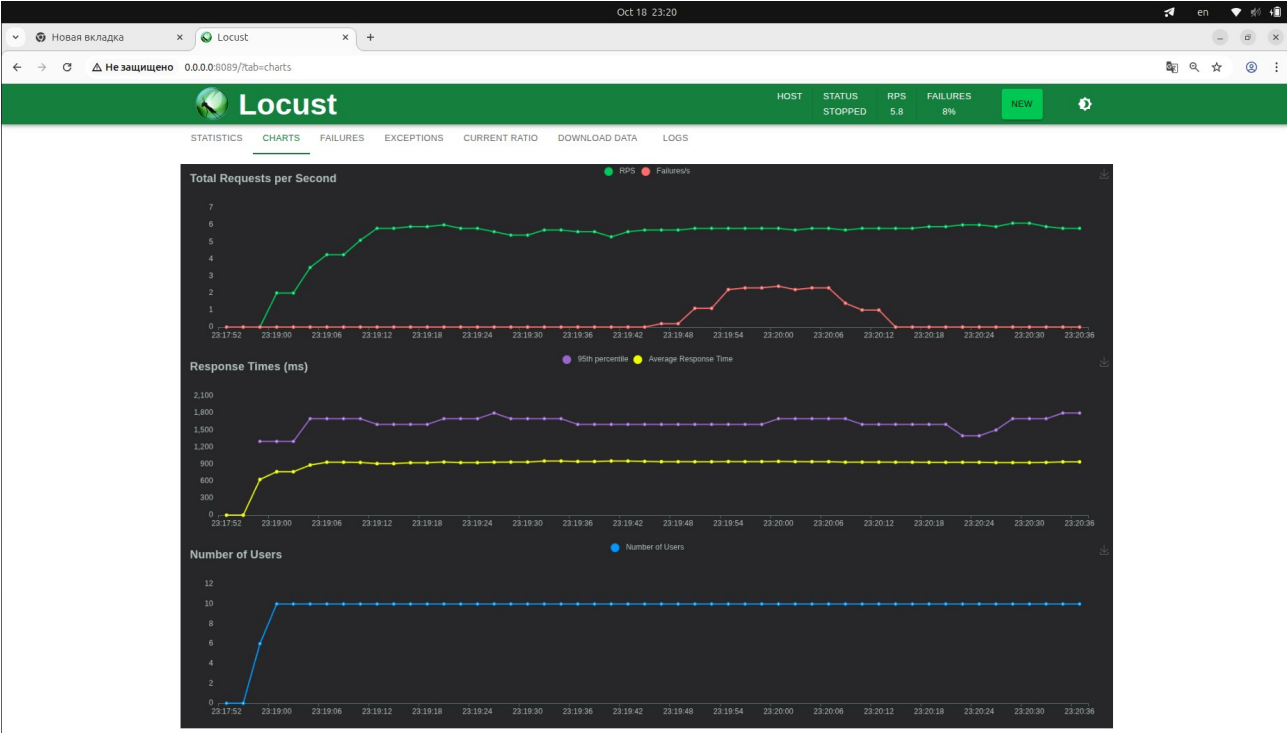
 **Locust**

HOSTSTATUSRPSFAILURESNEW  
STOPPED5.88%

STATISTICSCHARTSFAILURESEXCEPTIONSCURRENT RATIODOWNLOAD DATALOGS

Type	Name	# Requests	# Fails	Median (ms)	95%ile (ms)	99%ile (ms)	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	Current RPS	Current Failures/s
GET	/animes	218	45	110	1000	1400	216.22	68	1829	34375.51	2.3	0
GET	/redfish/v1/Systems/system	340	0	1400	1700	1800	1402.77	359	1883	2578	3.5	0
Aggregated		558	45	1300	1600	1800	939.21	68	1883	15000.68	5.8	0





Oct 18 23:22

Новая вкладка x Locust

← → ↻ Не защищено 0.0.0.0:8089/tab=failures

# Locust

HOST STATUS STOPPED RPS 5.8 FAILURES 8% NEW

STATISTICS CHARTS FAILURES EXCEPTIONS CURRENT RATIO DOWNLOAD DATA LOGS

# Failures	Method	Name	Message
45	GET	/animes	HTTPError(429 Client Error: Too Many Requests for url: /animes)



# Locust

[STATISTICS](#)[CHARTS](#)[FAILURES](#)[EXCEPTIONS](#)[CURRENT RATIO](#)

## Ratio Per Class

- 50.0% openBmcTester
  - 66.7% powerInfo
  - 33.3% sysInfo
- 50.0% shikimoriTester
  - 33.3% animeList
  - 66.7% getFirst

## Total Ratio

- 50.0% openBmcTester
  - 33.3% powerInfo
  - 16.7% sysInfo
- 50.0% shikimoriTester
  - 16.7% animeList
  - 33.3% getFirst



# Locust

[HOST](#)[STATUS](#)  
[STOPPED](#)[STATISTICS](#)[CHARTS](#)[FAILURES](#)[EXCEPTIONS](#)[CURRENT RATIO](#)[DOWNLOAD DATA](#)[LOGS](#)

## Logs

- [2025-10-18 23:16:54,106] gastello123/INFO/locust.main: Starting web interface at http://0.0.0.0:8089
- [2025-10-18 23:16:54,114] gastello123/INFO/locust.main: Starting Locust
- [2025-10-18 23:18:54,707] gastello123/INFO/locust.runners: Ramping to 10 users at a rate of 2.00 per second
- [2025-10-18 23:18:58,710] gastello123/INFO/locust.runners: All users spawned: {"openBmcTester": 5, "shikimoriTester": 5} (10 total users)

```
KeyboardInterrupt
2025-10-18T16:21:38Z
[2025-10-18 23:21:38,313] gastello123/INFO/locust.main: Shutting down (exit code 1)
Type      Name                                     # reqs      # fails      Avg      Min      Max      Med      req/s      failures/s
-----
GET      /animés                                     224      45 (20.09%)    213      68      1829     110      2.22      0.45
GET      /redfish/v1/Systems/system                 350      0 (0.00%)     1404     358      1883     1400      3.47      0.00
-----
Aggregated                                     574      45 (7.84%)    939      68      1883     1300      5.69      0.45

Response time percentiles (approximated)
Type      Name                                     50%      66%      75%      80%      90%      95%      98%      99%      99.9%      99.99%      100%      # reqs
-----
GET      /animés                                     110      130      140      150      540      960      1300     1400     1800     1800     1800     224
GET      /redfish/v1/Systems/system                 1400     1500     1500     1500     1600     1700     1800     1800     1900     1900     1900     350
-----
Aggregated                                     1300     1400     1400     1500     1600     1600     1700     1800     1900     1900     1900     574


Error report
# occurrences      Error
-----
45      GET /animés: HTTPError('429 Client Error: Too Many Requests for url: /animés')
```

Попробуем тест с большим количеством пользователей:

Новая вкладка

Locust

← → ↻ Не защищено 0.0.0.0:8089

 **Locust**

HOST: 0.0.0.0:8089 STATUS: READY RPS: 0 FAILURES: 0%

Start new load test

Number of users (peak concurrency)

500

Ramp Up (users started/second)

1

Host (setting this will override the host for the User classes)

Advanced options


▼

START SWARM

Новая вкладка

Locust

← → ↻ Не защищено 0.0.0.0:8089/?tab=stats

 **Locust**

HOST: 0.0.0.0:8089 STATUS: STOPPED RPS: 46 FAILURES: 83% NEW

STATISTICS

CHARTS

FAILURES

EXCEPTIONS

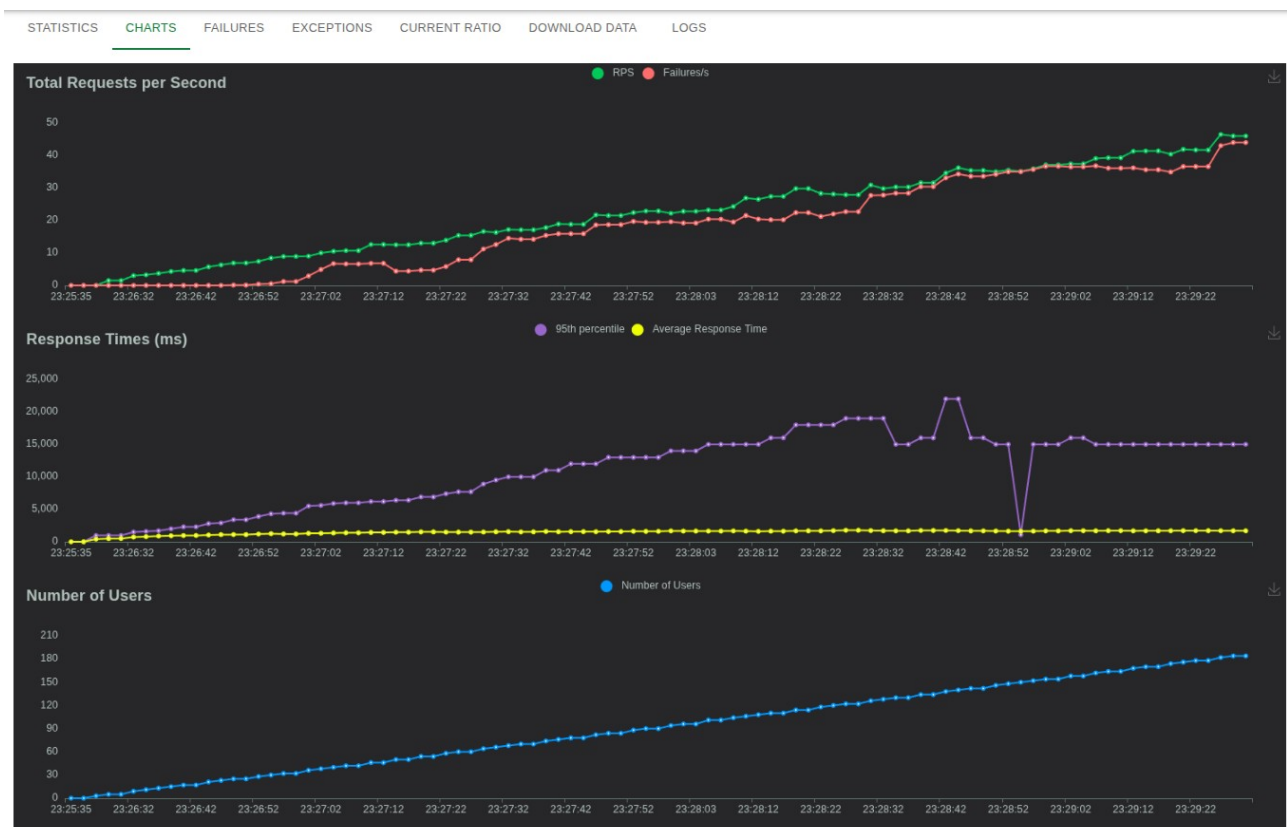
CURRENT RATIO

DOWNLOAD DATA

LOGS

Type	Name	# Requests	# Fails	Median (ms)	95%ile (ms)	99%ile (ms)	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	Current RPS	Current Failures/s
GET	/animés	3949	3589	71	890	1500	163.44	62	5839	3959.51	42.5	40.5
GET	/redfish/v1/Systems/system	540	156	14000	33000	43000	12901.69	295	47188	1926	3.5	3.5
Aggregated		4489	3745	72	15000	22000	1695.77	62	47188	3714.89	46	44





STATISTICS

CHARTS

FAILURES

EXCEPTIONS

CURRENT RATIO

DOWNLOAD DATA

LOGS

# Failures	Method	Name	Message
3589	GET	/animes	HTTPError(429 Client Error: Too Many Requests for url: /animes)
146	GET	/redfish/v1/Systems/system	RemoteDisconnected('Remote end closed connection without response')
10	GET	/redfish/v1/Systems/system	HTTPError(500 Server Error: Internal Server Error for url: /redfish/v1/Systems/system)

## Ratio Per Class

- 50.3% openBmcTester
  - 66.7% powerInfo
  - 33.3% sysInfo
- 49.7% shikimoriTester
  - 33.3% animeList
  - 66.7% getFirst

## Total Ratio

- 50.3% openBmcTester
  - 33.5% powerInfo
  - 16.8% sysInfo
- 49.7% shikimoriTester
  - 16.6% animeList
  - 33.2% getFirst

## Logs

- [2025-10-18 23:25:33,050] gastello123/INFO/locust.main: Starting web interface at http://0.0.0.0:8089
- [2025-10-18 23:25:33,057] gastello123/INFO/locust.main: Starting Locust
- [2025-10-18 23:26:22,756] gastello123/INFO/locust.runners: Ramping to 500 users at a rate of 1.00 per second

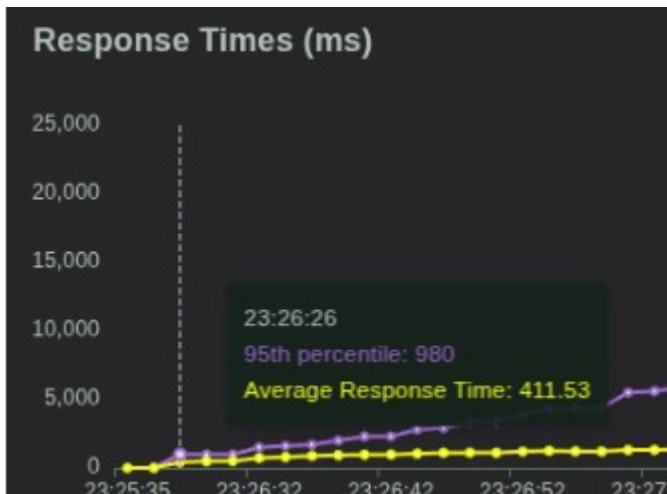
```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Shikimori API DOESN'T ANSWER
Shikimori API DOESN'T ANSWER
Shikimori API DOESN'T ANSWER
Shikimori API DOESN'T ANSWER
Shikimori API DOESN'T ANSWER
Shikimori API DOESN'T ANSWER
Shikimori API DOESN'T ANSWER
Shikimori API DOESN'T ANSWER
Shikimori API DOESN'T ANSWER
/usr/lib/python3/dist-packages/urllib3/connectionpool.py:1100: InsecureRequestWarning: Unverified HTTPS request is being made to host 'shikimori.one'. Adding certificate verification is strongly advised. See: https://urllib3.readthedocs.io/en/latest/advanced-usage.html#tls-warnings
warnings.warn(
KeyboardInterrupt
2025-10-18T16:30:55Z
[2025-10-18 23:30:55,743] gastello123/INFO/locust.main: Shutting down (exit code 1)
Type Name # reqs # fails Avg Min Max Med req/s failures/s
GET /animés 4059 3699(91.13%) 167 62 5838 71 21.76 19.83
GET /redfish/v1/Systems/system 549 165(30.05%) 13005 294 48534 14000 2.94 0.88
Aggregated 4608 3864(83.85%) 1697 62 48534 72 24.70 20.71

Response time percentiles (approximated)
Type Name 50% 66% 75% 80% 90% 95% 98% 99% 99.9% 99.99% 100% # reqs
GET /animés 71 74 86 100 330 890 1200 1500 2500 5800 5800 4059
GET /redfish/v1/Systems/system 14000 15000 16000 16000 20000 33000 41000 44000 49000 49000 49000 549
Aggregated 72 86 120 430 4300 15000 17000 22000 44000 49000 49000 4608

Error report
# occurrences Error
3699 GET /animés: HTTPError('429 Client Error: Too Many Requests for url: /animés')
154 GET /redfish/v1/Systems/system: RemoteDisconnected('Remote end closed connection without response')
11 GET /redfish/v1/Systems/system: HTTPError('500 Server Error: Internal Server Error for url: /redfish/v1/Systems/system')
```

## Часть 4: Анализ результатов.

- Среднее время отклика API (метрики Locust).



Чуть больше 1 секунды.

- Процент ошибок (если API не выдерживает нагрузку).

При 10 людях с добавлением 2 в секунду ошибок 8%.

При 500 людях с добавлением 1 в секунду ошибок 83%

В первом случае все, а во втором 96% ошибок приходится на запросы к открытому API

- Запросы к открытому API Shikimori начинают возвращать ошибки от перегрузки уже после 25 человек, в то время как у openBMC они начинаются когда людей становится уже 100 и больше.

Все графики можно наблюдать выше в части 3.

Ссылка на GitHub: <https://github.com/NekitD/testPO/tree/main/Lab6>