

TESTIRANJE APLIKACIJE - STUDENT 1

UVODNE NAPOMENE

Testiranje je izvršeno sa računara sledećih karakteristika:

- **Procesor:** Intel i5-8265U sa 4 jezgara i brzine 1.60 GHz
- **RAM:** kapaciteta 8GB brzine 2400MHz

Aplikacija je pokrenuta na računaru sledećih karakteristika:

- **Procesor:** AMD Ryzen 5 4600H sa 6 jezgara i brzine 3.0 GHz
- **RAM:** kapaciteta 16GB brzine 3200MHz

Računari su bili međusobno povezani u lokalnoj mreži preko *ethernet*-a prosečne brzine 20 Mbps.

Za potrebe testiranja opterećenja sistema je korišćen *Locust* radni okvir. Testiranje performansi prilikom rada sa simulatorima je obavljeno ručno – programski su generisane nove konekcije ka platformi preko posebne stranice u okviru klijentske aplikacije sa posebne putanje. Podaci o odzivu sistema su takođe ručno prikupljeni beleženjem vremenskih odrednica kada su podaci poslati/primljeni.

TESTIRANJE OPTEREĆENJA SISTEMA

Objašnjenje kolona tabela koje su priložene u ovoj sekciji:

- **Method** - Tip metode akcije koja je izvršena
- **Name** - Putanja ili ime same akcije u okviru scenarija koja je izvršena
- **# Requests** - Ukupan broj zahteva koji je poslat u okviru testa po akciji
- **# Fails** - Broj greški koje su se javile prilikom izvršavanja akcije
- **Average (ms)** - Prosečno vreme izvršavanja akcije u milisekundama
- **Min (ms)** - Minimalno zabeleženo vreme izvršavanja akcije u milisekundama
- **Max (ms)** - Maksimalno zabeleženo vreme izvršavanja akcije u milisekundama
- **Average size (bytes)** - Prosečna veličina povratne vrednosti u bajtovima
- **RPS (Requests Per Second)** - Prosečan broj zahteva po sekundi koji je poslat za akciju
- **Failure/s** - Procentualno izražen broj greški prilikom izvršavanja akcije
- **Aggregated** - Agregirane vrednosti navedenih kolona

Testiranje je izvršeno u cilju određivanja krajnjih limita serverske aplikacije. Maksimalan broj korisnika koji su slali zahteve i njihovo povećanje u vremenu su odabrani tako, da za većinu scenarija maksimalno koriste resurse (CPU i memoriju) računara, na kome je aplikacija pokrenuta. Vremensko ograničenje testa je ustanovljeno i zbog uniformnosti tumačenja rezultata testa, ali i da spreči veliko i štetno pregrevanje računara.

Scenarija koja su priložena prate tokove korisničke aplikacije. To znači, da akcije koje se izvršavaju u okviru scenarija su skoro uvek sekvencijalne, tj. izvršavanje jedne uzrokuje izvršavanje druge, slično kao što bi to bilo prilikom korišćenja korisničke aplikacije. Situacije, gde se neke akcije ne izvršavaju sekvencijalno su istaknute u kratkom opisu scenarija.

Svaki scenario ima 2 testa, koja se razlikuju po broju vrednosti koje se nalaze u bazi podataka u trenutku testiranja i po tome da li su podaci keširani, ili nisu. Keširanje se odnosi **SAMO** na podatke iz relacione baze podataka - obratiti pažnju na to u scenariju 2, 9 i 10, gde se dobavljaju podaci iz *time-series* baze.

U okviru informacija o scenariju se nalazi i kratka analiza dobijenih vrednosti. Ova analiza služi da se istaknu uzroci određenih problema koji su bili uočeni, ali i priloži rešenje koje je implementirano ili odluka koja je doneta. Imajući u vidu da je zadato opterećenje sistema veliko, neki uočeni problemi su hardverske prirode - najčešće zbog nedostatka radne memorije.

Scenario 1: Registracija novog naloga -> aktivacija naloga -> logovanje na novokreirani nalog

Maksimalan broj korisnika koji su istovremeno slali zahteve: 300

Povećanje broja korisnika po sekundi: 10

Trajanje testa: 2 minute

Test 1: Broj postojećih korisnika u bazi: 100 000 (bez keširanja)

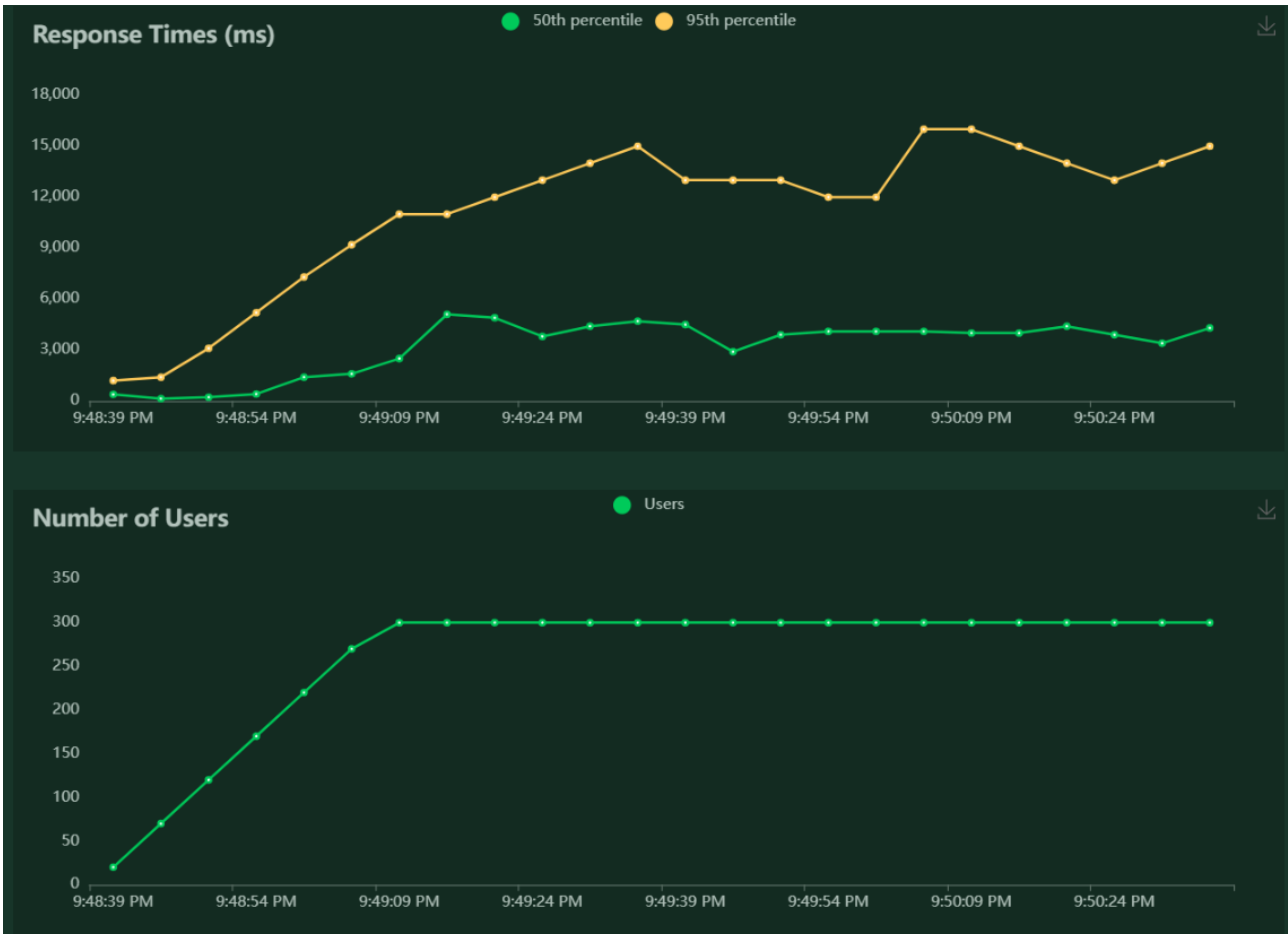
U tabeli 1 se nalaze osnovni podaci o izvršenom testu za Test 1.

Tabela 1

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
POST	/api/user/login	1672	0	2497	94	11993	282	13.9	0.0
POST	/api/user/register	1672	0	9520	486	24303	6	13.9	0.0
GET	Account Activation	1672	0	2737	13	12563	51	13.9	0.0
Aggregated		5016	0	4918	13	24303	113	41.8	0.0

Na grafiku 1 se nalazi prikaz prosečnog vremena potrebnog za izvršenje akcija u milisekundama i povećanje broja korisnika u toku vremena za Test 1.

Grafik 1



Test 2: Broj postojećih korisnika u bazi: 1 000 000 (sa keširanjem)

U tabeli 2 se nalaze osnovni podaci o izvršenom testu za Test 2.

Tabela 2

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
POST	/api/user/login	1047	0	5288	189	24088	282	8.7	0.0
POST	/api/user/register	1047	0	13015	1147	29444	7	8.7	0.0
GET	Account Activation	1047	0	5291	89	22798	51	8.7	0.0
Aggregated		3141	0	7865	89	29444	113	26.2	0.0

Na grafiku 2 se nalazi prikaz prosečnog vremena potrebnog za izvršenje akcija u milisekundama i povećanje broja korisnika u toku vremena za Test 2.

Grafik 2



Uočeno je veće čekanje prilikom registracije zahteva. Razlog tome je i komunikacija sa eksternim serverima - *Send Grid* mejl serverom i *min.io* fajl serverom za smeštanje slika. Slike se kompresuju pre slanja na fajl server, što takođe traje određeno vreme. Izbacivanje kompresije ubrzava proces, ali redukuje memorijsku efikasnost fajl servera. S tim u vezi, odlučeno je da se kompresija zadrži kao korak registracije.

Scenario 2: Dobavljanje istorijskih podataka o temperaturi i vlažnosti vazduha za jedan termometar. Vremenski interval se bira nasumično - postoji 50% šansa da se zatraži najveći

moćući period (mesec dana) i 50% šansa za manji (prethodna 3 sata, 6 sati, 12 sati, 24 sata, ili nedelja)

Maksimalan broj korisnika koji su istovremeno slali zahteve: 200

Povećanje broja korisnika po sekundi: 10

Trajanje testa: 2 minute

Test 1: Broj postojećih merenja u bazi: ~ 500 000 za uređaj (bez keširanja)

U tabeli 3 se nalaze osnovni podaci o izvršenom testu za Test 1.

Tabela 3

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
GET	Temperature	178	0	8392	33	49456	912972	1.9	0.0
GET	Humidity	191	0	7252	84	31711	930098	2.0	0.0
	Aggregated	369	0	7802	33	49456	921837	3.8	0.0

Na grafiku 3 se nalazi prikaz prosečnog vremena potrebnog za izvršenje akcija u milisekundama i povećanje broja korisnika u toku vremena za Test 1.

Grafik 3



Test 2: Broj postojećih merenja u bazi: ~ 1 500 000 za uređaj (sa keširanjem)

Test 2 je bio izvršen sa maksimalnih 300 korisnika koji su istovremeno slali zahteve

U tabeli 4 se nalaze osnovni podaci o izvršenom testu za Test 2.

Tabela 4

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
GET	Humidity	165	0	13389	1109	68396	128842	1.6	0.0
GET	Temperature	151	0	13394	871	46293	13500	1.4	0.0
	Aggregated	316	0	13392	871	68396	73726	3.0	0.0

Na grafiku 4 se nalazi prikaz prosečnog vremena potrebnog za izvršenje akcija u milisekundama i povećanje broja korisnika u toku vremena za Test 2.

Grafik 4



Uočeno je da je keširanje omogućilo povećanje maksimalnog broja korisnika. Prilikom testiranja testnog slučaja 1 je platforma sa opterećenjem oko 220 korisnika bila blokirana, tj. ulazila je u stanje iz kog se nije mogla oporaviti. To je zato što se u okviru dobavljanja merenja takođe proverava da li prosleđeni uređaj (sa njegovim ID-jem) postoji u bazi podataka, što je dodatno opteretilo sistem da se obrati i relacionoj bazi podataka i tako iskoristi maksimalan broj mogućih konekcija ka njoj. Keširanje je nakon prvog zahteva

izbacilo potrebu za time. Uočen je veliki problem performantnosti upita koji se šalje ka *time-series* bazi podataka koja skladišti merenja, za koji još uvek nije pronađeno optimalno rešenje. Implementirano rešenje podrazumeva paginaciju prilikom dobavljanja podataka, kao i slanje merenja korisniku u vidu *stream*-a, sa ciljem da se sva merenja šalju u manjim paketima preko mreže.

Scenario 3: Dobavljanje korisnika kojima su dodeljena prava pristupa za uređaj ili nekretninu. Ovde dobavljanje **nije** sekvencijalno, tj. nasumično se bira da li će se dobiti za uređaj ili nekretninu. Biranje koji uređaj/nekretnina je u pitanju je nasumično.

Maksimalan broj korisnika koji su istovremeno slali zahteve: 300

Povećanje broja korisnika po sekundi: 10

Trajanje testa: 2 minute

Test 1: Broj postojećih uređaja u bazi: 90 000 (bez keširanja)

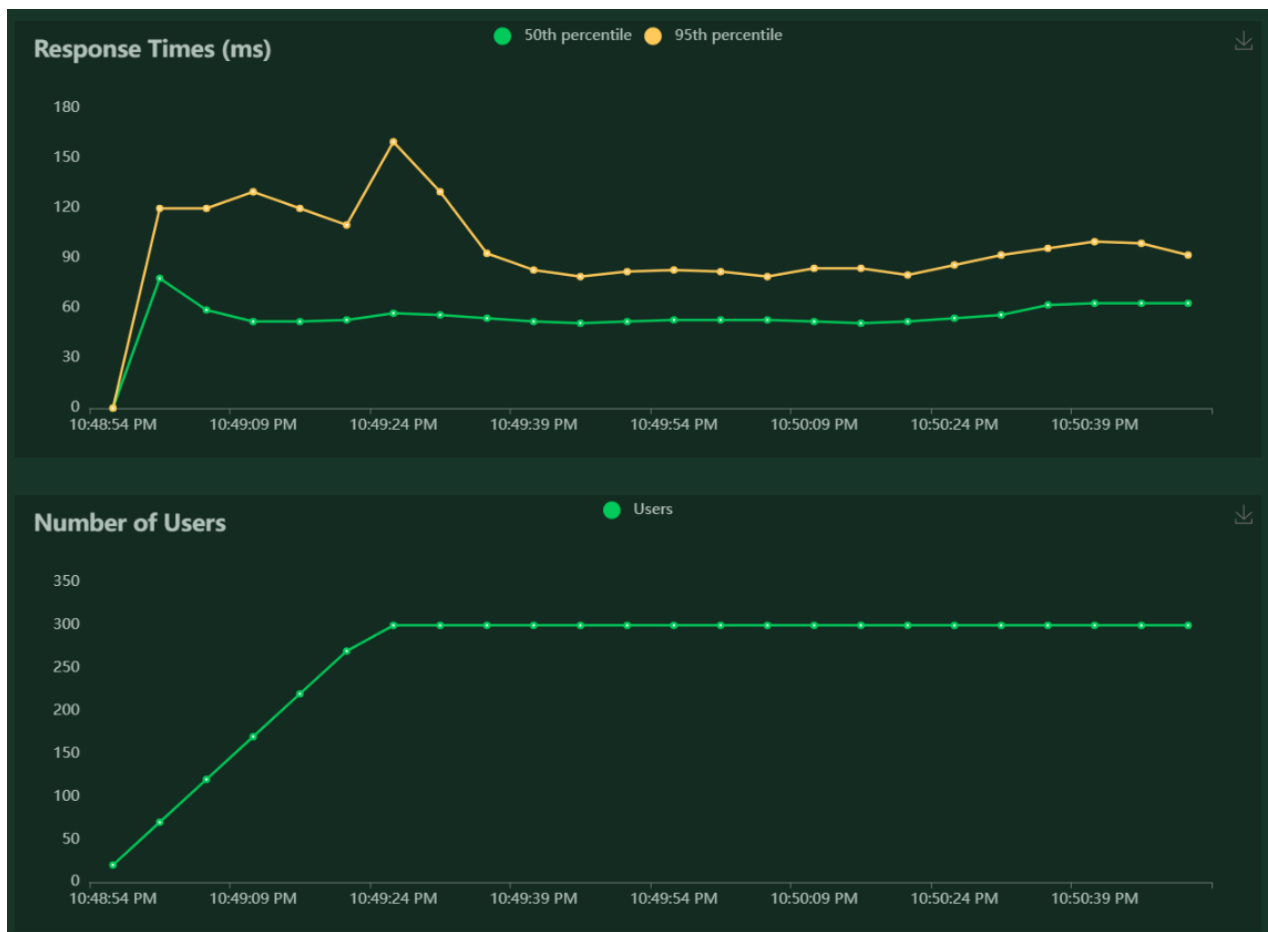
Broj postojećih permisija u bazi: 10 000 (bez keširanja)

U tabeli 5 se nalaze osnovni podaci o izvršenom testu za Test 1.

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
GET	Shared Permissions For Device	5329	0	49	30	243	9	44.3	0.0
GET	Shared Permissions For Property	5162	0	69	41	271	2	42.9	0.0
Aggregated		10491	0	59	30	271	5	87.3	0.0

Tabela 5

Na grafiku 5 se nalazi prikaz prosečnog vremena potrebnog za izvršenje akcija u milisekundama i povećanje broja korisnika u toku vremena za Test 1.



Test 2: Broj postojećih uređaja u bazi: 900 000 (sa keširanjem)

Broj postojećih permisija u bazi: 100 000 (sa keširanjem)

U tabeli 6 se nalaze osnovni podaci o izvršenom testu za Test 2.

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
GET	Shared Permissions For Device	1149	0	9294	544	43634	9	9.6	0.0
GET	Shared Permissions For Property	1137	0	10947	800	42107	2	9.5	0.0
	Aggregated	2286	0	10116	544	43634	5	19.0	0.0

Na grafiku 6 se nalazi prikaz prosečnog vremena potrebnog za izvršenje akcija u milisekundama i povećanje broja korisnika u toku vremena za **Test 2**.



Grafik 6

Uočeno znatno povećanje vremena čekanja između testnog slučaja 1 i 2. Parcijalno rešenje problema je implementirano u vidu uvođenja indeksa u tabeli permisija i po korisniku, i po ID-u uređaja.

Scenario 4: Logovanje korisnika i dobavljanje njegovih informacija. Biranje korisnika se vrši nasumično.

Maksimalan broj korisnika koji su istovremeno slali zahteve: 300

Povećanje broja korisnika po sekundi: 10

Trajanje testa: 2 minute

Test 1: Broj postojećih korisnika u bazi: 100 000 (bez keširanja)

U tabeli 7 se nalaze osnovni podaci o izvršenom testu za Test 1.

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
GET	/api/user/info	9651	0	66	16	801	203	80.4	0.0
POST	/api/user/login	9651	0	240	79	1274	289	80.4	0.0
	Aggregated	19302	0	153	16	1274	246	160.9	0.0

Tabela 3

Na grafiku 7 se nalazi prikaz prosečnog vremena potrebnog za izvršenje akcija u milisekundama i povećanje broja korisnika u toku vremena za Test 1 .



Grafik 7

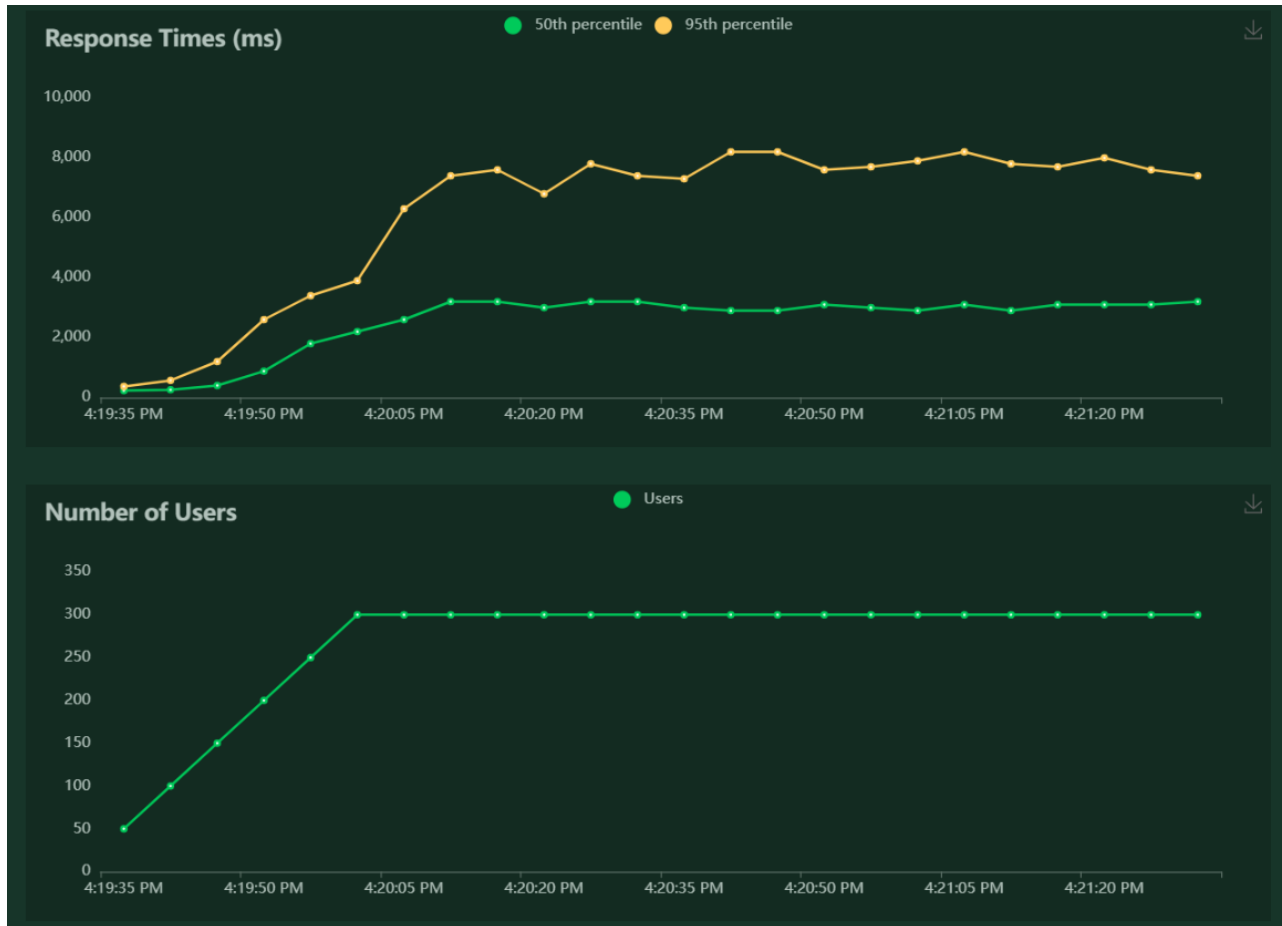
Test 2: Broj postojećih korisnika u bazi: 1 000 000 (sa keširanjem)

U tabeli 8 se nalaze osnovni podaci o izvršenom testu za Test 2.

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
GET	/api/user/info	3373	0	3657	121	21131	205	28.1	0.0
POST	/api/user/login	3373	0	2530	141	26485	290	28.1	0.0
	Aggregated	6746	0	3093	121	26485	248	56.1	0.0

Tabela 8

Na grafiku 8 se nalazi prikaz prosečnog vremena potrebnog za izvršenje akcija u milisekundama i povećanje broja korisnika u toku vremena za Test 2 .



Grafik 8

Uočeno je proporcionalno povećanje čekanja između dva testna slučaja.

Scenario 5: Izmena korisničke lozinke -> logovanje sa novom lozinkom

Maksimalan broj korisnika koji su istovremeno slali zahteve: 300

Povećanje broja korisnika po sekundi: 10

Trajanje testa: 2 minute

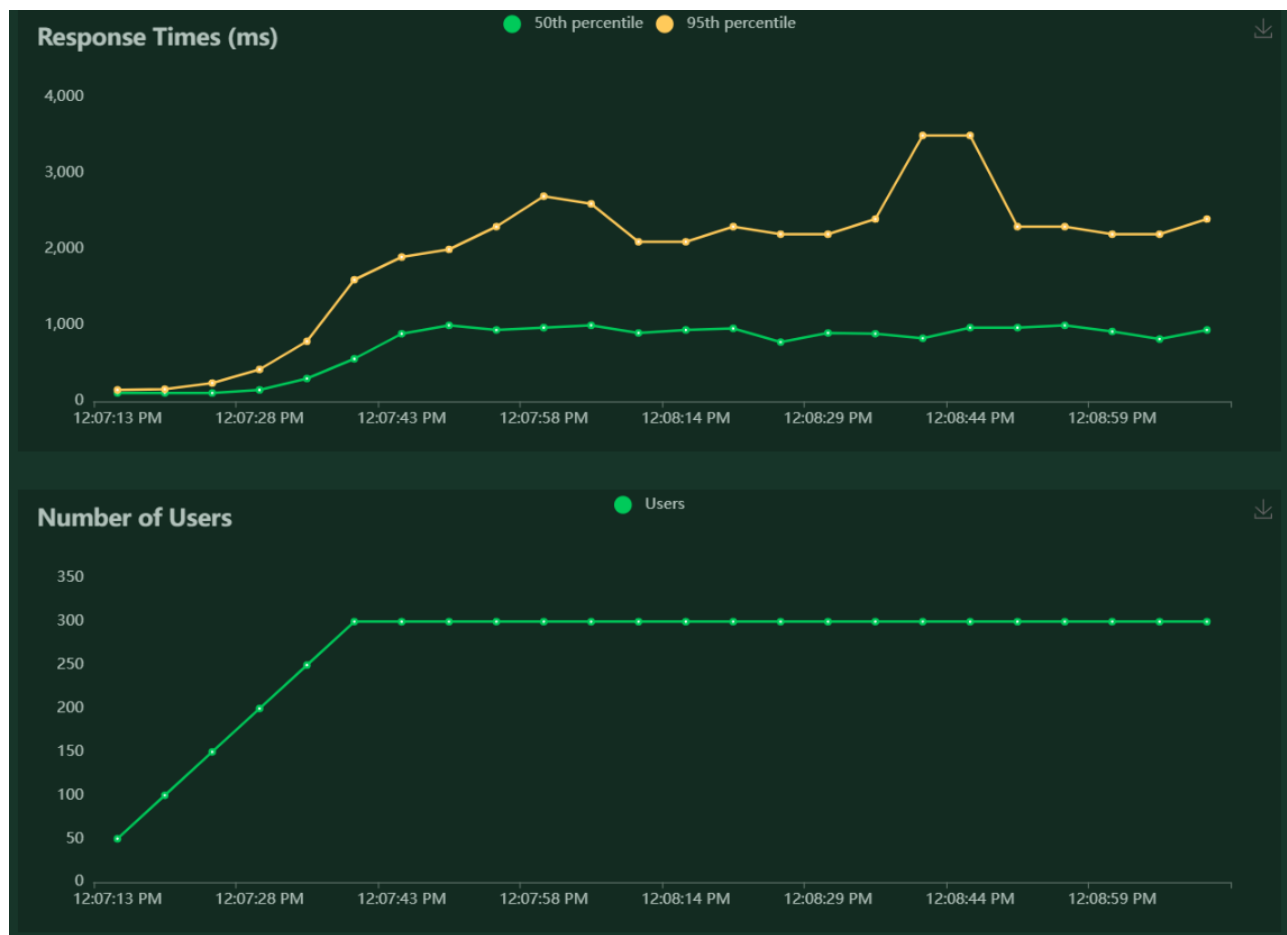
Test 1: Broj postojećih korisnika u bazi: 100 000 (bez keširanja)

U tabeli 9 se nalaze osnovni podaci o izvršenom testu za Test 1.

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
POST	/api/user/login	6506	0	1015	77	7458	289	54.2	0.0
POST	/api/user/passwordReset	6506	0	865	79	4899	28	54.2	0.0
Aggregated		13012	0	940	77	7458	158	108.3	0.0

Tabela 9

Na grafiku 9 se nalazi prikaz prosečnog vremena potrebnog za izvršenje akcija u milisekundama i povećanje broja korisnika u toku vremena za Test 1.



Grafik 9

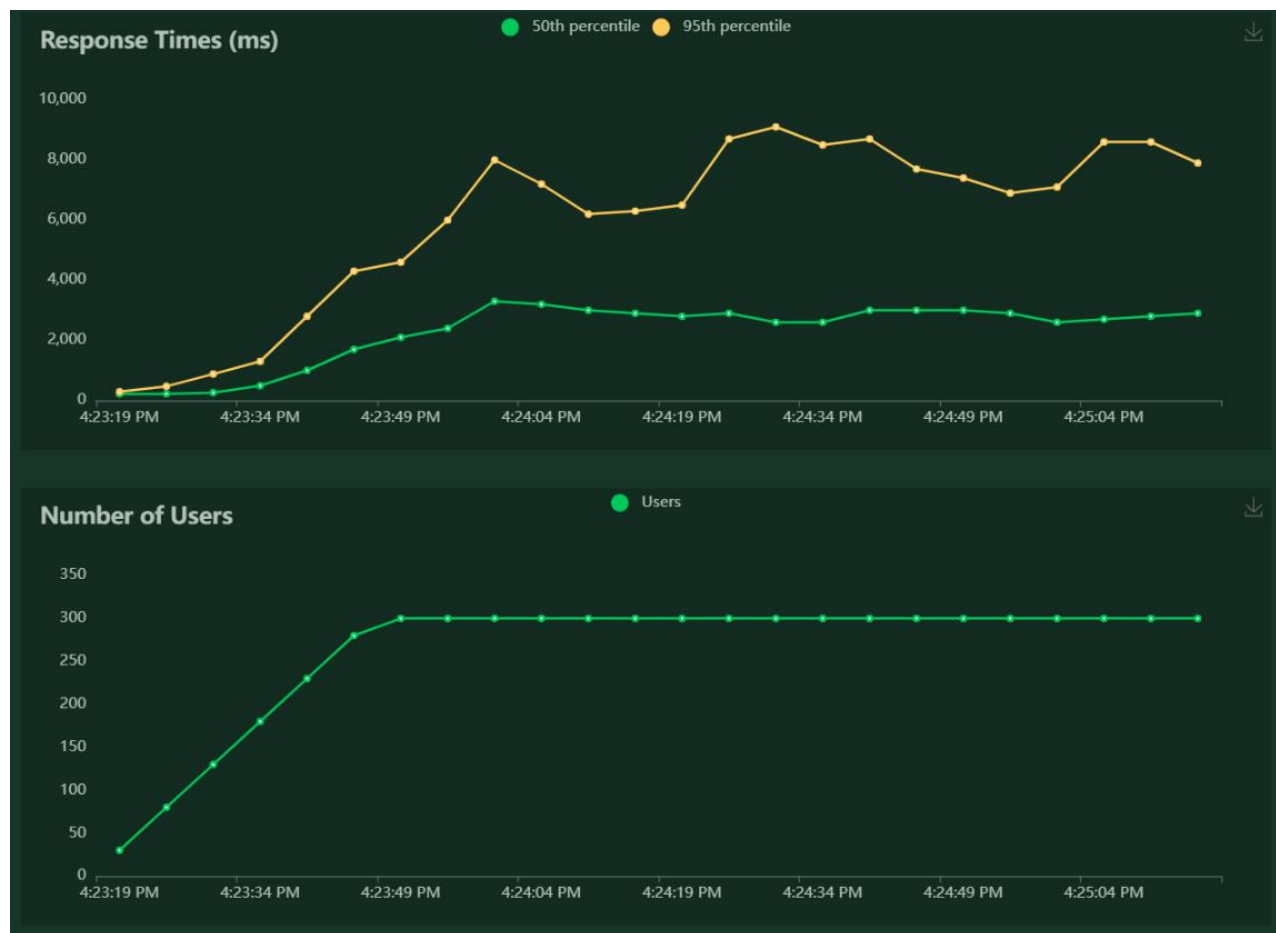
Test 2: Broj postojećih korisnika u bazi: 1 000 000 (sa keširanjem)

U tabeli 10 se nalaze osnovni podaci o izvršenom testu za Test 2.

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
POST	/api/user/login	3495	0	2848	130	22634	290	29.2	0.0
POST	/api/user/passwordReset	3495	0	2965	139	23292	28	29.2	0.0
Aggregated		6990	0	2907	130	23292	159	58.4	0.0

Tabela 10

Na grafiku 10 se nalazi prikaz prosečnog vremena potrebnog za izvršenje akcija u milisekundama i povećanje broja korisnika u toku vremena za Test 2 .



Grafik 10

Uočeno je proporcionalno povećanje čekanja između dva testna slučaja.

Scenario 6: Logovanje -> dobavljanje korisničkih uređaja -> dobavljanje uređaja za koje korisnik ima permisije. Biranje korisnika za kojeg se scenario vrši je nasumično.

Maksimalan broj korisnika koji su istovremeno slali zahteve: 300

Povećanje broja korisnika po sekundi: 10

Trajanje testa: 2 minute

Test 1: Broj postojećih korisnika u bazi: 100 000 (bez keširanja)

Broj postojećih permisija i nekretnina u bazi: 10 000 (bez keširanja)

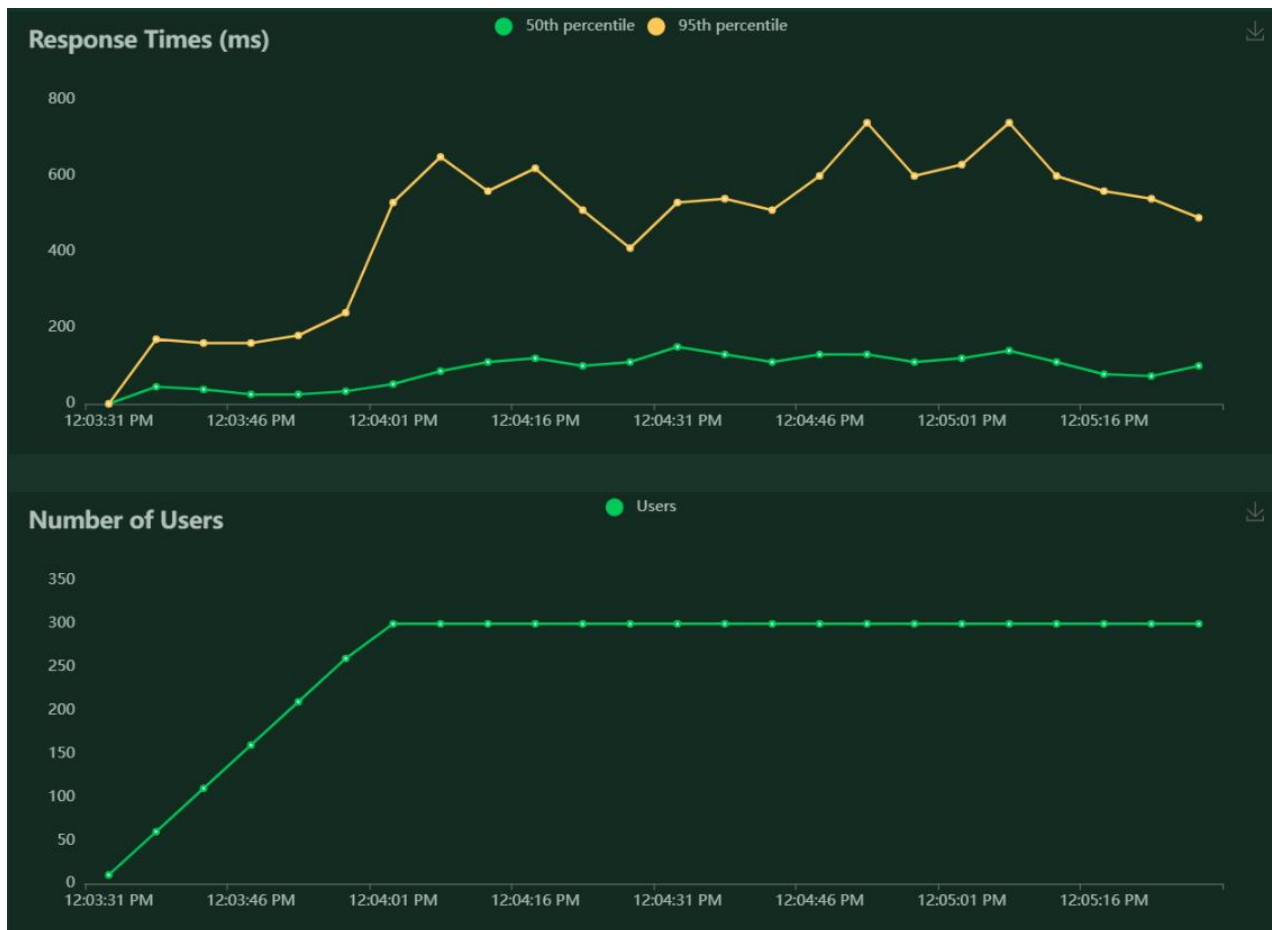
Broj postojećih uređaja u bazi: 90 000 (bez keširanja)

U tabeli 11 se nalaze osnovni podaci o izvršenom testu za Test 1.

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
GET	/api/devices/ownerAll	9157	0	96	10	5187	170	76.1	0.0
GET	/api/devices/shared	9157	0	87	10	2798	22	76.1	0.0
POST	/api/user/login	9157	0	307	77	2334	289	76.1	0.0
	Aggregated	27471	0	163	10	5187	161	228.4	0.0

Tabela 11

Na grafiku 11 se nalazi prikaz prosečnog vremena potrebnog za izvršenje akcija u milisekundama i povećanje broja korisnika u toku vremena za Test 1.



Grafik 11

Test 2: Broj postojećih korisnika u bazi: 1 000 000 (sa keširanjem)

Broj postojećih permisija i nekretnina u bazi: 100 000 (sa keširanjem)

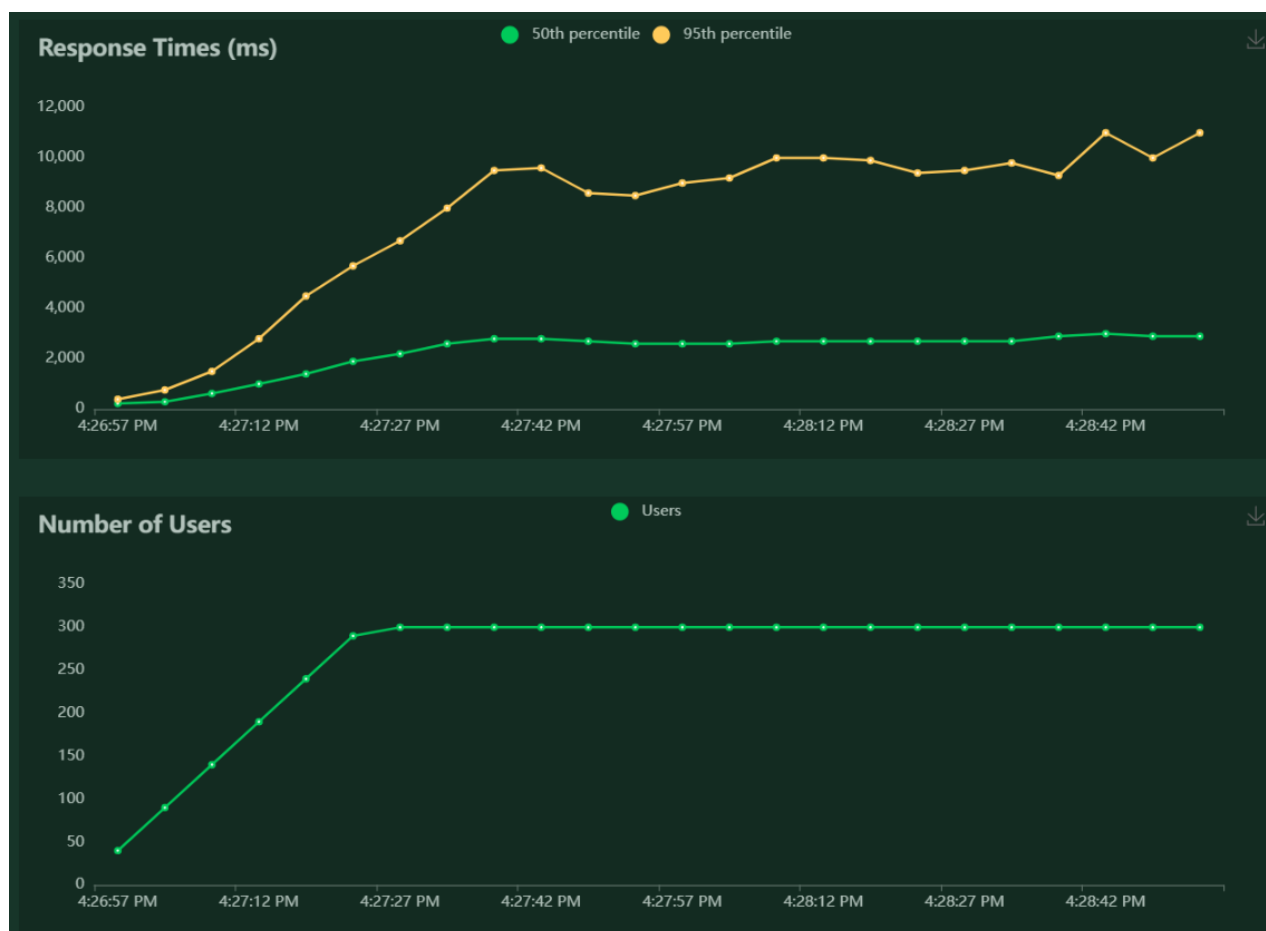
Broj postojećih uređaja u bazi: 900 000 (sa keširanjem)

U tabeli 12 se nalaze osnovni podaci o izvršenom testu za Test 2.

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
GET	/api/devices/ownerAll	2350	0	3562	113	36848	178	19.6	0.0
GET	/api/devices/shared	2350	0	3438	67	25552	22	19.6	0.0
POST	/api/user/login	2350	0	2763	172	20035	290	19.6	0.0
Aggregated		7050	0	3254	67	36848	163	58.9	0.0

Tabela 12

Na grafiku 12 se nalazi prikaz prosečnog vremena potrebnog za izvršenje akcija u milisekundama i povećanje broja korisnika u toku vremena za Test 2.



Grafik 12

Uočeno je proporcionalno povećanje čekanja između dva testna slučaja.

Scenario 7: Dobavljanje korisnika kojima su dodeljene permisije za neki uređaj -> izmena dobavljenih permisija. U okviru ovog scenarija se takođe vrši pretraga korisnika po ključu jednom ili dva puta i **nije** sekvencijalno sa dobavljanjem i izmenom.

Maksimalan broj korisnika koji su istovremeno slali zahteve: 300

Povećanje broja korisnika po sekundi: 10

Trajanje testa: 2 minute

Test 1: Broj postojećih korisnika u bazi: 100 000 (bez keširanja)

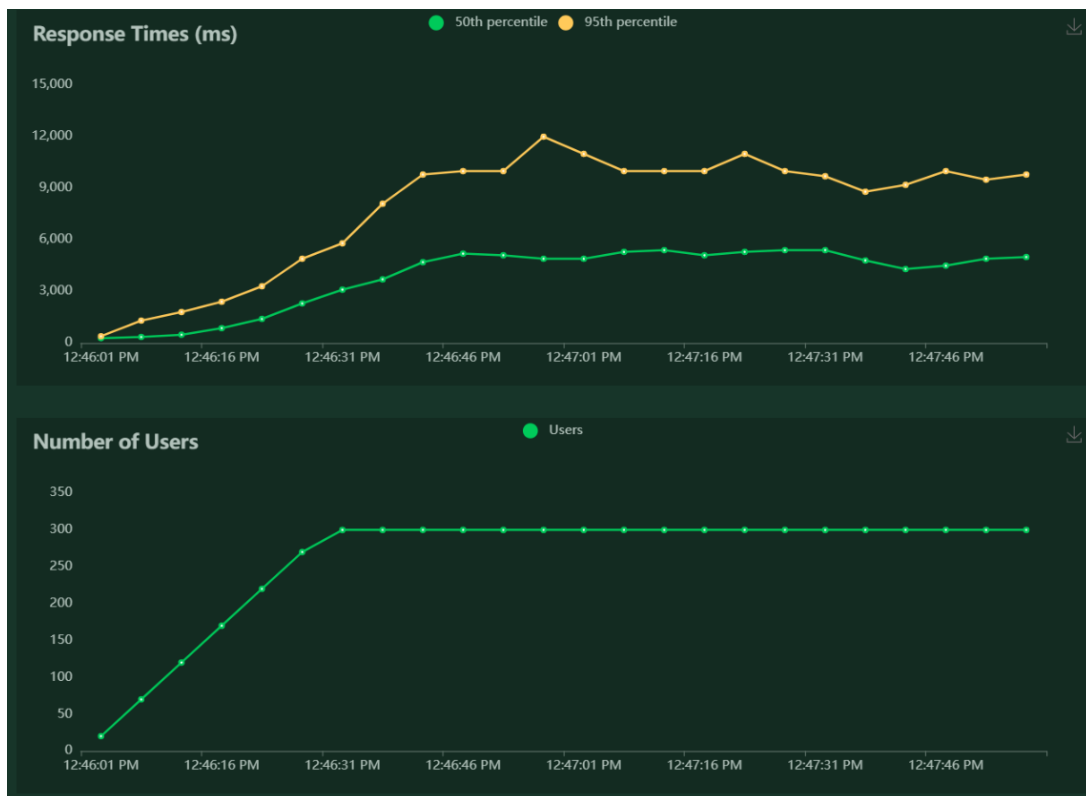
Broj postojećih permisija u bazi: 10 000 (bez keširanja)

U tabeli 13 se nalaze osnovni podaci o izvršenom testu za **Test 1**.

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
PUT	Permission Editing	1068	0	3766	59	51770	60	8.9	0.0
GET	Shared Permissions For Device	1068	0	3990	47	51762	12	8.9	0.0
GET	User Search	3443	0	5259	213	25734	74	28.7	0.0
Aggregated		5579	0	4730	47	51770	59	46.5	0.0

Tabela 13

Na grafiku 13 se nalazi prikaz prosečnog vremena potrebnog za izvršenje akcija u milisekundama i povećanje broja korisnika u toku vremena za **Test 1**.



Grafik 13

Test 2: Broj postojećih korisnika u bazi: 1 000 000 (sa keširanjem)

Broj postojećih permisija u bazi: 100 000 (sa keširanjem)

U tabeli 14 se nalaze osnovni podaci o izvršenom testu za Test 2.

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
PUT	Permission Editing	485	0	11124	1793	19855	60	4.0	0.0
GET	Shared Permissions For Device	557	0	6237	734	13331	9	4.6	0.0
GET	User Search	728	0	18215	852	34663	151	6.1	0.0
Aggregated		1770	0	12503	734	34663	81	14.7	0.0

Tabela 14

Na grafiku 14 se nalazi prikaz prosečnog vremena potrebnog za izvršenje akcija u milisekundama i povećanje broja korisnika u toku vremena za Test 2.



Grafik 14

Uočeno je znatno manje povećanje prosečnog vremena čekanja za dobavljanje korisnika koji imaju permisije između dva testna slučaja. Ovo je posledica većeg procenta pogodaka podataka, koji su keširani. Rezultati iz testa 2 su dobijeni optimizacijom upita ka bazi koji se tiču pretrage korisnika po ključu. Naime, u originalnoj formi, ovaj upit je poredio postojanje ključa u imenu, prezimenu ili korisničkom imenu korisnika, što je znatno otežalo pretragu. Optimizacija je implementirana u vidu dodavanja posebnog polja u tabelu korisnika koje predstavlja konkatenirane vrednosti ova tri polja, što originalni upit po 3 polja svodi na upit nad jednim poljem. U tabeli 15 su prikazane početne informacije o testnom slučaju 2 bez ove optimizacije, dok je u tabeli 14 prikazano stanje nakon nje.

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
PUT	Permission Editing	154	0	38008	4983	98231	60	1.3	0.0
GET	Shared Permissions For Device	265	0	21258	789	64096	9	2.2	0.0
GET	User Search	228	0	44340	8248	78778	76	1.9	0.0
Aggregated		647	0	33379	789	98231	45	5.4	0.0

Tabela 15

Scenario 8: Dobavljanje korisnika kojima su dodeljene permisije za neku nekretninu -> izmena dobavljenih permisija. U okviru ovog scenarija se takođe vrši pretraga korisnika po ključu jednom ili dva puta i **nije** sekvencijalno sa dobavljanjem i izmenom.

Maksimalan broj korisnika koji su istovremeno slali zahteve: 300

Povećanje broja korisnika po sekundi: 10

Trajanje testa: 2 minute

Test 1: Broj postojećih korisnika u bazi: 100 000 (bez keširanja)

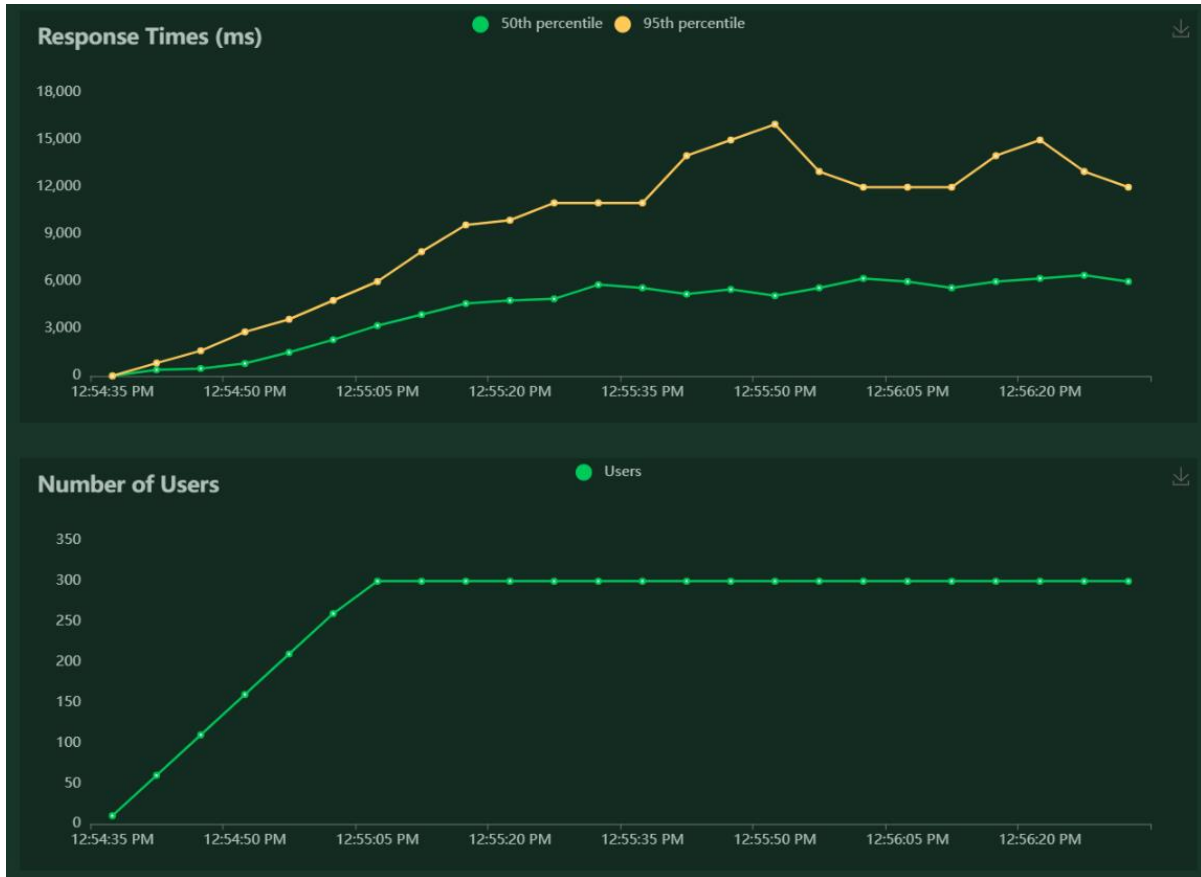
Broj postojećih permisija u bazi: 10 000 (bez keširanja)

U tabeli 16 se nalaze osnovni podaci o izvršenom testu za Test 1.

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
PUT	Permission Editing	933	0	4748	71	56033	60	7.8	0.0
GET	Shared Permissions For Property	933	0	4453	65	56025	9	7.8	0.0
GET	User Search	3125	0	5713	325	27612	74	26.0	0.0
Aggregated		4991	0	5297	65	56033	59	41.6	0.0

Tabela 16

Na grafiku 15 se nalazi prikaz prosečnog vremena potrebnog za izvršenje akcija u milisekundama i povećanje broja korisnika u toku vremena za Test 1.



Grafik 15

Test 2: Broj postojećih korisnika u bazi: 1 000 000 (sa keširanjem)

Broj postojećih permisija u bazi: 100 000 (sa keširanjem)

U tabeli 17 se nalaze osnovni podaci o izvršenom testu za Test 2.

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
PUT	Permission Editing	432	0	14086	1384	26971	60	3.6	0.0
GET	Shared Permissions For Property	432	0	14509	1487	26776	2	3.6	0.0
GET	User Search	598	0	19334	1286	44961	144	5.0	0.0
Aggregated		1462	0	16357	1286	44961	77	12.2	0.0

Tabela 17

Na grafiku 16 se nalazi prikaz prosečnog vremena potrebnog za izvršenje akcija u milisekundama i povećanje broja korisnika u toku vremena za Test 2.



Grafik 16

Uočeno je znatno smanjenje maksimalnog vremena čekanja između dva testna slučaja, iako je prosečno vreme veće. Optimizacija u vezi pretrage korisnika po ključu se i ovde primenjuje. U tabeli 18 su prikazane početne informacije o testnom slučaju 2 bez ove optimizacije, dok je u tabeli 17 prikazano stanje nakon nje.

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
PUT	Permission Editing	154	0	35604	1481	80233	60	1.3	0.0
GET	Shared Permissions For Property	154	0	22886	845	61865	4	1.3	0.0
GET	User Search	213	0	51003	16801	70866	76	1.8	0.0
Aggregated		521	0	38140	845	80233	50	4.3	0.0

Tabela 18

Scenario 9: Logovanje -> dobavljanje korisničkih uređaja -> dobavljanje poslednjih 100 komandi izvršenih nad uređajem. U sklopu dobavljanja komandi se dobavlja i lista svih korisnika koji su u bilo kom trenutku poslali komandu na uređaj. Odabir korisnika i korisničkog uređaja se vrši nasumično.

Maksimalan broj korisnika koji su istovremeno slali zahteve: 300

Povećanje broja korisnika po sekundi: 10

Trajanje testa: 2 minute

Test 1: Broj postojećih komandi u bazi: ~ 150 000 (bez keširanja)

Broj postojećih korisnika u bazi: 100 000 (bez keširanja)

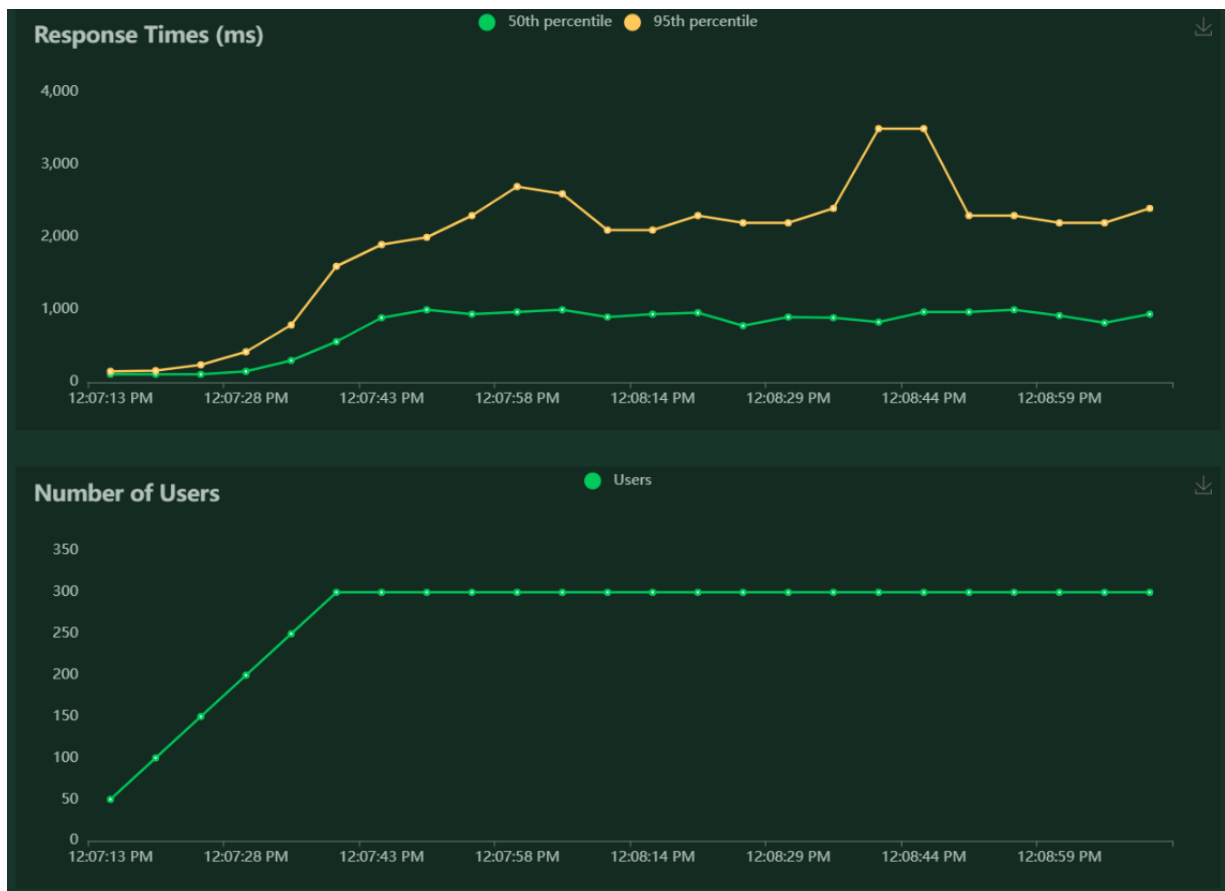
Broj postojećih uređaja u bazi: 90 000 (bez keširanja)

U tabeli 19 se nalaze osnovni podaci o izvršenom testu za Test 1.

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
GET	/api/devices/ownerAll	5825	0	643	12	10352	157	48.4	0.0
POST	/api/user/login	5825	0	1156	78	9100	289	48.4	0.0
GET	Commands	968	0	3960	319	14535	9543	8.0	0.0
	Aggregated	12618	0	1134	12	14535	938	104.8	0.0

Tabela 19

Na grafiku 17 se nalazi prikaz prosečnog vremena potrebnog za izvršenje akcija u milisekundama i povećanje broja korisnika u toku vremena za Test 1.



Grafik 17

Test 2: Broj postojećih komandi u bazi: ~ 770 000 (sa keširanjem)

Broj postojećih korisnika u bazi: 1 000 000 (sa keširanjem)

Broj postojećih uređaja u bazi: 900 000 (sa keširanjem)

U tabeli 20 se nalaze osnovni podaci o izvršenom testu za Test 2.

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
GET	/api/devices/ownerAll	2379	0	2997	107	22451	62	19.8	0.0
POST	/api/user/login	2379	0	2540	126	8506	290	19.8	0.0
GET	Commands	176	0	32700	1024	63823	1897	1.5	0.0
	Aggregated	4934	0	3836	107	63823	237	41.1	0.0

Tabela 20

Na grafiku 18 se nalazi prikaz prosečnog vremena potrebnog za izvršenje akcija u milisekundama i povećanje broja korisnika u toku vremena za Test 2.



Grafik 18

Uočen je veliki problem performantnosti upita koji se šalje ka *time-series* bazi podataka koja skladišti izvršene komande, za koji još uvek nije pronađeno optimalno rešenje. Implementirano rešenje podrazumeva paginaciju prilikom dobavljanja podataka.

Scenario 10: Dobavljanje komandi za uređaj i njihovo filtriranje. Prvo se dobavljaju poslednjih 100 komandi i lista korisnika koja je u bilo kom trenutku izvršila komandu, a nakon toga se nasumično bira stranica i korisnik radi filtracije. Vremenski opseg je postavljen na najveći mogući.

Maksimalan broj korisnika koji su istovremeno slali zahteve: 300

Povećanje broja korisnika po sekundi: 10

Trajanje testa: 2 minute

Test 1: Broj postojećih komandi u bazi: ~ 150 000 (bez keširanja)

U tabeli 21 se nalaze osnovni podaci o izvršenom testu za Test 1.

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
GET	Filtered	918	0	16504	726	76938	251	7.6	0.0
GET	First Fetch	387	0	28143	2206	92413	2035	3.2	0.0
	Aggregated	1305	0	19956	726	92413	780	10.9	0.0

Tabela 21

Na grafiku 19 se nalazi prikaz prosečnog vremena potrebnog za izvršenje akcija u milisekundama i povećanje broja korisnika u toku vremena za Test 1.



Grafik 19

Test 2: Broj postojećih komandi u bazi: ~ 770 000 (sa keširanjem)

U tabeli 22 se nalaze osnovni podaci o izvršenom testu za Test 2.

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
GET	Filtered	999	0	12754	1457	60323	262	8.4	0.0
GET	First Fetch	439	0	30324	1498	63842	2043	3.7	0.0
	Aggregated	1438	0	18118	1457	63842	806	12.0	0.0

Tabela 22

Na grafiku 20 se nalazi prikaz prosečnog vremena potrebnog za izvršenje akcija u milisekundama i povećanje broja korisnika u toku vremena za Test 2 .



Grafik 20

Uočeno je znatno manje povećanje vremena čekanja između dva testna slučaja.

TESTIRANJE PERFORMANSI PRILIKOM RADA SA SIMULATORIMA

Ovaj vid testiranja je podrazumevao praćenje vremenskih odrednica kada je određena informacija poslata sa jedne, i primljena sa druge strane. Sva komunikacija između aplikacije i simulatora se izvodila preko *WebSocket* protokola i MQTT protokola. *WebSocket* je povezivao korisničku aplikaciju sa *socket* serverom, a on je sa simulacionim povezan preko MQTT-a. Akcenat u ovoj sekciji jeste merenje brzine odziva sistema na korisničke komande sa i ka uređaju - u ovom procesu ne učestvuje REST server, već samo *socket* server.

TERMOMETAR

Merene su performanse sistema usled povećanja broja termometara koji šalju izmerenu temperaturu i vlažnost vazduha na platformu.

Test 1: **Broj aktivnih termometara:** 1000

Trajanje testa: ~ 3 minute

Uočeno je stabilno ponašanje aplikacije sa 1000 pokrenutih niti termometra. Merenja koja su poslata sa simulatora su iste sekunde primljena sa strane korisničke aplikacije, čime je izmereno kašnjenje mere milisekundi.

Test 2: **Broj aktivnih termometara:** 10 000

Trajanje testa: 2 minute

Uočeno je stabilno ponašanje aplikacije i sa 10 000 pokrenutih niti termometra. Merenja koja su poslata sa simulatora su iste sekunde primljena sa strane korisničke aplikacije, čime je izmereno kašnjenje mere milisekundi.

VEŠ MAŠINA

Funkcionalnosti koje su bile testirane:

- Uključivanje uređaja
- Dobavljanje karakteristika (mogućnosti) uređaja - dobija od *socket* servera i traži se samo jednom
- Dobavljanje zakazanih paljenja

- Zakazivanje uključivanja uređaja
- Brisanje zakazanog uključivanja uređaja

Merene su performanse sistema u dva test slučaja - prvi test slučaj testira performanse usled postepenog povećanja aktivnih veš mašina, dok drugi testira performanse usled povećanja korisnika koji su povezani na *socket* server i periodično šalju komande na aktivne veš mašine. U prvom test slučaju se broj veš mašina kreće postepeno ka **1000**, pri čemu se svake sekunde aktiviraju po **2** nove veš mašine. U drugom test slučaju je broj aktivnih veš mašina **1000**, a broj konektovanih korisnika se povećava za 2 svake sekunde. Ovo je nastavljeno dokle god je bilo dozvoljeno kreiranje novih *websocket* konekcija na računaru.

Test 1: Trajanje testa: ~7 minuta

Uočeno je stabilno ponašanje aplikacije usled povećanja broja aktivnih veš mašina i podataka koje šalju. Izvršeno je po 15 komandi dobavljanja, zakazivanja i brisanja zakazanog uključivanja uređaja, i 6 (maksimum) komandi uključivanja uređaja po minuti. Komande koje su poslate su iste sekunde pristigle na simulator, kao i podaci koje je simulator periodično i uzročno slao ka platformi.

Test 2: Trajanje testa: ~10 minuta

Dostignut broj konekcija: 304 - svaka je na 15 sekundi periodično slala komande ka svom uređaju (veš mašini).

Uočeno je identično ponašanje aplikacije kao i sa prethodnog test slučaja. Održan je isti broj poslatih komandi kao u prethodnom slučaju.

KLIMA UREĐAJ

Funkcionalnosti koje su bile testirane:

- Uključivanje i isključivanje uređaja
- Menjanje načina rada/ciljne temperature/brzine strujanja vazduha
- Dobavljanje karakteristika (mogućnosti) uređaja - dobija od *socket* servera i traži se samo jednom
- Dobavljanje zakazanih uključivanja
- Zakazivanje uključivanja uređaja
- Brisanje zakazanog uključivanja uređaja

Merene su performanse sistema u dva test slučaja - prvi test slučaj testira performanse usled postepenog povećanja aktivnih klima uređaja, dok drugi testira performanse usled povećanja korisnika koji su povezani na *socket* server i periodično šalju komande na aktivne klima uređaje. U prvom test slučaju se broj klima uređaja kreće postepeno ka **1000**, pri čemu se svake sekunde aktiviraju po **2** nove. U drugom test slučaju je broj aktivnih klima

uređaja **1000**, a broj konektovanih korisnika se povećava za 2 svake sekunde. Ovo je nastavljeno dokle god je bilo dozvoljeno kreiranje novih *websocket* konekcija na računaru.

Test 1: Trajanje testa: ~7 minuta

Uočeno je stabilno ponašanje aplikacije usled povećanja broja aktivnih klima uređaja i podataka koje šalju. Izvršeno je po 10 komandi uključivanja, menjanja načina rada i isključivanja uređaja, i po 5 komandi zakazivanja i brisanja zakazanog uključivanja uređaja po minuti. Komande koje su poslate su iste sekunde pristigle na simulator, kao i podaci koje je simulator periodično i uzročno slao ka platformi.

Test 2: Trajanje testa: ~6 minuta

Dostignut broj konekcija: 318 - svaka je na 15 sekundi periodično slala komande ka svom uređaju (klima).

Uočeno je identično ponašanje aplikacije kao i sa prethodnog test slučaja. Održan je isti broj poslatih komandi kao u prethodnom slučaju.

Problemi koji mogu nastati kod rada sa simulatorima jesu problemi kod dobavljanja mogućnosti uređaja za uređaje kojima se može upravljati, s obzirom da se taj podatak pretražuje u bazi podataka na strani *socket* servera. Povećanjem broja uređaja u bazi se direktno povećava vreme čekanja, ukoliko podatak nije prethodno keširan. Pored toga, postoji ograničenje u vidu mogućeg broja povezanih korisnika na jedan *websocket* server, što bi zahtevalo njegovo multipliciranje u slučaju velikog broja korisnika.

Velika količina podataka koje se šalju sa simulatora na *time-series* bazu podataka takođe može predstavljati problem. S obzirom da se svaka pristignuta vrednost beleži pojedinačno, puno resursa se troši na uspostavljanje veze sa bazom i upis u nju. Optimizacioni korak koji je potrebno implementirati jeste upis u *time-series* bazu u paketima tj. *batch*-evima.