Politechnika Wrocławska

Rekonfigurowalność e-systemów

Testowanie wydajnościowe aplikacji webowej InnoPoint

Prowadzący: Dr inż. Dariusz Caban

Jakub Pawłowski 235002

HR EXCELLENCE IN RESEARCH

Aplikacja do testowania wydajnościowego

InnoPoint jest aplikacją do zarządzania projektami zespołowymi

Narzędzia do testowania wydajnościowego

1. Laptop Lenovo ThinkPad T510
   System operacyjny Windows 7

Procesor:	Intel(R) Core(TM) i5 CPU M 520 @ 2.40GHz 2.40 GHz
Zainstalowana pamięć (RAM):	4,00 GB (dostępne: 3,80 GB)
Typ systemu:	64-bitowy system operacyjny

2. Locust - tworzy scenariusze obciążenia przy użyciu Pythona oraz obsługuje obciążenie rozproszone.

3. Psutil - biblioteka do Python'a, pozwala na pomiary obciążenia procesora i pamięci dla wybranego procesu.

Scenariusze użycia

1. Scenariusz standardowego użycia:
   a. Użytkownik loguje się do aplikacji,
   b. Tworzy grupę projektową,
   c. Wybiera projekt,

2. Scenariusz możliwości prowadzący:
   a. Prowadzący loguje się do aplikacji,
   b. Dodaje nowy projekt,
   c. Powiadamia o tym użytkowników, pisząc post na tablicy (w aplikacji),
   d. Przeprowadza weryfikację projektu.

3. Scenariusz testujący serwer losowymi zapytaniami w celu poprawnego obciążenia.

Scenariusz użycia - przykład

```
class AdminScenario(TaskSet):
    def on_start(self):
        self.login()
    def login(self):
        response = self.client.put("/user", {"token": token})
        print("Resposne code:", response.status_code)
        print("Response plain text: ", response.text)
    @task(1)
    def admin_scenario(self):
        project_id = self.add_project()
        self.verify_project(project_id)
    def add_project(self):
        project_data = {"project": {
            "name": "Performance Testing",
            "short_description": "Prepare performance tests for web application.",
            "long_description": "Vulputate odio ut enim blandit volutpat maecenas",
            "number_of_members": 5,
            "technology": "Python, Locust",
            "tags": "WebDev, WebTest",
            "requirements": "none",
            "theme_color": "#3f51b5",
            "verified": 1}, "token": token}
        response = self.client.post("/projects", json=project_data)
        print("Resposne code:", response.status_code)
        print("Response plain text: ", response.text)
        response = json.loads(response.text)
        return response[0]['id']
    def verify_project(self, project_id=1):
        self.client.put("/projects/verify/{0}".format(project_id), {"token": token})
```

Obszar badań

- Na podstawie scenariusz przeprowadzono badania,
- Badania dotyczyły pomiaru:
  - stabilności liczby żądań na sekundę,
  - czas odpowiedzi serwera na żądanie,
  - zmiennej liczby użytkowników,
  - obciążenie procesora,
  - obciążenie pamięci.

Problemy podczas testowania

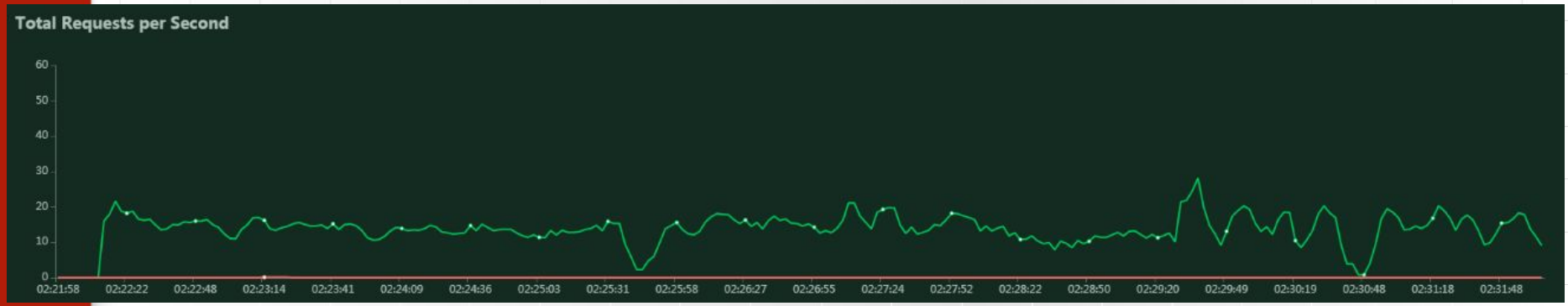
- Uruchomienie i konfiguracja serwera,
- Stworzenie nowej bazy danych.

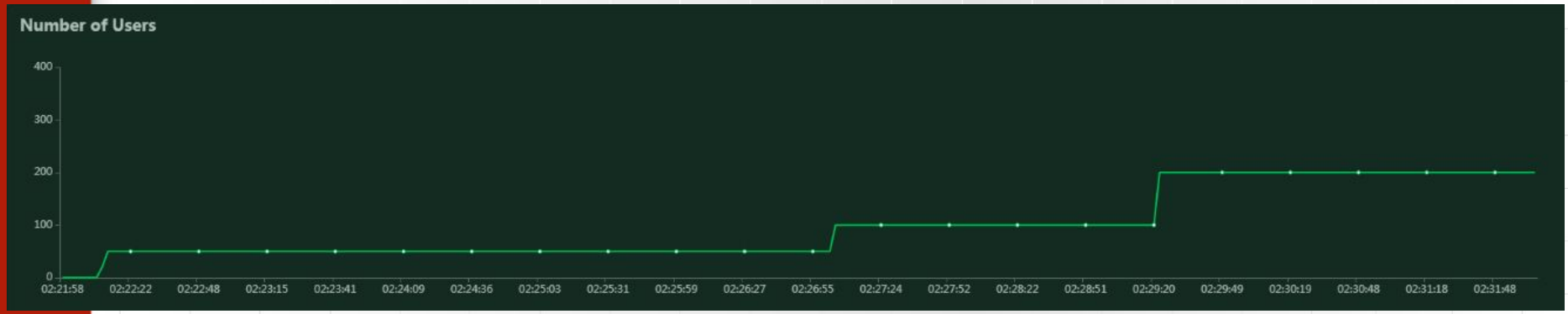
requests

# fails	Method	Name	Type
1	POST	/projects	ConnectionError(ProtocolError('Connection aborted.', ConnectionAbortedError(10053, 'Nawiązane połączenie zostało przerwane przez oprogramowanie zainstalowane w komputerze-hoście', None, 10053, None)))

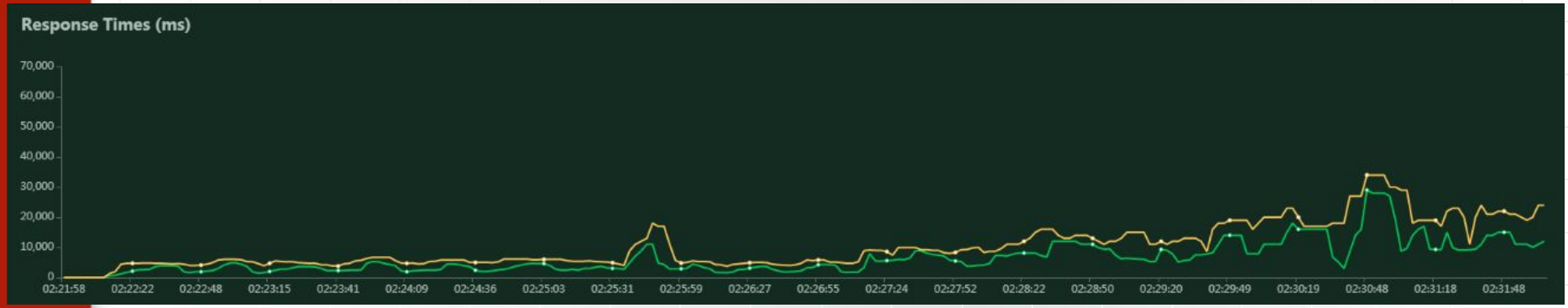
```
    'Cannot add or update a child row: a foreign key constraint fails (`inno-point`.`news`
, CONSTRAINT `news_ibfk_1` FOREIGN KEY (`user_id`) REFERENCES `user` (`id`))',
    sql:
     "INSERT INTO `news` (`id`,`title`,`body`,`date`,`user_id`) VALUES (0,'Performance Test
ing','New project has been added. You can check this out in the projects section!','2019-11-
19 21:43:27',101010101);" },
  sql:
   "INSERT INTO `news` (`id`,`title`,`body`,`date`,`user_id`) VALUES (0,'Performance Testing
','New project has been added. You can check this out in the projects section!','2019-11-19
21:43:27',101010101);",
  fields: [ 'user_id' ],
  table: 'user',
  value: undefined,
  index: 'news_ibfk_1',
  reltype: 'child' }
```

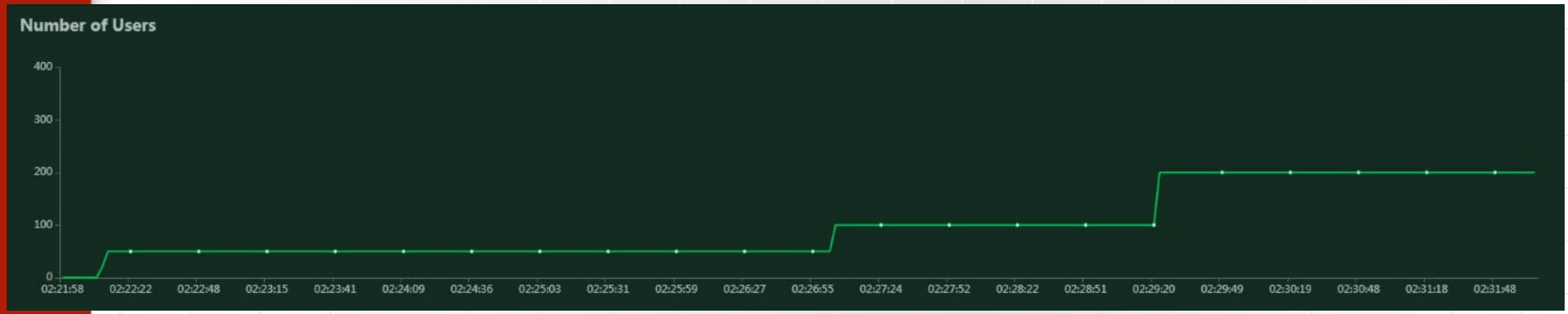
Wyniki badań - Scenariusz 1.



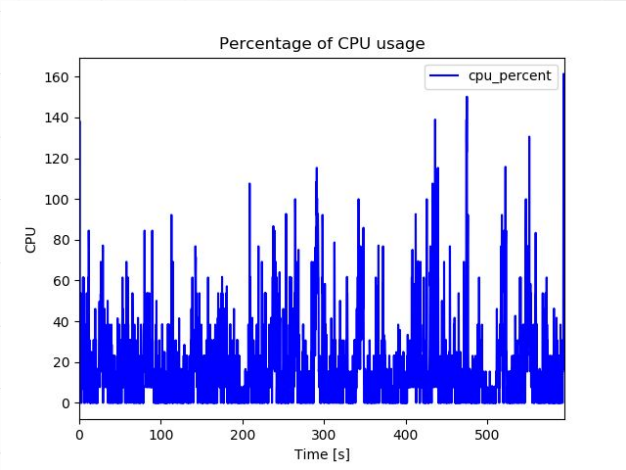


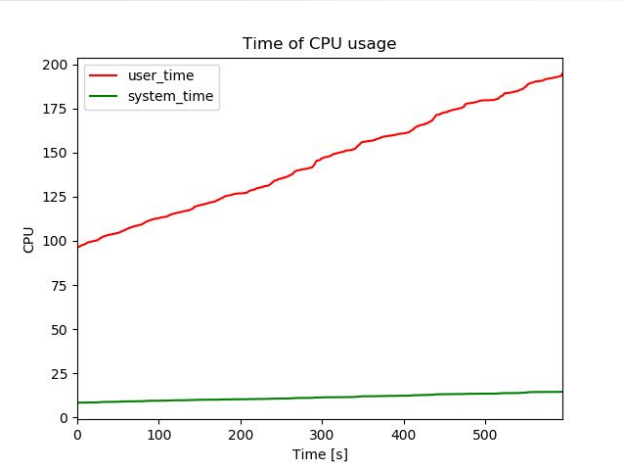
Wyniki badań - Scenariusz 1.



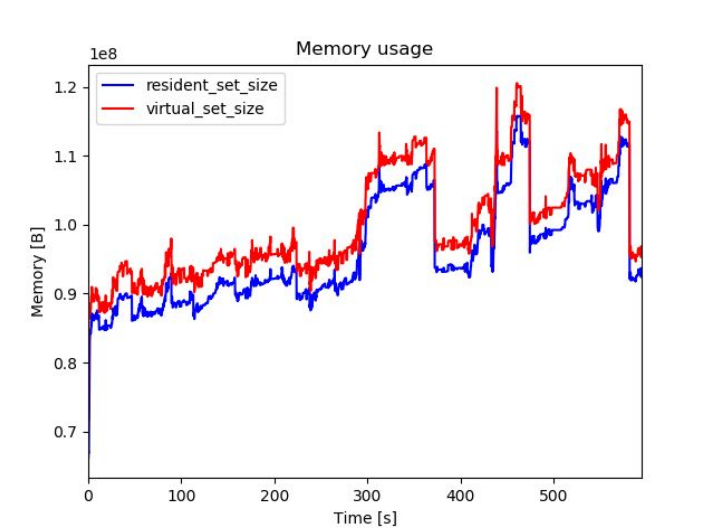


Wyniki badań - Scenariusz 1.



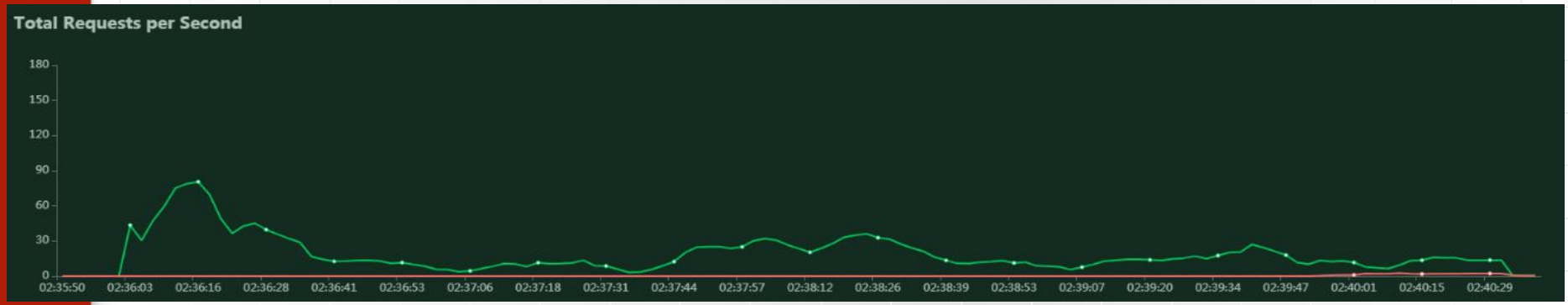


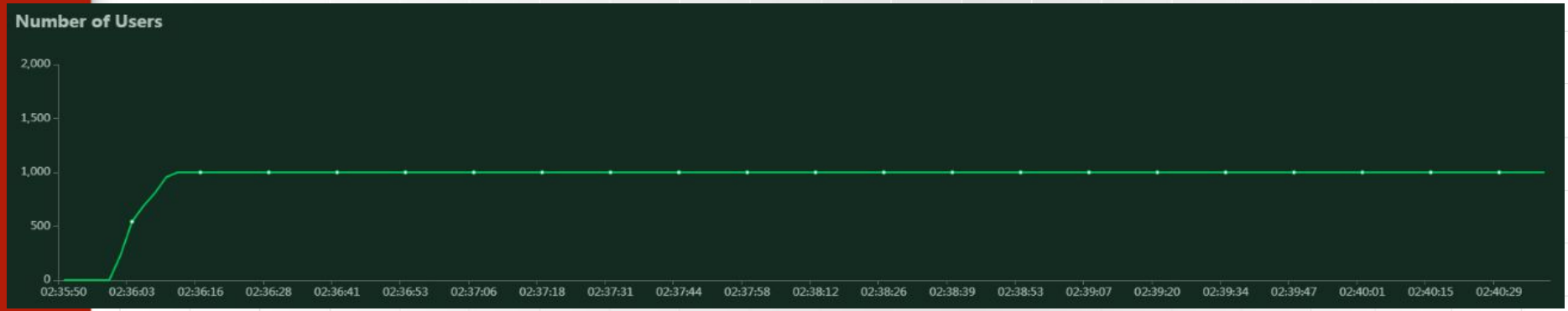
Wyniki badań - Scenariusz 1.



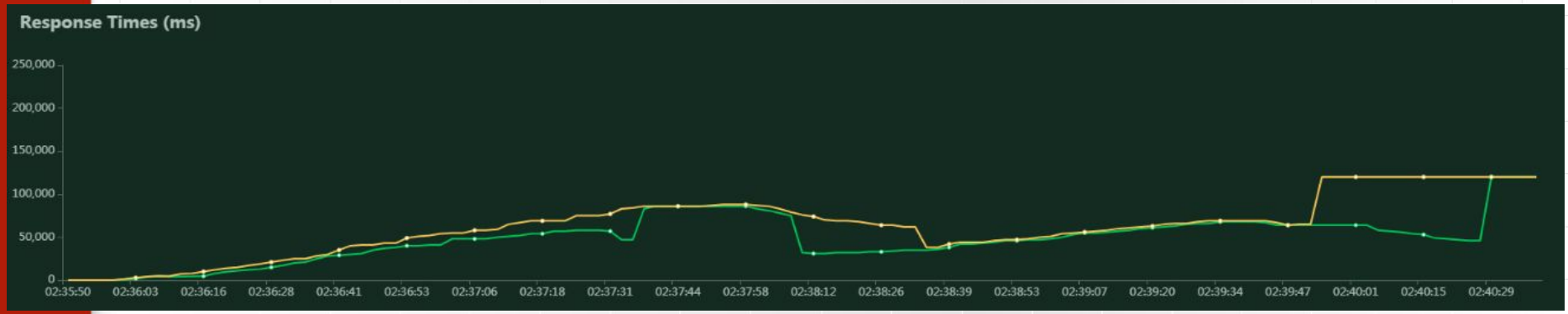
	Method	Name	# requests	# failures	Median response time	Average response time	Min response time	Max response time	Average Content Size	Requests/s
0	DELETE	/projects/	1.0	0.0	2742.175497	2742.175497	2742.175497	2742.175497	2.0	0.00
1	DELETE	/teams/	1.0	0.0	8736.218302	8736.218302	8736.218302	8736.218302	2.0	0.00
2	None	Aggregated	8354.0	1.0	4700.000000	6752.000000	4.000000	34553.000000	40.0	14.01
3	POST	/news	1100.0	0.0	4400.000000	5638.000000	556.000000	23180.000000	2.0	1.84
4	POST	/projects	1104.0	1.0	3400.000000	5076.000000	4.000000	22882.000000	244.0	1.85
5	POST	/teams	1102.0	0.0	3000.000000	5161.000000	673.000000	21325.000000	4.0	1.85
6	PUT	/projects//leave	1.0	0.0	9748.015890	9748.015890	9748.015890	9748.015890	2.0	0.00
7	PUT	/projects/apply/	1.0	0.0	10057.706775	10057.706775	10057.706775	10057.706775	2.0	0.00
8	PUT	/projects/verify/	1.0	0.0	8079.921960	8079.921960	8079.921960	8079.921960	2.0	0.00
9	PUT	/user	200.0	0.0	1100.000000	2265.000000	12.000000	4083.000000	279.0	0.34

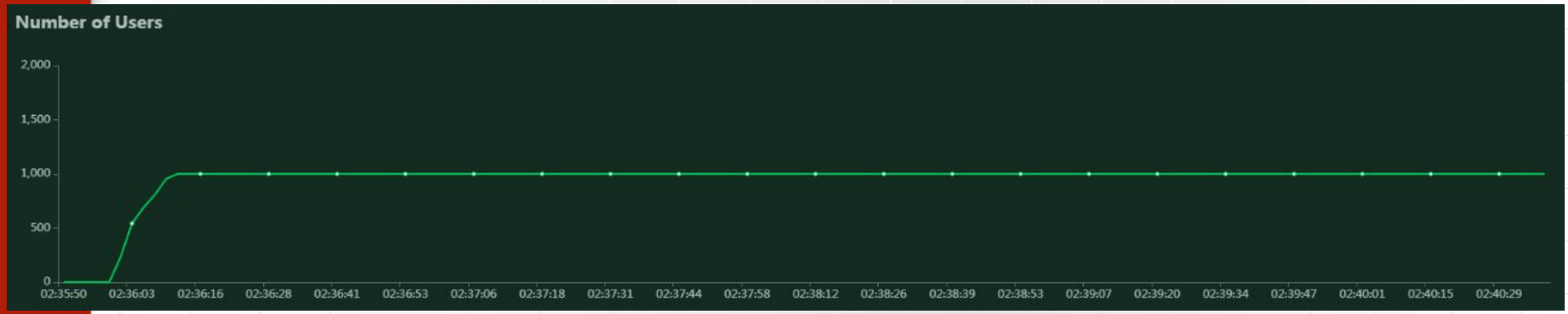
Wyniki badań - Scenariusz 1. - modyfikowany



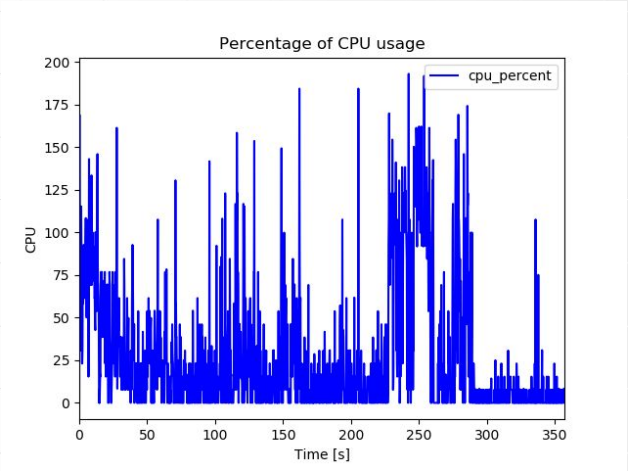


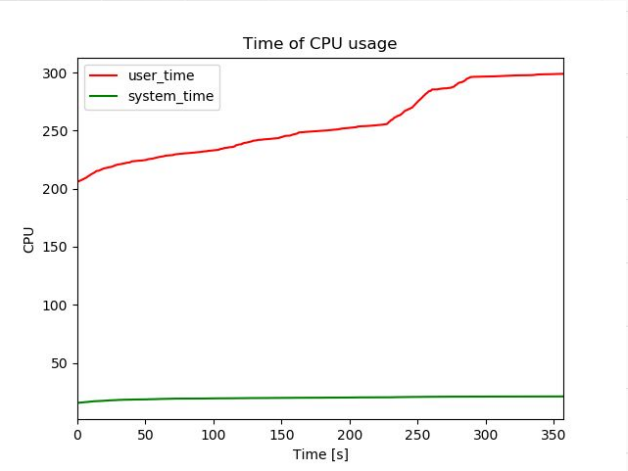
Wyniki badań - Scenariusz 1. - modyfikowany





Wyniki badań - Scenariusz 1. - modyfikowany

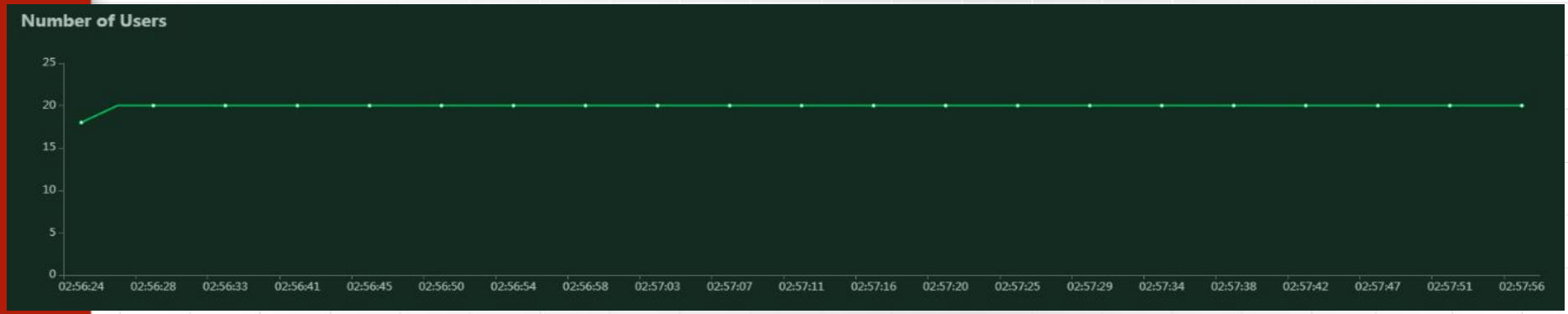




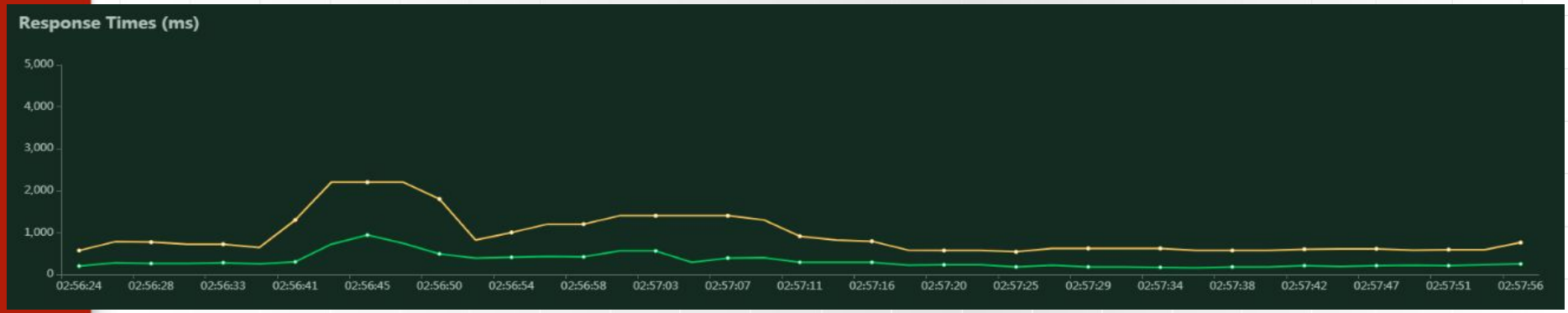
Wyniki badań - Scenariusz 1. - modyfikowany

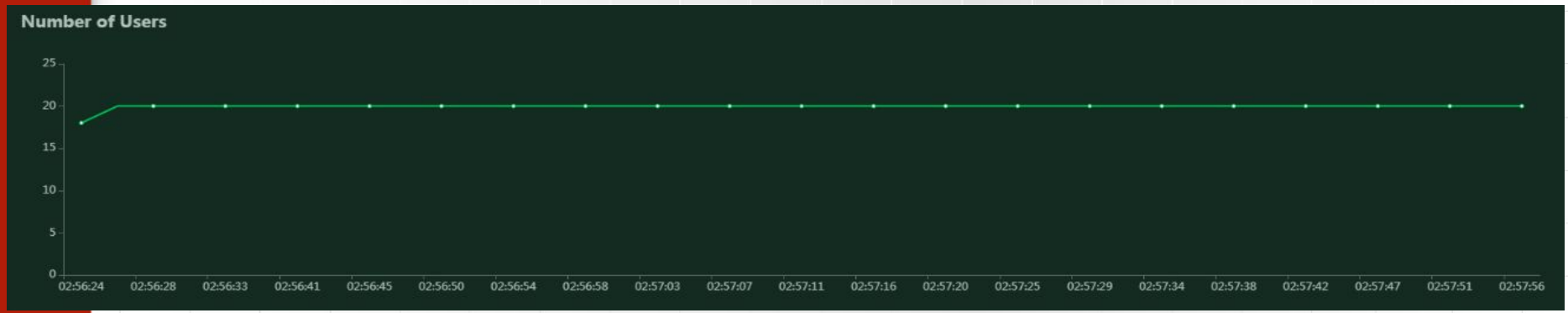
	Method	Name	# requests	# failures	Median response time	Average response time	Min response time	Max response time	Average Content Size	Requests/s
0	None	Aggregated	5075.0	75.0	44000.000	42928.000	188.000	120129.000	105.0	18.29
1	POST	/news	1000.0	0.0	63000.000	57755.000	29005.000	69197.000	2.0	3.60
2	POST	/projects	1000.0	0.0	21000.000	26431.000	673.000	58681.000	245.0	3.60
3	POST	/teams	1000.0	0.0	36000.000	42517.000	28326.000	69159.000	4.0	3.60
4	PUT	/projects/apply/	1.0	1.0	120027.440	120027.440	120027.440	120027.440	0.0	0.00
5	PUT	/projects/verify/	1.0	0.0	77018.898	77018.898	77018.898	77018.898	2.0	0.00
6	PUT	/user	1000.0	0.0	4400.000	5135.000	188.000	13498.000	282.0	3.60

Wyniki badań - Scenariusz 2.

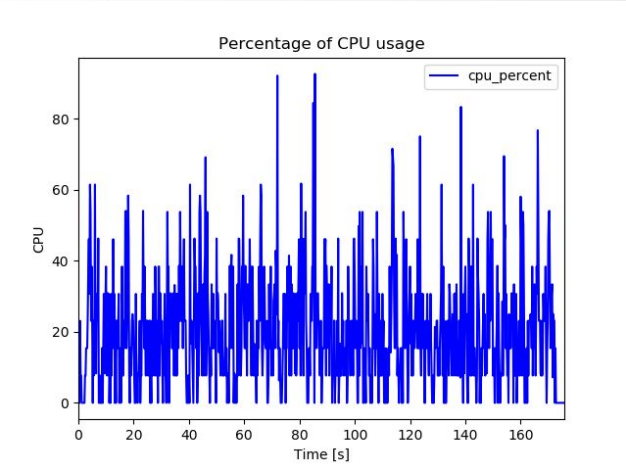


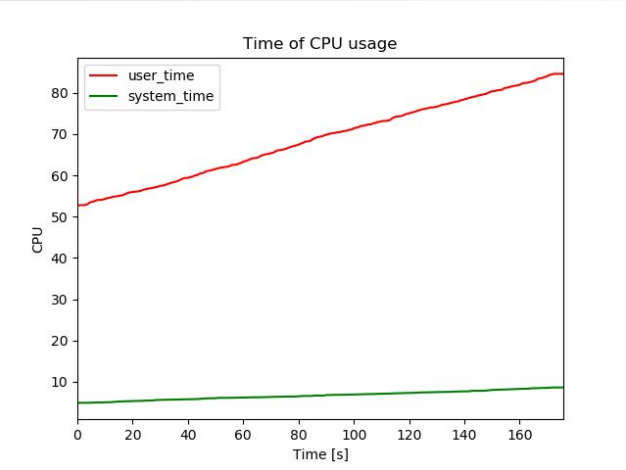
Wyniki badań - Scenariusz 2.



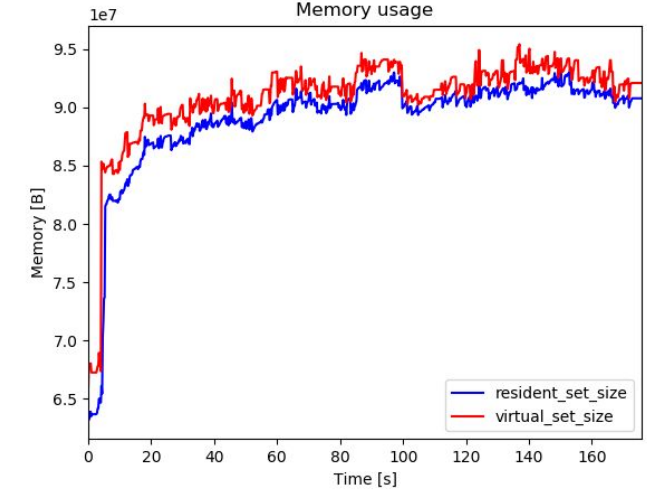


Wyniki badań - Scenariusz 2.



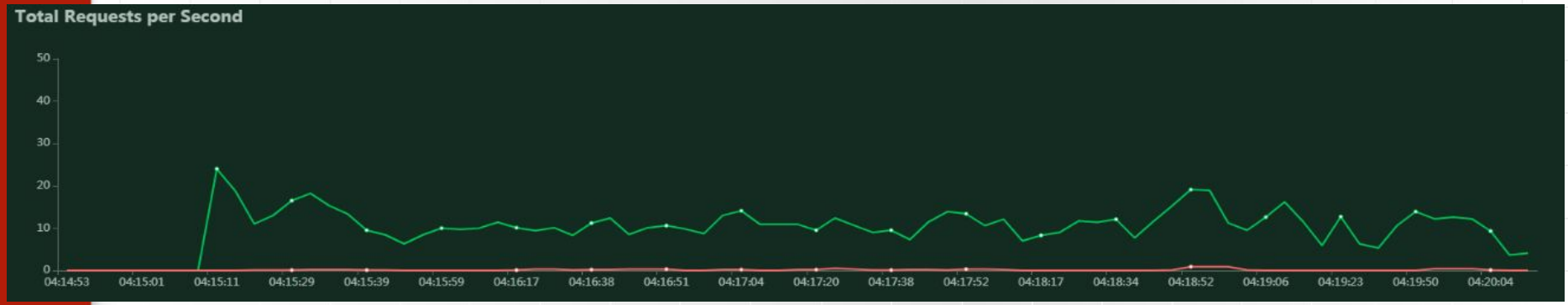


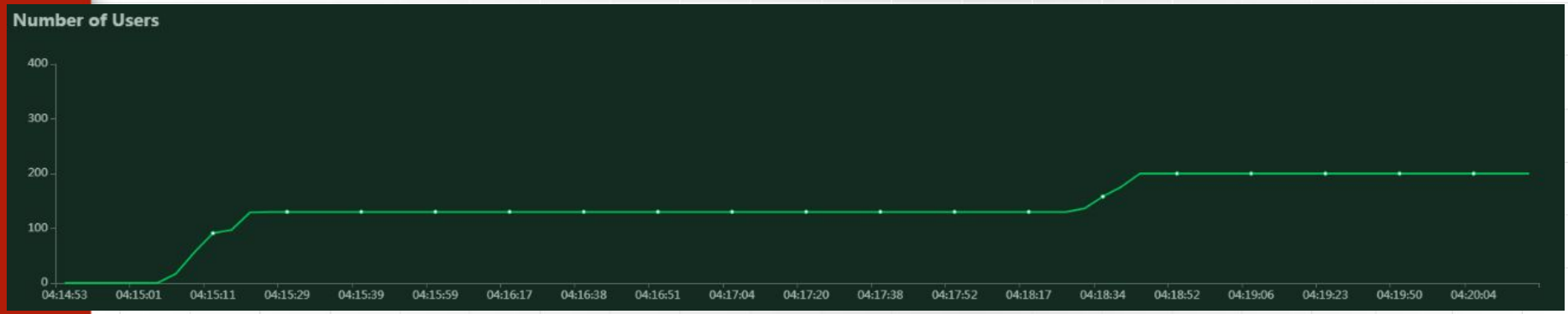
Wyniki badań - Scenariusz 2.



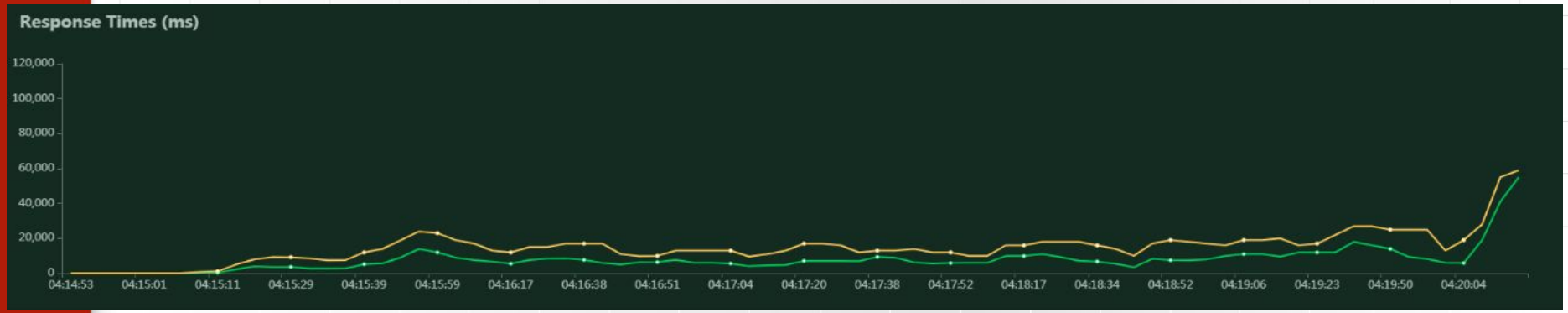
	Method	Name	# requests	# failures	Median response time	Average response time	Min response time	Max response time	Average Content Size	Requests/s
0	DELETE	/projects/	1.0	0.0	253.150000	253.150000	253.150000	253.150000	2.0	0.01
1	None	Aggregated	1100.0	0.0	270.000000	372.000000	12.000000	2783.000000	86.0	11.70
2	POST	/projects	360.0	0.0	230.000000	313.000000	35.000000	1501.000000	245.0	3.83
3	PUT	/projects/verify/	1.0	0.0	566.088889	566.088889	566.088889	566.088889	2.0	0.01
4	PUT	/user	20.0	0.0	94.000000	111.000000	12.000000	308.000000	279.0	0.21

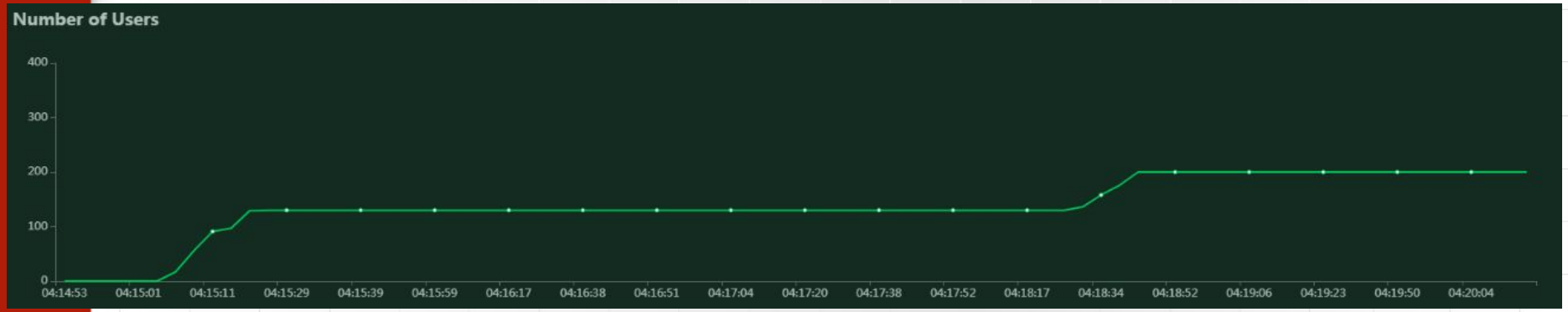
Wyniki badań - Scenariusz 3.



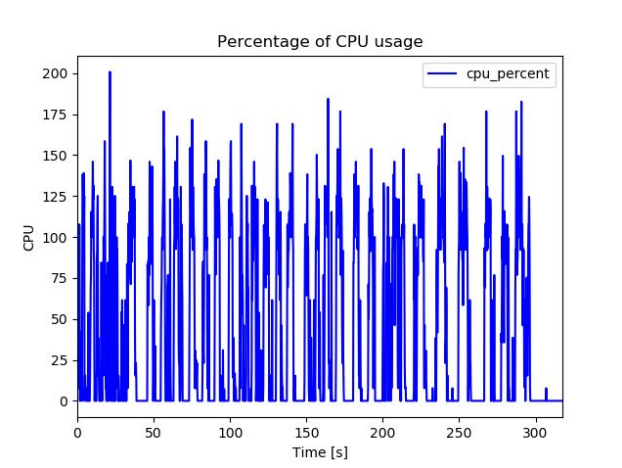


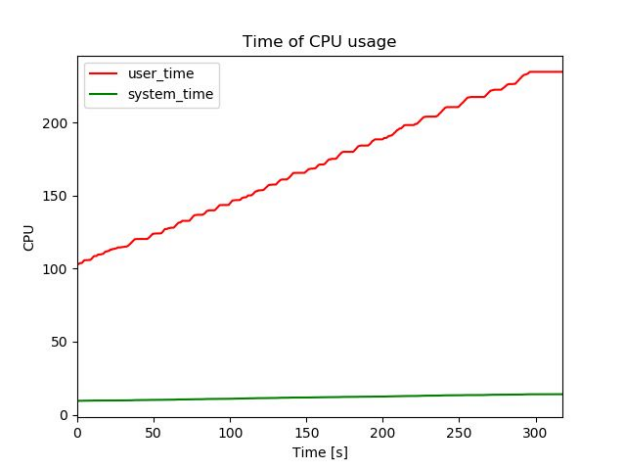
Wyniki badań - Scenariusz 3.

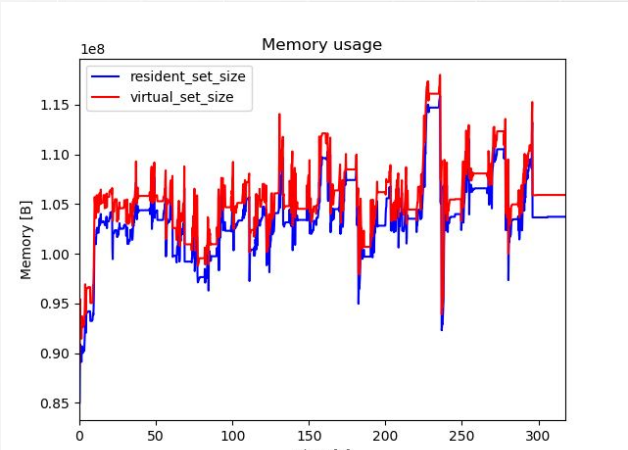




Wyniki badań - Scenariusz 3.







Wyniki badań - Scenariusz 3.

	Method	Name	# requests	# failures	Median response time	Average response time	Min response time	Max response time	Average Content Size	Requests/s
0	GET	/projects/	389.0	5.0	6800.0	8374.0	39.0	41258.0	1533.0	1.13
1	GET	/teams/	445.0	7.0	7700.0	8877.0	23.0	49230.0	75.0	1.29
2	None	Aggregated	3493.0	42.0	7200.0	8871.0	10.0	60190.0	72579.0	10.11
3	POST	/news	216.0	3.0	11000.0	13486.0	38.0	60190.0	1.0	0.63
4	POST	/projects	107.0	2.0	9400.0	11278.0	474.0	50879.0	240.0	0.31
5	POST	/teams	318.0	3.0	9100.0	11500.0	265.0	53534.0	3.0	0.92
6	PUT	/projects	410.0	4.0	7000.0	8834.0	379.0	58640.0	516074.0	1.19
7	PUT	/teams	327.0	3.0	7400.0	8775.0	179.0	51796.0	122932.0	0.95
8	PUT	/teams//status	203.0	5.0	6700.0	8686.0	39.0	52499.0	1.0	0.59
9	PUT	/user	657.0	7.0	5500.0	6723.0	10.0	48728.0	276.0	1.90
10	PUT	/users	421.0	3.0	6300.0	7913.0	46.0	45485.0	2119.0	1.22

Wnioski

- Serwer Express jest wydajnym narzędziem do zastosowań backendowych,
- Kluczowym elementem testowania wydajności są:
  - zaplanowanie testów odpowiadającym realnym wymaganiom aplikacji,
  - poprawna konfiguracja podsystemów aplikacji,
- Locust jest łatwym w obsłudze i przyjaznym użytkownikowi narzędziem.