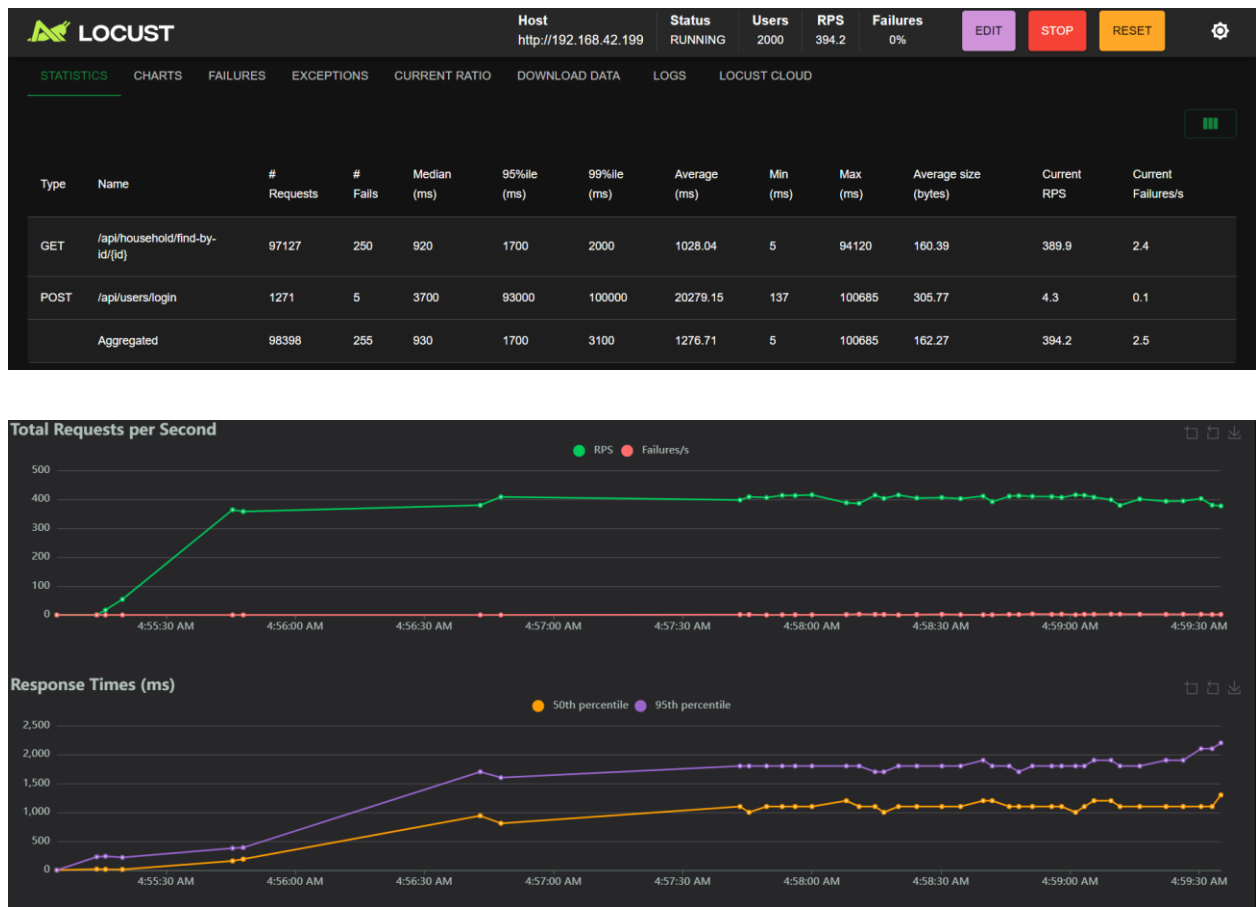


Izveštaj – Mila Milović SV22/2021

Svi testovi su pokrenuti na laptopu sa 64 GB RAM memorije i Intel(R) Core(TM) Ultra 5 125U 1.30 GHz procesorom. Svi testovi su podešeni tako da gađaju NGINX servis i simuliraju 2000 korisnika sa povećavanjem od po 10 korisnika po sekundi da bi se simuliralo realno korišćenje.

1. Dobavljanje detalja domaćinstva


Ovo je scenario gde se korisnik prvo prijavi na sistem a zatim traži podatke o konkretnom domaćinstvu. Scenario je podešen tako da se kreće od 1 korisnika i njihov broj se povećava za 10 svake sekunde. Maksimalan broj korisnika koji u isto vreme izvršavaju zahteve je 2000. Na slici ispod se može videti da funkcionalnost dobro radi pod ovakvim opterećenjem.



2. Pretraga domaćinstava

Ovo je scenario gde se korisnik prvo uloguje na sistem a zatim pretražuje sva domaćinstva na osnovu opštine i adrese. Scenario je podešen tako da kreće od 1 korisnika i njihov broj se povećava za 10 svake sekunde. Maksimalan broj korisnika koji u isto vreme izvršavaju zahteve je 2000. Na slici ispod se može videti da funkcionalnost pod ovim opterećenjem

radi dobro do nekih 1700 istovremenih korisnika, a nakon toga performanse počnu da opadaju.

 LOCUST

Host
http://192.168.42.199

Status
SPAWNING

Users
1650


RPS
309.3

Failures
3%

EDIT

STOP

RESET



STATISTICS


CHARTS

FAILURES


EXCEPTIONS

CURRENT RATIO


DOWNLOAD DATA

 LOGS

LOCUST CLOUD



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	/api/household/search/{municipality}/{address}	74741	2597	840	1700	2100	926.29	6	69359	4329.08	307.4	8.6
POST	/api/users/login	1128	14	2000	54000	67000	11976.14	187	68436	303.27	1.9	0
Aggregated		75869	2611	840	1800	5100	1090.58	6	69359	4269.22	309.3	8.6

 LOCUST

Host

http://192.168.42.199

Status

RUNNING

Users

2000

RPS

187.5


Failures

10%

EDIT

STOP

RESET



STATISTICS


CHARTS

FAILURES


EXCEPTIONS

CURRENT RATIO

DOWNLOAD DATA

 LOGS

LOCUST CLOUD

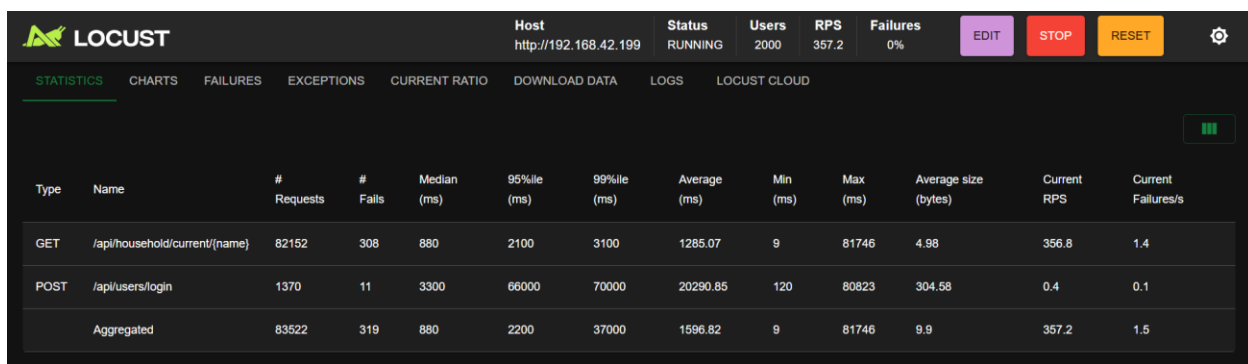
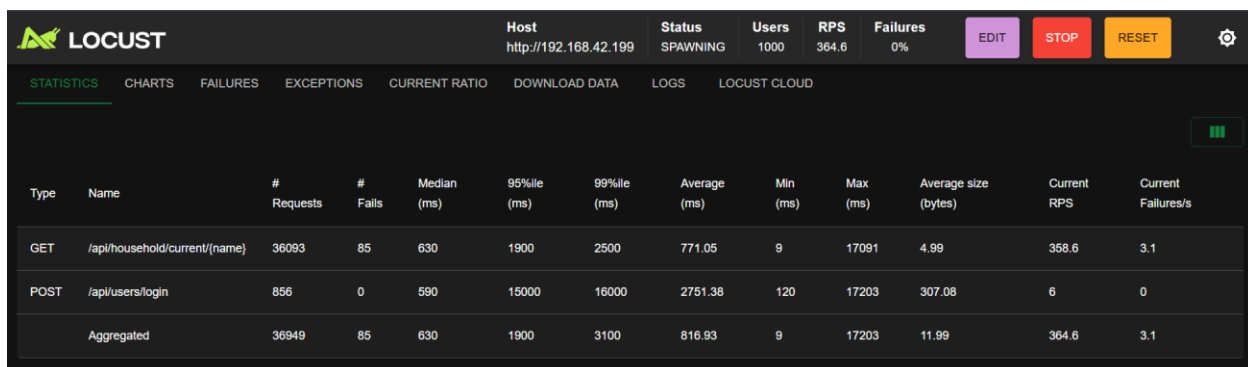


Type	Name	# Requests	# Fails	Median (ms)	95%ile (ms)	99%ile (ms)	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	Current RPS	Current Failures/s
GET	/api/household/search/{municipality}/{address}	81795	7637	840	3100	16000	1453.2	6	82164	4065.12	180.3	130.9
POST	/api/users/login	1886	523	11000	67000	81000	20874.82	187	90238	221.91	7.2	3.9
Aggregated		83681	8160	850	6400	34000	1890.93	6	90238	3978.5	187.5	134.8



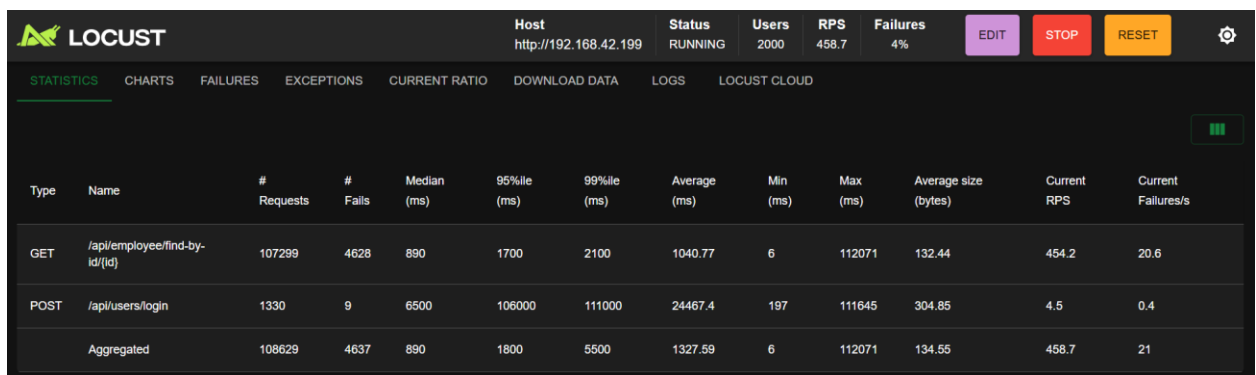
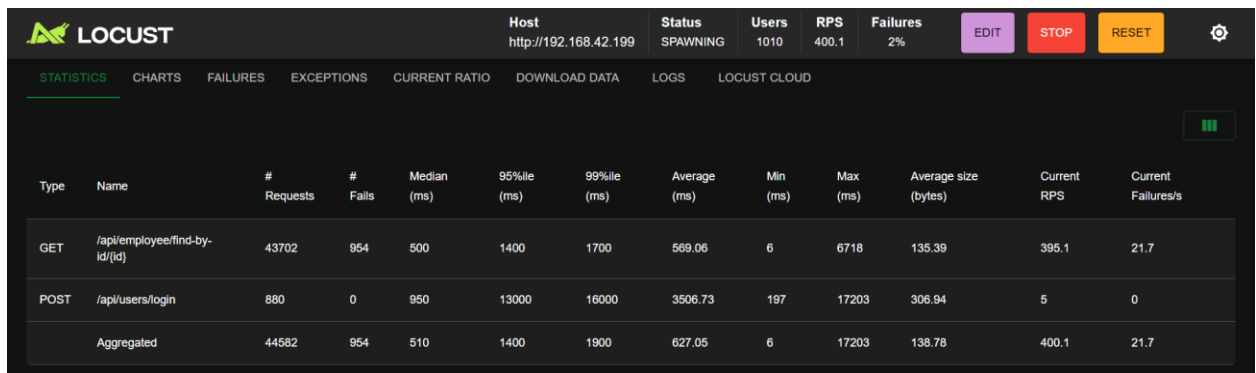
3. Trenutni status simulatora

Ovo je scenario gde se korisnik prvo uloguje na sistem a zatim proverava trenutno stanje simulatora. Scenario je podešen tako da kreće od 1 korisnika i njihov broj se povećava za 10 svake sekunde. Maksimalan broj korisnika koji u isto vreme izvršavaju zahteve je 2000. Na slici ispod se može videti da funkcionalnost pod ovim opterećenjem radi odlično.



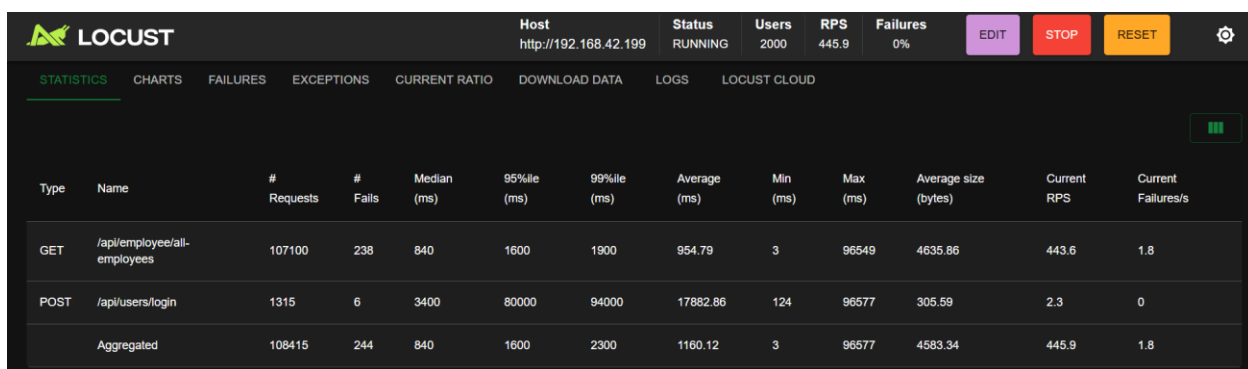
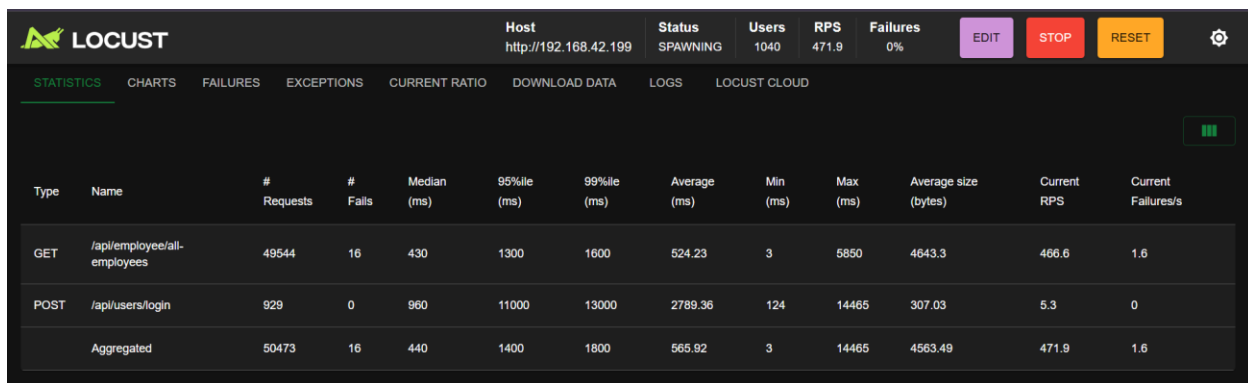
4. Dobavljanje detalja o službeniku

Ovo je scenario gde se korisnik prvo uloguje na sistem a zatim dobavi detalje o službeniku. Scenario je podešen tako da kreće od 1 korisnika i njihov broj se povećava za 10 svake sekunde. Maksimalan broj korisnika koji u isto vreme izvršavaju zahteve je 2000. Na slici ispod se može videti da funkcionalnost pod ovim opterećenjem radi dobro do nekih 1500 istovremenih korisnika, a nakon toga performanse počnu blago da opadaju.



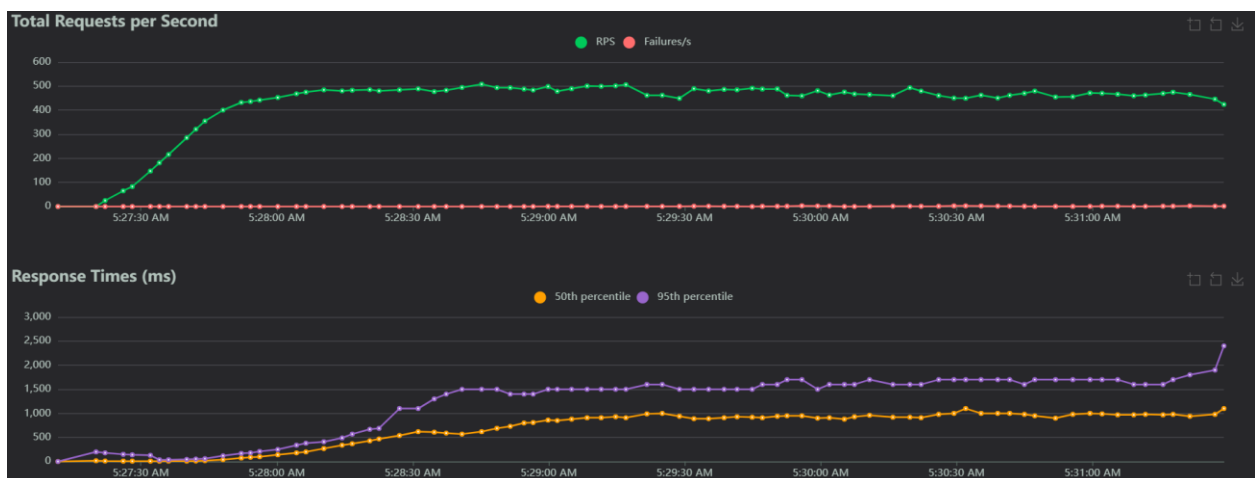
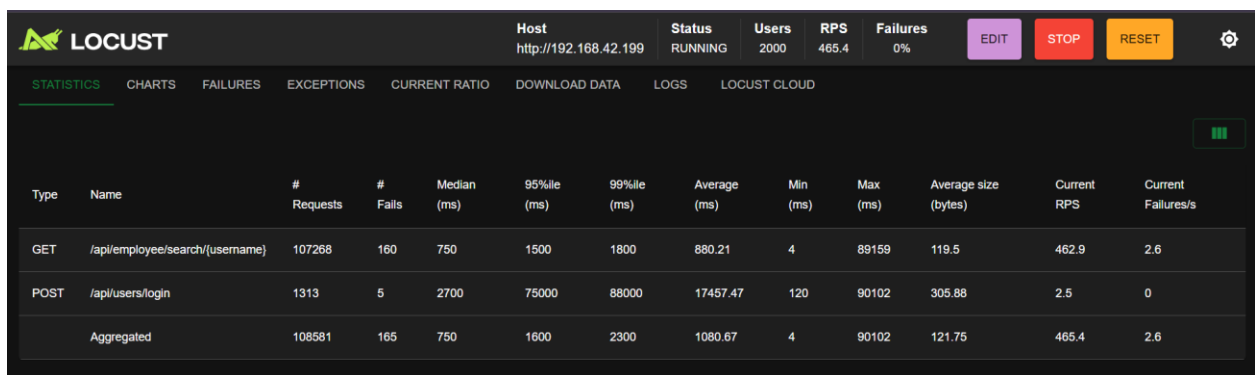
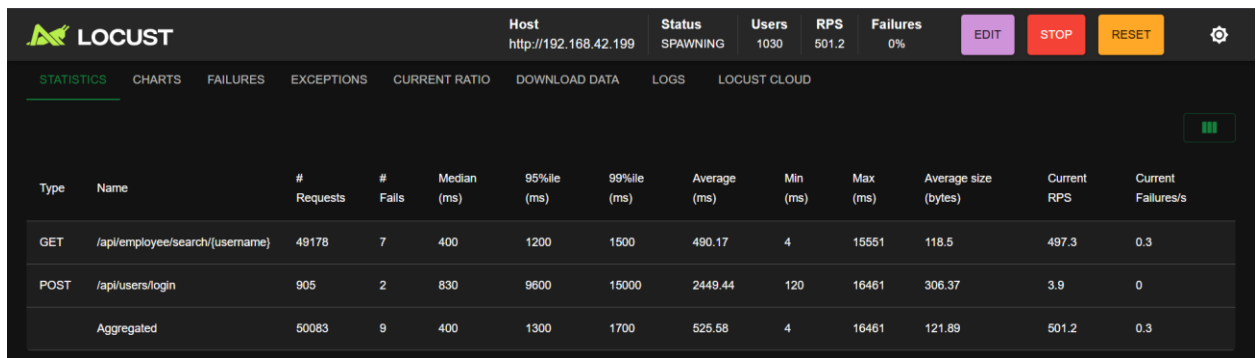
5. Dobavljanje svih službenika

Ovo je scenario gde se korisnik prvo uloguje na sistem a zatim dobavi sve službenike (određenu stranicu tog zahteva). Scenario je podešen tako da kreće od 1 korisnika i njihov broj se povećava za 10 svake sekunde. Maksimalan broj korisnika koji u isto vreme izvršavaju zahteve je 2000. Na slici ispod se može videti da funkcionalnost pod ovim opterećenjem radi odlično.



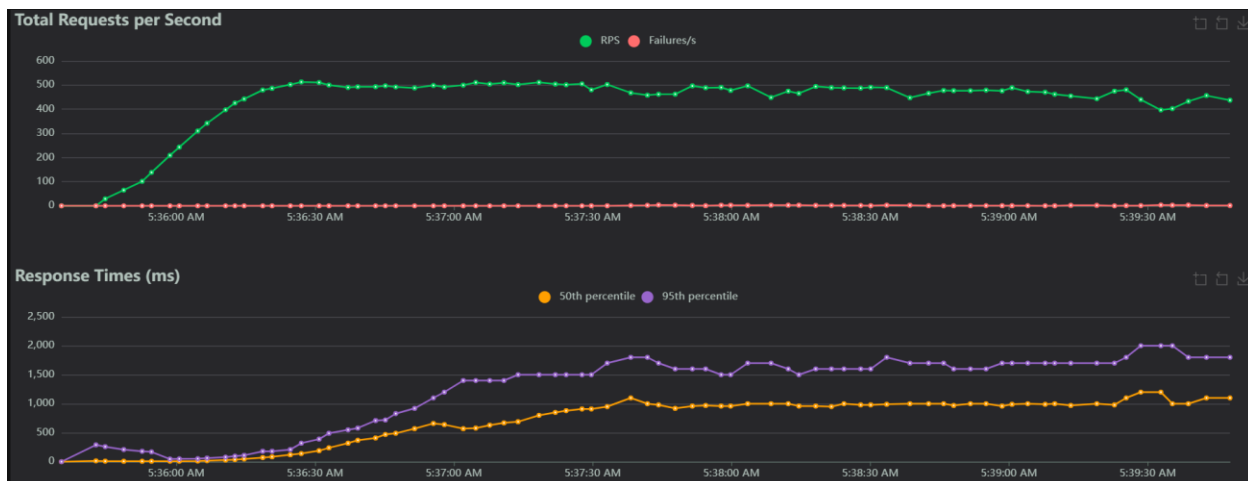
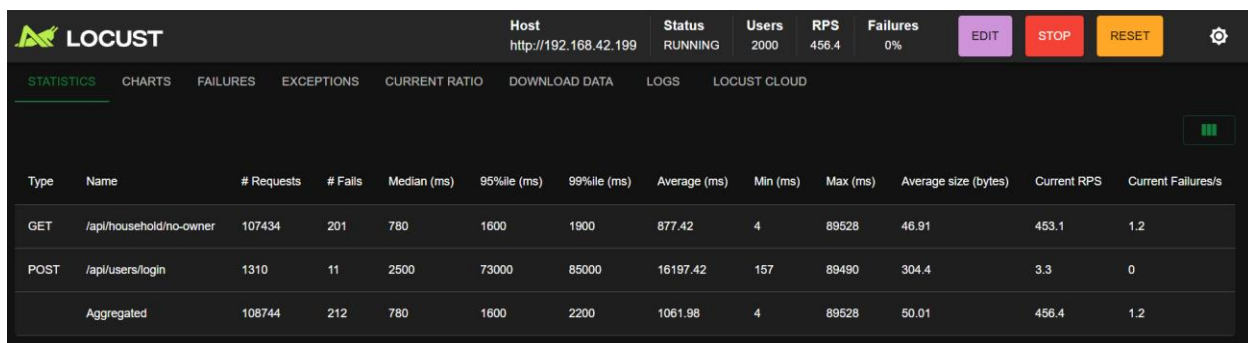
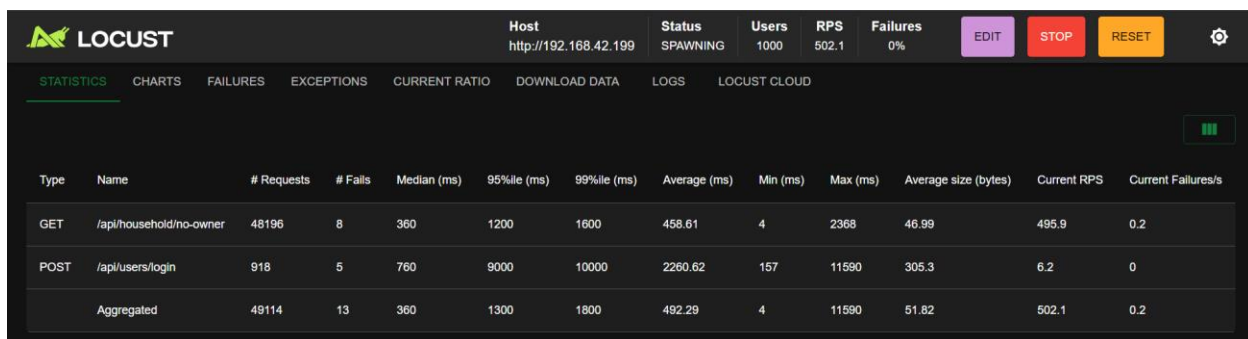
6. Pretraga službenika po mejlu

Ovo je scenario gde se korisnik prvo uloguje na sistem a zatim pretražuje sve službenike (na osnovu njihove e-mail adrese). Scenario je podešen tako da kreće od 1 korisnika i njihov broj se povećava za 10 svake sekunde. Maksimalan broj korisnika koji u isto vreme izvršavaju zahteve je 2000. Na slici ispod se može videti da funkcionalnost pod ovim opterećenjem radi odlično.



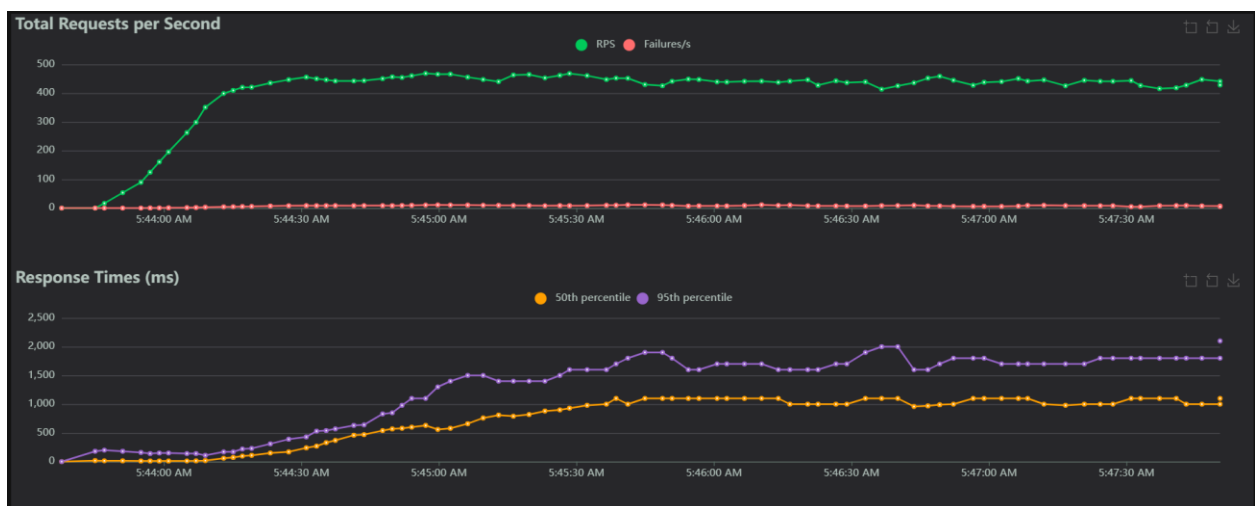
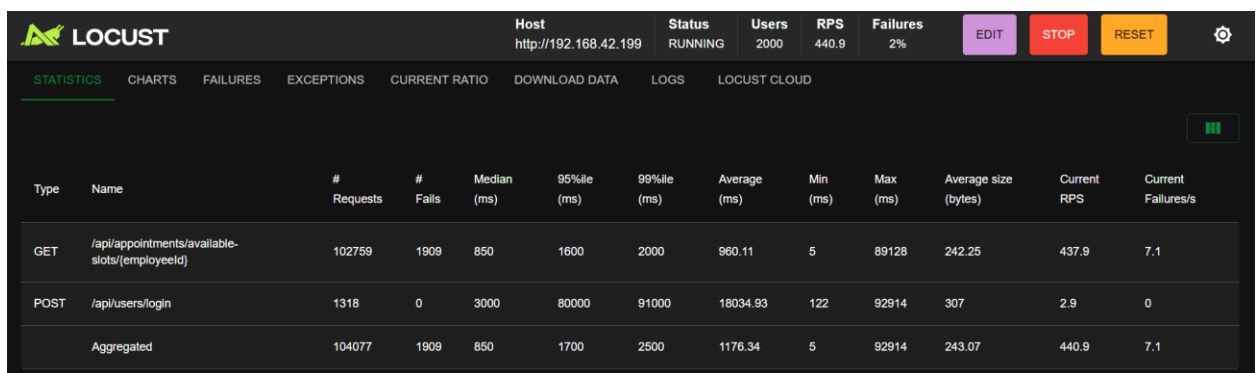
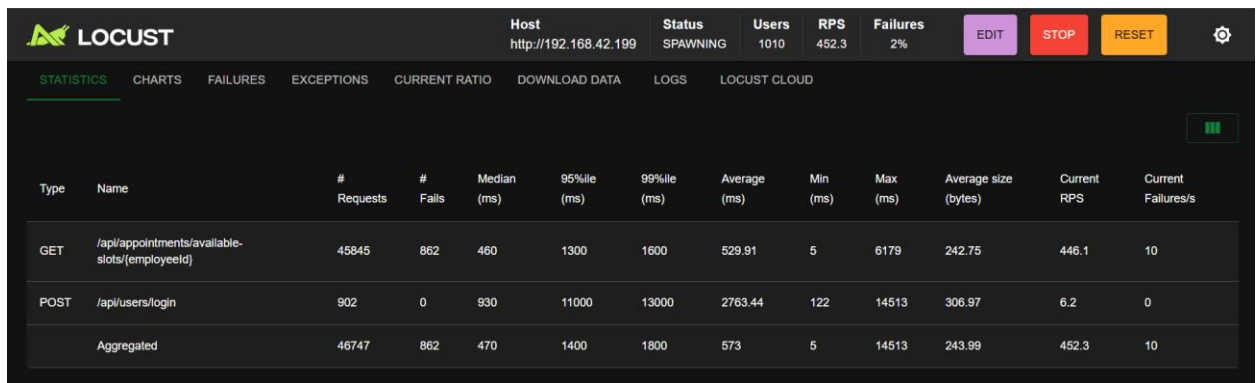
7. Pretraga domaćinstava bez vlasnika

Ovo je scenario gde se korisnik prvo uloguje na sistem a zatim pretražuje sva domaćinstva bez vlasnika. Scenario je podešen tako da kreće od 1 korisnika i njihov broj se povećava za 10 svake sekunde. Maksimalan broj korisnika koji u isto vreme izvršavaju zahteve je 2000. Na slici ispod se može videti da funkcionalnost pod ovim opterećenjem radi odlično.



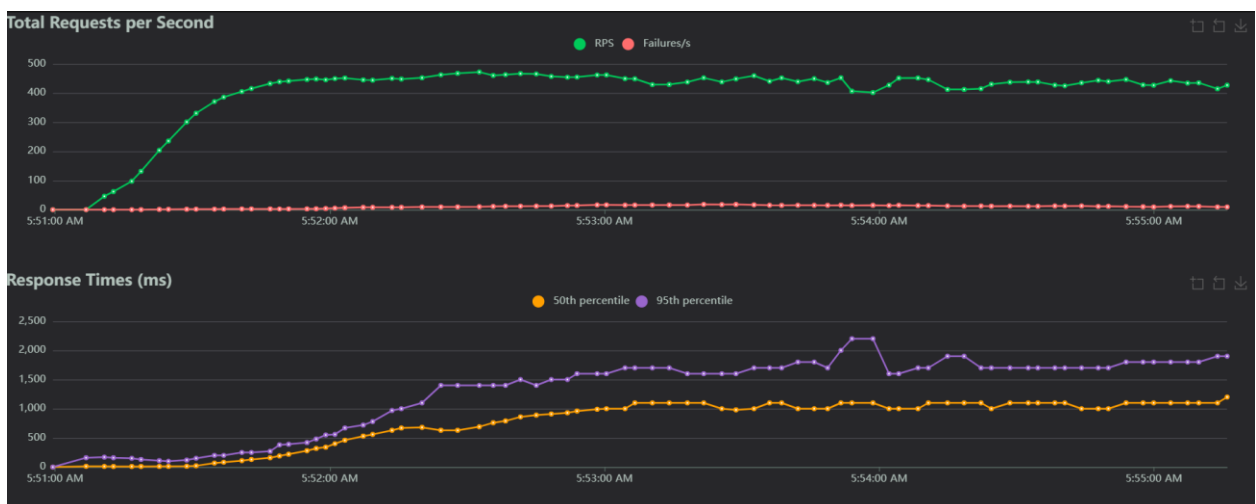
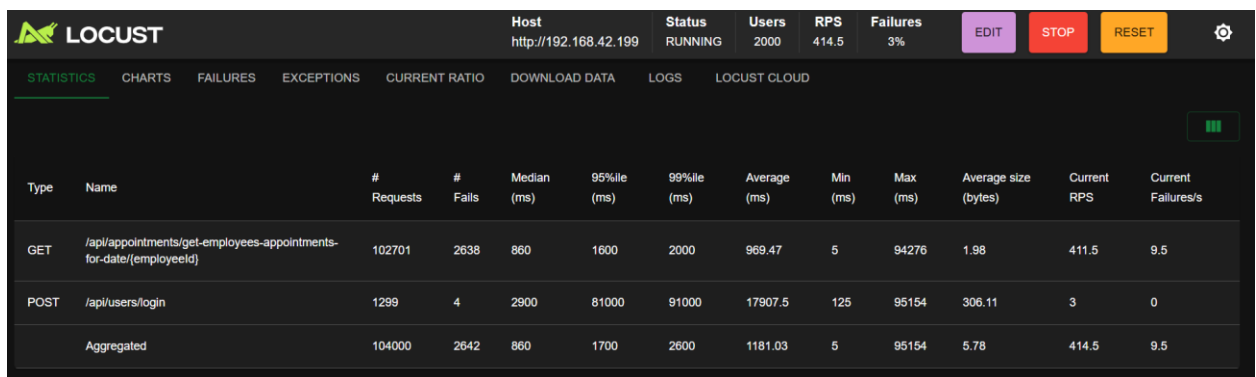
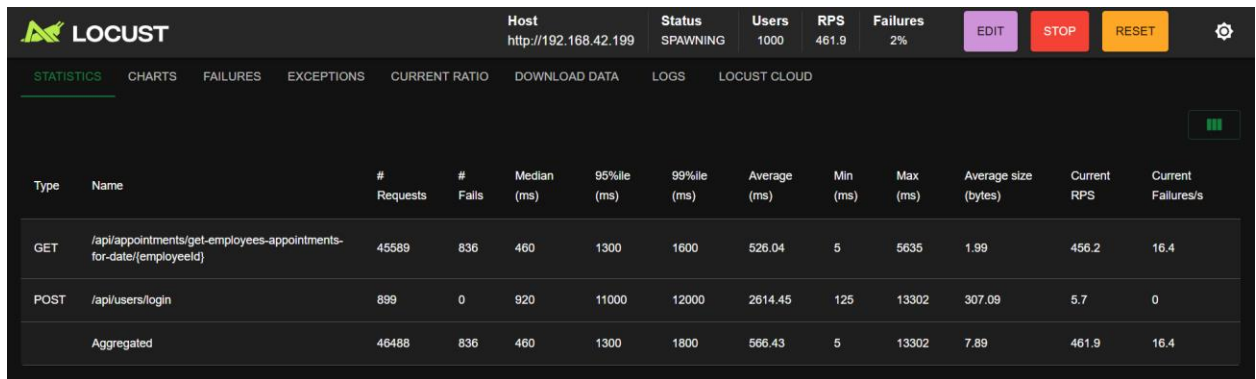
8. Dobavljanje slobodnih termina od službenika

Ovo je scenario gde se korisnik prvo uloguje na sistem a zatim dobavlja sve slobodne termine od službenika za određeni datum. Scenario je podešen tako da kreće od 1 korisnika i njihov broj se povećava za 10 svake sekunde. Maksimalan broj korisnika koji u isto vreme izvršavaju zahteve je 2000. Na slici ispod se može videti da funkcionalnost pod ovim opterećenjem radi zadovoljavajuće.



9. Dobavljanje službenikovih zauzetih termina

Ovo je scenario gde se korisnik prvo uloguje na sistem a zatim dobavlja sve zauzete termine za službenika. Scenario je podešen tako da kreće od 1 korisnika i njihov broj se povećava za 10 svake sekunde. Maksimalan broj korisnika koji u isto vreme izvršavaju zahteve je 2000. Na slici ispod se može videti da funkcionalnost pod ovim opterećenjem radi dobro.



10. Podaci iz influx-a

Ovo je scenario gde se korisnik prvo uloguje na sistem a zatim dobavlja podatke iz time series baze za domaćinstvo. Scenario je podešen tako da kreće od 1 korisnika i njihov broj se povećava za 10 svake sekunde. Maksimalan broj korisnika koji u isto vreme izvršavaju zahteve je 2000. Na slici ispod se može videti da funkcionalnost pod ovim opterećenjem radi dobro.

